



# Πανεπιστήμιο Πειραιώς

## Τμήμα Ψηφιακών Συστημάτων

Π.Μ.Σ. Διδακτική της Τεχνολογίας και  
Ψηφιακά Συστήματα  
Κατεύθυνση Δικτυοκεντρικών Συστημάτων

### Διπλωματική Εργασία

Μελέτη και Ανάπτυξη Πληροφοριακού Συστήματος Υγείας με χρήση  
SOA και Web Services

**Στεφανία Π. Μαυρογιαννέα**

Πειραιάς  
Δεκέμβριος 2011



**Πανεπιστήμιο Πειραιώς**

**Τμήμα Ψηφιακών Συστημάτων**

Π.Μ.Σ. Διδακτική της Τεχνολογίας και Ψηφιακά Συστήματα

Κατεύθυνση Δικτυοκεντρικών Συστημάτων

Επιβλέπων Καθηγητής:

**Μαρίνος Θεμιστοκλέους**, Επίκουρος καθηγητής στο Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς.

Τριμελής Εξεταστική Επιτροπή:

Μ. Θεμιστοκλέους, Επίκουρος καθηγητής στο Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς.

Α. Πρέτζα, Επίκουρος καθηγήτρια στο Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς.

Α. Αλεξίου, Επίκουρος καθηγήτρια στο Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς.

Η Διπλωματική Εργασία υποβάλλεται προς μερική κάλυψη των απαιτήσεων για την απόκτηση του Μεταπτυχιακού Διπλώματος Σπουδών του Μεταπτυχιακού Προγράμματος Διδακτικής της Τεχνολογίας και Ψηφιακών Συστημάτων.





## Πίνακας περιεχομένων

Περίληψη .....	6
Εισαγωγή.....	7
Ενότητα Α Πληροφοριακά Συστήματα Υγείας και Υπηρεσιοστρεφής Αρχιτεκτονική .....	14
Κεφάλαιο 1 Πληροφοριακά Συστήματα στον τομέα της Υγείας.....	15
Εισαγωγή.....	15
1.1 Η πληροφορία στην Υγεία .....	15
1.2 Πληροφοριακά Συστήματα Υγείας.....	17
1.3 Ιατρικός Φάκελος Υγείας.....	20
1.3.1 Σκοπός του Ιατρικού Φακέλου .....	21
1.3.2 Περιεχόμενα του Ιατρικού Φακέλου.....	21
1.3.3 Προσβασιμότητα Ιατρικού Φακέλου.....	24
1.4 Χειρόγραφος Φάκελος Υγείας .....	25
1.4.1 Πλεονεκτήματα χειρόγραφου φακέλου .....	26
1.4.2 Μειονεκτήματα χειρόγραφου φακέλου .....	26
1.5 Ηλεκτρονικός Φάκελος Υγείας.....	27
1.5.1 Ορισμός.....	27
1.5.2 Είδη του Ηλεκτρονικού Φακέλου .....	27
1.5.3 Σύγκριση με τον χειρόγραφο φάκελο.....	28
1.6 Πρότυπα και κωδικοποιήσεις.....	29
1.6.1 Πρότυπα επικοινωνίας.....	29
1.6.2 Τυποποίηση κλινικών δεδομένων .....	31
1.6.3 Πρότυπα Ταυτοποίησης.....	32
1.6.4 Πρότυπα εξασφάλισης του απορρήτου των δεδομένων .....	33
1.7 Πλεονεκτήματα της εφαρμογής του Ηλεκτρονικού Φακέλου .....	33
1.8 Ζητήματα που σχετίζονται με την εφαρμογή του Ηλεκτρονικού Φακέλου .....	35



1.8.1	Τεχνολογικά ζητήματα .....	35
1.8.2	Ζητήματα ασφάλειας και εμπιστευτικότητας των δεδομένων .....	37
1.8.3	Ζητήματα αποδοχής .....	38
1.8.4	Ζητήματα οργάνωσης .....	39
1.8.5	Ζητήματα χρηματοδότησης .....	40
1.9	Προσωπικός Φάκελος Υγείας .....	40
1.9.1	Microsoft Health Vault .....	41
1.9.2	Google Health .....	42
	Συμπεράσματα .....	43
	Κεφάλαιο 2 Υπηρεσίες Διαδικτύου και Υπηρεσιοστρεφής Αρχιτεκτονική .....	45
	Εισαγωγή .....	45
2.1	Υπηρεσίες Διαδικτύου .....	46
2.1.1	Πλεονεκτήματα των web services .....	47
2.1.2	Τεχνικά στοιχεία των web services .....	48
2.1.3	Τα επίπεδα των web services .....	53
2.1.4	Τύποι Υπηρεσιών Διαδικτύου .....	55
2.1.5	Ασφάλεια στις Υπηρεσίες Διαδικτύου .....	56
2.2	Υπηρεσιοστρεφής Αρχιτεκτονική .....	59
2.2.1	Τι είναι η SOA .....	59
2.2.2	Προέλευση της υπηρεσιοστρεφούς αρχιτεκτονικής .....	61
2.2.3	Η στοίβα της SOA .....	64
2.2.4	Πλεονεκτήματα της SOA .....	65
2.2.5	Ασφάλεια σε περιβάλλοντα αρχιτεκτονικής SOA .....	67
2.3	Διαχείριση Επιχειρηματικών Διαδικασιών .....	74
2.3.1	Η ανάγκη για επιχειρηματικές διαδικασίες .....	74
2.3.2	Ενορχήστρωση και Χορογραφία υπηρεσιών .....	76
2.3.3	BP4WS .....	78



Συμπεράσματα .....	82
Ενότητα Β' Έρευνα και Ερευνητική Μεθοδολογία .....	83
Κεφάλαιο 3 Έρευνα και Ερευνητική Μεθοδολογία.....	84
Εισαγωγή.....	84
3.1 Στόχος της έρευνας.....	84
3.2 Δείγμα της έρευνας.....	84
3.3 Ερευνητική μεθοδολογία.....	85
3.3.1 Πρωτογενής έρευνα (Ποιοτική).....	85
3.3.2 Δευτερογενής έρευνα (Ποσοτική).....	86
3.4 Τα αποτελέσματα της ποσοτικής έρευνας .....	87
Συμπεράσματα .....	95
Ενότητα Γ' Πληροφοριακό Σύστημα Υγείας .....	96
Κεφάλαιο 4 Ανάλυση του Συστήματος - Λειτουργικές Απαιτήσεις και Σχεδιασμός Αρχιτεκτονικής.....	97
Εισαγωγή.....	97
4.1 Περιγραφή του προς υλοποίηση συστήματος.....	97
4.2 Επιχειρησιακές διαδικασίες και λειτουργικές προδιαγραφές .....	99
4.3 Σχεδιασμός Αρχιτεκτονικής .....	101
4.3.1 Υποσύστημα Διεπαφής Χρήστη.....	103
4.3.2 Υποσύστημα Επιχειρηματικών Διαδικασιών .....	104
4.3.3 Υποσύστημα Σύνθετων Υπηρεσιών .....	104
4.3.4 Υποσύστημα Απλών Υπηρεσιών .....	105
4.3.5 Υποσύστημα Βάσης Δεδομένων.....	105
4.4 Οι Υπηρεσίες του συστήματος .....	105
4.4.1 Επιχειρησιακές Υπηρεσίες(Business Services) .....	106
4.4.2 Τοπικές Υπηρεσίες(Domain Services) .....	120
4.4.3 Υπηρεσίες Ολοκλήρωσης(Integration Services).....	125

4.4.4	Υπηρεσίες Υποστήριξης (Utility Services) .....	128
4.4.5	Βασικές Υπηρεσίες (Foundation Services) .....	130
4.5	Η Βάση Δεδομένων του συστήματος .....	131
	Κεφάλαιο 6 Πρότυπα, Εργαλεία, Τεχνολογίες και Γλώσσες Υλοποίησης .....	149
	Κεφάλαιο 7 Συμπεράσματα .....	154
	Παράρτημα Α Ηλεκτρονικός Ιατρικός Φάκελος Ασθενούς –Ερωτηματολόγιο.....	158
	Παράρτημα Β Παράδειγμα χρήσης του συστήματος .....	165
	Παράρτημα Γ Παρουσίαση μιας ολοκληρωμένης επιχειρηματικής διαδικασίας.....	183
	Βιβλιογραφία και Αναφορές.....	196

## Λίστα Εικόνων

<i>Εικόνα 1: Ροή πληροφορίας στην Υγεία</i> <sup>[E2]</sup> .....	17
<i>Εικόνα 2: Συστατικά πληροφοριακού συστήματος</i> <sup>[E3]</sup> .....	18
<i>Εικόνα 3: Σύστημα Υγείας ως παραγωγική διαδικασία</i> <sup>[E4]</sup> .....	18
<i>Εικόνα 4: Συνολικό διάγραμμα ενός Ολοκληρωμένου Π.Σ. Υγείας</i> <sup>[E5]</sup> .....	20
<i>Εικόνα 5: Το μοντέλο των web services</i> <sup>[E7]</sup> .....	49
<i>Εικόνα 6: Η αρχιτεκτονική των web services</i> <sup>[E8]</sup> .....	49
<i>Εικόνα 7: Τεχνολογίες των web services</i> <sup>[E8]</sup> .....	50
<i>Εικόνα 8: Βήματα για την ανάπτυξη μίας εφαρμογής web services.</i> <sup>[E8]</sup> .....	53
<i>Εικόνα 9: Στρώμα Υπηρεσίας Διαδικτύου</i> <sup>[E9]</sup> .....	53
<i>Εικόνα 10: Προέλευση και επιρροές της SOA</i> <sup>[E10]</sup> .....	61
<i>Εικόνα 11: Η στοίβα της SOA</i> <sup>[E11]</sup> .....	64
<i>Εικόνα 12: Top Down Σχεδιασμός Αρχιτεκτονικής</i> <sup>[E14]</sup> .....	102
<i>Εικόνα 13: Η αρχιτεκτονική του συστήματος</i> .....	103
<i>Εικόνα 14: Business Service – Login_WebService</i> .....	107
<i>Εικόνα 15: Business Service – AddDoctor_WebService</i> .....	109
<i>Εικόνα 16: Business Service – WFDoctorEdit_WebService</i> .....	110
<i>Εικόνα 17: Business Service – WFDoctorDelete_WebService</i> .....	111
<i>Εικόνα 18: Business Service – SearchDoctor_WebService</i> .....	112
<i>Εικόνα 19: Business Service – AddPatient_WebService</i> .....	113





Εικόνα 20: Business Service – EditPatient_WebService .....	114
Εικόνα 21: Business Service – SearchPatient_WebService .....	115
Εικόνα 22: Business Service – WFPatientDataForSecretary_WebService .....	116
Εικόνα 23: Business Service – WFPatientDataForDoctor_WebService .....	117
Εικόνα 24: Business Service – WFPatientDataForMedicalRecord_WebService .....	118
Εικόνα 25: Business Service – WFPatientDataForPHR_WebService .....	119
Εικόνα 26: Domain Service – CISClinicsWS .....	120
Εικόνα 27: Domain Service – CISDoctorWS .....	121
Εικόνα 28: Domain Service – CISPatientWS .....	121
Εικόνα 29: Domain Service – CISPatientAppointmentsWS .....	122
Εικόνα 30: Domain Service – CISPatientTreatmentPlansWS .....	123
Εικόνα 31: Domain Service – CISPatientFinancialDataWS .....	124
Εικόνα 32: Domain Service – CISPatientMedicationsWS .....	124
Εικόνα 33: Domain Service – CISPatientHistoryWS .....	125
Εικόνα 34: Integration Service – PatientDataWS .....	125
Εικόνα 35: Integration Service – PatientAllergiesWS .....	126
Εικόνα 36: Integration Service – PatientAppointmentsWS .....	126
Εικόνα 37: Integration Service – PatientFinancialDataWS .....	127
Εικόνα 38: Integration Service – PatientHistoryWS .....	127
Εικόνα 39: Integration Service – PatientMedicationsWS .....	128
Εικόνα 40: Integration Service – PatientTreatmentPlansWS .....	128
Εικόνα 41: Utility Service – UsersWS .....	129
Εικόνα 42: Utility Service – AppointmentsWS .....	129
Εικόνα 43: Utility Service – LookupsWS .....	130
Εικόνα 44: Utility Service – FieldCaptionsWS .....	130
Εικόνα 45: Foundation Service – ExecuteQRYWS .....	131
Εικόνα 46: Foundation Service – RunProcedureWS .....	131

## Περίληψη

Είναι πλέον κοινά αποδεκτό ότι ο όγκος της πληροφορίας που σχετίζεται με την ιατρική φροντίδα ενός ασθενούς έχει αυξηθεί κατά πολύ τα τελευταία χρόνια. Οι κλασικοί χειρόγραφοι φάκελοι ασθενών, που βασίζονται στην καταγραφή των δεδομένων του ασθενούς σε χαρτί, αδυνατούν να συγκρατήσουν τον μεγάλο όγκο των πληροφοριών, με αποτέλεσμα την απώλεια ιατρικών δεδομένων και την αδυναμία διατήρησης της χρονικής συσχέτισης των διαφόρων εξετάσεων με το ιστορικό του ασθενούς. Έτσι προκύπτει η ανάγκη δημιουργίας και διατήρησης ηλεκτρονικών φακέλων, οι οποίοι θα έχουν συγκεντρωμένες όλες τις ιατρικές πληροφορίες του ασθενούς, με σκοπό την παροχή καλύτερης και ποιοτικότερης ιατροφαρμακευτικής περίθαλψης.

Αν και έχουν γίνει προσπάθειες ανάπτυξης και εφαρμογής τέτοιων Πληροφοριακών Συστημάτων Υγείας, πολλά από αυτά δεν έχουν εισαγάγει επιτυχώς τον Ηλεκτρονικό Φάκελο Υγείας. Οι λόγοι που συμβάλουν σε αυτή την κατάσταση είναι πολλοί, με σημαντικότερους την έλλειψη διαλειτουργικότητας, διάφορα ζητήματα αποδοχής και τεχνολογίας και την ανησυχία για την ασφάλεια των ιατρικών δεδομένων.

Λύση σε πολλά από αυτά τα ζητήματα έρχεται να δώσει μια σχετικά νέα τεχνολογία στον τομέα της πληροφορικής. Η Υπηρεσιοστρεφής Αρχιτεκτονική (Service Oriented Architecture - SOA) επιτρέπει τη δυναμική δημιουργία σύνθετων εφαρμογών και επιχειρηματικών διαδικασιών σε πραγματικό χρόνο με μοναδικά οφέλη. Συγκεκριμένα, προσφέρει οφέλη στο κόστος ανάπτυξης και συντήρησης των εφαρμογών, απλοποιημένη ενοποίηση σε ετερογενή περιβάλλοντα, μεγαλύτερη αξιοποίηση των υπάρχοντων πόρων πληροφορικής και αποτελεσματική διαχείριση και ασφάλεια. Η αρχιτεκτονική SOA αντιλαμβάνεται κάθε οργανισμό σαν ένα σύνολο από συνδεδεμένες υπηρεσίες και με μια προσέγγιση βασισμένη σε ανοικτά πρότυπα οδηγεί σε πιο αποτελεσματικές και αποδοτικές διαδικασίες.

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ:** Υπηρεσιοστρεφής Αρχιτεκτονική

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:** Ηλεκτρονικός Φάκελος Ασθενούς, Υπηρεσίες Διαδικτύου, Υπηρεσιοστρεφής Αρχιτεκτονική, Πληροφοριακά Συστήματα Υγείας, Ιατρική Πληροφορική



## Εισαγωγή

Ο βαθμός αρτιότητας της λειτουργίας ενός οργανισμού προσδιορίζεται από τις τιμές που παίρνουν ορισμένα βασικά χαρακτηριστικά, τα κυριότερα από τα οποία είναι:

- **Αποτελεσματικότητα (Effectiveness)** : Εκφράζει το μέτρο κατά το οποίο ο οργανισμός ανταποκρίνεται στην αποστολή του, δηλαδή κατά πόσο πετυχαίνει τους στόχους του και κατά πόσο οι υπηρεσίες που προσφέρει είναι οι αναμενόμενες, τόσο ποιοτικά όσο και ποσοτικά.
- **Αποδοτικότητα (Efficiency)**: Εκφράζει το μέτρο επιτυχίας του συνδυασμού των πόρων (ανθρώπων, μηχανών, κεφαλαίων) που διαθέτει ο οργανισμός για την επίτευξη της αποστολής του.
- **Αξιοπιστία (Reliability)**: Εκφράζει το μέτρο κατά το οποίο ο Οργανισμός πετυχαίνει τους στόχους του με συνέπεια.
- **Ευελξία (Flexibility) - Ικανότητα Προσαρμογής (Adaptability)**: Εκφράζει την ικανότητα του Οργανισμού να ανταποκρίνεται και να προσαρμόζεται σε αλλαγές και τροποποιήσεις που προέρχονται από το περιβάλλον του, όπως επίσης και την ικανότητα εκτέλεσης έκτακτων έργων χωρίς αξιόλογη διαταραχή της λειτουργίας του.

Ένας Οργανισμός έχει πληροφοριακό πρόβλημα, όταν ο μετασχηματισμός των «γεγονότων» σε πληροφορίες καθυστερεί χρονικά ή ακόμη όταν δεν έχει τα αναμενόμενα ποιοτικώς αποτελέσματα. Δηλαδή, όταν δεν παρέχονται έγκαιρα οι απαιτούμενες πληροφορίες, κατάλληλα συσχετισμένες στους εκτελεστικούς και ελεγκτικούς μηχανισμούς, αλλά και σε όλα τα επίπεδα της διοίκησης. Αυτό συνήθως έχει τα εξής επακόλουθα:

- Τη διατάραξη της εύρυθμης λειτουργίας του Οργανισμού
- Την επιδείνωση της ποιότητας της εξυπηρέτησης που προσφέρεται.
- Την ελλιπή λειτουργία του συστήματος ελέγχου.
- Την διατάραξη των σχέσεων του Οργανισμού με το περιβάλλον του.

Μία από τις συνηθισμένες αιτίες δημιουργίας πληροφοριακού προβλήματος βρίσκεται στο χάσμα που υπάρχει μεταξύ ενός στατικού πληροφοριακού συστήματος και μιας μεταβαλλόμενης οργανωτικής δομής, αφού τα συστήματα που εξετάζουμε είναι δυναμικά. Συνηθισμένες ενδείξεις, που δημιουργούν στη Διοίκηση την ανάγκη να προχωρήσει σε διερεύνηση του ανασχεδιασμού του πληροφοριακού συστήματος και της χρήσης πληροφορικής, είναι:

- Η αύξηση του όγκου των πρωτογενών πληροφοριών που απαιτούν επεξεργασία
- Η αλλαγή των απαιτήσεων σε πληροφορίες
- Η απαίτηση για τη μείωση του χρόνου επεξεργασίας, που συνήθως εμφανίζεται σε συνδυασμό με την απαίτηση για βελτίωση της ποιότητας των αποτελεσμάτων της επεξεργασίας (μείωση λαθών).
- Η προβληματική λειτουργία του συστήματος επικοινωνίας των εσωτερικών λειτουργιών, τόσο μεταξύ τους όσο και με το περιβάλλον.
- Το υψηλό υπαλληλικό κόστος, τα λάθη των επεξεργασιών και η εκτέλεση περιττών επεξεργασιών.
- Η μείωση της αποτελεσματικότητας του Οργανισμού.
- Η αδυναμία εκτέλεσης εργασιών ή ολόκληρων λειτουργιών με τα συμβατικά υπαλληλικά μέσα.

### Αντιμετώπιση του προβλήματος

Γενικά, το πρόβλημα της επεξεργασίας των πληροφοριών μπορεί να λυθεί είτε με χειρόγραφη επεξεργασία είτε με τη χρήση πληροφορικής και εισαγωγή Η/Υ. Οι παράγοντες που οδηγούν στην επιλογή της πιο κατάλληλης από τις δύο λύσεις είναι οι τιμές που παίρνουν στην κάθε περίπτωση τα ακόλουθα χαρακτηριστικά:

- Ο όγκος των πληροφοριών για επεξεργασία
- Η πολυπλοκότητα της επεξεργασίας.
- Η συχνότητα της επεξεργασίας.
- Η αλγοριθμική επαναληπτική μορφή της επεξεργασίας.
- Το κόστος επεξεργασίας και αποθήκευσης.
- Οι απαιτήσεις σε χρόνο αντίδρασης.
- Η σημαντικότητα και συχνότητα των λαθών.

### Ερευνητική Περιοχή

Η Ιατρική Πληροφορική (Healthcare Informatics) είναι το σημείο τομής της πληροφορικής, της επιστήμης των υπολογιστών, και της υγειονομικής περίθαλψης. Συνδυάζει τους τομείς της πληροφορικής και της υγείας για την ανάπτυξη των συστημάτων που απαιτούνται για την διαχείριση των αυξανόμενων πληροφοριών, των πολύπλοκων κλινικών διαδικασιών και την βελτίωση της ασφάλειας του συστήματος υγείας. Ασχολείται με τους πόρους, τις συσκευές



και τις μεθόδους που απαιτούνται για τη βελτίωση της αποθήκευσης, της ανάκτησης και της χρήσης των πληροφοριών στον τομέα της υγείας και της βιοϊατρικής.

Τα εργαλεία της Ιατρικής Πληροφορικής δεν περιλαμβάνουν μόνο υπολογιστές αλλά και κλινικές κατευθυντήριες γραμμές, τυποποιημένες ιατρικές ορολογίες, καθώς και συστήματα επικοινωνίας. Εφαρμόζεται στους τομείς της νοσηλευτικής, της κλινικής φροντίδας, της οδοντιατρικής, της φαρμακευτικής, της δημόσιας υγείας και της (βιο) ιατρικής έρευνας.

Η ιατρική πληροφορική χρησιμοποιεί υπολογιστές, εξειδικευμένο λογισμικό και συσκευές επικοινωνίας διαμορφώνοντας ένα πολύπλοκο δίκτυο, με σκοπό την συλλογή, την ανάλυση και τη διαβίβαση ιατρικών διαδικασιών. Τα εργαλεία για τη δημιουργία συστημάτων πληροφορικής για την υγεία δεν περιορίζονται μόνο στην τεχνολογία της πληροφορικής. Τα συστήματα αυτά θα πρέπει, επίσης, να επιτρέπουν την αφομοίωση των κλινικών οδηγιών, την κατανόηση της επίσημης ιατρικής ορολογίας, την αποθήκευση των δεδομένων και την σαφή επικοινωνία. Η ιατρική πληροφορική μπορεί να εφαρμοστεί σε όλα τα είδη των περιβαλλόντων της υγείας, συμπεριλαμβανομένης της πρωτοβάθμιας φροντίδας, τη γενική ιατρική, τη νοσοκομειακή περίθαλψη και την αποκατάσταση.

Ένας από τους βασικούς στόχους της ιατρικής πληροφορικής είναι να διαμορφώσει μια τυποποιημένη προσέγγιση για την υγειονομική περίθαλψη σε διεθνές επίπεδο. Η ιδέα είναι να επωφεληθούν οι ερευνητές, οι πάροχοι, και οι ασθενείς από τα εργαλεία πληροφορικής, τις τεχνικές, τις ιδέες και τα πρωτόκολλα που αναδιαμορφώνουν την παροχή της υγειονομικής περίθαλψης και να προωθηθούν οι βέλτιστες πρακτικές σε αυτόν τον τομέα. Πολλοί ειδικοί συμφωνούν ότι μία από τις κύριες προκλήσεις της ιατρικής πληροφορικής είναι να πεισθούν οι πάροχοι της ιατρικής φροντίδας, ώστε να αποδεχθούν και να χρησιμοποιήσουν την συγκεκριμένη τεχνολογία. Μόνο τότε μπορεί να δημιουργηθεί ένα ομοιογενές και ολοκληρωμένο σύστημα υγειονομικής περίθαλψης, το οποίο θα βοηθήσει τους επαγγελματίες του κλάδου στην επίλυση προβλημάτων, τη λήψη αποφάσεων, την παροχή καλύτερων υπηρεσιών στους ασθενείς και την εκτέλεση των καθηκόντων τους πιο αποτελεσματικά και αποδοτικά.

Τα πληροφοριακά συστήματα υγείας μπορούν να χρησιμοποιηθούν για να αυξήσουν την αποδοτικότητα σε τρεις βασικούς τομείς της υγειονομικής περίθαλψης: κλινικά, διοικητικά και οικονομικά. Για παράδειγμα, μπορούν να συμβάλλουν στην περικοπή του κόστους και την ενίσχυση της φροντίδας του ασθενούς, μέσω της εφαρμογής ενός τυποποιημένου συστήματος Ηλεκτρονικού Φακέλου Υγείας. Είναι επίσης το κλειδί για την επέκταση των

συστημάτων υγείας ως προς την τιμολόγηση, την κλινική έρευνα, τον προγραμματισμό των ραντεβού, καθώς και την ανταλλαγή ιατρικών πληροφοριών.

## Ερευνητικό Πρόβλημα

Είναι πλέον κοινά αποδεκτό ότι ο όγκος της πληροφορίας που σχετίζεται με την ιατρική φροντίδα ενός ασθενούς έχει αυξηθεί κατά πολύ τα τελευταία χρόνια. Το γεγονός αυτό οφείλεται σε μεγάλο βαθμό στην ενσωμάτωση μεγάλου αριθμού εργαστηριακών και παραϊατρικών εξετάσεων στους φακέλους των ασθενών. Επιπλέον, τα καθήκοντα ιατρών και νοσηλευτών γίνονται διαρκώς περισσότερα, καθώς αυξάνει η πολυπλοκότητα των ιδρυμάτων παροχής υπηρεσιών υγείας.

Οι κλασικοί χειρόγραφοι φάκελοι ασθενών που βασίζονται στην καταγραφή των δεδομένων του ασθενούς σε χαρτί αδυνατούν να συγκρατήσουν τον μεγάλο όγκο των πληροφοριών, με αποτέλεσμα την απώλεια ιατρικών δεδομένων και την αδυναμία διατήρησης της χρονικής συσχέτισης των διαφόρων εξετάσεων με το ιστορικό του ασθενούς. Έτσι προκύπτει η ανάγκη δημιουργίας και διατήρησης ηλεκτρονικών φακέλων για τους ασθενείς, οι οποίοι θα έχουν συγκεντρωμένες όλες τις ιατρικές πληροφορίες του ασθενούς, με σκοπό την παροχή καλύτερης και ποιοτικότερης ιατροφαρμακευτικής περίθαλψης στον ασθενή.

Με τη ραγδαία ανάπτυξη της τεχνολογίας της πληροφορικής τις τελευταίες δεκαετίες και ειδικότερα των τεχνολογιών για την ιατροφαρμακευτική περίθαλψη, έχουν προταθεί πολλά και διαφορετικά μοντέλα Ηλεκτρονικού Φακέλου Υγείας. Σε πολλές χώρες, μάλιστα, έχει προγραμματιστεί η εισαγωγή ενός ηλεκτρονικού αρχείου υγείας σε εθνικό επίπεδο, ενώ σε άλλες έχει εφαρμοστεί σε κάποια μορφή του. Παρ' όλα αυτά, υπάρχει ποικιλομορφία ως προς τον τύπο, τη δομή και την υλοποίηση των αρχείων αυτών. Σε μερικές περιπτώσεις μπορεί να είναι ένας φάκελος διαθέσιμος ευρέως σε διάφορα ιδρύματα, ενώ σε άλλες μπορεί να αποτελεί ένα περιορισμένο σύστημα, το οποίο είναι διαθέσιμο μόνο σε μια συγκεκριμένη μονάδα ή τμήμα.

Αν και έχουν γίνει προσπάθειες ανάπτυξης και εφαρμογής τέτοιων πληροφοριακών συστημάτων υγείας, πολλά από αυτά δεν έχουν εισαγάγει επιτυχώς τον Ηλεκτρονικό Φάκελο Υγείας. Οι λόγοι που συμβάλουν σε αυτή την κατάσταση είναι πολλοί, με σημαντικότερους την έλλειψη διαλειτουργικότητας, διάφορα ζητήματα αποδοχής και τεχνολογίας και την ανησυχία για την ασφάλεια και την προστασία των ιατρικών δεδομένων.



Λύση σε πολλά από αυτά τα ζητήματα έρχεται να δώσει μια σχετικά νέα τεχνολογία στον τομέα της πληροφορικής. Η υπηρεσιοστρεφής αρχιτεκτονική (Service Oriented Architecture - SOA) επιτρέπει τη δυναμική δημιουργία σύνθετων εφαρμογών και επιχειρηματικών διαδικασιών σε πραγματικό χρόνο με μοναδικά οφέλη. Συγκεκριμένα, προσφέρει οφέλη στο κόστος ανάπτυξης και συντήρησης των εφαρμογών, απλοποιημένη ενοποίηση σε ετερογενή περιβάλλοντα, μεγαλύτερη αξιοποίηση των υπάρχοντων πόρων πληροφορικής και αποτελεσματική διαχείριση και ασφάλεια. Η αρχιτεκτονική SOA αντιλαμβάνεται κάθε οργανισμό σαν ένα σύνολο από συνδεδεμένες υπηρεσίες και με μια προσέγγιση βασισμένη σε ανοικτά πρότυπα οδηγεί σε πιο αποτελεσματικές και αποδοτικές διαδικασίες.

### **Σκοπός και Αντικειμενικοί Στόχοι**

Στην παρούσα διπλωματική εργασία θα ασχοληθούμε με τα Πληροφοριακά Συστήματα Υγείας και συγκεκριμένα με την ενσωμάτωση του Ηλεκτρονικού Φακέλου Υγείας σε αυτά.

Σκοπός μας είναι να προσδιορίσουμε τα πλεονεκτήματα του Ηλεκτρονικού Φακέλου Υγείας και να προτείνουμε μια υλοποίησή του, βασιζόμενοι στην Υπηρεσιοστρεφή Αρχιτεκτονική και τις Υπηρεσίες Διαδικτύου. Πιο συγκεκριμένα:

- ✓ Θα αναφερθούμε στην βιβλιογραφία, τόσο σχετικά με τον Ηλεκτρονικό Φάκελο Υγείας, όσο και με τις Υπηρεσίες Διαδικτύου και την Υπηρεσιοστρεφή Αρχιτεκτονική.
- ✓ Θα εντοπίσουμε τα διάφορα προβλήματα και εμπόδια που σχετίζονται με την εφαρμογή του Ηλεκτρονικού Φακέλου Υγείας.
- ✓ Θα δούμε πως οι Υπηρεσίες Διαδικτύου και η Υπηρεσιοστρεφής Αρχιτεκτονική μπορούν να ξεπεράσουν κάποια από τα παραπάνω εμπόδια.
- ✓ Θα διεξάγουμε μια έρευνα και θα παρουσιάσουμε τα αποτελέσματά της, σχετικά με την αποδοχή των συστημάτων Ηλεκτρονικού Φακέλου Υγείας από το ευρύ κοινό.
- ✓ Θα σχεδιάσουμε και θα υλοποιήσουμε ένα Πληροφοριακό Σύστημα Υγείας που θα επικοινωνεί με ένα σύστημα Ηλεκτρονικού Φακέλου Υγείας.

### **Δομή της Εργασίας**

Η παρούσα εργασία χωρίζεται σε τρεις ενότητες:

- Στην πρώτη ενότητα θα παρουσιάσουμε το θεωρητικό υπόβαθρο σχετικά με τα πληροφοριακά Συστήματα Υγείας, τον Ηλεκτρονικό Φάκελο Υγείας, τις Υπηρεσίες Διαδικτύου και την Υπηρεσιοστρεφή Αρχιτεκτονική.

- Στην δεύτερη ενότητα θα παρουσιάσουμε τα αποτελέσματα της έρευνας που διεξήχθη, σχετικά με την αποδοχή των συστημάτων Ηλεκτρονικού Φακέλου υγείας από το ευρύ κοινό.
- Στην τελευταία ενότητα θα παρουσιάσουμε τον σχεδιασμό και την υλοποίηση ενός Πληροφοριακού Συστήματος Υγείας που θα επικοινωνεί με ένα σύστημα Ηλεκτρονικού Φακέλου Υγείας.

Αναλυτικότερα, η εργασία χωρίζεται στα παρακάτω κεφάλαια:

### **Κεφάλαιο 1 - Πληροφοριακά Συστήματα στον τομέα της Υγείας**

Στο κεφάλαιο αυτό θα κάνουμε μια συνοπτική αναφορά στα Πληροφοριακά Συστήματα Υγείας και στον Ηλεκτρονικό Φάκελο Υγείας. Θα συγκρίνουμε τον ηλεκτρονικό και τον χειρόγραφο φάκελο υγείας, αναφέροντας τα πλεονεκτήματα και τα μειονεκτήματά τους, θα μιλήσουμε για τα πρότυπα και τις κωδικοποιήσεις που έχουν αναπτυχθεί σχετικά με τον ηλεκτρονικό φάκελο και θα αναλύσουμε τα διάφορα ζητήματα που σχετίζονται με την εφαρμογή του.

### **Κεφάλαιο 2 – Υπηρεσίες Διαδικτύου και Υπηρεσιοστρεφής Αρχιτεκτονική**

Στο κεφάλαιο αυτό θα αναφερθούμε στην βιβλιογραφία σχετικά με τις Υπηρεσίες Διαδικτύου και την Υπηρεσιοστρεφή Αρχιτεκτονική. Θα μιλήσουμε για τα πλεονεκτήματά τους, τα τεχνικά στοιχεία τους και τα επίπεδα ασφαλείας που προσφέρουν. Στη συνέχεια θα μιλήσουμε για τις Επιχειρηματικές Διαδικασίες καθώς και για την Ενορχήστρωση και την Χορογραφία των Υπηρεσιών Διαδικτύου.

### **Κεφάλαιο 3 - Έρευνα και Ερευνητική Μεθοδολογία**

Στο κεφάλαιο αυτό θα αναφερθούμε στα αποτελέσματα της έρευνας που διεξήχθη, σχετικά με την αποδοχή των συστημάτων Ηλεκτρονικού Φακέλου υγείας από το ευρύ κοινό. Θα αναφερθούμε στην ερευνητική μεθοδολογία που ακολουθήθηκε και θα παρουσιάσουμε τα πιο ενδιαφέροντα στατιστικά αποτελέσματα.

### **Κεφάλαιο 4 - Ανάλυση του Συστήματος – Λειτουργικές Απαιτήσεις και Σχεδιασμός Αρχιτεκτονικής**

Στο κεφάλαιο αυτό θα παρουσιάσουμε την ανάλυση και τον σχεδιασμό του Πληροφοριακού Συστήματος που αναπτύχθηκε. Θα περιγράψουμε το σύστημα που υλοποιήθηκε, την αρχιτεκτονική του και θα αναφερθούμε στις Υπηρεσίες που υλοποιήθηκαν. Τέλος, θα





περιγράψουμε την βάση δεδομένων του συστήματος, μέσω Διαγραμμάτων Οντοτήτων - Συσχετίσεων και Σχισιακών Μοντέλων.

### **Κεφάλαιο 5 – Πρότυπα, εργαλεία, Τεχνολογίες και Γλώσσες Υλοποίησης**

Στο κεφάλαιο αυτό θα αναφερθούμε συνοπτικά στα εργαλεία και τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος.

### **Κεφάλαιο 6 - Συμπεράσματα**

Στο τελευταίο κεφάλαιο θα αναφέρουμε τα συμπεράσματα που προέκυψαν από την εκπόνηση της διπλωματικής εργασίας και θα αναφερθούμε σε μελλοντικές προτάσεις.

### **Παράρτημα Α – Ηλεκτρονικός Ιατρικός Φάκελος Ασθενούς (Ερωτηματολόγιο)**

Στο Παράρτημα Α παρουσιάζεται το ερωτηματολόγιο της έρευνας

### **Παράρτημα Β – Περίπτωση χρήσης του συστήματος**

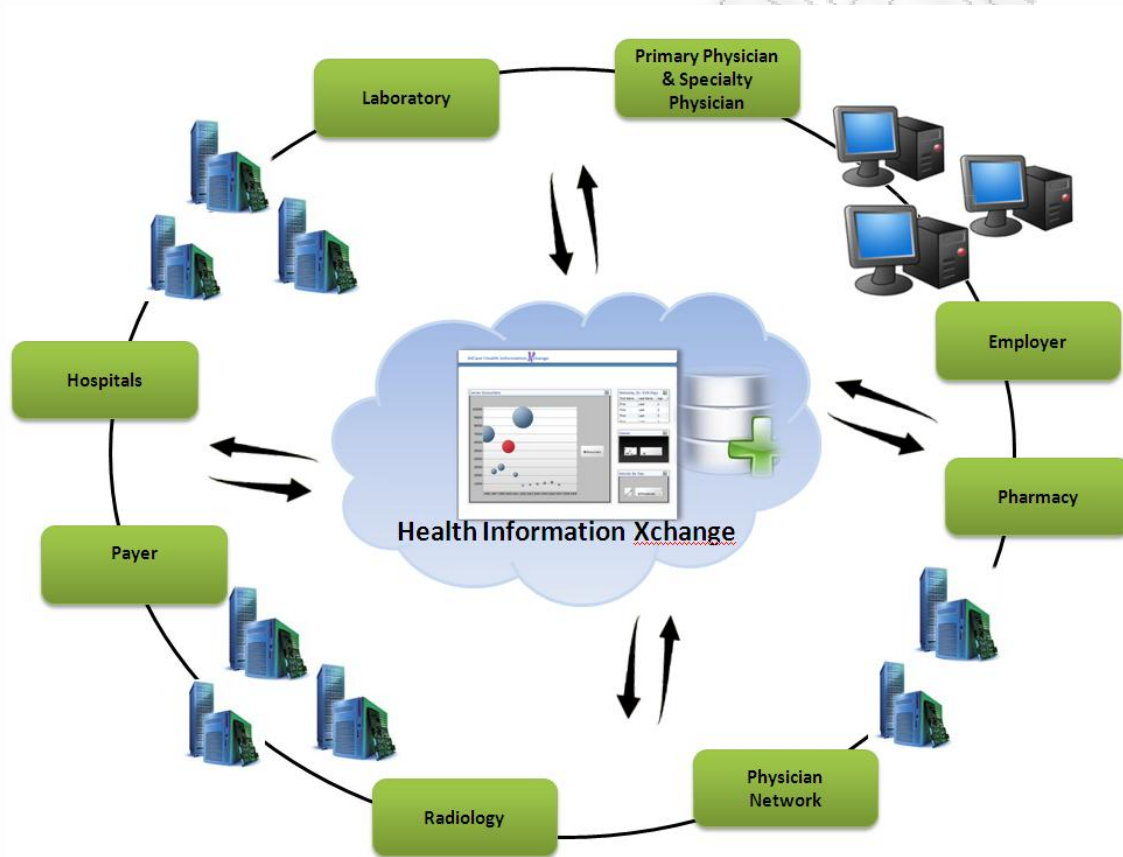
Στο Παράρτημα Β παρουσιάζεται ένα παράδειγμα χρήσης του συστήματος που αναπτύχθηκε.

### **Παράρτημα Γ – Παρουσίαση μιας ολοκληρωμένης επιχειρηματικής διαδικασίας**

Στο Παράρτημα Γ παρουσιάζεται ο κώδικας και η πλήρης διαδικασία υλοποίησης της επιχειρηματικής διαδικασίας «Εισαγωγή στο Σύστημα».

## Ενότητα Α Πληροφοριακά Συστήματα Υγείας και Υπηρεσιοστρεφής Αρχιτεκτονική

### Επισκόπηση της Θεωρίας



(SOA for Health Informatics, από Business Intelligence Technologies) <sup>[E1]</sup>

# Κεφάλαιο 1

## Πληροφοριακά Συστήματα στον τομέα της Υγείας

### Εισαγωγή

Στις μέρες μας το διαδίκτυο και ο παγκόσμιος ιστός έχουν επηρεάσει ουσιαστικά κάθε πλευρά της κοινωνικής ζωής και της επιστήμης. Ειδικότερα στην ιατρική, οι νέες αυτές τεχνολογικές δυνατότητες έχουν επιδράσει θετικά την καθημερινή της πρακτική και έχουν επιβάλει την σταδιακή ενσωμάτωση και την αξιοποίηση τους. Αυτό έχει ως αποτέλεσμα να γίνεται όλο και περισσότερο λόγος για τις υπηρεσίες και τη χρησιμότητα των πληροφοριακών και τηλεπικοινωνιακών συστημάτων στο πεδίο των ιατρικών εφαρμογών. Όροι όπως η ιατρική πληροφορική και ο ηλεκτρονικός φάκελος υγείας, η τηλείατρική και η τηλεπαρακολούθηση των ασθενών, εμφανίστηκαν πρόσφατα δια μέσου της τεχνολογικής εξέλιξης, ως κρίσιμες προκλήσεις της ανθρωπότητας, με στόχο την βελτίωση και αυτοματοποίηση των ιατρικών υπηρεσιών.

Στο κεφάλαιο αυτό θα κάνουμε μια συνοπτική αναφορά στα Πληροφοριακά Συστήματα Υγείας και στον Ηλεκτρονικό Φάκελο Υγείας. Θα συγκρίνουμε τον ηλεκτρονικό και τον χειρόγραφο φάκελο υγείας, αναφέροντας τα πλεονεκτήματα και τα μειονεκτήματά τους, θα μιλήσουμε για τα πρότυπα και τις κωδικοποιήσεις που έχουν αναπτυχθεί σχετικά με τον ηλεκτρονικό φάκελο και θα αναλύσουμε τα διάφορα ζητήματα που σχετίζονται με την εφαρμογή του.

### 1.1 Η πληροφορία στην Υγεία

Μπορούμε να διακρίνουμε τρεις βασικές έννοιες που σχετίζονται με την πληροφορία στην Υγεία:

- **Δεδομένο:** Οποιαδήποτε μοναδική παρατήρηση ή γεγονός. Ένα ιατρικό δεδομένο γενικά μπορεί να θεωρηθεί ως η τιμή μιας συγκεκριμένης παραμέτρου (π.χ. ενός ασθενούς) σε συγκεκριμένο χρόνο.
- **Πληροφορία:** Σύνολο εννοιών που περιγράφουν τα αντικείμενα και τις μεταξύ τους σχέσεις. Οργανωμένα δεδομένα ή γνώση που παρέχουν τη βάση για τη λήψη της απόφασης.
- **Γνώση:** μπορεί να θεωρηθεί ως η απόσταση πληροφοριών που έχουν συλλεχθεί, ταξινομηθεί, οργανωθεί, ολοκληρωθεί, απομονωθεί, και αποτιμηθεί. Η γνώση

βρίσκεται σε ένα επίπεδο υψηλότερο από το δεδομένο και την πληροφορία, στα οποία βασίζεται, και μπορεί να χρησιμοποιηθεί για την παραγωγή καινούργιας πληροφορίας και γνώσης. Η γνώση χρησιμοποιείται συνήθως με την έννοια της ανθρώπινης εξειδίκευσης που χρησιμοποιείται για την επίλυση προβλημάτων.

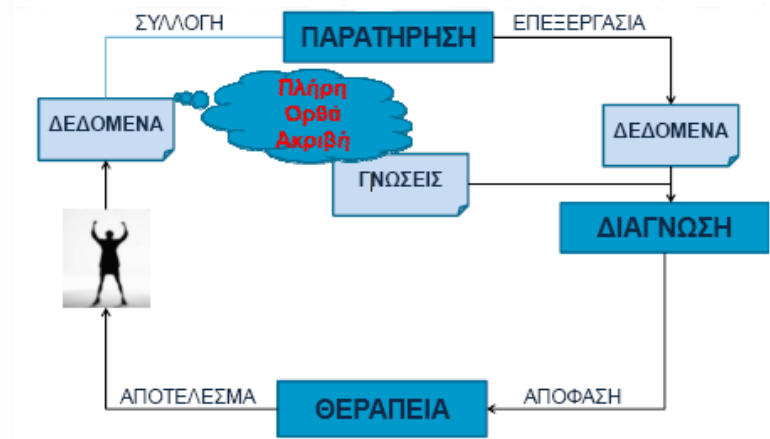
Στο χώρο της υγείας υπάρχουν διαδικασίες υψηλής πολυπλοκότητας που αναφέρονται σε ζωντανούς οργανισμούς και στις λειτουργίες τους. Οι πληροφορίες που συλλέγονται διακινούνται σε διάφορα στάδια της φροντίδας υγείας και προϋποθέτουν τη συμμετοχή διαφόρων επαγγελματιών (Ιατροί, Νοσηλευτές, Φαρμακοποιοί, Τεχνολόγοι εργαστηρίων, Διοικητικό προσωπικό, κ.λπ.). Οι υπολογιστές δεν αντικαθιστούν τις διαδικασίες σκέψης, αλλά ενισχύουν την ικανότητα επεξεργασίας δεδομένων που έχει ο εγκέφαλος, όπως για παράδειγμα τη μνήμη, την ακρίβεια, την ταχύτητα και την οπτικοποίηση σύνθετων δομών.

Μια πληροφορία μπορεί να εμφανιστεί σε μια ή περισσότερες από τις παρακάτω μορφές:

- **Κείμενο:** αναφορά, επιστολή, ιατρικός φάκελος
- **Ήχος:** Φωνή (ζωντανή λήψη και μετάδοση / ηχογραφημένο), Ιατρικός ήχος (ήχος που προέρχεται από ένα υπερηχογράφημα)
- **Εικόνα:** Εικόνα που έχουμε πάρει από ιατρικά μηχανήματα (υπερηχογράφημα, μικροσκόπιο) – εικόνα ή γράφημα σήματος, εικόνα που έχουμε πάρει από scanner, βίντεο, ψηφιακή κάμερα.
- **Βίντεο:** διάσκεψη μέσω βίντεο – Τηλεδιάσκεψη, ιατρικό Βίντεο (ζωντανή λήψη και μετάδοση / βιντεοσκοπημένο).

Αφού εξαχθεί, η πληροφορία διακινείται σε τρία διακριτά στάδια:

- **Παρατήρηση:** συλλογή δεδομένων για την κατάσταση του ασθενή, άτυπο «πρωτόκολλο» στη συλλογή:
  - Ιστορικό ασθενή,
  - Κλινική εξέταση,
  - Διεκπεραίωση εργαστηριακών εξετάσεων
- **Διάγνωση:** επεξεργασία δεδομένων παρατήρησης
- **Θεραπεία:** επεξεργασία δεδομένων διάγνωσης

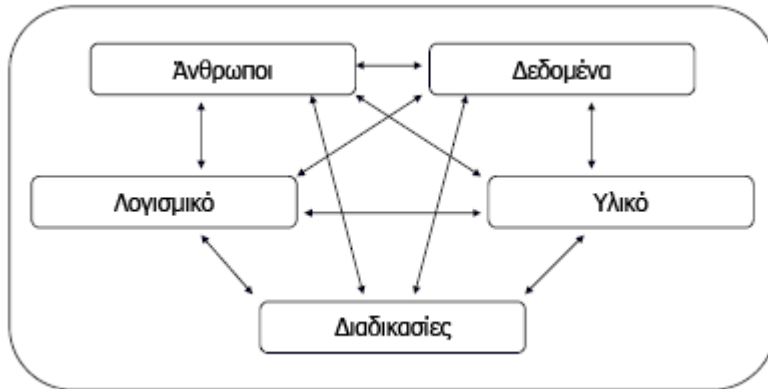


Εικόνα 1: Ροή πληροφορίας στην Υγεία [E2]

## 1.2 Πληροφοριακά Συστήματα Υγείας

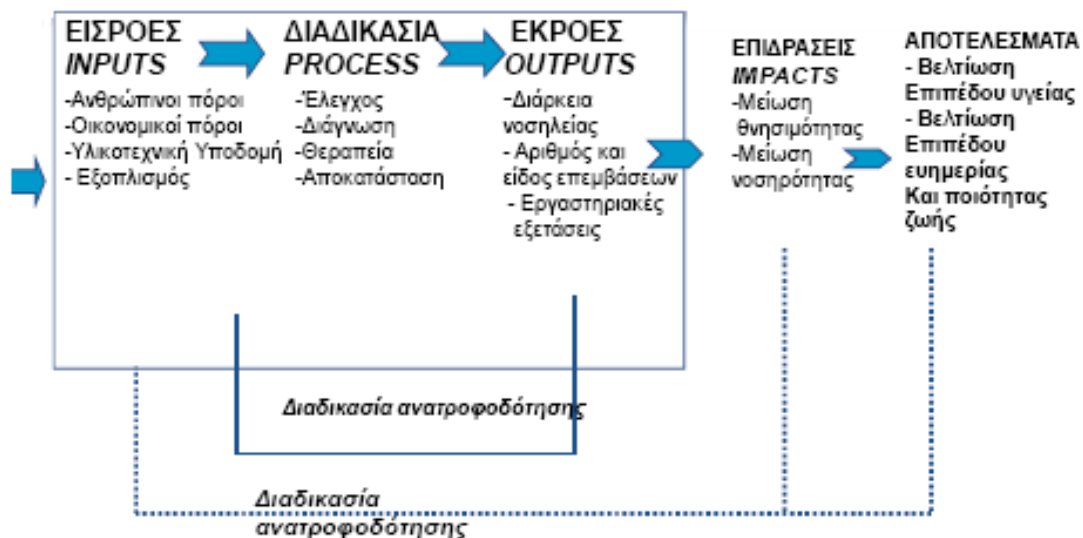
Ως «Σύστημα» ορίζεται ένα σύνολο συστατικών στοιχείων (π.χ. άνθρωποι – μηχανές – διαδικασίες) που διέπονται από συγκεκριμένους τρόπους λειτουργίας και τα οποία αλληλεπιδρούν μεταξύ τους για τη διεκπεραίωση κάποιου έργου. Κάθε σύστημα μπορεί να περιλαμβάνει έναν αριθμό υποσυστημάτων που και αυτά με τη σειρά τους μπορούν να περιλαμβάνουν άλλα μικρότερα υποσυστήματα. Όλα αυτά τα υποσυστήματα συνδέονται και συνεργάζονται μεταξύ τους, ανταλλάσσοντας δεδομένα για την επίτευξη των στόχων και των σκοπών του συστήματος, το οποίο απαρτίζουν. Τα συστατικά στοιχεία του συστήματος (άνθρωποι – μηχανές – διαδικασίες) καθορίζουν το εσωτερικό περιβάλλον του. Ότι βρίσκεται εκτός του εσωτερικού περιβάλλοντος ονομάζεται εξωτερικό περιβάλλον. Αυτά τα δύο περιβάλλοντα βρίσκονται σε συνεχή επικοινωνία ανταλλάσσοντας δεδομένα (Είσοδοι / Έξοδοι).

Πληροφοριακό σύστημα είναι το σύνολο διασυνδεδεμένων στοιχείων αποτελούμενο από ανθρώπους, δεδομένα, διαδικασίες, υλικό και λογισμικό, τα οποία αλληλεπιδρούν τόσο μεταξύ τους όσο και με το εξωτερικό περιβάλλον με σκοπό την παραγωγή και την επεξεργασία πληροφοριών για την υποστήριξη ανθρώπινων δραστηριοτήτων (π.χ. έλεγχος, λήψη απόφασης) σε έναν οργανισμό. Ένα Πληροφοριακό Σύστημα μετατρέπει τα δεδομένα (είσοδοι) σε πληροφορίες (έξοδοι).



Εικόνα 2: Συστατικά πληροφοριακού συστήματος <sup>[E3]</sup>

Τρεις είναι οι δραστηριότητες με τις οποίες ένα πληροφοριακό σύστημα παράγει τις πληροφορίες που χρειάζεται ο οργανισμός για να παίρνει αποφάσεις, να ελέγχει λειτουργίες, να αναλύει προβλήματα και να δημιουργεί νέα προϊόντα ή υπηρεσίες. Αυτές οι δραστηριότητες είναι η είσοδος, η επεξεργασία και η έξοδος. Η είσοδος συλλαμβάνει ή συλλέγει πρωτογενή δεδομένα μέσα από τον οργανισμό ή το περιβάλλον του. Η επεξεργασία μετατρέπει αυτή την πρωτογενή είσοδο σε μια πιο κατανοητή μορφή. Η έξοδος μεταφέρει τις επεξεργασμένες πληροφορίες στους ανθρώπους ή στις δραστηριότητες που θα τις χρησιμοποιήσουν. Στα πληροφοριακά συστήματα συχνή είναι η διαδικασία της ανατροφοδότησης, κατά την οποία η έξοδος του συστήματος επιστρέφει στα κατάλληλα μέλη του οργανισμού για να τα βοηθήσει να αξιολογήσουν ή να διορθώσουν τη φάση της εισόδου. Στην εικόνα 3 παρουσιάζεται το Σύστημα Υγείας ως παραγωγική διαδικασία.



Εικόνα 3: Σύστημα Υγείας ως παραγωγική διαδικασία <sup>[E4]</sup>



Κάθε οργανισμός επενδύει στην τεχνολογία και στα πληροφοριακά συστήματα επειδή αυτά προσφέρουν οικονομική αξία στον οργανισμό. Η απόφαση για τη δημιουργία ή τη συντήρηση ενός πληροφοριακού συστήματος ξεκινά από την αφετηρία ότι η απόδοση της επένδυσης αυτής θα είναι ανώτερη από άλλες επενδύσεις σε κτίρια, μηχανήματα ή άλλα στοιχεία ενεργητικού. Αυτή η ανώτερη απόδοση θα εκδηλωθεί με τη μορφή αύξησης της παραγωγικότητας, αύξηση των εσόδων (ή μείωση των δαπανών), ή ίσως ως ανώτερη μακροπρόθεσμη ανταγωνιστική τοποθέτηση του οργανισμού.

Ένα πληροφοριακό σύστημα δεν είναι κατ' ανάγκη μηχανογραφημένο. Όμως τα περισσότερα σύγχρονα πολύπλοκα συστήματα πληροφοριών δύσκολα μπορούν να υλοποιηθούν χωρίς την κάποιες μορφής υποστήριξη από υπολογιστές και τηλεπικοινωνίες. Η υποστήριξη του πληροφοριακού συστήματος από υπολογιστές και τηλεπικοινωνίες δημιουργεί προστιθέμενη αξία στον οργανισμό.

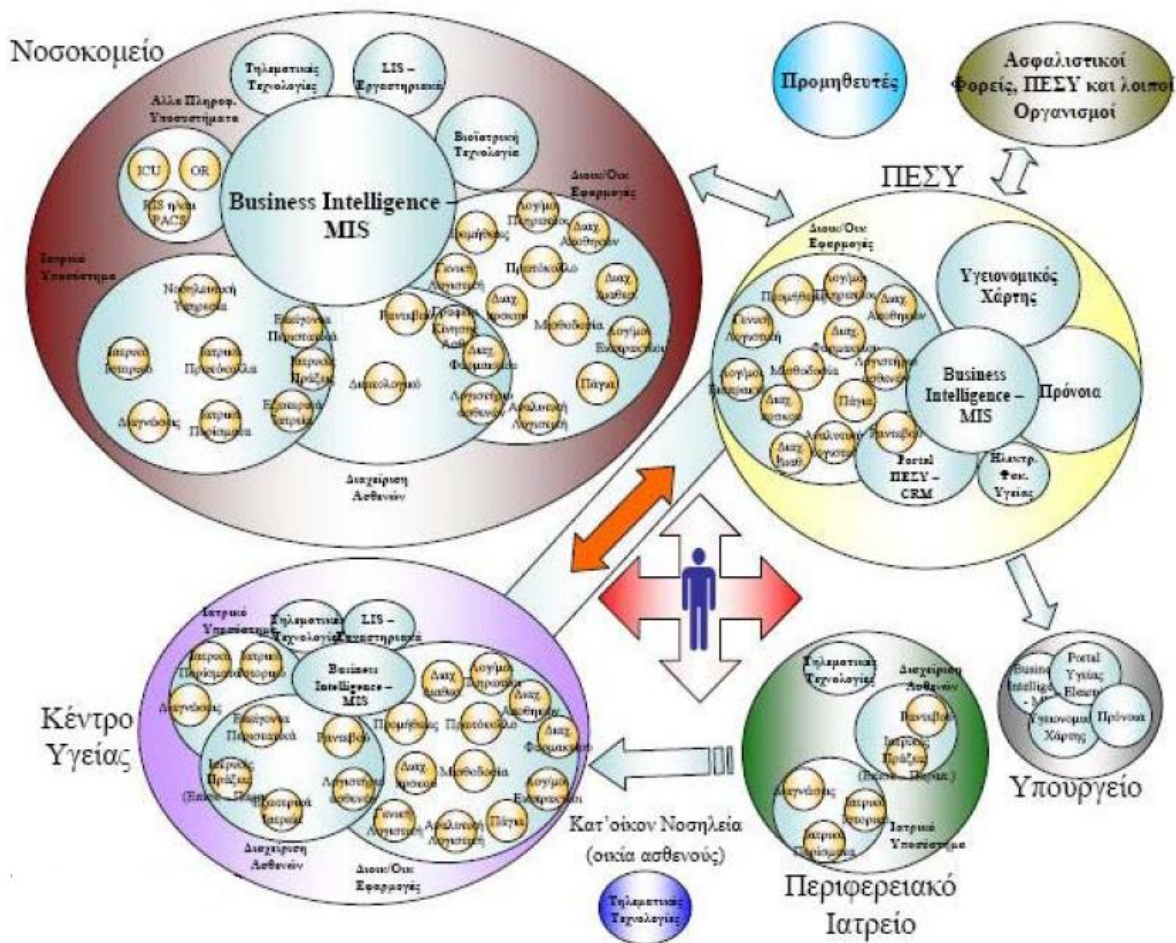
Τα πληροφοριακά συστήματα εντοπίζονται σε όλους τους χώρους παροχής φροντίδας, συμπεριλαμβανομένων των νοσοκομείων, κλινικών, κέντρων υγείας, ερευνητικών κέντρων και εκπαιδευτικών οργανισμών. Οι δυνατότητες και οι λειτουργίες τους ποικίλουν σε μεγάλο βαθμό ανάλογα με το πώς χρησιμοποιούνται και τι είδους εργασία πραγματοποιείται στον οργανισμό.

Τα Πληροφοριακά Συστήματα Υγείας ορίζονται ως: «Πληροφοριακά Συστήματα που έχουν σχεδιαστεί να υποστηρίζουν τη διοίκηση και τη λειτουργία όλων των τεχνικών και διαχειριστικών δεδομένων, για ολόκληρο το σύστημα υγείας, για ένα νοσοκομείο ή για ένα τμήμα νοσοκομείου». Ο σκοπός ενός συστήματος υγείας είναι να διασφαλίσει καλή υγεία και καλές υπηρεσίες υγείας σε όλα τα μέλη ενός πληθυσμού.

Ένα Πληροφοριακό Σύστημα Υγείας περιέχει:

- Πληροφορίες για τον ασθενή (εξετάσεις, ακτινογραφίες, κόστος)
- Πληροφορίες για τα τμήματα του νοσοκομείου (διαχειριστικές, οικονομικές, στατιστικές)
- Πληροφορίες για τους υπαλλήλους (γιατροί, νοσοκόμοι, γραμματεία, διοίκηση)
- Πληροφορίες για το φαρμακείο(συνταγές)
- Πληροφορίες για την αποθήκη

Κάθε πληροφοριακό Σύστημα Υγείας αποτελείται από αρκετά υποσυστήματα, καθένα από τα οποία αναλαμβάνει να διαχειριστεί συγκεκριμένα θέματα.



Εικόνα 4: Συνολικό διάγραμμα ενός Ολοκληρωμένου Π.Σ. Υγείας [E5]

### 1.3 Ιατρικός Φάκελος Υγείας

Η έννοια του ιατρικού φακέλου είναι τόσο παλιά όσο η ιατρική όπως την ξέρουμε στο σύγχρονο πολιτισμό. Οι ιατροί εκπαιδεύονται από τα πρώτα χρόνια των σπουδών τους στην συγκέντρωση στοιχείων που αφορούν τον ασθενή. Άλλωστε, η ίδια η φύση της ιατρικής πρακτικής βασίζεται στη σωστή συγκέντρωση στοιχείων και στην αξιολόγηση αυτών.

Ο ιατρικός φάκελος αφορά σε έγγραφα σχετικά με την κατάσταση της υγείας ενός ασθενή, όπως παραπεμπτικά εξετάσεων, αποτελέσματα εξετάσεων, καταγραφή στοιχείων νοσηλείας, ακτινογραφίες, καρδιογραφήματα κλπ. Ο ιατρικός φάκελος μπορεί να θεωρηθεί σαν ένας αποθηκευτικός χώρος όλων των πληροφοριών που αφορούν το ιστορικό υγείας του ασθενούς και επομένως αποτελεί το βασικό συστατικό για τη διαδικασία της διάγνωσης και της αποτελεσματικής θεραπείας του. Επιπρόσθετα, μπορεί να αποτελέσει βάση επιδημιολογικών ερευνών και να παρέχει πληροφορίες πέρα από την υγεία του ασθενούς, όπως για παράδειγμα οικονομικής και στατιστικής φύσεως.



Αποτελεί ένα μέσο επικοινωνίας μεταξύ του ιατρικού και παραϊατρικού προσωπικού που ασχολείται με την ιατρική περίθαλψη ενός ασθενή. Δια μέσου του ιατρικού φακέλου διάφοροι εμπλεκόμενοι μπορούν να αποκτήσουν πληροφορίες σχετικά με διαγνώσεις, θεραπευτικές αγωγές, καταγραφή πορείας ασθενούς κλπ.

### 1.3.1 Σκοπός του Ιατρικού Φακέλου

Στους χρήστες των Ιατρικών Φακέλων περιλαμβάνονται οι ασθενείς, οι επαγγελματίες της υγείας (ιατρικό, νοσηλευτικό και παραϊατρικό προσωπικό), το διοικητικό προσωπικό και άλλοι. Κάθε χρήστης του φακέλου έχει και διαφορετικές απαιτήσεις. Επομένως, ένας ιατρικός φάκελος πρέπει να υποστηρίζει διαφορετικούς σκοπούς, όπως:

- **σκοπούς ασθενών:** χρήση του φακέλου στη διάγνωση, την πρόγνωση, τη θεραπεία και την παρακολούθηση του ασθενούς, όπως επίσης στον σχεδιασμό και τον καθορισμό της φροντίδας που του παρέχεται. Επιπλέον, πρέπει να προστατεύει τα δικαιώματα του ασθενή σχετικά με τις κλινικές και προσωπικές του πληροφορίες.
- **επαγγελματικούς σκοπούς:** αυξάνει την ικανότητα για διάγνωση, πρόγνωση και παροχή ικανοποιητικής θεραπείας. Παρέχει ένα νόμιμο περιβάλλον για τη συλλογή δεδομένων και καθιστά ικανό κάθε πάροχο φροντίδας υγείας να πιστοποιεί την αυθεντικότητα των καταχωρήσεών του με υπογραφή. Χρησιμοποιείται σαν ημερολόγιο για τους παρόχους φροντίδας, στην παρακολούθηση και αξιολόγηση της ποιότητας της παρεχόμενης φροντίδας και στην καθημερινή διαχείριση του συστήματος υγείας.
- **διοικητικούς σκοπούς:** παρέχει δεδομένα για τη διοίκηση σε διαφορετικά επίπεδα του συστήματος υγείας, για να εξασφαλίζεται καλύτερη ποιότητα φροντίδας και να εξυπηρετούνται οι χρεώσεις. Τα δεδομένα αυτά λειτουργούν, επίσης, σαν βάση για την καταμέτρηση της χρήσης των πόρων ανά ασθενή και βοηθούν στον καλύτερο σχεδιασμό του συστήματος υγείας.
- **στατιστικούς και ερευνητικούς σκοπούς:** παρέχει πληροφορίες που μπορούν να χρησιμοποιηθούν για να εξαχθούν στατιστικά συμπεράσματα, ενώ βοηθά στην βελτιστοποίηση της τεχνολογίας που χρησιμοποιείται για την φροντίδα υγείας.

### 1.3.2 Περιεχόμενα του Ιατρικού Φακέλου

Ο ιατρικός φάκελος ασθενούς περιέχει μια σειρά στοιχείων και πληροφοριών που τον αφορούν, όπως δημογραφικά στοιχεία, κλινικά δεδομένα και διαγνώσεις, τιμές

εργαστηριακών εξετάσεων, ιατρονοσηλευτικές πράξεις, παραπεμπτικά, διαχειριστικά και οικονομικά στοιχεία ιατρονοσηλευτικών πράξεων και άλλα.

Αναλυτικότερα αναφέρουμε:

**Ιατρικό ιστορικό :** Το ιατρικό ιστορικό είναι μια διαχρονική καταγραφή του τι έχει συμβεί στον ασθενή από τη γέννησή του έως σήμερα. Αποτελεί ιστορική καταγραφή των παθήσεων, κύριων και δευτερευόντων ασθενειών, καθώς και των ορόσημων της ανάπτυξης. Έτσι, δίνει στον ιατρό μια αίσθηση για το τι έχει συμβεί μέχρι τώρα στον ασθενή, ενώ συχνά δίνει και ενδείξεις για την τρέχουσα κατάσταση των ασθενειών. Το ιατρικό ιστορικό χωρίζεται σε διάφορα υποσύνολα που αναλύονται παρακάτω:

- **Χειρουργικό ιστορικό:** είναι μια χρονική καταγραφή των χειρουργείων στα οποία έχει επιβληθεί ο ασθενής. Μπορεί να περιλαμβάνει τις ημερομηνίες των χειρουργείων, αναφορές για τα χειρουργεία , ή / και την λεπτομερή αφήγηση για την διαδικασία που ακολούθησε ο χειρουργός.
- **Μαιευτικό ιστορικό:** περιλαμβάνει προηγούμενες εγκυμοσύνες και τα αποτελέσματά τους. Περιλαμβάνει, επίσης, τυχόν επιπλοκές από αυτές τις κυήσεις.
- **Φάρμακα και αλλεργίες σε φάρμακα:** Ο ιατρικός φάκελος μπορεί να περιέχει μια σύνοψη της τρέχουσας φαρμακευτικής αγωγής και των προηγούμενων φαρμακευτικών αγωγών του ασθενούς, καθώς και κάθε φαρμακευτική αλλεργία.
- **Οικογενειακό ιστορικό:** απαριθμεί την κατάσταση της υγείας των στενών μελών της οικογένειας, καθώς και τις αιτίες του θανάτου τους (εφόσον είναι γνωστές). Περιλαμβάνει κοινές νόσους που έχουν εμφανιστεί στην οικογένεια ή κάποιες που έχουν βρεθεί μόνο στο ένα ή το άλλο φύλο. Μπορεί, επίσης, να περιλαμβάνει ένα γενεαλογικό διάγραμμα, το οποίο αποτελεί πολύτιμο στοιχείο για την πρόβλεψη μερικών επιπτώσεων για τον ασθενή.
- **Κοινωνικό ιστορικό:** αποτελεί ένα χρονικό της ανθρώπινης αλληλεπίδρασης. Αναφέρει τις κοινωνικές σχέσεις του ασθενούς, την σταδιοδρομία και την εκπαίδευσή του, καθώς και την θρησκευτική του εκπαίδευση. Αν και δεν συναντάται συχνά σε ιατρικούς φακέλους, είναι χρήσιμο για το γιατρό να γνωρίζει τι είδους κοινωνική υποστήριξη μπορεί να αναμένει ο ασθενής κατά τη διάρκεια μιας σοβαρής ασθένειας. Μπορεί, επίσης, να εξηγήσει τη συμπεριφορά του ασθενούς σε σχέση με μια ασθένεια ή μια απώλεια ή ακόμα και να παρέχει ενδείξεις ως προς την αιτία της ασθένειας.

- **Συνήθειες:** περιλαμβάνει διάφορες συνήθειες που έχουν αντίκτυπο στην υγεία, όπως το κάπνισμα, η χρήση αλκοόλ, η άσκηση και η διατροφή. Το τμήμα αυτό μπορεί να περιλαμβάνει και περισσότερες προσωπικές λεπτομέρειες, όπως οι σεξουαλικές συνήθειες και ο σεξουαλικός προσανατολισμός.
- **Ανοσοποιητικό ιστορικό:** αναφέρει το ιστορικό των εμβολιασμών. Στο τμήμα αυτό περιλαμβάνονται και τυχόν εξετάσεις αίματος που αποδεικνύουν ανοσία.
- **Αναπτυξιακό γράφημα και ιστορικό:** για τα παιδιά και τους εφήβους περιλαμβάνονται τα διαγράμματα που τεκμηριώνουν την ανάπτυξη, όπως αυτή συγκρίνεται με άλλα παιδιά της ίδιας ηλικίας, έτσι ώστε οι ιατροί να μπορούν να παρακολουθήσουν την ανάπτυξη του παιδιού με την πάροδο του χρόνου. Πολλές ασθένειες και κοινωνικές πιέσεις μπορούν να επηρεάσουν την ανάπτυξη και συνεπώς το αναπτυξιακό γράφημα μπορεί να παρέχει μια ένδειξη για μια υποκείμενη νόσο. Επιπλέον, η συμπεριφορά του παιδιού (όπως το πότε μίλησε, περπάτησε, κ.λπ.), συγκρίνεται με άλλα παιδιά της ίδιας ηλικίας και καταγράφεται στο ιατρικό ιστορικό, για τους ίδιους λόγους.

**Ιατρική αντιμετώπιση:** Εντός του ιατρικού φακέλου περιέχονται όλες οι ατομικές ιατρικές επισκέψεις του ασθενούς. Σε κάθε τέτοια επίσκεψη συμπληρώνεται μια μικρή περίληψη του ιατρικού ιστορικού του ασθενούς από τον ιατρό, το νοσηλευτικό προσωπικό ή τον βοηθό του ιατρού, η οποία μπορεί να έχει διαφορετική μορφή ανά περίπτωση. Για παράδειγμα, αν ο ασθενής χρειαστεί να εισαχθεί στο νοσοκομείο ή να επισκεφτεί κάποιον ειδικευμένο ιατρό, το ιστορικό συχνά λαμβάνει πλήρη μορφή, προσδιορίζοντας λεπτομερώς την πρότερη κατάσταση της υγείας του ασθενούς και την ιατροφαρμακευτική περίθαλψή του. Ενώ για τις επισκέψεις ενός οικογενειακού γιατρού, το ιστορικό περιλαμβάνει μόνο το εκάστοτε πρόβλημα του ασθενούς, την διάγνωση και την ακολουθούμενη θεραπεία. Σε γενικές γραμμές μπορούμε να πούμε ότι καταγράφονται τα παρακάτω:

- **Το πρόβλημα του ασθενούς:** είναι το πρόβλημα που έχει οδηγήσει τον ασθενή στο να επισκεφτεί τον ιατρό. Αναφέρονται οι πληροφορίες σχετικά με τη φύση και τη διάρκεια του προβλήματος.
- **Ιστορικό της παρούσας ασθένειας:** μια λεπτομερής διερεύνηση των συμπτωμάτων του ασθενούς που τον έχουν οδηγήσει στο να αναζητήσει ιατρική βοήθεια.
- **Σωματική εξέταση:** Η σωματική εξέταση οδηγεί στην καταγραφή των παρατηρήσεων στον ασθενή. Περιλαμβάνει την εξέταση διαφόρων συστημάτων οργάνων, ειδικά

αυτών που θα μπορούσαν να είναι άμεσα υπεύθυνα για τα συμπτώματα του ασθενούς.

- **Αξιολόγηση και σχέδιο θεραπείας:** Η αξιολόγηση είναι μια γραπτή σύνοψη για το ποιες είναι οι πιο πιθανές αιτίες των συμπτωμάτων του ασθενούς. Το σχέδιο θεραπείας περιγράφει την αποφασισμένη πορεία δράσης για την αντιμετώπιση των συμπτωμάτων (διάγνωση, θεραπεία, κ.λπ.).

**Ιατρικές εντολές:** Οι γραπτές εντολές από φορείς ιατρικών υπηρεσιών περιλαμβάνονται στον ιατρικό φάκελο. Σε αυτές προσδιορίζονται οι οδηγίες που έχουν δοθεί σε άλλα μέλη της ομάδας της υγειονομικής περίθαλψης από τον θεράποντα ιατρό.

**Σημειώσεις προόδου:** Όταν ένας ασθενής εισάγεται στο νοσοκομείο, ο ιατρικός φάκελος ενημερώνεται καθημερινά με κλινικές αλλαγές, νέες πληροφορίες κλπ. Αυτές οι πληροφορίες έχουν συνήθως την μορφή σημείωσης και εγγράφονται από όλα τα μέλη της ομάδας της υγειονομικής περίθαλψης (ιατροί, νοσηλευτές, θεραπευτές, διαιτολόγοι, κλινικοί φαρμακοποιοί κ.λπ.). Φυλάσσονται με χρονολογική σειρά για να τεκμηριώσουν την ακολουθία των γεγονότων που οδήγησαν στην τρέχουσα κατάσταση της υγείας του ασθενούς.

**Αποτελέσματα εξετάσεων:** Τα αποτελέσματα των εξετάσεων, όπως εξετάσεις αίματος (π.χ., γενική αίματος) ακτινολογικές εξετάσεις (π.χ. ακτίνες Χ), παθολογικές εξετάσεις (π.χ., αποτελέσματα βιοψίας), ή εξειδικευμένες εξετάσεις (π.χ., έλεγχος πνευμονικής λειτουργίας) περιλαμβάνονται στον ιατρικό φάκελο. Συχνά οι εξετάσεις αυτές συνοδεύονται από γραπτές αναφορές σχετικά με τα ευρήματα.

**Δημογραφικά στοιχεία:** Τα Δημογραφικά στοιχεία περιλαμβάνουν πληροφορίες των ασθενών που δεν είναι ιατρικής φύσης. Είναι πληροφορίες για τον εντοπισμό του ασθενούς, όπως αριθμοί μητρώου, διευθύνσεις και τηλέφωνα επικοινωνίας. Μπορεί, επίσης, να περιλαμβάνουν πληροφορίες σχετικά με τη φυλή και τη θρησκεία του ασθενούς, το χώρο εργασίας, το είδος της απασχόλησής και την ιατρική ασφάλισή του.

### 1.3.3 Προσβασιμότητα Ιατρικού Φακέλου

Οι βασικοί κανόνες που διέπουν την πρόσβαση σε ένα ιατρικό αρχείο υπαγορεύουν ότι μόνο ο ασθενής και οι πάροχοι υπηρεσιών υγειονομικής περίθαλψης, που εμπλέκονται άμεσα στην παροχή περίθαλψης, έχουν το δικαίωμα να δουν το αρχείο. Επιπλέον, ο ασθενής μπορεί να

χορηγεί άδεια σε κάθε φυσικό ή νομικό πρόσωπο να αξιολογήσει τον ιατρικό φάκελο. Οι κανόνες γίνονται πιο περίπλοκοι σε ειδικές περιπτώσεις, όπως οι παρακάτω:

- **Έλλειψη ικανότητας:** Όταν ένας ασθενής δεν έχει την ικανότητα (δεν μπορεί νομικά) να αποφασίζει για τη δική του φροντίδα, ορίζεται ένας νόμιμος κηδεμόνας (είτε μέσω των πλησιέστερων συγγενών ή από το δικαστήριο). Οι νόμιμοι κηδεμόνες μπορούν να έχουν πρόσβαση στον ιατρικό φάκελο, ώστε να παίρνουν ιατρικές αποφάσεις για λογαριασμό του ασθενούς. Στην κατηγορία αυτή ανήκουν οι άνθρωποι που είναι σε κώμα, οι ανήλικοι, τα άτομα που πάσχουν από ψυχιατρικές ασθένειες και οι τοξικομανείς.
- **Επείγουσα ιατρική κατάσταση:** Σε περίπτωση έκτακτης ιατρικής ανάγκης που αφορά μη επικοινωνιακό ασθενή, θεωρείται αυτόματα ότι υπάρχει η συναίνεση για την πρόσβαση στον ιατρικό φάκελο, εκτός αν υπάρχει πρότερη έγγραφη τεκμηρίωση.
- **Έρευνα, έλεγχος και αξιολόγηση:** Τα άτομα που εμπλέκονται στην ιατρική έρευνα, σε δημοσιονομικούς και διαχειριστικούς ελέγχους ή στην αξιολόγηση του προγράμματος περίθαλψης, έχουν πρόσβαση στα ιατρικά αρχεία. Ωστόσο, δεν τους επιτρέπεται η πρόσβαση σε οποιοσδήποτε πληροφορίες αναγνώρισης.
- **Κίνδυνος θανάτου ή βλάβης:** Οι πληροφορίες που περιέχονται στους ιατρικούς φακέλους μπορούν να δοθούν στις αρχές χωρίς άδεια, σε περίπτωση που η άρνηση αυτή θα είχε ως αποτέλεσμα το θάνατο ή τη βλάβη, είτε του ασθενούς είτε τρίτων.

#### 1.4 Χειρόγραφος Φάκελος Υγείας

Δεν υπάρχει αμφιβολία ότι για δεκαετίες η λύση του χειρόγραφου φακέλου υγείας έχει χρησιμοποιηθεί με αρκετή επιτυχία. Η μέθοδος που χρησιμοποιεί ως βάση το χαρτί, υλικό με το οποίο είναι εξοικειωμένοι οι περισσότεροι άνθρωποι και σίγουρα αυτοί που ενασχολούνται με τα ιατρικά θέματα, παρουσιάζει ως βασικά πλεονεκτήματα αφενός την πιο πάνω εξοικείωση και αφετέρου το αυτόνομο της μεθόδου.

Οι χειρόγραφοι φάκελοι ασθενών διαφέρουν από νοσοκομείο σε νοσοκομείο και σε μερικά νοσοκομεία από τμήμα σε τμήμα. Επίσης, διαφέρουν για τις διάφορες υπηρεσίες που εμπλέκονται στην φροντίδα υγείας. Η επεξεργασία και η πρόσβαση σε χειρόγραφους φακέλους ασθενών είναι αρκετά δύσκολη και επίπονη εργασία. Ο χειρόγραφος Φάκελος Υγείας έχει τόσο πλεονεκτήματα όσο και μειονεκτήματα:

#### 1.4.1 Πλεονεκτήματα χειρόγραφου φακέλου

- Είναι οικείοι στους χρήστες, οι οποίοι δεν χρειάζεται να αποκτήσουν καινούργιες δεξιότητες ή συμπεριφορά για να τους χρησιμοποιήσουν.
- Είναι φορητοί και μπορούν να μεταφερθούν στο σημείο παροχής φροντίδας.
- Εφόσον βρεθούν, δεν αντιμετωπίζουν χρόνο διακοπής λειτουργίας όπως τα συστήματα υπολογιστών.
- Επιτρέπουν ευελιξία (έλλειψη πρότυπου λεξικού και κωδικοποίησης) στην καταγραφή των δεδομένων και είναι ικανοί να καταγράψουν εύκολα «ασταθή» (δηλ. υποκειμενικά) δεδομένα.
- Μπορούν να ξεφυλλιστούν και να μελετηθούν (εάν δεν είναι πολύ μεγάλοι). Αυτό το χαρακτηριστικό επιτρέπει στους χρήστες να οργανώνουν τα δεδομένα με διάφορους τρόπους και να ψάχνουν για τύπους ή τάσεις που δεν είναι σαφώς δηλωμένα.

#### 1.4.2 Μειονεκτήματα χειρόγραφου φακέλου

- Προβλήματα με το περιεχόμενό τους (ελλιπή, δυσανάγνωστα, υπερβολικά, πλεονάζοντα ή ανακριβή δεδομένα)
- Προβλήματα με τη μορφή (τα δεδομένα είναι διάσπαρτα μέσα στο φάκελο και δεν είναι ταξινομημένα ώστε να εξυπηρετούν την ικανότητα άντλησης πληροφοριών, αλλά συνήθως είναι οργανωμένα ανάλογα με τις πηγές και τη χρονολογία των δεδομένων)
- Προβλήματα με την πρόσβαση, τη διαθεσιμότητα και την ανάκτηση. Είναι διαθέσιμος μόνο σε ένα μέρος την ίδια στιγμή.
- Προβλήματα με διασυνδέσεις και την ολοκλήρωση/ενοποίηση (ασυνέχεια της παροχής φροντίδας ανάμεσα στους παρόχους)
- Προβλήματα με τους εξωνοσοκομειακούς φακέλους (είναι πολλοί σε αριθμό, διασκορπισμένοι σε ιδιωτικά ιατρεία και συνήθως παρουσιάζουν μεγαλύτερη διακύμανση στην ποιότητα)
- Μπορούν να καταστραφούν εύκολα, ενώ είναι αρκετά επίπονη και χρονοβόρα η διαδικασία δημιουργίας αντιγράφων ασφαλείας.
- Δεν είναι δυνατή η άμεση χρήση του περιεχομένου πληθώρας ιατρικών φακέλων για έρευνα, μιας και κάτι τέτοιο απαιτεί όχι μόνο ιδιαίτερη προσπάθεια (ανάγνωση των φακέλων, κωδικοποιημένη καταγραφή των στοιχείων και άλλα), αλλά κυρίως χρόνο.

## 1.5 Ηλεκτρονικός Φάκελος Υγείας

### 1.5.1 Ορισμός

Ο Ηλεκτρονικός Ιατρικός Φάκελος είναι ένας ψηφιακά αποθηκευμένος φάκελος φροντίδας υγείας (ή υποσύνολο αυτού) για όλη τη διάρκεια ζωής του ατόμου, με στόχο την υποστήριξη της συνέχειας της φροντίδας υγείας (ποιότητα, πρόσβαση, αποδοτικότητα), την εκπαίδευση και την έρευνα. Ο Ηλεκτρονικός Ιατρικός Φάκελος περιλαμβάνει στοιχεία του ασθενούς σύμφωνα με τις επισκέψεις του σε παρόχους υγείας και τα επεισόδια που έχουν δημιουργηθεί κατά τη διάρκεια αυτών. Περιλαμβάνει, επίσης, όλα τα δημογραφικά και ιατρικά στοιχεία του σύμφωνα με την ιστορικότητα αυτών, στοιχεία για την πιθανή ύπαρξη αλλεργιών, λήψης φαρμάκων κτλ. Παράλληλα εμπεριέχει σημειώσεις για την παρούσα κατάσταση του ασθενούς, τις κλινικές σημειώσεις και τη διάγνωση που προκύπτει από τον εκάστοτε επαγγελματία υγείας.

Ένας Ηλεκτρονικός Ιατρικός διευκολύνει:

- την πρόσβαση των στοιχείων των ασθενών από το κλινικό προσωπικό σε οποιαδήποτε δεδομένη τοποθεσία
- την διεξαγωγή αυτόματων ελέγχων για τα φάρμακα και τις αλλεργικές αλληλεπιδράσεις
- την τήρηση κλινικών σημειώσεων
- την συνταγογράφηση
- τον προγραμματισμό των ραντεβού
- την αποστολή και προβολή εξετάσεων προς και από τα εργαστήρια

Παρόλο που σε ένα σύστημα Ηλεκτρονικού Ιατρικού Φακέλου υπάρχει η πιθανότητα παραβίασης των ιδιωτικών δεδομένων, εάν ακολουθούνται αποτελεσματικά οι πολιτικές ασφαλείας, τα συστήματα αυτά είναι τόσο ασφαλή όσο, για παράδειγμα, τα ηλεκτρονικά αρχεία των τραπεζών.

### 1.5.2 Είδη του Ηλεκτρονικού Φακέλου

Σύμφωνα με το Ινστιτούτο Ιατρικών Φακέλων (Medical Records Institute) , μπορούν να διακριθούν πέντε επίπεδα ενός Ηλεκτρονικού Φακέλου Φροντίδας της Υγείας (Electronic HealthCare Record - EHCR):

- Ο **Αυτοματοποιημένος Ιατρικός Φάκελος** (Automated Medical Record) είναι ένα χειρόγραφο αρχείο, στο οποίο περιλαμβάνονται και κάποια έγγραφα που έχουν παραχθεί από υπολογιστή.
- Ο **Μηχανογραφημένος Ιατρικός Φάκελος** (Computerized Medical Record - CMR) καθιστά τα στοιχεία του Αυτοματοποιημένου Ιατρικού Φακέλου ηλεκτρονικά διαθέσιμα.
- Ο **Ηλεκτρονικός Ιατρικός Φάκελος** (Electronic Medical Record - EMR) αναδομεί και βελτιστοποιεί τα έγγραφα των προηγούμενων επιπέδων, διασφαλίζοντας τη διαλειτουργικότητα όλων των συστημάτων τεκμηρίωσης.
- Ο **Ηλεκτρονικός Φάκελος Ασθενούς** (Electronic Patient Record - EPR) είναι ένα βασισμένος στον ασθενή ιατρικός φάκελος, που περιέχει πληροφορίες από πολλούς οργανισμούς.
- Ο **Ηλεκτρονικός Φάκελος Υγείας** (Electronic Health Record - EHR) προσθέτει στον Ηλεκτρονικό Φάκελο Ασθενούς πληροφορίες σχετικά με την γενικότερη κατάσταση της υγείας του ασθενούς, οι οποίες δεν σχετίζονται κατ' ανάγκη με μια ασθένεια.

### 1.5.3 Σύγκριση με τον χειρόγραφο φάκελο

Στον παρακάτω πίνακα επιχειρούμε να κάνουμε μια σύγκριση ανάμεσα στον χειρόγραφο και τον ηλεκτρονικό φάκελο του ασθενούς.

Χειρόγραφος Φάκελος	Ηλεκτρονικός Φάκελος
Δύσκολη πρόσβαση (ένας χρήστης, μία τοποθεσία συχνά στο ίδιο μέρος), μέρη του φακέλου είναι γεωγραφικά κατανεμημένα	Αποκεντρωμένη, ταυτόχρονη πρόσβαση κάθε στιγμή.
Παθητικός: ανίκανος να προκαλέσει συγκεκριμένες ενέργειες	Ενεργός: μπορεί να προκαλέσει ενέργειες σύμφωνα με τα δεδομένα.
Μη τυποποιημένος χειρισμός της πληροφορίας	Τυποποιημένη αποθήκευση πληροφοριών, ευανάγνωστα δεδομένα, επιτρέπει επικοινωνία μεταξύ συστημάτων και μπορεί να οδηγήσει σε βελτίωση και διασφάλισης της ποιότητας
“Χειροκίνητες” συνδέσεις	“Αυξημένη” διασύνδεση με εξωτερικούς παρόχους φροντίδας υγείας
Περισσότεροι κίνδυνοι να είναι ελλιπής - ατελής (π.χ. λόγω δανεισμού)	Περισσότερο πλήρης με επιπρόσθετους μηχανισμούς ελέγχου κατά την εισαγωγή δεδομένων.



Χειρόγραφος Φάκελος	Ηλεκτρονικός Φάκελος
Μη εύκαμπτη αναπαράσταση δεδομένων (μία αναπαράσταση μόνο)	Εύκαμπτη αναπαράσταση δεδομένων ακολουθώντας διάφορες προοπτικές για την εκπλήρωση των εκάστοτε αναγκών
Η εξερεύνησή τους για τη διεξαγωγή κλινικών και οικονομικών μελετών είναι χρονοβόρα	Άριστη βάση για τη διεξαγωγή κλινικών και οικονομικών μελετών
Χαμηλός κίνδυνος τεχνικής αποτυχίας	Κίνδυνος τεχνικής αποτυχίας και μη διαθεσιμότητα ανάλογα με το υλικό/λογισμικό
Μικρότερο κόστος (μόνο κόστος καταχώρησης και όχι συνολικό κόστος)	Υψηλότερο κόστος εγκατάστασης, εκπαίδευσης και διαχείρισης συστήματος
Χειρόγραφη εισαγωγή δεδομένων	Πιθανή αντίσταση και φόβος εισαγωγής δεδομένων στον υπολογιστή
Ασφάλεια και εμπιστευτικότητα επιτυγχάνονται εύκολα	Ασφάλεια και εμπιστευτικότητα επιτυγχάνονται πιο δύσκολα

## 1.6 Πρότυπα και κωδικοποιήσεις

Η δημιουργία ενός πληροφοριακού συστήματος που θα ανταποκρίνεται πλήρως στις ανάγκες του χώρου της υγείας απαιτεί την αρμονική συνύπαρξη και συνεργασία μεθόδων παροχής υπηρεσιών του παρελθόντος και τεχνολογιών του σήμερα. Τα παραπάνω προϋποθέτουν την θέσπιση και εφαρμογή κωδικών και προτύπων, τα οποία θα καθορίζουν τον τρόπο συλλογής και παρουσίασης των δεδομένων από διαφορετικά πληροφοριακά συστήματα. Τα πρότυπα αυτά αναφέρονται στον τρόπο με τον οποίο πρέπει να γίνεται η μεταφορά πληροφορίας από ένα σύστημα σε κάποιο άλλο, αλλά και στο είδος της πληροφορίας που μπορεί να μεταφέρεται.

Υπάρχει τεράστια έρευνα σε παγκόσμιο επίπεδο στο θέμα της ανάπτυξης προτύπων στο χώρο της ιατρικής πληροφορικής. Αυτός είναι και ο λόγος που υφίστανται πληθώρα προτύπων σε όλα τα επίπεδα. Βασικές περιοχές στις οποίες εστιάζονται οι προσπάθειες, είναι οι ακόλουθες:

### 1.6.1 Πρότυπα επικοινωνίας

Παρακάτω παρουσιάζονται τα σημαντικότερα διεθνή standards που έχουν αναπτυχθεί για την επικοινωνία μεταξύ Πληροφοριακών Συστημάτων ή/και ιατρικών μηχανημάτων. Τα πρότυπα

αυτής της κατηγορίας είναι από τα πλέον ώριμα και εξελίσσονται συνεχώς για να επιτευχθεί η καλύτερη επικοινωνία μεταξύ των συστημάτων.

- **HL7:** Το HL7 (Health Level Seven) είναι σήμερα το πλέον ευρέως χρησιμοποιημένο πρότυπο ανταλλαγής πληροφοριών μέσω ηλεκτρονικών μηνυμάτων στο χώρο της υγείας. Σχεδόν όλα τα ευφυή διαγνωστικά μηχανήματα (ιατροτεχνολογικός εξοπλισμός) υποστηρίζουν το HL7 και σχεδόν όλα τα ιατρικά πληροφοριακά συστήματα υψηλού επιπέδου είναι σε θέση να στείλουν και να λάβουν τα κατάλληλα HL7 μηνύματα, χρησιμοποιώντας τους κανόνες ανταλλαγής μηνυμάτων του πρωτοκόλλου του HL7.

Το HL7 είναι ένα σύνολο από ανοιχτά πρότυπα, που επιτρέπει σε ετερογενή ιατρικά πληροφοριακά συστήματα να επικοινωνούν μεταξύ τους. Ο οργανισμός έχει αναπτύξει ένα ευρύ σύνολο τυποποιήσεων για την μεταφορά δεδομένων σχετικά με την καταχώρηση του ασθενούς, την εισαγωγή, την μεταφορά και το εξιτήριο ασθενούς, την μεταφορά εξετάσεων, εντολών και άλλα.

- **CEN/TC 251:** Η Ευρωπαϊκή Επιτροπή Τυποποίησης (European Standards Committee – CEN) έχει δημοσιεύσει ένα PreStandard για την αρχιτεκτονική ηλεκτρονικού ιατρικού φακέλου (Electronic HealthCare Record) με την ονομασία ENV 13606. Αυτό ορίζει γενικές δομές πληροφορίας και χαρακτηριστικά κοινά σε κάθε ηλεκτρονικό ιατρικό φάκελο, δηλαδή ένα λογικό μοντέλο, χωρίς να καθορίζει ακριβώς τι ιατρική πληροφορία θα περιέχει ή πως θα υλοποιηθεί. Το ENV 13606 είναι το μόνο πρότυπο που δεν έχει υλοποιηθεί σε κάποιο σύστημα, αποτελεί όμως αναφορά για την υλοποίηση άλλων προτύπων, όπως το HL7.
- **ISO/TC 215:** Ο οργανισμός τυποποίησης ISO έχει ιδρύσει την Τεχνική Επιτροπή 215 (TC 215) με στόχο την προτυποποίηση στον τομέα της ιατρικής πληροφορικής (Health Informatics). Ειδικότερα, το WG1 στοχεύει στην ανάπτυξη προτύπων για τη διαχείριση της ιατρικής πληροφορίας και των ιατρικών διαδικασιών. Η επιδίωξη είναι ένα πρότυπο ιατρικού φακέλου, όπου η κατάλληλη πληροφορία θα είναι διαθέσιμη όταν και όπου απαιτείται η υποστήριξη αποφάσεων.
- **DICOM-3** (Digital Imaging and Communications). Αποτελεί το βασικό πρότυπο για μεταφορά διαγνωστικών ιατρικών εικόνων και υποστηρίζεται από όλους τους κατασκευαστές συστημάτων PACS (Picture Archiving and Communication Systems).

### 1.6.2 Τυποποίηση κλινικών δεδομένων

Τα πρότυπα αυτά αφορούν την κωδικοποιημένη αναπαράσταση ιατρικών δεδομένων, όπως διαγνώσεις, κλινικές εξετάσεις κλπ. Οι κυριότερες κωδικοποιήσεις προέρχονται από την Αμερική και την Αγγλία και είναι κοινή πρακτική να μεταφράζονται για τις υπόλοιπες χώρες. Σήμερα υπάρχουν παραπάνω από 150 γνωστά συστήματα κωδικοποίησης αλλά αυτά με την ευρύτερη αποδοχή είναι:

- **ICD (*International Classification of Diseases*):** Η κωδικοποίηση νόσων – διαγνώσεων κατά ICD αναπτύχθηκε και συντηρείται από τον Παγκόσμιο Οργανισμό Υγείας (WHO) και είναι αποδεκτή παγκοσμίως. Η WHO έχει αναπτύξει το ICD-10 και το HCFA έχει δημιουργήσει μια εθελοντική ομάδα για να βοηθήσει στην ανάπτυξη του συστήματος κωδικοποίησης διαδικασιών (Procedure Coding System ICD-10-PCS). Χρησιμοποιείται κυρίως για ασφαλιστικούς λόγους (πολλές ασφαλιστικές εταιρείες, απαιτούν τη χρήση του ICD-9-CM(Clinical Modification) για να δώσουν τις όποιες αποζημιώσεις). Στη χώρα μας έχει γίνει μετάφραση του ICD-10 στα Ελληνικά μέσα από σχετικό έργο του Υπουργείου Υγείας (Γ' Κοινοτικό Πλαίσιο Στήριξης).
- **RCC (*Read Clinical Classification ή Read Codes*):** Οι Read Codes αναπτύχθηκαν στη Μεγάλη Βρετανία και είναι ουσιαστικά μια εκτενής λίστα όρων που χρησιμοποιούνται στο χώρο της υγείας και είναι δυνατόν να ενταχθούν σε φάκελο ασθενούς. Καλύπτουν θέματα όπως επαγγέλματα, σημάδια και συμπτώματα, εξετάσεις, διαγνώσεις, θεραπευτικές αγωγές και θεραπείες, φάρμακα και άλλα. Αυτό μπορεί να κάνει δυνατή την αποθήκευση σε έναν υπολογιστή σχεδόν οποιουδήποτε σχετικού με την υγεία κειμένου, από την συνοπτική περιγραφή κάποιου επεισοδίου μέχρι έναν πλήρη ηλεκτρονικό φάκελο ασθενή.
- **SNOMED (*Systematized Nomenclature of Human & Veterinary Medicine*):** Αποτελεί κωδικοποίηση για διάφορους κλινικούς σκοπούς. Δημιουργήθηκε συντηρείται από το College of American Pathologists (CAP) και είναι ευρέως αποδεκτή για την περιγραφή αποτελεσμάτων παθολογικών εξετάσεων. Έχει πολύ-αξονική δομή κωδικοποίησης (έντεκα πεδία), η οποία επιτρέπει μεγαλύτερη σαφήνεια σε σχέση με την κωδικοποίηση ICD και έχει σημαντική αξία όσον αφορά το κλινικό κομμάτι. Το SNOMED αποτελεί ένα από τους πρώτους υποψήφιους για να γίνει το πρότυπο για τον ιατρικό φάκελο βασισμένο σε υπολογιστή.

- **DRG's (Diagnosis Related Groups):** Ένα DRG είναι η κατηγοριοποίηση μιας επίσκεψης σε κάποιο νοσοκομείο από την άποψη του ποιο ήταν το πρόβλημα και πως αντιμετωπίστηκε σε κάποιον ασθενή. Συνήθως το DRG προσδιορίζει το ποσό που θα κοστίσει μια επίσκεψη (με την ευρεία έννοια) ανεξάρτητα από τις χρεώσεις που έχουν προκύψει και χρησιμοποιείται κυρίως για τον έλεγχο της απόδοσης των δαπανών που έχει κάνει ο ασφαλισμένος. Οι κωδικοί DRG έχουν αξία κυρίως για να διευκολύνουν τέτοιου είδους οικονομικές αναλύσεις και όχι για κλινικές έρευνες ή θεραπευτική αγωγή στους ασθενείς, καθώς δεν έχουν την απαραίτητη κλινική σαφήνεια.

### 1.6.3 Πρότυπα Ταυτοποίησης

Αυτά τα πρότυπα καλύπτουν την ανάγκη για ύπαρξη κάποιων κωδικών που να προσδιορίζουν με μοναδικό τρόπο κάθε ασθενή, παροχέα, οργανισμό ή προϊόν.

- **Αναγνωριστικά Ασθενών:** Όταν ένας ασθενής προσέρχεται σε κάποιον φορέα υγείας για να του παρασχεθεί φροντίδα, υπάρχει η ανάγκη για αναζήτηση του φακέλου του σχετικά με οποιαδήποτε φροντίδα είχε λάβει στο παρελθόν εντός του ίδιου οργανισμού. Επίσης, είναι επιθυμητό να μπορεί να γίνει συνδυασμός των τμημάτων του ιατρικού του φακέλου που βρίσκονται σε άλλους φορείς. Ο αριθμός κοινωνικής ασφάλισης (Social Security Number-SSN) θεωρείται σαν ένας αριθμός ο οποίος προσδιορίζει μοναδικά κάθε ασθενή. Εντούτοις, οι κριτικές επιμένουν ότι δεν μπορεί να θεωρηθεί σαν τέτοιος, γιατί δεν έχει ο καθένας έναν SSN, μπορεί να χρησιμοποιούν τον ίδιο αριθμό SSN διάφοροι πολίτες ή τέλος, λόγω της ευρείας χρήσης που έχει αυτός ο αριθμός, υπάρχουν μεγάλοι κίνδυνοι σε ότι έχει να κάνει με το ιατρικό απόρρητο και την ασφάλεια. Στην χώρα μας έχει ξεκινήσει μια προσπάθεια θεσμοθέτησης ενός «εθνικού κωδικού ασθενούς» που επικεντρώνεται στο ασφαλιστικό σύστημα, με την δημιουργία του Αριθμού Μητρώου Κοινωνικής Ασφάλισης (ΑΜΚΑ).
- **Αναγνωριστικά Παρόχων Υγείας, προϊόντων και προμηθευτών:** Αντίστοιχα με τα αναγνωριστικά ασθενών, υπάρχει η ανάγκη ταυτοποίησης των εμπλεκόμενων στην παροχή υπηρεσιών υγείας, καθώς και για τα βασικά προϊόντα και υλικά που χρησιμοποιούνται. Έτσι έχουν δημιουργηθεί μια σειρά κωδικών για την αναγνώριση του ιατρικού και νοσηλευτικού προσωπικού (HCFA - Health Care Financing Administration, UPIN - Universal Physician Identifier Number), κωδικοί για τους φορείς υγείας, όπως νοσοκομεία, κλινικές, εργαστήρια κ.α., (HIN - Health Industry Number,

HIBCC - Health Industry Business Communications Council)), κωδικοποίηση υγειονομικών υλικών, βιοϊατρικού εξοπλισμού, φαρμάκων και άλλα. Στην κατεύθυνση αυτή έχουν ήδη δημιουργηθεί στη χώρα μας οι παρακάτω κωδικοποιήσεις:

- Κωδικοποίηση φαρμάκων από τον ΕΟΦ
- Κωδικοποίηση αντιδραστηρίων και υγειονομικών Υλικών από το Υπουργείο Υγείας
- Κωδικοποίηση βιοϊατρικού εξοπλισμού κατά UMDNS από το Ινστιτούτο Βιοϊατρικής Τεχνολογίας.

#### 1.6.4 Πρότυπα εξασφάλισης του απορρήτου των δεδομένων

Η ανάπτυξη συστημάτων ιατρικού φακέλου βασισμένων σε υπολογιστή καθώς και δικτύων υπολογιστών ανάμεσα σε οργανισμούς υγείας, δημιούργησαν την ανάγκη για ανάπτυξη προτύπων και μεθόδων που θα εξασφαλίσουν το ιατρικό απόρρητο και την ασφάλεια των δεδομένων. Όσον αφορά το θέμα της ασφάλειας προσωπικών δεδομένων υγείας, υπάρχουν μια σειρά από οδηγίες και προτάσεις από διάφορους φορείς:

- Η σύσταση N. R(97) της Ευρωπαϊκής Ένωσης για την προστασία ιατρικών δεδομένων.
- Η τεχνική αναφορά του CEN/TC 251 περί πλαισίου για την ασφαλή προστασία στις επικοινωνίες της υγείας.
- Η American Society for Testing and Materials-ASTM μέσω των υποεπιτροπών του απευθύνεται σε ανάλογα ζητήματα. Συγκεκριμένα:
- Η υποεπιτροπή E31.12 του ASTM για τον Ιατρικό Φάκελο Ασθενή βασισμένο σε Υπολογιστή (Computer-based Patient Records) έχει αναπτύξει το "Guidelines for Minimal Data Security Measures for the Protection of Computer-based Patient Records."
- Η υποεπιτροπή E31.17 του ASTM εργάζεται πάνω σε πρότυπα για την πρόσβαση και το απόρρητο των Ιατρικών Φακέλων.
- Η υποεπιτροπή E31.20 του ASTM έχει αναπτύξει τις προδιαγραφές που πρέπει να έχουν τα πρότυπα για τον έλεγχο της πρόσβασης στην ιατρική πληροφορία.

### 1.7 Πλεονεκτήματα της εφαρμογής του Ηλεκτρονικού Φακέλου

Ο πρωταρχικός σκοπός του ηλεκτρονικού φακέλου είναι να υποστηρίξει τη συνέχεια, την αποτελεσματικότητα και την ποιότητα της ολοκληρωμένης ιατρικής φροντίδας. Επιπλέον,



είναι μια πηγή κλινικών πληροφοριών για την ανθρώπινη φροντίδα, για στατιστικές αναλύσεις και συμπεράσματα, όπως και για ερευνητικούς σκοπούς.

Ο Ηλεκτρονικός Φάκελος παρέχει πληθώρα πλεονεκτημάτων τόσο για τους ασθενείς, όσο και για άλλους φορείς. Βελτιώνει την ακρίβεια και την ποιότητα των δεδομένων που καταγράφονται, δίνοντας λύση στα προβλήματα ασάφειας και δυσνόητης πληροφορίας των χειρόγραφων φακέλων. Βελτιώνει τις υπηρεσίες υγείας και την αποτελεσματικότητά τους, με το να επιτρέπεται ο διαμοιρασμός της ιατρικής πληροφορίας μεταξύ των επαγγελματιών του κλάδου υγείας και η άμεση πρόσβαση στα δεδομένα από όσους εμπλέκονται στην διαδικασία της περίθαλψης, ανεξάρτητα από την γεωγραφική περιοχή.

Η χρήση του Ηλεκτρονικού Φακέλου μειώνει τα λάθη στις ιατρικές αναφορές. Είναι γεγονός ότι τα χειρόγραφα αρχεία ευνοούν τα ανθρώπινα λάθη, λόγω του δυσανάγνωστου χαρακτήρα και των διαφορετικών ορολογιών που χρησιμοποιούνται. Με την χρήση της τυποποίησης στα ηλεκτρονικά αρχεία η μείωση των λαθών είναι εφικτή. Η εισαγωγή του ηλεκτρονικού φακέλου ασθενή μπορεί να συμβάλλει σημαντικά στη βελτίωση των κλινικών διαδικασιών και στην αποδοτικότερη ροή τους. Ταυτόχρονα παρέχει καλύτερη και ποιοτικότερη συλλογή και διαχείριση της κλινικής πληροφορίας. Αποτελεί εργαλείο που διευκολύνει την κλινική απόφαση και κατά συνέπεια βελτιώνει την ποιότητα των υπηρεσιών που δέχεται ο ασθενής.

Οι ασθενείς μπορούν να έχουν πρόσβαση στα δεδομένα υγείας τους οποιαδήποτε στιγμή και να πραγματοποιήσουν δραστηριότητες, όπως να ανανεώσουν κάποιες συνταγές, να ορίσουν ένα ραντεβού κλπ. Λόγω του ότι τα δεδομένα τους μπορούν να προσπελαστούν από οποιοδήποτε σημείο μπορούν να λάβουν μια δεύτερη γνώμη από κάποιον ιατρό που βρίσκεται σε διαφορετικό νοσοκομείο, αποφεύγοντας τις χρονοβόρες διαδικασίες.

Τα ηλεκτρονικά αρχεία μπορούν να αποθηκευτούν ουσιαστικά για πάντα και μπορούν να διατηρηθούν για πολύ μεγαλύτερο χρονικό διάστημα από τους χειρόγραφους φακέλους. Διατηρούν πληροφορίες υγείας που οι ασθενείς τείνουν να ξεχνούν με το χρόνο, όπως εμβολιασμοί, προηγούμενες ασθένειες, φάρμακα κ.α. Περιορίζουν την επαναλαμβανόμενη εκτέλεση ιατρικών εξετάσεων, μειώνοντας έτσι τις δαπάνες τόσο στους ασθενείς, όσο και στο σύστημα υγείας.

## 1.8 Ζητήματα που σχετίζονται με την εφαρμογή του Ηλεκτρονικού Φακέλου

Εφόσον έχει αποφασιστεί η ανάπτυξη και εφαρμογή ενός συστήματος Ηλεκτρονικού Φακέλου, το επόμενο βήμα που πρέπει να πραγματοποιηθεί είναι ο προσδιορισμός και η επίλυση των διαφόρων ζητημάτων και προκλήσεων που προκύπτουν και τα οποία μπορεί να διαδραματίσουν μεγάλο ρόλο στην επιτυχή εφαρμογή του συστήματος. Δεδομένου ότι η εισαγωγή αυτού του συστήματος θα επιφέρει ριζικές αλλαγές, τα ζητήματα που προκύπτουν είναι τόσο τεχνολογικά, όσο πολιτικά, οργανωτικά και κοινωνικά, όπως η αναθεώρηση και αναδιάρθρωση των ροών εργασίας και η εξέταση της αντίστασης των ιατρών στην αλλαγή.

### 1.8.1 Τεχνολογικά ζητήματα

Οι τεχνικές προκλήσεις μπορούν να διακριθούν πολύ πιο εύκολα και είναι από τις πρώτες που εντοπίζονται. Αποτελούνται από τα κατανοητά προβλήματα, όπως είναι η έλλειψη τυποποίησης των κλινικών δεδομένων και των μηνυμάτων, μέχρι πιο λεπτομερείς προκλήσεις, όπως είναι η ανάγκη για εξαιρετικά ευέλικτα και εύκολα περιβάλλοντα ως προς την υποστήριξη της διαχείρισης των διαδικασιών.

Η υιοθέτησή ενός ηλεκτρονικού φακέλου ασθενή απαιτεί την θέσπιση προτύπων για να διευκολύνει την εύκολη ανταλλαγή των δεδομένων από ένα σύστημα σε ένα άλλο, και από έναν επαγγελματία υγειονομικής περίθαλψης σε έναν άλλο. Η πρόσφατη ανάπτυξη συστημάτων πληροφοριών και τεχνολογιών επικοινωνιών (ICT) καθώς και η ανάπτυξη διαφόρων υπηρεσιών υγείας, οδήγησαν στον πολλαπλασιασμό των ανόμοιων συστημάτων καθώς και των προτύπων που συναντούνται μεταξύ φορέων ιατρικής περίθαλψης. Η επίλυση των προβλημάτων επικοινωνίας που αντιμετωπίζουν τέτοια συστήματα υγείας μπορεί να γίνει εφικτή με την ύπαρξη της διαλειτουργικότητας.

Η ύπαρξη της διαλειτουργικότητας επιτρέπει την εύκολη και άμεση πρόσβαση στις σχετικές, ενημερωμένες πληροφορίες του ασθενούς, με αποτέλεσμα να βελτιώνεται η ποιότητα των υπηρεσιών που δέχεται ο ασθενής. Για την επίτευξη της διαλειτουργικότητας απαιτείται η συνεργασία μεταξύ προμηθευτών, αγοραστών και ρυθμιστικών αρχών του κλάδου των υπηρεσιών υγείας.

Για να επιτευχθεί η διαλειτουργικότητα σε σημασιολογικό επίπεδο, πρέπει να γίνει επισκόπηση των προτύπων κωδικοποίησης και ορολογίας, τα οποία θα οδηγήσουν στην κατανόηση του νοήματος των πληροφοριών που ανταλλάχθηκαν κατά τη μεταφορά

δεδομένων μεταξύ δύο ή περισσότερων συστημάτων υγείας. Τα πρότυπα είναι εκείνα τα οποία θα επιτρέψουν τη δημιουργία της απαραίτητης υποδομής για την παροχή ιατρικής περίθαλψης σε διασυνοριακό επίπεδο, εφόσον τα συστήματα συχνά δεν είναι συμβατά ως προς τη «γλώσσα» που μιλούν και δεν διευκολύνουν την πρόσβαση σε ζωτικές πληροφορίες για την παροχή ασφαλούς και ποιοτικής ιατρικής περίθαλψης.

Οι χρήστες των ιατρικών συστημάτων θα πρέπει να εκπαιδευτούν, ώστε να κατανοήσουν την ανάγκη χρήσης των τυποποιημένων μεθόδων. Αντιλαμβανόμενοι τα οφέλη της χρήσης της τυποποιημένης ορολογίας στις διαδικασίες λήψης αποφάσεων, θα βελτιώσουν τις μέχρι τώρα πρακτικές που ακολουθούσαν στην τεκμηρίωση των αρχείων. Ως εκ τούτου, θα οδηγηθούν στην αποδοτικότερη χρήση των συστημάτων και στην πιο ποιοτική διαχείριση της πληροφορίας.

Ένα βασικό θέμα το οποίο καθυστερεί την εισαγωγή των κωδικοποιήσεων, είναι η έλλειψη κατάρτισης και εκπαίδευσης των ατόμων που θα αναλάβουν το δύσκολο έργο της μετατροπής των διαδικασιών περίθαλψης, ιατρικής ορολογίας, κλινικών δοκιμών και της φαρμακολογίας σε ένα σύνολο αποδεκτών ηλεκτρονικών κωδικών. Το ιατρικό προσωπικό θα πρέπει να συνεργαστεί μαζί με τα άτομα που θα αναπτύξουν τις κωδικοποιήσεις, έτσι ώστε να μπορέσουν να οδηγηθούν στη δημιουργία μιας περιεκτικής κωδικοποίησης που θα χαρακτηρίζεται για τη σαφήνιά της αλλά και για τη λειτουργικότητά της.

Επιπλέον παραδείγματα τεχνικών προκλήσεων είναι τα παρακάτω:

- Η δυνατότητα να αντιμετωπιστεί ο τεράστιος όγκος των δεδομένων που παράγονται με την καταγραφή ακόμα και ενός ελάχιστου ποσού στοιχείων για κάθε ασθενή.
- Η ανάγκη να παρέχονται συστήματα τα οποία υποστηρίζουν τη διάθεση των δεδομένων 24x7, όπου αυτό άλλωστε δικαιολογεί την αντικατάσταση του χειρόγραφου ιατρικού φακέλου .
- Διεπαφές χρήστη οι οποίες είναι αποτέλεσμα μελετών και είναι φιλικές προς τον χρήστη, αλλά ταυτόχρονα καλύπτουν τις απαιτήσεις για λειτουργικότητα και απόδοση, θα καταστήσουν τα συστήματα ηλεκτρονικού φακέλου ασθενή πιο αποδοτικά με μεγαλύτερη αποδοχή από τους χρήστες.
- Η ανάγκη για ύπαρξη μεθόδων επικοινωνίας μεταξύ του ηλεκτρονικού φακέλου ασθενή και των εκάστοτε τοπικών συστημάτων, μέσα από τη δυνατότητα ανταλλαγής ασφαλών μηνυμάτων.



### 1.8.2 Ζητήματα ασφάλειας και εμπιστευτικότητας των δεδομένων

Η μεγαλύτερη ανησυχία που διέπει τους επαγγελματίες υγείας, τους ασθενείς και γενικά τον ιατρικό χώρο, αλλά και ολόκληρη την κοινωνία είναι η παροχή εγγυήσεων ότι όλες οι προσωπικές πληροφορίες διατηρούνται σε ένα ασφαλές περιβάλλον. Παρά το γεγονός ότι σε ένα σύστημα ηλεκτρονικού φακέλου ασθενή οι ευαίσθητες πληροφορίες της υγείας είναι προσβάσιμες μόνο από τους επαγγελματίες του χώρου, μπορεί να προσελκύσουν το ενδιαφέρον τρίτων, όπως ασφαλιστικών εταιρειών. Από την άποψη της προστασίας των δεδομένων, με τη συγκέντρωση των ιατρικών πληροφοριών που βρίσκονται διανεμημένες σε διάφορα ιδρύματα ιατρικής περίθαλψης και με την ενημέρωσή ενός ενιαίου φακέλου, τα συστήματα ηλεκτρονικών φακέλων υγείας εισάγουν νέους κινδύνους σχετικά με την κατάχρηση των ιατρικών πληροφοριών των ασθενών.

Στις μέρες μας αντιμετωπίζονται μεγάλες αδυναμίες στα θέματα ασφάλειας και συνεχώς νέα θεσμικά πλαίσια και πιέσεις στην αγορά οδηγούν στον επανασχεδιασμό των ιατρικών πληροφοριακών συστημάτων, δίνοντας έμφαση στην ασφάλεια. Οι κυριότερες πτυχές ασφαλείας αναφέρονται παρακάτω:

- **Πιστοποίηση:** Έλεγχος της αυθεντικότητας της ταυτότητας των μερών μιας ανταλλαγής δεδομένων
- **Εξουσιοδότηση:** Η πρόσβαση του χρήστη πρέπει να είναι εξουσιοδοτημένη και να βασίζεται στα δικαιώματα πρόσβασης του χρήστη. Η πρόσβαση πρέπει να απαγορεύεται σε μη εξουσιοδοτημένα άτομα
- **Εμπιστευτικότητα:** Η τήρηση του απορρήτου των δεδομένων – η πληροφορία διατίθεται μόνο σε εκείνους τους χρήστες που είναι εξουσιοδοτημένοι
- **Ακεραιότητα:** Τα δεδομένα θα πρέπει να παραμείνουν ακέραια, δηλαδή να μην υποστούν αλλοίωση
- **Δυνατότητα ελέγχου:** Κάθε τροποποίηση ή επεξεργασία των δεδομένων πρέπει να μπορεί να ελεγχθεί, δηλαδή από ποιον έγινε και πότε
- **Ευθύνη:** Καθορισμός της ευθύνης για την εισαγωγή, πρόσβαση ή τροποποίηση κάθε δεδομένου
- **Διαφάνεια:** Τεκμηρίωση των διαδικασιών της επεξεργασίας ώστε να μπορούν να ελεγχθούν
- **Διαθεσιμότητα:** Τα δεδομένα και οι υπηρεσίες πρέπει να είναι διαθέσιμα όταν χρειάζεται

Η τεχνική υποδομή πρέπει να στηρίζεται σε ένα ευέλικτο πρότυπο ασφάλειας, το οποίο μπορεί να ρυθμιστεί και να παρέχει δικαιώματα πρόσβασης ανάλογα με το ρόλο και τη λειτουργία που θα έχουν οι ασθενείς και οι νοσοκομειακοί ιατροί. Το πρότυπο ασφάλειας πρέπει να χαρακτηρίζεται από ποικιλομορφία, για να υποστηρίξει τη χορήγηση ευρείας πρόσβασης στα αρχεία υγείας στους εξουσιοδοτημένους χρήστες, καθώς επίσης και τη δυνατότητα να περιορίσει την πρόσβαση στα «ευαίσθητα» δεδομένα στους χρήστες που δεν θα έχουν τα κατάλληλα δικαιώματα πρόσβασης.

Θα πρέπει να υιοθετηθεί ένα περιεκτικό νομικό πλαίσιο το οποίο θα υποστηρίζει τη διαλειτουργικότητα του συστήματος, λαμβάνοντας μέτρα τεχνικής και οργανωτικής φύσεως, που θα αποτρέπουν την υποκλοπή της ταυτότητας και άλλες επιθέσεις στα προσωπικά δεδομένα. Η ακεραιότητα του συστήματος θα πρέπει να εγγυηθεί από την ύπαρξη τεχνολογιών αιχμής. Επιπλέον, θα πρέπει να ενσωματωθούν τεχνολογίες ασφάλειας και εμπιστευτικότητας πριν την εφαρμογή του συστήματος, ώστε να επιτραπεί η προστασία των προσωπικών δεδομένων.

### 1.8.3 Ζητήματα αποδοχής

Ακόμη και σήμερα, σε μια κοινωνία όπου η έκρηξη της πληροφορίας αποτελεί γεγονός, ένας μεγάλος αριθμός επαγγελματιών του χώρου της υγείας, αρνείται πεισματικά να αποδεχτεί τη χρήση των υπολογιστών κατά τη διάρκεια παροχής ιατρικών υπηρεσιών. Δείχνουν την προτίμησή τους στην καταγραφή δεδομένων με το χέρι και θεωρούν δύσκολη, βαρετή και μη ιδιαίτερα φιλική τη χρήση των ηλεκτρονικών μέσων. Παράλληλα, η απροθυμία των επαγγελματιών υγείας να συνεργαστούν και να μοιραστούν τα δεδομένα με άλλους φορείς ή και με τον ίδιο τον ασθενή, θα μπορούσε να αποτελέσει μεγαλύτερο πρόβλημα.

Ένα μέτρο το οποίο μπορεί να ληφθεί, είναι η δημιουργία κινήτρων για τη χρήση της τεχνολογίας στα πλαίσια της υγείας. Να ενημερωθούν, δηλαδή, οι χρήστες σχετικά με τα οφέλη και την ανάγκη δημιουργίας συστημάτων ηλεκτρονικού φακέλου ασθενή. Μια άλλη καλή στρατηγική για την αντιμετώπιση αυτού του προβλήματος είναι η συμμετοχή των τελικών χρηστών του συστήματος στην διαδικασία της σχεδίασης και ανάπτυξης του συστήματος. Αυτό θα βοηθήσει σημαντικά στην αποδοχή του συστήματος, εφόσον θα έχουν ενεργή συμμετοχή στην υλοποίησή του.

Επίσης, κάποια μερίδα των επαγγελματιών υγειονομικής περίθαλψης αμφισβητεί την ποιότητα και την ακρίβεια των δεδομένων που διαχειρίζονται τα συστήματα ηλεκτρονικών



φακέλων ασθενών. Σε σχέση όμως με τα χειρόγραφα αρχεία, η ποιότητα των ηλεκτρονικών δεδομένων είναι καλύτερη, δεδομένης πάντα της ύπαρξης μέτρων που στοχεύουν στην εξασφάλιση της ακρίβειας.

Τα χαρακτηριστικά που δηλώνουν την ανώτερη ποιότητα των ηλεκτρονικών δεδομένων και τα οποία πρέπει να λαμβάνονται υπόψη σοβαρά είναι:

- Η ακρίβεια και ακεραιότητα των δεδομένων
- Η αξιοπιστία των δεδομένων εφόσον οι πληροφορίες καλούνται να είναι συνεπείς και κατανοητές
- Η πληρότητα των δεδομένων, δηλαδή να υπάρχουν συγκεντρωμένα όλα τα απαραίτητα δεδομένα.
- Η εύκολη αναγνωσιμότητα των δεδομένων.
- Η δυνατότητα πρόσβασης στα δεδομένα, τα οποία πρέπει να είναι διαθέσιμα στα εξουσιοδοτημένα πρόσωπα όταν και όπου απαιτούνται.

#### 1.8.4 Ζητήματα οργάνωσης

Ως προς το οργανωτικό επίπεδο, θα πρέπει πρώτα να ληφθούν τα απαραίτητα οργανωτικά μέτρα και να γίνουν οι απαραίτητες αλλαγές στο τρόπο λειτουργίας του τοπικού συστήματος υγείας. Ένα από τα βασικά οργανωτικά ζητήματα είναι να καθοριστεί κατά πόσο η υλοποίηση του συστήματος θα είναι πλήρης ή θα βασιστεί σε μια σταδιακή και σε βήματα προσέγγιση. Μερικοί από τους παράγοντες που μπορεί να επηρεάσουν τη μορφή της υλοποίησης είναι η ύπαρξη της βασικής υποδομής, η ετοιμότητα όλων των χρηστών να αποδεχτούν τις αλλαγές και τα διαθέσιμα κονδύλια που μπορούν να διατεθούν για την υλοποίηση.

Η διαδικασία εισαγωγής ενός συστήματος ηλεκτρονικού φακέλου ασθενή, απαιτεί μεγάλες αλλαγές και είναι μια τεράστια διαδικασία. Συνήθως προτιμάται η σταδιακή εισαγωγή του συστήματος και για κάποιο χρονικό διάστημα λειτουργεί παράλληλα με το παλιό σύστημα μέχρι να γίνει πλήρης εισαγωγή του.

Ένα δεύτερο σημαντικό θέμα είναι η ποσότητα των αναλογικών δεδομένων που θα ενσωματωθούν στο καινούργιο σύστημα. Οι πληροφορίες που βρίσκονται καταγραμμένες στους χειρόγραφους φακέλους είναι σημαντικές και πολύτιμες για την περίθαλψη ενός ασθενή, αλλά ο τρόπος με τον οποίο θα ενσωματωθούν, εάν θα ενσωματωθούν, μπορεί να έχει σημαντικές επιπτώσεις στις κλινικές αποφάσεις. Η απόφαση που πρέπει να ληφθεί είναι κατά πόσο η περίθαλψη θα στηριχτεί μόνο στα νέα στοιχεία που θα προκύψουν, ή κατά πόσο

θα βασιστεί και στα προηγούμενα δεδομένα. Η ενσωμάτωση των προηγούμενων δεδομένων είναι ένας στόχος επίπονος και σύνθετος, ο οποίος θα πρέπει να προνοήσει και την ύπαρξη των απαραίτητων κεφαλαίων.

Τέλος πρέπει να ληφθεί υπόψη η ποιότητα και η διαθεσιμότητα χώρου που θα χρησιμοποιηθεί για την εγκατάσταση της υλικοτεχνικής υποδομής, όπως είναι οι υπολογιστές και οι servers. Πρέπει και πάλι να γίνουν οι κατάλληλες μελέτες για αναθεώρηση των υπάρχοντων χώρων για να καθοριστεί κατά πόσο ικανοποιητικές και κατάλληλες είναι οι εγκαταστάσεις. Θα πρέπει να γίνει η καταγραφή της υπάρχουσας υποδομής ώστε να διαπιστωθεί αν κάποιο μέρος του υπάρχοντος εξοπλισμού είναι ικανό να υποστηρίξει το καινούργιο σύστημα.

### 1.8.5 Ζητήματα χρηματοδότησης

Η δημιουργία ενός συστήματος ηλεκτρονικού φακέλου ασθενή, απαιτεί την ύπαρξη υψηλών κεφαλαίων και δαπανών που θα καλύψουν τα κόστη της υλικοτεχνικής υποδομής. Τα κονδύλια που αφιερώνονται στην υγειονομική περίθαλψη είναι μηδαμικά για να καλύψουν το αυξημένο και τεράστιο κόστος μια τέτοιας προσπάθειας. Οι υπεύθυνοι της υγείας αλλά και η κυβέρνηση αντιλαμβάνονται ένα τέτοιο εγχείρημα σαν μια οικονομική επένδυση. Ενδιαφέρονται για τα οικονομικά οφέλη και τους οικονομικούς όρους, αλλά τα κέρδη από ένα σύστημα σαν και αυτό πρέπει είναι η ποιοτικότερη περίθαλψη των ασθενών. Η επένδυση αυτή θεωρείται ότι πρέπει να είναι ικανή να επιβιώνει και να συντηρείται μέσα από τους δικούς της εσωτερικούς μηχανισμούς, σε μια περίοδο που το γενικό κόστος της υγείας αυξάνεται και η διαθέσιμη χρηματοδότηση είναι περιορισμένη. Σε πολλές προσπάθειες υλοποίησης ενός τέτοιου συστήματος, διαπιστώθηκε ότι παρά τα αρχικά κεφάλαια που είχαν διοχετευθεί για τον εκσυγχρονισμό, το καινούργιο σύστημα φάνηκε να είναι ασύμφορο οικονομικά ως προς τη χρήση και τελικά εγκαταλείφθηκε.

## 1.9 Προσωπικός Φάκελος Υγείας

Ο Προσωπικός Φάκελος Υγείας (Personal Healthcare Record - PHR) είναι ένα εργαλείο που μπορεί να χρησιμοποιηθεί για την παρακολούθηση και τον διαμοιρασμό παλαιών και νέων δεδομένων σχετικά με την υγεία ενός ατόμου.

Ο Ιατρικός Φάκελος και ο Προσωπικός Φάκελος Υγείας δεν είναι το ίδιο πράγμα. Ο γενικά συμφωνημένος διαχωρισμός των όρων αυτών αφορά κυρίως την ιδιοκτησία των δεδομένων. Ο Ιατρικός Φάκελος περιλαμβάνει δεδομένα που δημιουργούνται και διατηρούνται από κάθε

έναν από τους παρόχους υγειονομικής περίθαλψης. Ο Προσωπικός Φάκελος Υγείας είναι συνήθως ένα αρχείο υγείας, το οποίο δημιουργείται και υποστηρίζεται από ένα άτομο. Τα δεδομένα του PHR προέρχονται από τον ιατρικό του φάκελο, αλλά δεν διατηρούνται στο σύστημα της μονάδας παροχής υγειονομικής περίθαλψης. Αντίθετα, ο ασθενής αποφασίζει τον τόπο διατήρησής τους και η πρόσβαση στις πληροφορίες αυτές ελέγχεται εξ ολοκλήρου από τον ίδιο.

Ένα ιδανικό PHR θα παρέχει πλήρη και ακριβή περίληψη του ιατρικού ιστορικού και της ιατρικής περίθαλψης του ατόμου, μέσω της συλλογής δεδομένων από πολλές πηγές και καθιστώντας τις πληροφορίες αυτές σε διαθέσιμες σε όποιον έχει τα απαραίτητα ηλεκτρονικά διαπιστευτήρια για να τις δει. Ο όρος Personal Healthcare Record δεν είναι νέος. Αναφέρθηκε για πρώτη φορά το 1978, ωστόσο τα περισσότερα επιστημονικά άρθρα που σχετίζονταν με αυτόν δημοσιεύτηκαν μετά το 2000. Αν και σαν έννοια μπορεί να συσχετιστεί τόσο με ηλεκτρονικά όσο και με χειρόγραφα συστήματα, η χρήση του στις μέρες μας υπονοεί την ύπαρξη μιας ηλεκτρονικής πηγής.

Δεν υπάρχει κανένα νομικό πλαίσιο που να υποχρεώνει τον ασθενή να διατηρεί τέτοιο προσωπικό αρχείο, παρόλα αυτά η χρήση του μπορεί να συμβάλει στην παρακολούθηση της υγείας ενός ατόμου, το συντονισμό της φροντίδας που λαμβάνει και την εξασφάλιση της ποιότητας της υγειονομικής περίθαλψης που του παρέχεται. Σήμερα, ο Προσωπικός Φάκελος Υγείας ενός ατόμου μπορεί να διατηρηθεί σε ειδικά διαμορφωμένες ιστοσελίδες στο διαδίκτυο. Οι δύο πιο διαδεδομένες ιστοσελίδες αυτού του τύπου έχουν αναπτυχθεί από την Microsoft και την Google και παρουσιάζονται συνοπτικά παρακάτω:

### 1.9.1 Microsoft Health Vault



Ένα βήμα μπροστά από την αναμενόμενη, αντίστοιχη κίνηση της Google, η Microsoft εγκαινίασε μια δωρεάν, διαδικτυακή υπηρεσία που επιτρέπει στους χρήστες να αποθηκεύουν και να διαχειρίζονται αποτελέσματα ιατρικών εξετάσεων και άλλα δεδομένα υγείας. Για τους χρήστες, η υπηρεσία HealthVault λειτουργεί ταυτόχρονα σαν φάκελος αρχειοθέτησης, ιατρική βιβλιοθήκη και συσκευή fax για επικοινωνία με τους γιατρούς.

Ο δικτυακός τόπος καταγράφει ιατρικά ιστορικά, εμβολιασμούς και πληροφορίες από επισκέψεις σε γιατρούς και νοσοκομεία, συμπεριλαμβανομένων των μετρήσεων από

συσκευές όπως καρδιακά μόνιτορ και μετρητές του σακχάρου του αίματος. Συνδέεται, επίσης, με μια νέα μηχανή αναζήτησης της Microsoft που ειδικεύεται σε ιατρικές πληροφορίες. Οι χρήστες μπορούν να αποστέλλουν ειδικά email σε φίλους, συγγενείς και γιατρούς και να τους δίνουν πρόσβαση σε όλα ή σε ορισμένα από τα αποθηκευμένα δεδομένα.

Προκειμένου η Microsoft να πετύχει τον στόχο της, συνεργάστηκε με ορισμένους κορυφαίους οργανισμούς υγείας αλλά και κλινικές, όπως η Αμερικάνικη καρδιολογική εταιρεία και η κλινική Mayo. Όσον αφορά τα προσωπικά δεδομένα, αυτά πρόκειται να αποθηκεύονται σε κρυπτογραφημένες βάσεις δεδομένων, με τα μέλη του portal να έχουν τον πρώτο λόγο όσον αφορά στον καθορισμό δικαιωμάτων ως προς την πρόσβαση των υπευθύνων του HealthVault σε αυτά.

Η Microsoft κάνει ένα επιπλέον βήμα, ενθαρρύνοντας τους γιατρούς να καταγράψουν τα αρχεία των ασθενών τους απευθείας στο HealthVault διευκολύνοντας έτσι την επικοινωνία τους με τους ασθενείς. Έτσι, οι τελευταίοι, σύμφωνα με το πιο αισιόδοξο σενάριο, δεν θα χρειάζεται ούτε να τηλεφωνήσουν ούτε φυσικά να επισκεφτούν το γραφείο του προσωπικού τους γιατρού αφού τα αποτελέσματα των τελευταίων εξετάσεων θα βρίσκονται αποθηκευμένα στο HealthVault.

Η Microsoft ελπίζει ότι στο μέλλον τα νοσοκομεία, οι γιατροί, οι ασφαλιστικές εταιρείες και οι οργανώσεις ασθενών θα δημιουργήσουν διαδικτυακές εφαρμογές για την αξιοποίηση των αποθηκευμένων δεδομένων κατά περίπτωση. Τέτοιες εφαρμογές αναπτύσσουν ήδη η καρδιολογική και η πνευμονολογική εταιρεία των ΗΠΑ, ενώ η Johnson & Johnson θα κυκλοφορήσει μετρητές σακχάρου που ανεβάζουν τις καταγραφές απευθείας στον δικτυακό τόπο. Παρά τις διαβεβαιώσεις της Microsoft, πάντως, η συγκέντρωση τόσων ευαίσθητων πληροφοριών στο Διαδίκτυο πολλαπλασιάζει τις ευκαιρίες για διαρροή ή κατάχρηση.

### 1.9.2 Google Health



Η Google κυκλοφόρησε μία νέα φιλόδοξη εφαρμογή που σκοπεύει να δώσει στους χρήστες ένα κεντρικό online χώρο για να αποθηκεύουν τα ιατρικά τους δεδομένα και να τα μοιράζονται με ιατρικούς φορείς. Το Google Health Beta μπορεί να δημιουργήσει προφίλ χρήστη που θα περιλαμβάνει ιατρικές πληροφορίες, όπως η τρέχουσα ιατρική κατάσταση, οι αλλεργίες και η φαρμακολογία που κάποιος πιθανόν λαμβάνει. Η αντιπρόεδρος της Google



Marissa Mayer αναφέρει «Το Google Health αναφέρεται στην διαδικασία καταγραφής των ιατρικών δεδομένων από το γραφείο του γιατρού, τους εργαστηριακούς ελέγχους και τα ληφθέντα φάρμακα, ώστε να δίνουν μία ολοκληρωμένη εικόνα για την υγεία ενός ατόμου».

Με την πρώτη ματιά βλέπουμε ότι το online βιβλιάριο χωρίζεται σε τέσσερις κατηγορίες: Ιατρικό Προφίλ (Health profile), Εισαγωγή Ιατρικών Δεδομένων (Import medical records), Αναζήτηση ιατρικών υπηρεσιών στο δίκτυο (Explore online health services) και Αναζήτηση ιατρών (Doctor search) αλλά μόνο το πρώτο βρίσκει εφαρμογή στην Ελλάδα. Αν και προς το παρόν το Google Health φαίνεται να είναι χρήσιμο μόνο για τους κατοίκους πέραν του Ατλαντικού δε θα αργήσει να γίνεται περισσότερο χρήσιμο για το μεγαλύτερο μέρος του κόσμου.

Η Google είναι μία από τις εταιρίες που προσπαθεί να «δημοκρατικοποιήσει» την υγεία επιτρέποντας στους χρήστες της υπηρεσίας να έχουν πλήρη έλεγχο στα προσωπικά τους ιατρικά δεδομένα. Φυσικά εγείρονται ζητήματα προστασίας των προσωπικών δεδομένων, στα οποία η Google απαντάει ότι θα χρησιμοποιήσει πιο ασφαλής servers από ότι συνήθως χρησιμοποιεί για τις υπόλοιπες υπηρεσίες καθώς και ότι ο χρήστης θα μπορεί να διαλέξει ποιοι θα μπορούν να δουν τα προσωπικά ιατρικά του δεδομένα. Το κατά πόσον διασφαλίζονται τα υπερευαίσθητα προσωπικά δεδομένα είναι ένα σημείο που το εγχείρημα θα βρεθεί αντιμέτωπο με τους σκεπτικιστές, αλλά από οικονομικής πλευράς είναι σχεδόν σίγουρο πως θα δώσει αρκετά στην Google, αφού στις υπηρεσίες υγείας δαπανώνται πολλά χρήματα.

## Συμπεράσματα

Δεδομένης της σημερινής πολυπλοκότητας των λειτουργιών των σύγχρονων παρόχων ιατρικών υπηρεσιών, καθώς και της άμεσης ανάγκης για περιορισμό των λειτουργικών εξόδων, ένας ηλεκτρονικός ιατρικός φάκελος δεν αρκεί, εκτός εάν:

- Διαθέτει εξαιρετικά οργανωμένη δομή, ώστε να διαχειρίζεται εξίσου αποτελεσματικά πραγματικά ιατρικά δεδομένα, ανακτώμενα από ιατρικά μηχανήματα, διαγνώσεις, αγωγές αλλά και διοικητικά και οικονομικά στοιχεία.
- Διασφαλίζει την προστασία των δεδομένων, περιλαμβάνει μηχανισμούς ταυτοποίησης των χρηστών και ελέγχει την πρόσβαση στα δεδομένα.
- Συνοδεύεται από εξελιγμένες δυνατότητες διαχείρισης πληροφοριών και προγραμματισμού εργασιών.



- Προσφέρει υψηλή ευελιξία, ώστε να υποστηρίζει διαφορετικά επιχειρηματικά μοντέλα.
- Υποστηρίζει την μεταφερισιμότητα μεταξύ συστημάτων, ανεξάρτητα από το υλικό, το λογισμικό και την εθνική γλώσσα που χρησιμοποιεί ο καθένας.

Από τα ζητήματα που αναφέραμε παραπάνω, τα ζητήματα χρηματοδότησης και οργάνωσης είναι εκτός του σκοπού της συγκεκριμένης εργασίας. Στα κεφάλαια που ακολουθούν θα ασχοληθούμε με τα τεχνολογικά ζητήματα και τα ζητήματα ασφαλείας που σχετίζονται με την εφαρμογή ενός συστήματος Ηλεκτρονικού Φακέλου Υγείας.





## Κεφάλαιο 2

# Υπηρεσίες Διαδικτύου και Υπηρεσιοστρεφής Αρχιτεκτονική

### Εισαγωγή

Όπως αναφέραμε στο προηγούμενο κεφάλαιο, σημαντικά εμπόδια για την εφαρμογή ενός συστήματος Ηλεκτρονικού Φακέλου Υγείας αποτελούν τα ζητήματα διαλειτουργικότητας τέτοιου είδους συστημάτων, αλλά και ζητήματα ασφάλειας των δεδομένων.

Τα web services, ήρθαν να δώσουν λύση στα προβλήματα ευχρηστίας και διαλειτουργικότητας των ηλεκτρονικών υπηρεσιών. Μέσα από την αρχιτεκτονική τους ορίζεται ένα σύνολο από προδιαγραφές και κανόνες, με τους οποίους είναι δυνατή η δημιουργία και η παροχή ηλεκτρονικών υπηρεσιών μέσω του Internet με αρκετά απλό τρόπο. Από την επιχειρηματική σκοπιά, τα Web Services επιτυγχάνουν την ολοκλήρωση μεταξύ των δεδομένων και των εφαρμογών και μειώνουν την απόσταση ανάμεσα στις τεχνικές λεπτομέρειες και τον επιχειρησιακό προσανατολισμό.

Η αρχιτεκτονική SOA, με τη σειρά της, επιτρέπει τη δυναμική δημιουργία σύνθετων εφαρμογών και επιχειρηματικών διαδικασιών σε πραγματικό χρόνο με μοναδικά οφέλη. Συγκεκριμένα, προσφέρει οφέλη στο κόστος ανάπτυξης και συντήρησης των εφαρμογών, απλοποιημένη ενοποίηση σε ετερογενή περιβάλλοντα, μεγαλύτερη αξιοποίηση των υπάρχοντων πόρων πληροφορικής και αποτελεσματική διαχείριση και ασφάλεια. Η αρχιτεκτονική SOA αντιλαμβάνεται την επιχείρηση σαν ένα σύνολο από συνδεδεμένες υπηρεσίες και, με προσέγγιση βασισμένη σε ανοικτά πρότυπα, μετατρέπει τις επιχειρηματικές διαδικασίες σε πιο αποτελεσματικές, αποδοτικές και συνεργατικές.

Στο κεφάλαιο αυτό θα αναφερθούμε στην βιβλιογραφία σχετικά με τις Υπηρεσίες Διαδικτύου και την Υπηρεσιοστρεφή Αρχιτεκτονική. Θα μιλήσουμε για τα πλεονεκτήματά τους, τα τεχνικά στοιχεία τους και τα επίπεδα ασφαλείας που προσφέρουν. Στη συνέχεια θα μιλήσουμε για τις Επιχειρηματικές Διαδικασίες καθώς και για την Ενορχήστρωση και την Χορογραφία των Υπηρεσιών Διαδικτύου.

## 2.1 Υπηρεσίες Διαδικτύου

Σύμφωνα με την IBM:

*«τα web services είναι μια τεχνολογία που επιτρέπει στις εφαρμογές να επικοινωνούν μεταξύ τους ανεξαρτήτως πλατφόρμας και γλώσσας προγραμματισμού. Ένα web service είναι μια διεπαφή λογισμικού (software interface) που περιγράφει μια συλλογή από λειτουργίες οι οποίες μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων XML. Χρησιμοποιεί πρότυπα βασισμένα στη γλώσσα XML για να περιγράψει μία λειτουργία (operation) προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή. Μια ομάδα από web services τα οποία αλληλεπιδρούν μεταξύ τους, καθορίζει μια εφαρμογή web services».*

Από μια λιγότερο τεχνική σκοπιά, τα web services είναι μια καινοτομική αρχιτεκτονική με την οποία παρέχεται η δυνατότητα δημιουργίας και χρήσης ηλεκτρονικών υπηρεσιών στο διαδίκτυο με απλό και οικονομικό τρόπο. Σύμφωνα με το Δίκτυο Προγραμματιστών της Microsoft (Microsoft Developer Network - MSDN), τα web services έχουν τρία κοινά χαρακτηριστικά:

- Εκθέτουν χρήσιμη λειτουργικότητα σε χρήστες του διαδικτύου μέσα από ένα πρότυπο δικτυακό πρωτόκολλο. Στις περισσότερες περιπτώσεις αυτό το πρωτόκολλο είναι το SOAP (Simple Object Access Protocol).
- Είναι αυτοπεριγραφόμενα. Παρέχουν ένα τρόπο να περιγράψουν τις διεπαφές τους με αρκετή λεπτομέρεια, ώστε να επιτρέψουν στο χρήστη τους να χτίσει μια εφαρμογή-πελάτη, η οποία να επικοινωνήσει μαζί τους. Η περιγραφή συνήθως παρέχεται σε ένα έγγραφο με χρήση της γλώσσας WSDL (Web Services Description Language).
- Είναι ανακαλύψιμα. Καταχωρούνται, ώστε οι δυνητικοί χρήστες να μπορούν να τα βρουν εύκολα. Αυτό γίνεται με το UDDI (Universal Discovery Description and Integration).

Τα web services επιτρέπουν σε ανόμιες εφαρμογές να επικοινωνήσουν και να ανταλλάξουν δεδομένα, χωρίς τη χρήση επιπρόσθετου υλικού και λογισμικού. Η ανταλλαγή αυτή είναι ανεξάρτητη της γλώσσας προγραμματισμού και της πλατφόρμας που έχουν αναπτυχθεί οι εφαρμογές, καθώς και του πρωτόκολλου διαδικτύου που θα χρησιμοποιηθεί για την



μεταφορά. Οι εφαρμογές που βασίζονται σε web services είναι αυτόνομες, ανεξάρτητες και αυτό-περιγραφικές.

Τα web services αποτελούν λογική συνέχεια των object oriented και component based εφαρμογών. Αποτελούν λογισμικές μονάδες που διατίθενται μέσω του διαδικτύου και δίνουν τη δυνατότητα για ολοκλήρωση εργασιών, επίλυση προβλημάτων ή διεξαγωγή συναλλαγών, εκ μέρους ενός χρήστη ή μιας εφαρμογής.

### 2.1.1 Πλεονεκτήματα των web services

Η αρχιτεκτονική των web services παρέχει αρκετά πλεονεκτήματα μερικά από τα οποία αναφέρονται παρακάτω:

**Ευκολότερος χειρισμός δεδομένων:** Παραδοσιακά το κυριότερο πρόβλημα στις καταναμημένες τεχνολογίες ήταν το λεγόμενο tight-coupling (ισχυρή συνδεσιμότητα). Μια εφαρμογή που καλούσε μια άλλη απομακρυσμένη ήταν αυστηρά δεμένη με αυτή, από την κλήση (function call) που εκτελούσε και τις παραμέτρους που περνούσε. Στα περισσότερα συστήματα, πριν από την έλευση των web services, η επικοινωνία πραγματοποιούταν με μια σταθερή διεπαφή με λίγη έως καθόλου ευελιξία ή προσαρμοστικότητα στα περιβάλλοντα ή τις ανάγκες που μεταβάλλονταν συνεχώς.

Τα web services χρησιμοποιούν τη γλώσσα XML, η οποία μπορεί να περιγράψει οποιαδήποτε δεδομένα με ένα πραγματικά ανεξάρτητο από πλατφόρμα τρόπο, για την ανταλλαγή αυτών των δεδομένων μεταξύ των συστημάτων. Με αυτόν τον τρόπο οδηγούμαστε σε εφαρμογές με χαλαρή συνδεσιμότητα (loosely-coupled). Επιπλέον, τα web services μπορούν να λειτουργήσουν σε πιο αφηρημένο επίπεδο, στο οποίο μπορούν να επαναξιολογήσουν, να τροποποιήσουν ή να χειριστούν τύπους δεδομένων δυναμικά κατά περίπτωση. Έτσι, σε τεχνικό επίπεδο, τα web services μπορούν να χειριστούν δεδομένα πολύ ευκολότερα και να επιτρέψουν στο λογισμικό να επικοινωνεί πιο ελεύθερα.

**Διαλειτουργικότητα:** Οι προηγούμενες καταναμημένες τεχνολογίες είχαν σοβαρά ζητήματα διαλειτουργικότητας, διότι κάθε προμηθευτής (vendor) υλοποιούσε το δικό του πρότυπο για μια καταναμημένη δρομολόγηση μηνυμάτων μεταξύ αντικειμένων που βρίσκονται σε διαφορετικά φυσικά περιβάλλοντα (distributed object messaging). Ένα web service παρέχει ανεξαρτησία τόσο από το λειτουργικό σύστημα όσο και από το υλικό (hardware). Οποιοδήποτε πρόγραμμα που συμβαδίζει με αυτή τη τεχνολογία μπορεί πολύ εύκολα να προσπελάσει μία τέτοια υπηρεσία. Με την XML ως το μόνο πρότυπο στα web services,

συστήματα φτιαγμένα από διαφορετικές τεχνολογίες όπως η Java και το .Net μπορούν να επικοινωνήσουν μεταξύ τους.

**Ενσωμάτωση:** Για ένα υπάρχον σύστημα που λειτουργεί στο Internet, η δημιουργία ενός web service δεν απαιτεί αλλαγές στον μηχανισμό του.

**Διαθεσιμότητα και Δημοσίευση:** Οι πληροφορίες για τα web services δημοσιεύονται, οπότε η εύρεση και η χρήση τους μπορεί να γίνει πολύ γρήγορα.

**Επέκταση:** Ένα έτοιμο web service είναι δυνατό να επεκταθεί με εύκολο τρόπο, παρέχοντας έτσι νέες υπηρεσίες στους χρήστες του.

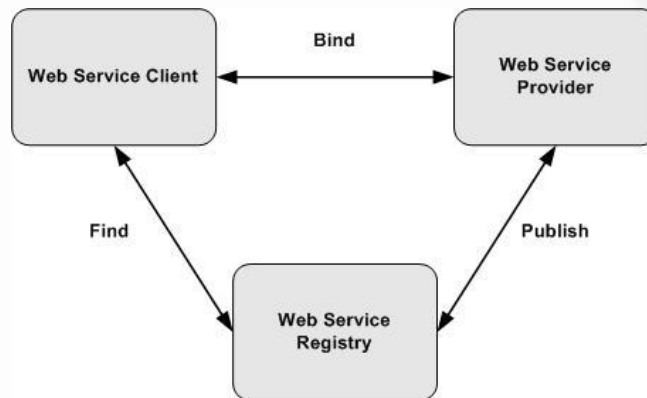
**Μικρό κόστος δημιουργίας και χρήσης:** Το κόστος ενσωμάτωσης ενός web service σε κάποια διαδικτυακή εφαρμογή είναι πάρα πολύ μικρό. Ακόμα και στις περιπτώσεις που η χρήση κάποιου web service γίνεται με ενοικίαση, σίγουρα το συνολικό κόστος της χρήσης είναι αρκετά πιο μικρό από το κόστος δημιουργίας της υπηρεσίας αυτής.

**Ευκολία στην επικοινωνία:** Με τις προηγούμενες τεχνολογίες η συνεργασία μεταξύ συστημάτων ήταν προβληματική, γιατί οι κατακευματισμένες τεχνολογίες, όπως οι CORBA και DCOM, χρησιμοποιούσαν μη πρότυπες πόρτες(ports). Συνεπώς, μια συνεργασία σήμαινε άνοιγμα "οπών" στα τείχη προστασίας (firewalls), κάτι που πολλές φορές δεν ήταν αποδεκτό από τους ανθρώπους της πληροφορικής σε έναν οργανισμό, αφού έθετε σε κίνδυνο την ασφάλεια των συστημάτων. Το γεγονός αυτό δεν επέτρεπε τη δυναμική συνεργασία μεταξύ διαφορετικών συστημάτων, λόγω του ότι απαιτούσε μια χειροκίνητη διαδικασία. Τα web services μπορούν να χρησιμοποιήσουν (μεταξύ άλλων) το HTTP ως πρωτόκολλο μεταφοράς και τα περισσότερα τείχη προστασίας επιτρέπουν την πρόσβασή του μέσω της θύρας 80 (πρότυπη θύρα για το HTTP). Με αυτόν τον τρόπο οδηγούμαστε σε ευκολότερες και πιο δυναμικές συνεργασίες μεταξύ των συστημάτων.

### 2.1.2 Τεχνικά στοιχεία των web services

Το μοντέλο των web services ακολουθεί το παράδειγμα δημοσίευση (publish), εύρεση(find) και σύνδεση(bind). Στο πρώτο βήμα, ο προμηθευτής της υπηρεσίας δημοσιεύει την υπηρεσία σε ένα κατάλογο υπηρεσιών. Στο δεύτερο βήμα, ο πελάτης, ο οποίος ψάχνει για μία υπηρεσία που να καλύπτει τις απαιτήσεις του, την αναζητεί στον κατάλογο. Αφού βρει πολλαπλές υπηρεσίες, επιλέγει μία βάσει των προτιμήσεών του. Τότε μεταφορτώνει την περιγραφή της

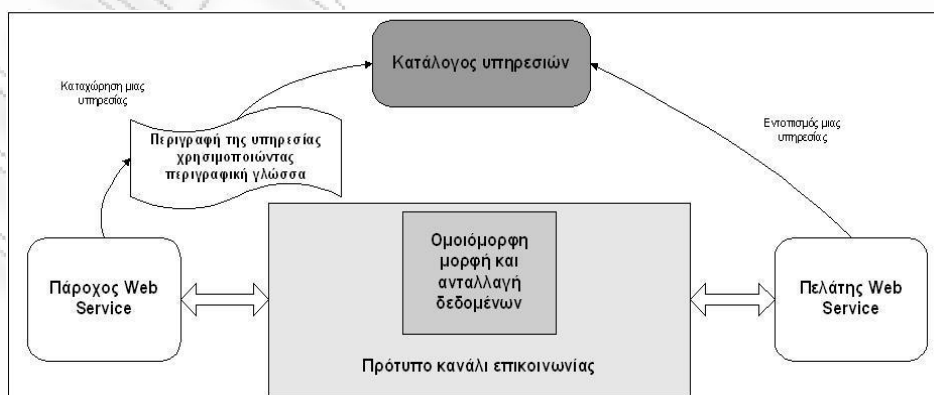
υπηρεσίας και συνδέεται (δεσμεύεται) με αυτήν, ώστε να μπορέσει να καλέσει και να εκτελέσει την υπηρεσία.



Εικόνα 5: Το μοντέλο των web services [E7]

Όταν μιλάμε, λοιπόν, για μία αρχιτεκτονική προσαρμοσμένη στις υπηρεσίες προκύπτουν ορισμένα ζητήματα. Η εφαρμογή που παρέχει την υπηρεσία και η εφαρμογή-πελάτης, η οποία χρησιμοποιεί την υπηρεσία, μιλάνε μεταξύ τους σε μια κοινή γλώσσα. Έπειτα, οι δύο εφαρμογές χρειάζονται ένα τρόπο να εντοπίζουν η μία την άλλη πριν ξεκινήσουν να μιλούν μεταξύ τους. Ως εκ τούτου, μπορούμε να πούμε ότι μια βασική αρχιτεκτονική για web services πρέπει να παρέχει:

- Έναν πρότυπο τρόπο για επικοινωνία.
- Ένα ομοίμορφο μηχανισμό για την περιγραφή και την ανταλλαγή των δεδομένων.
- Μια πρότυπη περιγραφική γλώσσα (meta language) για να περιγράψει τις υπηρεσίες που προσφέρονται.
- Ένα μηχανισμό για να καταχωρούνται και να εντοπίζονται οι εφαρμογές που βασίζονται σε web services.



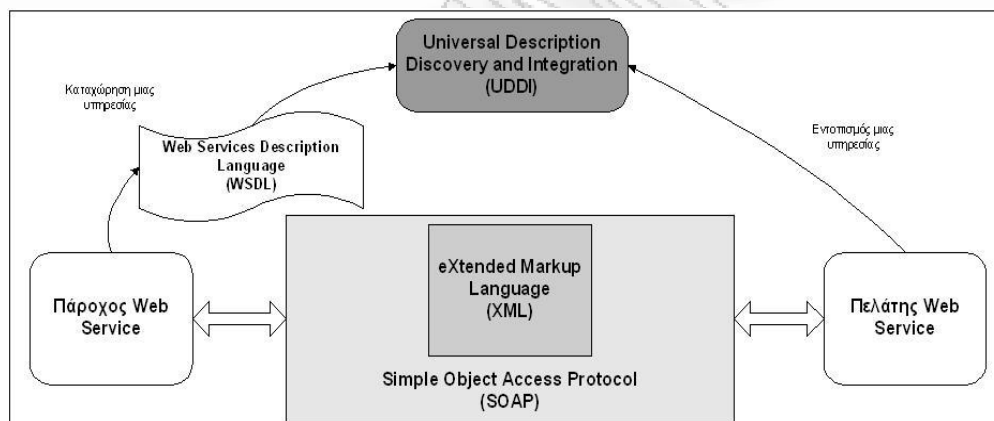
Εικόνα 6: Η αρχιτεκτονική των web services [E8]

Για την επικοινωνία χρησιμοποιείται συνήθως το πρωτόκολλο HTTP, το ίδιο δηλαδή πρωτόκολλο που χρησιμοποιούν και οι κοινοί browsers για την πλοήγηση στο Internet. Πιο απλά τα δεδομένα μεταφέρονται όπως ακριβώς μεταφέρονται και οι ιστοσελίδες.

Βασιζόμενο πάνω στο HTTP πρωτόκολλο (όχι αναγκαστικά) χρησιμοποιείται ένα άλλο πρωτόκολλο, το SOAP. Με τη χρήση του πρωτοκόλλου αυτού υλοποιείται ένας εξυπηρετητής (server) ο SOAP server.

Για κάθε μέθοδο της υπηρεσίας που θέλουμε να προσφέρουμε, ορίζουμε μία αντιστοιχία με μία λειτουργία του SOAP Server. Κατόπιν, το σύνολο των λειτουργιών του SOAP Server καθώς και τα υπόλοιπα χαρακτηριστικά του, περιγράφονται σε ένα WSDL αρχείο, το οποίο δημοσιεύεται στο Internet, έτσι ώστε να δίνεται η δυνατότητα σε κάθε ενδιαφερόμενο χρήστη να μπορεί άμεσα να χρησιμοποιήσει την υπηρεσία.

Έτσι, σύμφωνα με τα παραπάνω, η εικόνα 6 μετατρέπεται στην εικόνα 7 που παρουσιάζεται παρακάτω.



Εικόνα 7: Τεχνολογίες των web services [E8]

Πιο αναλυτικά, οι βασικές τεχνολογίες στις οποίες βασίζονται τα web services παρουσιάζονται παρακάτω :

### **XML - Ομοιόμορφος ορισμός και ανταλλαγή δεδομένων**

Η Extended Markup Language (XML) είναι μια μέτα-γλώσσα (περιγραφική γλώσσα) η οποία έχει καλά καθορισμένη σύνταξη και σημασιολογία. Τα «αυτοπεριγραφικά» χαρακτηριστικά της XML την μετατρέπουν σε έναν απλό, αλλά δυνατό, μηχανισμό για τη σύλληψη και την ανταλλαγή των στοιχείων μεταξύ διαφορετικών εφαρμογών.



### **SOAP - Πρότυπο κανάλι επικοινωνίας**

Το Simple Object Access Protocol (SOAP) είναι το κανάλι που χρησιμοποιείται για την επικοινωνία μεταξύ μιας εφαρμογής - προμηθευτή web services και μιας εφαρμογής-πελάτη. Η απλότητα του SOAP είναι το ότι δεν καθορίζει κανένα νέο πρωτόκολλο μεταφοράς. Αντίθετα, επαναχρησιμοποιεί μεταξύ άλλων το Hyper Text Transfer Protocol (HTTP) ή το Simple Mail Transfer Protocol (SMTP) για μεταφορά δεδομένων σαν μηνύματα. Αυτή η χρήση του HTTP ή του SMTP σαν πρωτόκολλο μεταφοράς εξασφαλίζει ότι οι εφαρμογές - προμηθευτές με τις εφαρμογές - πελάτες μπορούν να επικοινωνήσουν χρησιμοποιώντας το διαδίκτυο σαν ραχοκοκαλιά. Έτσι η χρήση του SOAP πολλαπλασιάζει τις ικανότητες των web services.

Το SOAP είναι standard της W3C και χρησιμοποιείται αρκετά με στόχο την αποστολή απλών αντικειμένων (αρχείων, εφαρμογών, κλπ.) σε XML μορφή. Γι' αυτό και κάθε web service που χρησιμοποιεί SOAP μπορεί να λάβει αιτήσεις για συγκεκριμένες λειτουργίες, απλά δεχόμενο αντικείμενα σε XML.

Πλέον, οι περισσότερες γλώσσες προγραμματισμού από την Delphi v7 μέχρι το Visual Studio .NET, την PHP, την JSP και άλλες πολλές, υποστηρίζουν τη δημιουργία SOAP Servers με πάρα πολύ απλό τρόπο. Το μόνο που πρέπει να κάνει κανείς είναι να καθορίσει τις λειτουργίες που πρέπει να γίνουν, όταν ο server δεχθεί κάποιο αίτημα προς εξυπηρέτηση.

### **WSDL - Πρότυπη περιγραφική γλώσσα για την περιγραφή των παρεχόμενων υπηρεσιών**

Οι εφαρμογές που παρέχουν web services διαφημίζουν τις διάφορες υπηρεσίες που παρέχουν χρησιμοποιώντας μια πρότυπη περιγραφική γλώσσα που ονομάζεται Web Services Description Language (WSDL). Η WSDL βασίζεται στην XML και χρησιμοποιεί ένα ειδικό σύνολο ετικετών (tags) για να περιγράψει ένα web service, τις υπηρεσίες που παρέχονται, που να εντοπιστεί και ούτω καθεξής. Οι εφαρμογές-πελάτες λαμβάνουν πληροφορίες για ένα web service πριν από την πρόσβασή τους σε αυτό και τελικά τη χρήση του. Οι προγραμματιστές και εδώ δεν θα πρέπει να ανησυχούν πολύ, αφού υπάρχουν πάρα πολλά διαθέσιμα εργαλεία που δημιουργούν αυτόματα ένα WSDL αρχείο παράλληλα με τη δημιουργία του SOAP server.

### UDDI - Καταχώρηση και εντοπισμός των παρεχόμενων υπηρεσιών

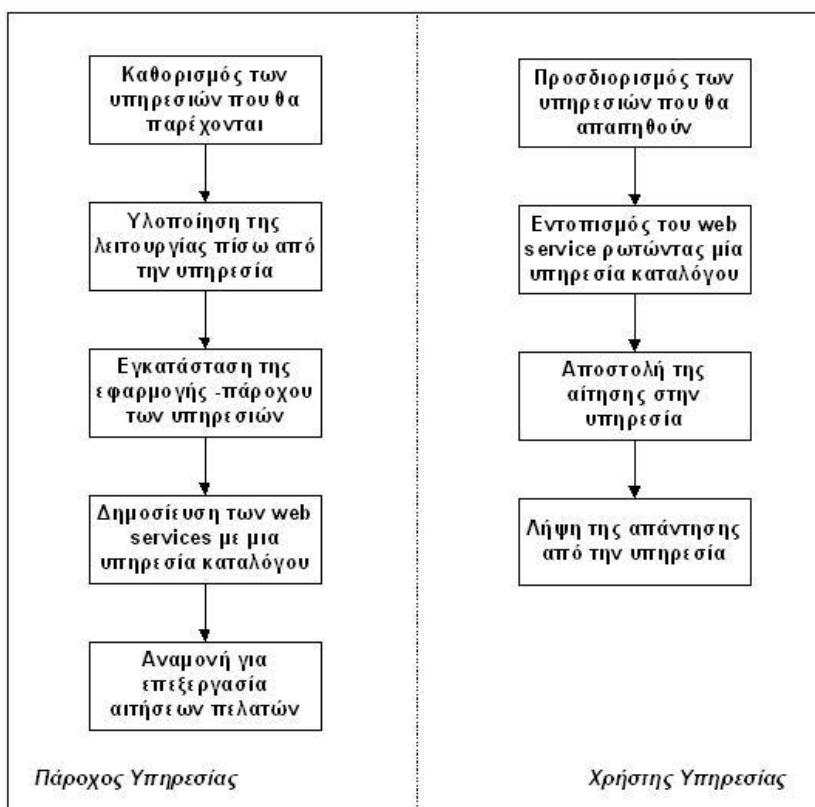
Ο «χρυσός οδηγός» των web services είναι το Universal Description Discovery and Integration (UDDI). Οι εφαρμογές που παρέχουν web services παρατίθενται σε ένα κατάλογο από παρόχους υπηρεσιών χρησιμοποιώντας το UDDI. Παρόμοια, οι εφαρμογές-πελάτες εντοπίζουν τους παρόχους εφαρμογών web services χρησιμοποιώντας UDDI. Όπως και στην περίπτωση της WSDL, το UDDI βασίζεται στην XML.

Κάθε καταχώρηση περιέχει το WSDL αρχείο και τη διεύθυνση που λειτουργεί η υπηρεσία στο Internet. Επιπρόσθετα, σε κάθε καταχώρηση υπάρχουν και διάφορες άλλες πληροφορίες για την υπηρεσία που σχετίζονται με τον ιδιοκτήτη της, την πολιτική του κ.α.

Υπάρχουν διαφορετικοί τύποι καταχωρήσεων μίας υπηρεσίας. Πιο συγκεκριμένα, υπάρχουν καταχωρήσεις που μπορούν να γίνουν για υπηρεσίες από όλο τον κόσμο και που απευθύνονται σε όλο τον κόσμο, αλλά και καταχωρήσεις που απευθύνονται μόνο σε εξειδικευμένες επιχειρήσεις προωθώντας έτσι και το B2B (Business to Business) μοντέλο συνεργασίας. Τέλος υπάρχουν και καταχωρήσεις υπηρεσιών για πιο εξειδικευμένες περιπτώσεις.

Στο παρακάτω σχήμα παρουσιάζονται τα βασικά βήματα για την ανάπτυξη μίας εφαρμογής web services.





Εικόνα 8: Βήματα για την ανάπτυξη μίας εφαρμογής web services. [E8]

Αυτά τα βήματα θα είναι τα ίδια ανεξαρτήτου τεχνολογίας και γλώσσας προγραμματισμού η οποία θα χρησιμοποιηθεί για την υλοποίηση των web services.

### 2.1.3 Τα επίπεδα των web services

Η αρχιτεκτονική των web services αποτελείται από 6 επίπεδα. Κάθε επίπεδο εκτελεί συγκεκριμένες λειτουργίες, τα αποτελέσματα των οποίων αποτελούν είσοδο για τα ανώτερα επίπεδα.

**Web Services Technology Stack**

Layer Description	Implementation(s)	Other Concerns		
Standard Messaging	Electronic Business XML Initiative (ebXML)	Quality of Service	Management	Security
Service Composition	Business Process Execution Service for Web Services (BPEL4WS)			
Service Registry	Universal Description, Discovery and Integration (UDDI) ebXML Registries			
Service Description	Web Services Description Language (WSDL)			
Service Messaging	Simple Object Access Protocol (SOAP)/Extensible Markup Language (XML)			
Service Transport	Hypertext Transfer Protocol (HTTP) Simple Mail Transfer Protocol (SMTP) File Transfer Protocol (FTP)			

Εικόνα 9: Στρώμα Υπηρεσίας Διαδικτύου [E9]



### ***Service Transport (Επίπεδο Μεταφοράς)***

Η κυριότερη λειτουργία του επιπέδου μεταφοράς είναι να μεταφέρει δεδομένα από ένα μηχάνημα σε ένα άλλο. Τα web services χρησιμοποιούν διάφορα πρωτόκολλα για να μεταφέρουν τα δεδομένα από υπηρεσία σε υπηρεσία, συμπεριλαμβανομένων των HTTP, SMTP, FTP και άλλων. Το πιο διαδεδομένο πρωτόκολλο, μέχρι στιγμής, είναι το HTTP. Χρησιμοποιείται περισσότερο λόγω της ευχρηστίας του (δεν απαιτεί ρυθμίσεις για να περάσει από τα firewalls), αλλά και της ασφάλειάς του (μέσω της χρήσης SSL) και αποτελεί το πρότυπο για διαλειτουργικά συστήματα.

### ***Service Messaging (Επίπεδο Μηνυμάτων)***

Το επίπεδο μηνυμάτων περιγράφει σε τι μορφή θα είναι τα δεδομένα ώστε να μπορούν να μεταφερθούν από τη μια υπηρεσία στην άλλη. Για το σκοπό αυτό χρησιμοποιούνται XML αρχεία τύπου SOAP. Τα δεδομένα περιγράφουν τον εαυτό τους κι αυτό γιατί έχουν ειδικές ετικέτες που τους δίνουν ένα όνομα καθώς και την θέση τους μέσα στη δενδρική μορφή του αρχείου. Το SOAP μήνυμα έχει τρεις περιοχές: τον φάκελο, την επικεφαλίδα και το σώμα.

### ***Service Description (Επίπεδο Περιγραφής)***

Το επίπεδο περιγραφής καθορίζει τις τρεις πλευρές του web service: τις μεθόδους που διαθέτει, τα μηνύματα που δέχεται και το πρωτόκολλο το, οποίο πρέπει να χρησιμοποιήσει ο πελάτης για να καλέσει το web service. Τα παραπάνω περιγράφονται σε μια γλώσσα περιγραφής, την WSDL (Web Services Description Language) και καθορίζουν την σύμβαση της υπηρεσίας.

Η σύμβαση υπηρεσίας είναι μια περιγραφή ενός συνόλου από τελικά σημεία (endpoints), στα οποία στέλνονται ή λαμβάνονται μηνύματα, μαζί με μια προδιαγραφή για το πώς πρέπει να διαμορφωθεί ένα XML μήνυμα για να σταλεί στο τελικό σημείο. Το τελικό σημείο (endpoint) είναι μια διεύθυνση στο διαδίκτυο, η οποία δέχεται τα ειδικά διαμορφωμένα μηνύματα σύμφωνα με τη προδιαγραφή που καθορίζεται στην WSDL. Η WSDL χρησιμοποιεί τον όρο *port* για να περιγράψει το τελικό σημείο το οποίο θα δεχτεί το μήνυμα.

### ***Service Registry (Κατάλογος Υπηρεσιών)***

Το επίπεδο αυτό αφορά την καταχώρηση, διαφήμιση και δυναμική εύρεση παρεχόμενων υπηρεσιών μέσω αποθηκών, όπως το UDDI (Universal Description, Discovery and Integration). Ο παροχέας υπηρεσίας χρησιμοποιεί την UDDI αποθήκη για να δημοσιεύσει πληροφορίες



σχετικά με τις υπηρεσίες που παρέχει και ο καταναλωτής μιας υπηρεσίας στέλνει ειδικά διαμορφωμένα XML μηνύματα προς την αποθήκη υπηρεσιών, για να παραλάβει μια λίστα από υπηρεσίες που ταιριάζουν στα κριτήριά του.

### ***Service Composition (Σύνθεση Υπηρεσιών)***

Η σύνθεση υπηρεσιών αναφέρεται στη δυνατότητα συνδυασμού υπηρεσιών σε μια ροή εργασίας, με σκοπό την παροχή μιας πιο σύνθετης υπηρεσίας (ενορχήστρωση και χορογραφία υπηρεσιών). Για την περιγραφή αυτού του επιχειρηματικού διαδικαστικού διαγράμματος χρησιμοποιούνται γλώσσες, όπως η WSFL (Web Services Flow Language), η XLANG (επέκταση της WSDL βασισμένη σε XML) ή, πιο πρόσφατα, η BPEL4WS (Business Process Execution Language for Web Services).

### ***Standard Messaging (Πρότυπη Μηνυματοδότηση)***

Το επίπεδο αυτό αφορά τεχνολογίες που σχετίζονται με την ανάκτηση business scenarios και την ολοκλήρωση του η-επιχειρείν. Μία από τις σημαντικότερες προσπάθειες για την κατάρτιση ενός κοινού πλαισίου ολοκλήρωσης είναι η ebXML (Electronic Business using eXtensible Markup Language), η οποία επεκτείνει το πρωτόκολλο SOAP με την προσθήκη χαρακτηριστικών απαραίτητων για την ανταλλαγή εταιρικών εγγράφων, όπως η ασφάλεια, η πιστοποίηση και η μη άρνηση παραλαβής (non-repudiation).

## **2.1.4 Τύποι Υπηρεσιών Διαδικτύου**

Τα web services χωρίζονται σε δύο κύριες κατηγορίες, ανάλογα με την πολυπλοκότητά τους.

### **Απλά ή Τύπου I web services**

Είναι τα web services που υποστηρίζουν μόνο απλές λειτουργίες, τύπου αίτημα – απόκριση. Στην κατηγορία αυτή το web service περιμένει ένα αίτημα. Αφού το λάβει, το επεξεργάζεται και στέλνει πίσω την απόκριση. Οι υπηρεσίες αυτού του τύπου υλοποιούν μια εργασία, χωρίς να αποθηκεύουν στην μνήμη τους την ροή δραστηριοτήτων μεταξύ διαδοχικών αιτημάτων. Οι υπηρεσίες Τύπου I ονομάζονται και Βασικές Υπηρεσίες (Foundation Services).

### **Περίπλοκα ή Τύπου II web services**

Είναι τα web services που υλοποιούν κάποιας μορφής συντονισμό μεταξύ των εισερχόμενων και εξερχόμενων εργασιών. Οι υπηρεσίες αυτού του τύπου έχουν αυξημένη λειτουργικότητα

και συνήθως χρησιμοποιούν άλλες υπηρεσίες Τύπου I ή Τύπου II, προκειμένου να συλλέξουν την απαραίτητη πληροφορία και να ολοκληρώσουν τον σκοπό χρήσης τους.

Οι υπηρεσίες Τύπου II, μπορούν να διαχωριστούν περαιτέρω στις παρακάτω κατηγορίες, ανάλογα με τον σκοπό και το εύρος της λειτουργικότητάς τους:

- **Επιχειρησιακές Υπηρεσίες (Business Services):** Είναι οι υπηρεσίες με τον υψηλότερο βαθμό λειτουργικότητας. Οι υπηρεσίες αυτές συνθέτουν την λειτουργικότητα τους από άλλες υπηρεσίες του επιπέδου αυτού, με χαμηλότερη λειτουργικότητα. Οι υπηρεσίες αυτές προσφέρουν επιχειρηματική λειτουργικότητα που μπορεί να χρησιμοποιηθεί από πολλές διαδικασίες και έτσι χρησιμοποιούνται από πολλές εφαρμογές, εξασφαλίζοντας συνέπεια στην συμπεριφορά τους.
- **Τοπικές Υπηρεσίες (Domain Services):** Είναι υπηρεσίες με μεσαίου βαθμού λειτουργικότητα, οι οποίες αφορούν διάφορα domains του συστήματος και χρησιμοποιούνται εντός των περιοχών αυτών.
- **Υπηρεσίες Ολοκλήρωσης (Integration Services):** Είναι υπηρεσίες με μεσαίου και χαμηλού βαθμού λειτουργικότητα, η οποία εξαρτάται από την φύση των εφαρμογών που τις χρησιμοποιούν. Φροντίζουν για την συνεπή προβολή δεδομένων ανάμεσα στις εφαρμογές που ολοκληρώνονται. Ουσιαστικά συλλέγουν δεδομένα που είναι κοινά και στα διάφορα συστήματα, ώστε η πληροφορία που παρουσιάζεται σε καθένα από αυτά να είναι πλήρης.
- **Εξωτερικές Υπηρεσίες (External Services):** Είναι και αυτές υπηρεσίες με μεσαίου και χαμηλού βαθμού λειτουργικότητα, η οποία εξαρτάται από την φύση των εφαρμογών που τις χρησιμοποιούν. Οι υπηρεσίες αυτές διασφαλίζουν την επικοινωνία του συστήματος με εφαρμογές που βρίσκονται στο εξωτερικό περιβάλλον του οργανισμού.
- **Υπηρεσίες Υποστήριξης (Utility Services):** Είναι υπηρεσίες με χαμηλού βαθμού λειτουργικότητα και παρέχουν βασική λειτουργικότητα που είναι σημαντική για αρκετές διαδικασίες.

#### 2.1.5 Ασφάλεια στις Υπηρεσίες Διαδικτύου

Οι τρεις βασικές τεχνολογίες των web services SOAP, WSDL, και UDDI μαζί με την XML, στην οποία βασίζονται, παρέχουν μόνο τις βασικές λειτουργίες των web services. Άλλα



χαρακτηριστικά, τα οποία είναι αναγκαία για μία ολοκληρωμένη καταναμεμημένη τεχνολογία, είναι η αξιοπιστία (reliability), η ασφάλεια (security) και οι συναλλαγές (transactions). Η παροχή αυτών των υπηρεσιών στο περιβάλλον των web services είναι ευθύνη άλλων προδιαγραφών, τις οποίες αναφέρουμε παρακάτω.

### **Web services και ασφάλεια**

Τα web services παρέχουν πολλά πλεονεκτήματα στις εφαρμογές, αλλά εκθέτουν σημαντικούς νέους κινδύνους στην ασφάλεια. Η δημιουργία και η διατήρηση ενός ασφαλούς περιβάλλοντος περιλαμβάνει και τη διαχείριση διαφόρων μηχανισμών για το διαδίκτυο, την XML και τα ίδια τα web services. Η γενική αντιμετώπιση αφορά τα παρακάτω:

- Ασφάλεια σε επίπεδο μεταφοράς, όπως τα τείχη προστασίας (firewalls), τα ιδεατά ιδιωτικά δίκτυα (VPNs), η επικύρωση (authentication), η μη άρνηση παραλαβής (non-repudiation) και η κρυπτογράφηση (encryption).
- Ασφάλεια σε επίπεδο μηνύματος, όπως τα στοιχεία επικύρωσης (authentication tokens) για να επικυρώσουν την ταυτότητα του αιτούντα και οι δηλώσεις έγκρισης (authorization assertions) για να ελέγξουν την πρόσβαση στις υπηρεσίες του παρόχου.
- Ασφάλεια σε επίπεδο δεδομένων, όπως είναι η κρυπτογράφηση (encryption) και η ηλεκτρονική υπογραφή (digital signature) για την προστασία ενάντια στις αθέμιτες αλλαγές των δεδομένων.
- Ασφάλεια σε επίπεδο περιβάλλοντος, όπως είναι η διαχείριση (management), η αναγραφή (logging) και ο έλεγχος (auditing) για την αναγνώριση των προβλημάτων που πρέπει να διορθωθούν.

Για την αντιμετώπιση όλων των παραπάνω δημιουργήθηκε μια ομάδα από προδιαγραφές, οι κυριότερες εκ των οποίων είναι οι παρακάτω:

- WS-Security : καθορίζει μια αρχιτεκτονική για ασφαλή επικοινωνία.
- WS-Policy : και οι σχετικές με αυτήν προδιαγραφές καθορίζουν μια πολιτική από κανόνες για το πώς οι υπηρεσίες θα αλληλεπιδρούν μεταξύ τους.
- WS-Trust : καθορίζει το μοντέλο εμπιστοσύνης για ασφαλείς συναλλαγές.
- WS-Privacy : καθορίζει πώς τηρείται η ιδιωτικότητα στις πληροφορίες.
- WS-Secure Conversation : καθορίζει πώς θα επιτευχθεί μια ασφαλή συνεδρία μεταξύ υπηρεσιών που ανταλλάσσουν δεδομένα με κανόνες ορισμένους στα WS-Policy, WS-Trust, WS-Privacy.

- WS-Federation : καθορίζει τους κανόνες σχετικά με την ταυτότητα σε κατακευματισμένο περιβάλλον.
- WS-Authorization : χειρίζεται την επεξεργασία για την επικύρωση που αφορά στην πρόσβαση και την ανταλλαγή δεδομένων.

Όλα τα παραπάνω χρησιμοποιούν το SOAP Header και έτσι εκμεταλλεύονται στο έπακρο την επεκτασιμότητα του πρωτοκόλλου SOAP.

### **Web services και μηνυματοδότηση (messaging)**

Υπάρχει ακόμη μία ομάδα προδιαγραφών που πραγματεύεται ζητήματα που έχουν να κάνουν με την αξιόπιστη αποστολή μηνυμάτων (reliable messaging) και τις ειδοποιήσεις (notification):

- WS-ReliableMessaging και WS-Reliability: καθορίζουν μηχανισμούς για την εξασφάλιση της παράδοσης των μηνυμάτων με τη σωστή σειρά και την εξάλειψη των διπλότυπων.
- WS-Eventing και WS-Notification: παρέχουν ένα μηχανισμό σχετικά με τα γεγονότα (events), ώστε οι συνδρομητές να ειδοποιούνται ανεξάρτητα από τη σχέση μεταξύ του παρόχου και του αιτούντα.

Και οι παραπάνω τεχνολογίες χρησιμοποιούν το SOAP Header.

### **Web services και συναλλαγές (transactions)**

Οι συναλλαγές αποτελούν ένα πολύ σημαντικό χαρακτηριστικό, το οποίο έχει αποδειχθεί πολύ χρήσιμο στο πέρασμα των χρόνων.

Οι συναλλαγές στο περιβάλλον των web services μας εξασφαλίζουν ότι μία ομάδα από web services πετυχαίνει ένα κοινό αποτέλεσμα. Τα web services συχνά εξαρτώνται το ένα από το άλλο για να ολοκληρώσουν πολύπλοκες αιτήσεις σε μία εφαρμογή, όπως η ενημέρωση μιας εγγραφής πελάτη (η οποία μπορεί να ενημερώνει πολλαπλές βάσεις δεδομένων με στοιχεία πελάτη) ή η επεξεργασία μίας παραγγελίας (η οποία μπορεί να ενημερώνει πολλαπλές βάσεις δεδομένων αποθηκών).

Οι πιο σημαντικές προσπάθειες που σχετίζονται με τις συναλλαγές στο περιβάλλον των web services είναι οι παρακάτω:

- WS-Transaction: μία οικογένεια προδιαγραφών που περιλαμβάνει:
  - WS-AtomicTransactions (WS-AT): Ένα πρωτόκολλο που ενεργεί σε δύο φάσεις (two-phase commit) για διαλειτουργικότητα των web services.

- WS-BusinessActivity (WS-BA): Ένα ανοικτό ενσωματωμένο πρωτόκολλο για μακροπρόθεσμες επιχειρησιακές διαδικασίες.
- WS-Coordination (WS-C): Ένα πλαίσιο συντονισμού που υποστηρίζει τα WS-AT και WS-C.
- WS-Composite Application Framework: μία οικογένεια προδιαγραφών που περιλαμβάνει:
  - WS-Context (WS-CTX): Ένας γενικός μηχανισμός διαχείρισης πλαισίου.
  - WS-CoordinationFramework (WS-CF): Ένα πλαίσιο συντονισμού που υποστηρίζει τα WS-AT, WS-BA, και τα τρία πρωτόκολλα στο WS-TXM.
  - WS-TransactionManagement (WS-TXM): Ένα πρωτόκολλο που ενεργεί σε δύο φάσεις, για διαλειτουργικότητα των Web Services (ACID), ένα πρωτόκολλο βασισμένο στη διόρθωση (compensation) για μακροπρόθεσμες διαδικασίες (LRA), και ένα πρωτόκολλο διαχείρισης επιχειρησιακών διαδικασιών (Business Processes)

## 2.2 Υπηρεσιοστρεφής Αρχιτεκτονική

Η μεγαλύτερη διαφορά μεταξύ της προσανατολισμένης στις υπηρεσίες αρχιτεκτονικής και των προηγούμενων προσεγγίσεων, είναι ότι η εφαρμογή της αρχιτεκτονικής σχεδιασμού σε σύγκριση με το SOA ήταν στενά συνδεδεμένη στα συγκεκριμένα περιβάλλοντα εκτέλεσης και η επαναχρησιμοποίηση περιοριζόταν συνήθως μέσα στα όρια του τμήματος IT. Με τη SOA ο στόχος της επαναχρησιμοποίησης έχει επεκταθεί πέρα από τα ετερογενή περιβάλλοντα εκτέλεσης, ξεπέρασε τα παραδοσιακά όρια του τμήματος IT και έφθασε στους τελικούς χρήστες.

### 2.2.1 Τι είναι η SOA

Από το όνομά της μπορούμε να πούμε, καταρχήν, ότι πρόκειται για μια αρχιτεκτονική λογισμικού, μια περιγραφή, δηλαδή, ενός συστήματος λογισμικού, από την άποψη των σημαντικών συστατικών του, των σχέσεών τους και των πληροφοριών που περνούν μεταξύ τους, προκειμένου να προσφέρουν την απαιτούμενη λειτουργικότητα ή υπηρεσία. Η διαφορά της σε σχέση με παλαιότερες αρχιτεκτονικές είναι ότι τα επιμέρους τμήματα δεν χαρακτηρίζονται από την τεχνολογική τους συνεισφορά στη λύση, αλλά από τη σημασία τους για την επιχείρηση. Η SOA αποτελεί μια εύκαμπτη αρχιτεκτονική πάνω στην οποία μπορούν να σχεδιαστούν εφαρμογές χρησιμοποιώντας μια συλλογή από επαναχρησιμοποιήσιμα

λειτουργικά τμήματα με ευκρινής διεπαφές. Οι εφαρμογές ολοκληρώνονται στις διεπαφές και όχι στο επίπεδο της υλοποίησης.

Η υπηρεσιοστρεφής αρχιτεκτονική αντιμετωπίζει το λογισμικό ως σύνολο υπηρεσιών. Βασίζεται στην απλή έννοια ενός εντελώς ανοιχτού περιβάλλοντος στο οποίο χρήστες υπηρεσιών ή πελάτες επικοινωνούν με εξυπηρετητές ή παρόχους υπηρεσιών, προκειμένου να αποκτήσουν πρόσβαση σε ένα σύνολο υπηρεσιών. Σύμφωνα με την προσέγγιση της IBM, οι υπηρεσίες της αρχιτεκτονικής SOA είναι οι εξής:

- Υπηρεσίες πληροφοριών, οι οποίες παρέχουν τις απαιτούμενες δυνατότητες για την ενοποίηση, αναπαραγωγή και μετατροπή από διαφορετικές πηγές δεδομένων.
- Υπηρεσίες διαδικασιών, οι οποίες παρέχουν τις απαιτούμενες υπηρεσίες ελέγχου για τη διαχείριση της ροής και της επικοινωνίας πολλαπλών υπηρεσιών με έναν τρόπο που να υλοποιεί μια επιχειρηματική διαδικασία.
- Υπηρεσίες εφαρμογών, οι οποίες προσφέρουν τις απαιτούμενες υπηρεσίες εκτέλεσης για νέες εφαρμογές και νέα επιχειρηματική λογική.
- Υπηρεσίες επικοινωνίας με χρήστες, στις οποίες περιλαμβάνονται οι απαιτούμενες δυνατότητες για την παράδοση των λειτουργικών και των δεδομένων πληροφορικής στους τελικούς χρήστες και τέλος
- Υπηρεσίες ενοποίησης κοινότητας, οι οποίες παρέχουν τις απαιτούμενες υπηρεσίες διαχείρισης εγγράφων, πρωτοκόλλων και συνεργατών για την αποτελεσματική υλοποίηση των επικοινωνιών B2B (business-to-business).

Κατά την υλοποίηση υπηρεσιοστρεφούς αρχιτεκτονικής, είτε δημιουργούμε εφαρμογές που χρησιμοποιούν υπηρεσίες (services), είτε εφαρμογές που λειτουργούν σαν υπηρεσίες προς όφελος άλλων εφαρμογών, είτε συνδυασμό και των δυο. Δηλαδή εφαρμογές που χρησιμοποιούν υπηρεσίες και αυτές με την σειρά τους αποτελούν υπηρεσίες για άλλες εφαρμογές.

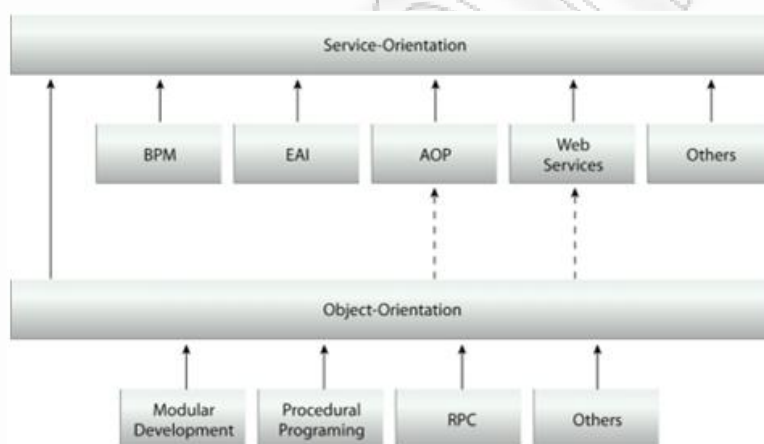
Αυτό που ξεχωρίζει την αρχιτεκτονική SOA από άλλες είναι το γεγονός ότι ο πελάτης (client) μιας υπηρεσίας είναι ουσιαστικά ανεξάρτητος από την ίδια την υπηρεσία. Ο τρόπος με τον οποίο ένα πρόγραμμα πελάτης (ή μια άλλη υπηρεσία) καλεί μια υπηρεσία δεν σχετίζεται με την υλοποίηση της υπηρεσίας αυτής. Πρακτικά αυτό σημαίνει ότι ο client δεν χρειάζεται να έχει καμία γνώση για την υπηρεσία που χρησιμοποιεί. Για παράδειγμα δεν χρειάζεται να ξέρει σε τι γλωσσά είναι γραμμένη η υπηρεσία ή σε τι λειτουργικό σύστημα τρέχει. Ο πελάτης επικοινωνεί με την υπηρεσία με τη βοήθεια μιας καθορισμένης και πολύ καλά ορισμένης



διεπαφής (interface) και από κει και πέρα αφήνει την ίδια την υπηρεσία να κάνει την όποια επεξεργασία απαιτείται. Αν η υλοποίηση της υπηρεσίας διαφοροποιηθεί χωρίς να υπάρξει διαφοροποίηση στο interface, τότε ένας πελάτης εξακολουθεί να καλεί την συγκεκριμένη υπηρεσία χωρίς κανένα πρόβλημα.

### 2.2.2 Προέλευση της υπηρεσιοστρεφούς αρχιτεκτονικής

Ο καλύτερος τρόπος για να καταλάβουμε αυτό το αρχιτεκτονικό ύφος είναι να γνωρίσουμε την ιστορία του. Η υπηρεσιοστρεφής αρχιτεκτονική προέκυψε «από το πουθενά». Αναπαριστά, ουσιαστικά, την εξέλιξη του IT και ως εκ τούτου βασίζεται σε προηγούμενες τεχνολογίες. Παράλληλα βρίσκεται και η ίδια σε μια εξελισσόμενη κατάσταση και έτσι επηρεάζεται από τις τρέχουσες τάσεις.



Εικόνα 10: Προέλευση και επιρροές της SOA <sup>[E10]</sup>

Παρακάτω αναφέρουμε κάποιες από τις πιο εμφανείς επιρροές της υπηρεσιοστρεφούς αρχιτεκτονικής:

#### Object Orientation

Στη δεκαετία του 1990 η κοινότητα του IT ενστερνίστηκε μια σχεδιαστική φιλοσοφία που άνοιξε το δρόμο για την υλοποίηση καταναμημένων λύσεων. Η φιλοσοφία αυτή ήταν ο αντικειμενοστραφής σχεδιασμός των εφαρμογών και ήρθε με το δικό της σύνολο αρχών, η εφαρμογή του οποίου συνέβαλε στη διασφάλιση της συνέπειας μεταξύ πολυάριθμων περιβαλλόντων. Οι αρχές αυτές όριζαν ένα συγκεκριμένο τύπο σχέσης μεταξύ των μονάδων της λογικής λύσης, που αναφέρονται ως αντικείμενα.

Ο προσανατολισμός σε υπηρεσίες συγκρίνεται δικαίως με τον προσανατολισμό σε αντικείμενα, αφού οι αρχές και τα πρότυπα πίσω από την αντικειμενοστρεφή ανάλυση αντιπροσωπεύουν μία από τις σημαντικότερες πηγές έμπνευσης της υπηρεσιοστρεφούς

προσέγγισης. Στην πραγματικότητα, ένα υποσύνολο των αρχών της υπηρεσιοστρεφούς αρχιτεκτονικής (επαναχρησιμοποίηση, αφαιρετικότητα, σύνθεση υπηρεσιών και άλλα) μπορούν άμεσα να αναχθούν στους αντικειμενοστρεφείς ομολόγους τους.

### **Web services**

Παρόλο που η υπηρεσιοστρεφής αρχιτεκτονική είναι ουδέτερης υλοποίησης, είναι κοινή η σύνδεσή της με τις Υπηρεσίες Ιστού. Οι περισσότεροι πάροχοι SOA λύσεων έχουν διαμορφώσει τις πλατφόρμες τους γύρω από την αξιοποίηση των τεχνολογιών των υπηρεσιών παγκόσμιου ιστού. Έτσι, το πλαίσιο των web services ενέπνευσε και προώθησε μια σειρά αρχών στην υπηρεσιοστρεφή αρχιτεκτονική, συμπεριλαμβανομένων της αφαιρετικότητας των υπηρεσιών, τη χαλαρή συνδεσιμότητα και την σύνθεση υπηρεσιών.

### **Business Process Management (BPM)**

Η διαχείριση επιχειρηματικών διαδικασιών δίνει ιδιαίτερη έμφαση στις επιχειρηματικές διαδικασίες μέσα στην επιχείρηση, τόσο από την άποψη του εξορθολογισμού της διαδικασίας, ώστε να βελτιωθεί η αποτελεσματικότητα, όσο και στο να θεσπίσει διαδικασίες που είναι προσαρμόσιμες και επεκτάσιμες, για να μπορούν να ανταποκριθούν σε μελλοντικές αλλαγές των επιχειρήσεων.

Το στρώμα των επιχειρηματικών διαδικασιών αποτελεί βασικό τμήμα κάθε service-oriented αρχιτεκτονικής. Από την άποψη της σύνθεσης υπηρεσιών, αναλαμβάνει το ρόλο του ελεγκτή και από την άποψη της εφαρμογής, αποτελεί την τεχνολογία ενορχήστρωσης. Ο πρωταρχικός στόχος του service-orientation είναι να θεσπιστεί ένα εξαιρετικά ευέλικτο περιβάλλον αυτοματοποίησης, με πλήρη ικανότητα προσαρμογής στις αλλαγές. Ο στόχος αυτός μπορεί να υλοποιηθεί με το να αφαιρεθεί η επιχειρηματική λογική διαδικασία στο δικό της επίπεδο, ελευθερώνοντας, έτσι, τις άλλες υπηρεσίες από την υποχρέωση να ενσωματώνουν κατ'επανάληψη την λογική διαδικασία.

Ενώ η ίδια η υπηρεσιοστρεφής προσέγγιση δεν ασχολείται τόσο με τη διαδικασία της επιχειρηματικής αναδιοργάνωσης, υποστηρίζει πλήρως τη βελτιστοποίηση των διαδικασιών ως πρωταρχική πηγή των αλλαγών που θα οδηγήσουν στην ανασύνθεση των υπηρεσιών.

### **Enterprise Application Integration (EAI)**

Η ενσωμάτωση των επιχειρηματικών διαδικασιών αποτέλεσε πρωταρχικό σημείο εστίασης στα τέλη της δεκαετίας του 90 και πολλές επιχειρήσεις ήταν καλά προετοιμασμένες για αυτό.



Τα περισσότερα συστήματα είχαν δημιουργηθεί χωρίς να δώσουν σημασία στο πως τα δεδομένα των εφαρμογών θα μπορούσαν να διαμοιραστούν εκτός των ορίων του συστήματος. Αυτό είχε ως αποτέλεσμα να δημιουργούνται κανάλια ολοκλήρωσης, σημείο προς σημείο, κάθε φορά που προέκυπτε ανάγκη ανταλλαγής δεδομένων, γεγονός που οδηγούσε στα γνωστά προβλήματα έλλειψης σταθερότητας, επεκτασιμότητας, καθώς και ανεπαρκών πλαισίων διαλειτουργικότητας.

Οι EAI πλατφόρμες εισήγαγαν την έννοια του ενδιάμεσου λογισμικού(middleware) που επέτρεπε την χρήση των εφαρμογών ενός συστήματος, μέσω προσαρμογέων(adapters), μεσολαβητών(brokers), και μηχανών ενορχήστρωσης(orchestration engines). Η προκύπτουσα αρχιτεκτονική ολοκλήρωσης ήταν, στην πραγματικότητα, πιο ισχυρή και επεκτάσιμη.

Πολλές καινοτομίες που έγιναν γνωστές με την EAI έχουν χρησιμοποιηθεί για να επιτευχθούν οι γενικοί στόχοι που συνδέονται με την δημιουργία SOA αρχιτεκτονικών με τη χρήση υπηρεσιών Ιστού. Ένα παράδειγμα αυτών είναι η χρήση του μεσολαβητή (broker), η οποία επιτρέπει σε υπηρεσίες που χρησιμοποιούν διαφορετικά σχήματα για το ίδιο είδος δεδομένων να εξακολουθούν να επικοινωνούν μέσω runtime μετασχηματισμών. Ένα άλλο παράδειγμα είναι η μηχανή ενορχήστρωσης, η οποία μπορεί να χρησιμοποιηθεί έτσι, ώστε να αντιπροσωπεύει ένα πλήρες στρώμα υπηρεσιών στα πλαίσια μεγαλύτερων υπηρεσιοστρεφών υλοποιήσεων. Αυτά τα στοιχεία της EAI πλατφόρμας υποστηρίζουν αρκετές αρχές της υπηρεσιοστρεφούς αρχιτεκτονικής, όπως η αφαιρετικότητα(abstraction), η απουσία κατάστασης(statelessness), η χαλαρή συνδεσιμότητα(loose coupling) και σύνθεση(composition) υπηρεσιών.

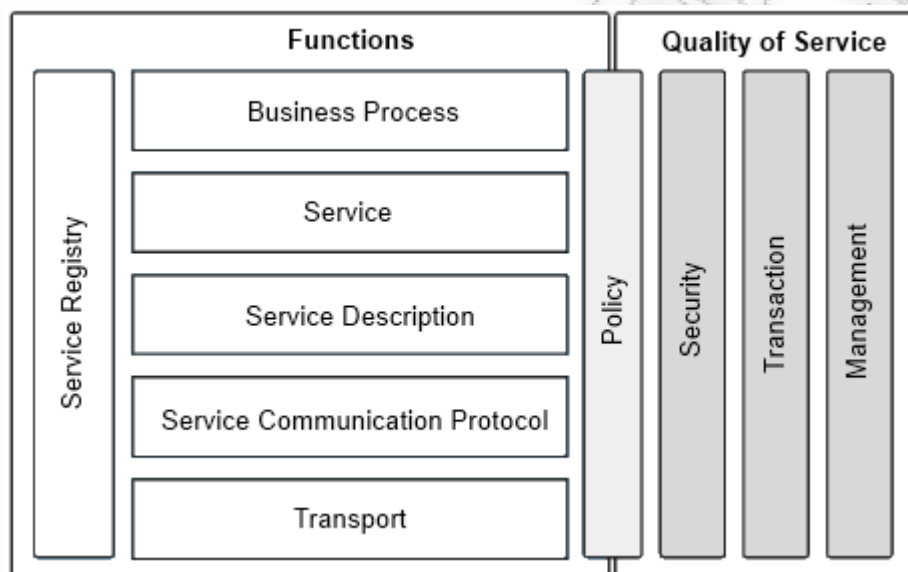
### **Aspect-Oriented Programming**

Ο προσανατολισμός σε περιπτώσεις(aspect orientation) προέκυψε από τον προσανατολισμό σε αντικείμενα(object orientation) βασιζόμενος στους αρχικούς στόχους για την δημιουργία επαναχρησιμοποιήσιμων αντικειμένων. Αν και δεν αποτελεί πρωταρχικό παράγοντα επιρροής του προσανατολισμού σε υπηρεσίες, ο AOP αποδεικνύει έναν κοινό στόχο, τονίζοντας τη σημασία δημιουργίες μονάδων λογικής (solution logic units), οι οποίες είναι μη ενήμερες(agnostic) των επιχειρηματικών διαδικασιών και εφαρμογών και ως εκ τούτου άκρως επαναχρησιμοποιήσιμες.

### 2.2.3 Η στοίβα της SOA

Η αρχιτεκτονική SOA παρουσιάζει έναν τρόπο δημιουργίας συστημάτων, στα όποια παντός είδους εφαρμογή και λειτουργικότητα παρουσιάζεται σαν υπηρεσία προς τον τελικό χρήστη ή προς κάποια άλλη εφαρμογή. Αποτελείται από στοιχεία τα όποια μπορούν να χωριστούν σε δυο κατηγορίες: στα λειτουργικά (functional) στοιχεία και σε αυτά που αφορούν την ποιότητα της υπηρεσίας (Quality of Service).

Στο παρακάτω σχήμα βλέπουμε τα στοιχεία που είναι δυνατόν να παρατηρηθούν σε μια τέτοια αρχιτεκτονική με τη μορφή στοίβας.



Εικόνα 11: Η στοίβα της SOA <sup>[E11]</sup>

#### Functions

- **Transport:** είναι ο μηχανισμός που είναι υπεύθυνος για τη μεταφορά της αίτησης μιας υπηρεσίας από τον καταναλωτή προς τον πάροχο και βέβαια της απόκρισης της υπηρεσίας προς την αντίθετη κατεύθυνση.
- **Service communication protocol:** είναι ο προσυμφωνημένος μηχανισμός που οι δυο πλευρές χρησιμοποιούν για να επικοινωνήσουν. Παράδειγμα τέτοιου πρωτοκόλλου αποτελεί το SOAP.
- **Service Description:** αποτελεί μια διαγραμματική παράσταση που περιγράφει τι ακριβώς είναι η εκάστοτε υπηρεσία, πως ενεργοποιείται και τι δεδομένα απαιτούνται για τη σωστή χρήση της. Χαρακτηριστικό παράδειγμα αποτελεί η WSDL.
- **Service:** περιγράφει την ίδια την υπηρεσία που έχει φτιαχτεί και δημοσιευθεί προς χρήση.

- **Business Process:** είναι ένα σύνολο υπηρεσιών που ενεργοποιείται με μια συγκεκριμένη σειρά, ακολουθώντας συγκεκριμένους κανόνες και εξυπηρετεί πολύπλοκες εταιρικές προδιαγραφές. Αξίζει να σημειωθεί ότι business process μπορεί να θεωρηθεί και ένα service από μόνο του, πράγμα που μας οδηγεί στο συμπέρασμα ότι οι business processes μπορούν να δημιουργούνται από ετερογενή services.
- **Service Registry:** είναι μια αποθήκη που περιέχει περιγραφές υπηρεσιών, οι οποίες μπορούν να χρησιμοποιηθούν από requestors ή providers για τη χρήση ή τη δημοσίευση υπηρεσιών αντίστοιχα.

#### Quality of Service

- **Policy:** είναι το σύνολο των κανόνων και των συνθηκών, σύμφωνα με τα όποια ένας πάροχος διαθέτει τις υπηρεσίες του προς τους καταναλωτές. Υπάρχουν θέματα πολιτικής που σχετίζονται με την λειτουργικότητα και θέματα πολιτικής που σχετίζονται με την ποιότητα των υπηρεσιών. Γι' αυτό στο σχήμα η πολιτική είναι τοποθετημένη στο ενδιάμεσο.
- **Security:** αναφέρεται στο σύνολο των κανόνων που σχετίζονται με την αναγνώριση, την εξουσιοδότηση και την προσβασιμότητα αυτών που προσπαθούν να ενεργοποιήσουν μια υπηρεσία. Οι κανόνες αυτοί καθορίζονται από τον πάροχο της υπηρεσίας.
- **Transaction:** αφορά το σύνολο των χαρακτηριστικών που πρέπει να εφαρμοσθούν σε ένα σύνολο υπηρεσιών, ώστε να έχουμε ένα εγγυημένα καλό αποτέλεσμα. Για παράδειγμα, στην περίπτωση που τρεις υπηρεσίες συνθέτουν μια επιχειρησιακή διαδικασία πρέπει να εκτελεστούν με επιτυχία και οι τρεις ή να μην εκτελεστεί καμία.
- **Management:** είναι το σύνολο των κανόνων που μπορεί να εφαρμοστούν για καλή διαχείριση των υπηρεσιών που προσφέρονται και καταναλώνονται.

#### 2.2.4. Πλεονεκτήματα της SOA

Στο σημείο αυτό είναι σκόπιμο να παρουσιάσουμε όλα εκείνα τα χαρακτηριστικά της SOA, τα οποία την καθιστούν μια πολύ ελκυστική μέθοδο ανάπτυξης λογισμικού.

**Επαναχρησιμοποίηση ( reusability ).** Η πρώτη σημαντική παράμετρος που δίνει μεγάλη ώθηση στην προτίμηση της SOA είναι η επαναχρησιμοποίηση πολλών διαδικασιών. Στα πλαίσια ενός project ή μεταξύ διαφορετικών, ένας προγραμματιστής έχει την δυνατότητα να πάρει έτοιμο κώδικα από άλλες εφαρμογές, να τον μετατρέψει σε υπηρεσία και στην συνέχεια να κάνει χρήση της υπηρεσίας αυτής για να υποστηρίξει τις δικές του ανάγκες. Η

δυνατότητα επαναχρησιμοποίησης λειτουργικότητας, η όποια είναι ήδη έτοιμη, στα πλαίσια ενός project επιφέρει πολλά οφέλη, όπως η μείωση του προγραμματιστικού φόρτου, του χρόνου ανάπτυξης, αλλά και των χρηματικών δαπανών. Το πλεονέκτημα της επαναχρησιμοποίησης αυξάνεται σημαντικά, καθώς όλο και περισσότερες επιχειρηματικές λειτουργίες ενσωματώνονται σε διάφορες προγραμματιστικές εφαρμογές.

**Διαλειτουργικότητα (interoperability).** Οι υπηρεσίες διαδικτύου περιλαμβάνουν μια ώριμη συλλογή πρωτοκόλλων και τεχνολογιών που είναι ευρύτατα διαδομένες και γενικά αποδεκτές, γεγονός που τις καθιστά ανεξάρτητες πλατφόρμας, συστήματος και γλωσσάς προγραμματισμού. Επιπρόσθετα αυτά τα πρωτόκολλα και οι τεχνολογίες δεν έχουν πρόβλημα με τα firewalls κάτι που δίνει την δυνατότητα στις εταιρίες να χρησιμοποιούν και να μοιράζονται πολύ απλά υγιείς υπηρεσίες.

**Κλιμάκωση (Scalability).** Επειδή οι υπηρεσίες που εντάσσονται στην SOA έχουν μικρή συνεκτικότητα μεταξύ τους, είναι πολύ εύκολο να κλιμακώνονται σε αντίθεση με υπηρεσίες που βρίσκονται σε περισσότερο συνεκτικά περιβάλλοντα. Το γεγονός ότι τα services στην SOA είναι coarse grained, δηλαδή μπορούν να προσφέρουν πολλές σχετιζόμενες μεταξύ τους υπηρεσίες, τους δίνει πλεονέκτημα σε σχέση με τα fine grained. Μια άλλη ιδιότητα των υπηρεσιών είναι ότι μπορεί να είναι document oriented. Δηλαδή η επικοινωνία μεταξύ πελάτη και υπηρεσίας μπορεί να γίνει με την ανταλλαγή ενός εγγράφου (XML στην περίπτωση των web services). Επιπλέον, τα services μπορεί να είναι ασύγχρονα. Αυτό σημαίνει ότι η υπηρεσία δεν χρειάζεται να περιμένει την απόκριση του πελάτη για να συνεχίσει και αντίστροφα. Παρόντων των τριών παραπάνω χαρακτηριστικών βλέπουμε ότι η αλληλεπίδραση ανάμεσα σε εφαρμογή και πελάτη είναι σχετικά περιορισμένη. Έτσι, είναι πιο εύκολο για την εφαρμογή να επεκταθεί, αφού τελικά το φορτίο κίνησης που επιφέρει στο δίκτυο είναι περιορισμένο.

**Ευελιξία (flexibility).** Τα χαλαρά συνδεδεμένα services είναι κατά κανόνα πιο ευέλικτα από τις πιο σφιχτές εφαρμογές. Οι εφαρμογές οι οποίες ακολουθούν την αρχιτεκτονική SOA, ακριβώς επειδή είναι χαμηλής συνοχής, είναι δυνατόν να παραμετροποιηθούν, να αλλάξουν μόνο μικρά κομμάτια κώδικα σε συγκεκριμένα σημεία και όλα αυτά με τον ελάχιστο προγραμματιστικό φόρτο. Έτσι προσαρμόζονται εύκολα στις αλλαγές των προδιαγραφών.

**Κόστος (cost efficiency).** Οποιαδήποτε αλλαγή πρέπει να γίνει σε μια εφαρμογή που ακολουθεί υπηρεσιοστρεφή αρχιτεκτονική, ακολουθεί συγκεκριμένους, γνωστούς κανόνες και όχι προσαρμοσμένες τεχνικές, πολύπλοκες αναλύσεις και εκτεταμένο προγραμματισμό.



Επιπλέον, η επαναχρησιμοποίηση ήδη έτοιμου κώδικα μειώνει ακόμα πιο δραματικά το κόστος.

### 2.2.5 Ασφάλεια σε περιβάλλοντα αρχιτεκτονικής SOA

Η ίδια η λέξη «ασφάλεια» δεν είναι μονοκατευθυντική. Υπάρχουν πολλαπλές πτυχές στις οποίες αναφερόμαστε όταν μιλάμε για την ασφάλεια στην Τεχνολογία Πληροφοριών (Information Technology - IT). Η διαχείριση ασφάλειας μπορεί να ταξινομηθεί γενικά στην ασφάλεια δικτύων, την ασφάλεια πλατφορμών και την ασφάλεια εφαρμογής. Από αυτούς, τα δύο πρώτα έχουν λίγη ως καμία σχέση με την ασφάλεια στην SOA. Ως εκ τούτου, τα ζητήματα που θα μας απασχολήσουν περιορίζονται σε αυτά που απαιτούνται στην ασφάλεια εφαρμογής.

#### Ζητήματα ασφάλειας

Στη συμβατική σφαίρα ασφάλειας μιας επιχειρηματικής εφαρμογής, υπάρχει η έννοια μιας εμπιστευόμενης βάσης υπολογισμού (trusted computing base - TCB). Η TCB παρέχει τους μηχανισμούς για μια πολιτική ασφαλείας που προστατεύει τους πόρους μέσα στο ελεγχόμενο περιβάλλον. Αυτό είναι ισοδύναμο με την παροχή μιας περιμέτρου ασφαλείας που προστατεύει τους πολύτιμους πόρους.

Οι υπηρεσίες Ιστού δεν έχουν σαφή την έννοια μιας περιμέτρου ασφαλείας. Η αρχιτεκτονική υπηρεσιών Ιστού επιτρέπει τη σύνθεση υπηρεσιών, η οποία μπορεί να δεσμεύσει πολλούς διαφορετικούς φορείς παροχής υπηρεσιών. Σε ένα τέτοιο ανοικτό περιβάλλον, η διάκριση των νόμιμων αιτημάτων των υπηρεσιών από τα παράνομα είναι μια πρόκληση. Η SOA πρέπει να αντιμετωπίσει τα ακόλουθα ζητήματα ασφάλειας:

**Ασφάλεια μηνυμάτων:** Ένα μήνυμα πρέπει να μεταφερθεί μέσω διάφορων μεσαζόντων μέχρι να φτάσει στον τελικό προορισμό. Είναι κρίσιμο να διατηρηθεί η εμπιστευτικότητα των ευαίσθητων πληροφοριών, έτσι ώστε καμία αναρμόδια οντότητα να μην μπορεί να αποκτήσει πρόσβαση σε αυτές. Είναι επίσης σημαντικό να αξιολογηθεί εάν το μήνυμα τροποποιήθηκε ή όχι κατά τη μεταφορά και να εξασφαλιστεί ότι διατηρείται η ακεραιότητά του. Στα περισσότερα από τα B2B σενάρια, είναι σημαντικό να καθιερωθεί η επικύρωση μηνυμάτων, η οποία εγγυάται ότι το μήνυμα δημιουργήθηκε από την ισχυριζόμενη οντότητα, αλλά και η μη άρνηση της αποδοχής τους (non-repudiation). Η αποδοχή εγγυάται ότι ο αποστολέας δεν μπορεί να αποκηρύξει το μήνυμα αργότερα και να υποστηρίξει ότι δεν το έστειλε ποτέ.

**Διαχείριση ταυτότητας:** Ένας πελάτης χρησιμοποιεί μια ταυτότητα (τις περισσότερες φορές είναι μια ταυτότητα χρήστη) για να αποκτήσει πρόσβαση στην υπηρεσία που χρειάζεται. Σε μια αρχιτεκτονική SOA μπορεί να εμπλέκονται πολλαπλές περιοχές ασφάλειας και να υπάρχουν διάφορες ταυτότητες συνδεδεμένες με έναν χρήστη, σε διαφορετικές περιοχές ασφάλειας. Επιπλέον, οι υποδομές ασφάλειας μπορεί να ποικίλουν μεταξύ των διάφορων συστημάτων, έτσι οι χρήστες πρέπει να επικυρωθούν για κάθε σύστημα. Ένα άλλο πρόβλημα συσχετίζεται με το στρώμα σύνθεσης υπηρεσιών, το οποίο δύναται να καλεί πολλές διαφορετικές ατομικές υπηρεσίες που εμπίπτουν σε διαφορετικές περιοχές ασφάλειας. Η απουσία γενικού πλαισίου ασφάλειας καθιστά δύσκολο να συνδεθούν οι πολλαπλές ταυτότητες χρηστών.

**Διανεμημένες πολιτικές:** Οι επιχειρήσεις θέλουν να σιγουρευτούν ότι οποιαδήποτε επικοινωνία με το εξωτερικό περιβάλλον είναι σύμμορφη με τις πολιτικές που έχουν καθιερώσει. Οι πολιτικές ασφαλείας είναι συνδεδεμένες με όλες τις σχετικές οντότητες, τον πελάτη, την υπηρεσία και την διαδικασία ανακάλυψης της υπηρεσίας. Χρησιμοποιούνται για να περιγράψουν μια σειρά απαιτήσεων και ικανοτήτων των υπηρεσιών. Παραδείγματος χάριν, ένας οργανισμός μπορεί να χρησιμοποιήσει μια πολιτική για να καθορίσει μια απαίτηση ασφάλειας, όπως η κρυπτογράφηση. Οι πολιτικές επικυρώνονται κατά την διάρκεια της αλληλεπίδρασης και μέσα στο πλαίσιο της αλληλεπίδρασης.

**Διαλειτουργικότητα:** Αν και η διαλειτουργικότητα δεν αποτελεί ζήτημα ασφάλειας, είναι ένας πολύ σημαντικός παράγοντας στην λειτουργία μιας λύσης SOA. Οι περισσότερες από τις προδιαγραφές υπηρεσιών Ιστού παρέχουν διάφορους μηχανισμούς για να κάνουν το ίδιο πράγμα, οι οποίοι οδηγούν σε προβλήματα διαλειτουργικότητας. Για παράδειγμα, ένας αποστολέας μπορεί να κρυπτογραφήσει τα στοιχεία χρησιμοποιώντας διάφορους τριπλούς αλγορίθμους ( π.χ DES, AES 128, AES 192, ή AES 256) και ο δέκτης να μπορεί να αποκρυπτογραφήσει μόνο με έναν από αυτούς.

### **Πρότυπα και τεχνολογίες ασφάλειας**

Οι οργανισμοί έχουν υπάρχουσες υποδομές ασφάλειας σε ισχύ, οι οποίες προστατεύουν τους πόρους στις διαφορετικές πλατφόρμες. Ο θεμελιώδης στόχος της SOA είναι να χρησιμοποιηθούν αυτές οι υποδομές για να επιτευχθεί η διαλειτουργικότητα. Η λύση είναι να επιτραπεί η επέκταση του στρώματος ασφαλείας πέρα από τις υπάρχουσες υποδομές, μέσω της SOA.





Η SOA επιτρέπει στους οργανισμούς να χτίσουν ένα σύνολο επαναχρησιμοποιήσιμων υπηρεσιών ασφάλειας που μπορούν να χρησιμοποιηθούν από τις επιχειρησιακές εφαρμογές. Η προκύπτουσα ικανότητα επαναχρησιμοποίησης εξασφαλίζει συνεκτικές πολιτικές ασφάλειας πέρα από την πλατφόρμα, με μειωμένες δαπάνες ανάπτυξης. Παραδείγματος χάριν, μια λειτουργία επικύρωσης μπορεί να προσφερθεί ως υπηρεσία.

**Εφαρμογή της ασφάλειας σε επίπεδο μηνυμάτων:** Η ανάγκη για την ασφάλεια σε επίπεδο μηνυμάτων είναι κοινή στα σενάρια ολοκλήρωσης. Στην ολοκλήρωση, υπάρχει συχνά η ανάγκη να συντεθούν οι ικανότητες πολλαπλών υπηρεσιών για να δημιουργήσουν μια υψηλότερου επιπέδου υπηρεσία. Σε μια τέτοια περίπτωση, τα διάφορα μέρη ενός μηνύματος που παραλαμβάνεται από την υψηλού επιπέδου υπηρεσία, μπορούν να υποβληθούν προς επεξεργασία στις διαφορετικές χαμηλότερου επιπέδου υπηρεσίες. Οι υπηρεσίες αυτές μπορεί να μην ανήκουν στην ίδια περιοχή ασφάλειας. Αυτό μεταφράζεται στην ανάγκη τα ίδια ή διαφορετικά μέρη ενός μηνύματος να προστατεύονται διαφορετικά, κατά τη χρήση τους από τα διαφορετικά συμβαλλόμενα μέρη. Αυτό, φυσικά, δεν μπορεί να γίνει χρησιμοποιώντας ασφάλεια επιπέδου δικτύων δεδομένου, δεδομένου ότι το δίκτυο δεν διακρίνει το ένα μέρος ενός μηνύματος από το άλλο. Αυτός ο περιορισμός δεν υπάρχει στην ασφάλεια επιπέδου μηνυμάτων και ως εκ τούτου, είναι μια καλή τεχνική που μπορεί να χρησιμοποιηθεί για την ασφάλεια ολοκλήρωσης.

Για να επιτευχθεί η ασφάλεια σε επίπεδο μηνυμάτων, το SOAP επιτρέπει την επέκτασή του, προκειμένου να μεταφέρει πρόσθετες πληροφορίες κατά την ανταλλαγή μηνυμάτων. Μια τυποποιημένη επέκταση του SOAP ονομάζεται WS-security (Ασφάλεια υπηρεσιών δικτύου - Web Services Security) και διευκρινίζει πώς το SOAP μπορεί να επεκταθεί για να μεταφέρει τις πληροφορίες ασφάλειας στα μηνύματα. Η WS-security μπορεί να χρησιμοποιηθεί για να μεταφερθούν πληροφορίες ταυτότητας, όπως τα ονόματα χρήστη και οι κωδικοί πρόσβασης, οι ψηφιακές υπογραφές, να φέρει τις πληροφορίες για τα κλειδιά και τους αλγορίθμους που χρησιμοποιούνται για να κρυπτογραφήσουν τα διαφορετικά μέρη ενός μηνύματος, όπως επίσης να κρατήσει τις υπογραφές στα διαφορετικά μέρη ενός μηνύματος, οι οποίες δημιουργήθηκαν από τα διαφορετικά συμβαλλόμενα μέρη που συμμετέχουν σε μια ανταλλαγή μηνυμάτων.

Η WS-security χρησιμοποιεί την υπογραφή XML (XML Signature) για να παρέχει την ακεραιότητα (integrity) και την επικύρωση (authentication) των μηνυμάτων, και την

κρυπτογράφηση XML (XML Encryption) για να παρέχει την εμπιστευτικότητα. Παρέχει επίσης έναν γενικής χρήσης μηχανισμό για τη σύνδεση των σημείων ασφάλειας με τα μηνύματα.

Οι υπογραφές μπορούν να χρησιμοποιηθούν και ως συστατικό αποδοχής (non-repudiation), μια σημαντική πτυχή της ασφάλειας που πρέπει να εξεταστεί σε ένα πλαίσιο SOA. Η αποδοχή αναφέρεται στη δυνατότητα ενός οργανισμού να επικυρώσει ότι πραγματοποιήθηκε μια συγκεκριμένη συναλλαγή και έτσι δεν δίνει στον αποστολέα την ευκαιρία να την αποκηρύξει.

Ο έλεγχος μηνυμάτων SOAP βασισμένος στην παρεμπόδιση SOAP ( SOAP interception) είναι ένας τρόπος να τοποθετηθεί η βάση μιας αποτελεσματικής λύσης ασφάλειας στη SOA. Η παρεμπόδιση SOAP περιλαμβάνει την τοποθέτηση ενός πρόσθετου τμήματος που ονομάζεται «αναχαιτιστής SOAP» (SOAP interceptor) στα μηνύματα που ανταλλάσσονται μεταξύ ενός καταναλωτή και μιας υπηρεσίας ιστού. Λόγω της δυνατότητάς τους να αφομοιώσουν, να ελέγξουν, να αντιγράψουν και να διαβιβάσουν τα μηνύματα SOAP, οι αναχαιτιστές SOAP είναι σημαντικοί στην ασφάλεια. Κάθε μήνυμα ελέγχεται, για να είναι σίγουρο ότι η οντότητα που κάνει την αίτηση επικυρώνεται και εγκρίνεται να χρησιμοποιήσει την υπηρεσία Ιστού. Αυτό επιτυγχάνεται με τον έλεγχο των στοιχείων του χρήστη που περιλαμβάνονται στην επικεφαλίδα των μηνυμάτων SOAP.

**Μετατροπή της ασφάλειας σε μια υπηρεσία:** Ένα από τα ενδιαφέροντα θέματα στην επιβολή ασφάλειας στην ολοκλήρωση είναι η παρουσία αυστηρώς κωδικοποιημένων προτύπων ασφαλείας σε κάθε εφαρμογή. Οι περισσότερες εφαρμογές σήμερα έχουν τις δικές τους συνηθισμένες λύσεις στα ζητήματα ασφάλειας. Παραδείγματος χάριν, οι εφαρμογές έχουν τους δικούς τους καταλόγους χρηστών/κωδικού πρόσβασης που χρησιμοποιούν για την επικύρωση. Η επικύρωση αυτή γίνεται βάση του ονόματος χρήστη, τον πόρο, τη δραστηριότητα, ίσως ακόμη και την εσωτερική κατάσταση του συστήματος. Το πρόβλημα με την παρούσα κατάσταση είναι ότι το πρότυπο ασφάλειας σε κάθε εφαρμογή μπορεί να κάνει υποθέσεις που ισχύουν μόνο όταν χρησιμοποιείται η εφαρμογή αυτή καθαυτή ξεχωριστά και όχι όταν η εφαρμογή ενσωματώνεται με σε άλλες.

Οι εφαρμογές μπορούν να χρησιμοποιήσουν τυποποιημένα πρωτόκολλα όπως το LDAP (Lightweight Directory Access Protocol) για να επικυρώσουν τους χρήστες. Το LDAP έχει δώσει στις εφαρμογές έναν τρόπο για απομακρυσμένη επικύρωση εκπροσώπων (και την έγκριση σε έναν μικρό βαθμό) και αυτό έχει μειώσει το κόστος της διαχείρισης ασφάλειας. Η ολοκλήρωση μπορεί σίγουρα να πετύχει την ανάθεση περισσότερων στόχων ασφάλειας από



τις εφαρμογές. Το ερώτημα είναι αν υπάρχουν οποιεσδήποτε τεχνολογίες που επιτρέπουν κάτι τέτοιο. Με το SOAP, η απάντηση σε αυτήν την ερώτηση αποδεικνύεται καταφατική.

Μια υπηρεσία ασφάλειας μπορεί να προσφέρει στις εφαρμογές (υπηρεσίες στην περίπτωση της SOA) τη δυνατότητα να επικυρώσουν, να εγκρίνουν, να κρυπτογραφήσουν / αποκρυπτογραφήσουν μηνύματα, να υπογράψουν τα μηνύματα, να ελέγχουν τις υπογραφές και να καταγράφουν τα μηνύματα για να προστατεύσουν τις εφαρμογές από τις γνωστές και άγνωστες ευπάθειες. Η υπηρεσία ασφάλειας όχι μόνο διευκολύνει την ολοκλήρωση, αλλά μπορεί, επίσης, να ρίξει δραστικά το κόστος της ανάπτυξης, του ελέγχου και της εφαρμογής της ασφάλεια σε κάθε επιχειρησιακή εφαρμογή.

Το SOAP επιτρέπει το σαφή διαχωρισμό μεταξύ των σημείων ασφάλειας και του υπολοίπου του μηνύματος, ώστε να είναι δυνατή η χρήση μιας γενικής υπηρεσίας ασφάλειας από όλες τις επιχειρησιακές υπηρεσίες. Περαιτέρω, το SOAP επιτρέπει τη δυνατότητα ύπαρξης μιας υπηρεσίας ασφάλειας ως μεσάζων σε μια πορεία μηνυμάτων, εάν έτσι θέλουμε να χρησιμοποιήσουμε μια υπηρεσία ασφάλειας. Εάν θέλουμε κάποιο εκ των σημείων τέλους να επικαλεστεί την υπηρεσία ασφάλειας, το πρότυπο WS-trust (WS-εμπιστοσύνη) μας παρέχει έναν τρόπο για να γίνει αυτό, χρησιμοποιώντας το SOAP.

Αν και η επιβολή ασφάλειας μπορεί να αποσυνδεθεί από τις επιχειρησιακές υπηρεσίες υπό τη μορφή υπηρεσίας ασφάλειας, οι επιχειρησιακές υπηρεσίες εντούτοις απαιτούν κάποιο πλαίσιο ασφάλειας για να εκτελέσουν την εργασία τους. Ως εκ τούτου, μια υπηρεσία ασφάλειας πρέπει να διαβιβάσει στις επιχειρησιακές υπηρεσίες μερικά από τα στοιχεία που εξακριβώνει κατά τη διάρκεια της επιβολής ασφάλειας. Η Γλώσσα Σήμανσης Ισχυρισμών Ασφάλειας (Security Assertions Markup Language - SAML), σε συνδυασμό με την WS-ασφάλεια, βοηθά το SOAP να μεταφέρει αυτά τα δεδομένα.

Η SAML είναι ένα XML-βασισμένο πρότυπο που παρέχει ένα πλαίσιο για την περιγραφή των πληροφοριών ασφάλειας ενός χρήστη με έναν τυποποιημένο τρόπο. Η WS-ασφάλεια είναι το όνομα που δίνεται στο σύνολο προτύπων ασφάλειας που έχουν επικυρωθεί μέχρι σήμερα. Πολλές λύσεις ασφάλειας SOA εκμεταλλεύονται αυτά τα αναδυόμενα πρότυπα ασφάλειας.

**Δηλωτική και βασισμένη σε πολιτική ασφάλεια:** Όπως είδαμε προηγουμένως, η SOA υποστηρίζει τους τεχνικούς μηχανισμούς για να ενσωματώσει ασφαλώς τις εφαρμογές. Εντούτοις, μόνο η εξασφάλιση της ολοκλήρωσης δεν είναι αρκετή. Ακόμα κι αν αναπτυχθεί μια



λύση για την ασφαλή ολοκλήρωση των περισσότερων εφαρμογών της επιχείρησης, αντιμετωπίζουμε ακόμα το πρόβλημα της διαχείρισης της προκύπτουσας λύσης.

Στην ενσωμάτωση των εφαρμογών, εκτός από όλα τα ανωτέρω, έχουμε πρόσθετη ευθύνη για τις αλληλεπιδράσεις μεταξύ των εφαρμογών. Παραδείγματος χάριν, μια αλλαγή κωδικού πρόσβασης σε ένα σημείο μπορεί να πρέπει να διαδοθεί σε πολλαπλές εφαρμογές ή μια αλλαγή στην πολιτική ασφάλειας ενός μέρους της εφαρμογής μπορεί να προσκρούσει στην πολιτική πρόσβασης μιας άλλης εφαρμογής.

Παραδοσιακά, αυτές οι ανάγκες ικανοποιούνται μόνο σε μια περιορισμένη έκταση από αποκλειστικές λύσεις, όπως η ύπαρξη πολλών SSO(Single Sign-On) λύσεων που διαχειρίζονται τους χρήστες σε διαφορετικές εφαρμογές. Εντούτοις, αυτές οι λύσεις δεν κατάφεραν να φτάσουν τις πλήρεις δυνατότητές τους λόγω της έλλειψης προτύπων. Οι περισσότερες από αυτές υποστήριζαν τις καθιερωμένες εφαρμογές ή πλατφόρμες, όμως δεν υπήρξε κανένας εύκολος και ομοιόμορφος τρόπος να υποστηριχθούν οποιεσδήποτε υπάρχουσες home-grown εφαρμογές.

Η SOA βάζει σε τάξη αυτή την κατάσταση. Παρέχει έναν δηλωτικό τρόπο για να ορίσει την ασφάλεια για τις εφαρμογές, ο οποίος έχει δύο σημαντικά αντίκτυπα - το πρώτο είναι η ύπαρξη εύκολης εφαρμογής των λύσεων ασφάλειας και το δεύτερο η ύπαρξη δυνατότητας για τους πελάτες και τους εξυπηρετητές να διαπραγματευτούν δυναμικά τις πολιτικές ασφάλειας.

Πρώτα ας δούμε εν συντομία τι σημαίνει η δηλωτική ασφάλεια. Μια δήλωση διευκρινίζει μόνο τι πρέπει να γίνει και όχι το πώς γίνεται. Στην περίπτωση της ασφάλειας, διευκρινίζει πώς η εφαρμογή εξασφαλίζεται, δηλαδή ποιο μέρος της εφαρμογής είναι προσιτό σε ποια ομάδα χρηστών και τα λοιπά. Διευκρινίζει, επίσης, που βρίσκονται οι μονάδες και πώς οι χρήστες συνδέονται με αυτές. Μια προηγμένη λύση ασφάλειας που στηρίζεται σε τέτοια πρότυπα, όχι μόνο υποστηρίζει τη διαχείριση χρηστών, αλλά επιτρέπει στην ασφάλεια της εφαρμογής να διαμορφωθεί, να επεκταθεί και να τροποποιηθεί, χωρίς την αλλαγή οποιουδήποτε κώδικα. Δηλαδή οι απαιτήσεις ασφάλειας των εφαρμογών είναι διαθέσιμες σε μια διοικητική λύση για τον πλήρη χειρισμό.

Η δηλωτική ασφάλεια περιορίζεται στους χρήστες και τους ρόλους σε μια εφαρμογή. Μπορεί να επεκταθεί με το χωρισμό της εφαρμογής σε διαφορετικές ζώνες, κάθε μια από τις οποίες έχει διαφορετικό έλεγχο πρόσβασης. Στη SOA διευκρινίζουμε περισσότερα πράγματα από τη



δήλωση ασφάλειας. Παραδείγματος χάριν, διευκρινίζουμε τις πολιτικές ασφάλειάς, τι είδους κρυπτογράφηση αναμένεται, τι είδους πιστοποιητικά γίνονται δεκτά, ποιο μέρος των μηνυμάτων πρέπει να κρυπτογραφηθεί και/ή να υπογραφεί και πολλά άλλα. Συνεπώς, τα πλαίσια ασφάλειας της SOA μπορούν να επιβάλουν τις πολιτικές ασφάλειας με συνέπεια σε όλες τις υπηρεσίες που προσφέρονται από μια επιχείρηση.

Μαζί με τον τύπο των δεδομένων, τις διεπαφές και τις πληροφορίες μεταφοράς, η περιγραφή μιας υπηρεσίας μπορεί να συμπεριλάβει τις πολιτικές ασφάλειας που συνδέονται με τη χρήση της υπηρεσίας αυτής. Όταν οι πελάτες έχουν πρόσβαση στην πολιτική ασφάλειας των εξυπηρετητών, μπορούν να σιγουρευτούν για τη διαλειτουργικότητα μεταξύ τους. Ένας φορέας παροχής υπηρεσιών μπορεί να δηλώσει τα είδη των σημείων ασφάλειας που μπορεί να δεχτεί, δεδομένου ότι ένας καταναλωτής υπηρεσιών μπορεί να δηλώσει τι μπορεί να παρέχει. Με τη σύγκριση αυτών των δηλώσεων, ένα πλαίσιο ασφάλειας μπορεί από κάθε πλευρά δυναμικά να εξασφαλίσει τη διαλειτουργικότητα. Στην πραγματικότητα, οι πελάτες μπορούν δυναμικά να αποφασίσουν τι είδους τις πληροφορίες ασφάλειας απαιτούνται, με βάση τις πληροφορίες που ζητούν.

**Διαχείριση συμβάσεων και έλεγχος επαναλαμβανόμενης επίθεσης:** Μια σύμβαση είναι ένα σύνολο κανόνων που διέπουν τη χρήση μιας υπηρεσίας Ιστού. Για παράδειγμα, μια σύμβαση ορίζει ότι ένας συγκεκριμένος χρήστης έχει το δικαίωμα να καλέσει μια συγκεκριμένη υπηρεσία Ιστού δέκα φορές ανά ημέρα. Πέρα από την επίκληση, το επίπεδο υπηρεσιών πρέπει να ελέγξει ορισμένες παραμέτρους, όπως ένας δεύτερος χρόνος απόκρισης.

Στην ασφάλεια, οι συμβάσεις είναι χρήσιμες στον καθορισμό εάν η SOA λειτουργεί κατάλληλα ή εάν χρησιμοποιείται κατ' άσχημο τρόπο, λόγω μιας παραβίασης ασφάλειας όπως μια επίθεση DoS (Denial of Service – Άρνηση παροχής υπηρεσιών). Η λύση ασφάλειας SOA πρέπει να παρέχει μια δυνατότητα για τον εντοπισμό των αιτημάτων SOAP, προκειμένου να περιοριστούν οι ζημιές από επιθέσεις. Συνήθως, ακολουθώντας ένα χαρακτηριστικό γνώρισμα, ελέγχεται ο αποστολέας του μηνύματος SOAP και ο χρόνος που δημιουργήθηκε. Σε μερικές περιπτώσεις, η λύση ασφάλειας SOA «θα σφραγίσει» το μήνυμα SOAP με έναν μοναδικό αντιπροσωπευτικό αριθμό. Εάν τίθεται ως στόχος η παρεμπόδιση των διπλών μηνυμάτων, είναι αδύνατο το ίδιο μήνυμα να σταλεί δύο φορές. Ο περιορισμός αυτής της δυνατότητας μειώνει την πιθανότητα να πλημμυρίσει η SOA με «διπλά αιτήματα», μια συνήθης τεχνική που χρησιμοποιείται στις επιθέσεις DoS.

## 2.3 Διαχείριση Επιχειρηματικών Διαδικασιών

### 2.3.1 Η ανάγκη για επιχειρηματικές διαδικασίες

Ο στόχος της προσπάθειας που έχει καταβληθεί για την ανάπτυξη και βελτίωση των Web Services είναι η επίτευξη της πλήρους διαλειτουργικότητας μεταξύ εφαρμογών, χρησιμοποιώντας πρότυπα του Παγκοσμίου Ιστού. Τα Web Services χρησιμοποιούν ένα μοντέλο ολοκλήρωσης βασισμένο στην χαλαρή συνδεσιμότητα, με σκοπό να καταστεί δυνατή η ευέλικτη ολοκλήρωση ετερογενών συστημάτων σε μια ποικιλία τομέων, συμπεριλαμβανομένων των επιχειρήσεων προς τους καταναλωτές (business-to-consumer), από επιχείρηση σε επιχείρηση (business-to-business) και την ενσωμάτωση επιχειρησιακών εφαρμογών (enterprise application integration).

Το SOAP, η WSDL και το UDDI ορίζουν, όπως έχουμε αναφέρει, το χώρο των web services. Το SOAP καθορίζει ένα πρωτόκολλο XML μηνυμάτων για τη βασική διαλειτουργικότητα μεταξύ υπηρεσιών, η WSDL εισάγει μια κοινή γραμματική για την περιγραφή των υπηρεσιών και το UDDI παρέχει την υποδομή που απαιτείται για να δημοσιεύονται και να ανακαλύπτονται υπηρεσίες με συστηματικό τρόπο. Μαζί, οι τρεις αυτές προδιαγραφές επιτρέπουν στις εφαρμογές να ανακαλύψουν η μία την άλλη και να αλληλεπιδρούν με ένα χαλαρής συνδεσιμότητας, ανεξάρτητο πλατφόρμας μοντέλο.

Η ολοκλήρωση των συστημάτων απαιτεί κάτι περισσότερο από τη δυνατότητα διεξαγωγής απλών αλληλεπιδράσεων με τη χρήση τυποποιημένων πρωτοκόλλων. Η πλήρης εκμετάλλευση των δυνατοτήτων των Web Services, ως πλατφόρμα ενσωμάτωσης, θα επιτευχθεί μόνον όταν οι εφαρμογές και οι επιχειρηματικές διαδικασίες είναι σε θέση να ενσωματώσουν τις πιο σύνθετες αλληλεπιδράσεις τους, χρησιμοποιώντας ένα τυποποιημένο μοντέλο ολοκλήρωσης διαδικασιών.

Το μοντέλο αλληλεπίδρασης που υποστηρίζεται άμεσα από την WSDL είναι ουσιαστικά ένα μοντέλο χωρίς καταστάσεις (stateless) σύγχρονων ή μη σχετιζόμενων ασύγχρονων αλληλεπιδράσεων. Στις επιχειρήσεις, τα μοντέλα αλληλεπιδράσεων περιλαμβάνουν συνήθως ακολουθίες μηνυμάτων peer-to-peer και συναλλαγών, τόσο σύγχρονα όσο και ασύγχρονα, εντός μακροχρόνιων αλληλεπιδράσεων που εμπλέκουν δύο ή περισσότερα μέρη. Για να οριστούν αυτές οι επιχειρηματικές συναλλαγές χρειάζεται μια τυπική περιγραφή των πρωτοκόλλων ανταλλαγής μηνυμάτων που χρησιμοποιούνται από τις επιχειρηματικές διαδικασίες στις μεταξύ τους αλληλεπιδράσεις. Ο ορισμός τέτοιων πρωτοκόλλων



συνεπάγεται τον ακριβή προσδιορισμό της συμπεριφοράς κατά την ανταλλαγή μηνυμάτων κάθε ενός εκ των μερών που εμπλέκονται στο πρωτόκολλο, χωρίς όμως να αποκαλύπτεται ο εσωτερικός τρόπος υλοποίησης τους.

Υπάρχουν δύο λόγοι για τους οποίους πρέπει να διαχωρίζονται τα θέματα της δημόσιας συμπεριφοράς μιας επιχειρηματικής διαδικασίας από τις εσωτερικές ή ιδιωτικές πτυχές της. Ο ένας είναι ότι οι επιχειρήσεις προφανώς δεν θέλουν να αποκαλύψουν όλες τις εσωτερικές τους αποφάσεις και τον τρόπο διαχείρισης των δεδομένων τους στους επιχειρηματικούς τους εταίρους. Ο δεύτερος είναι ότι ο διαχωρισμός μεταξύ των δημόσιων και των ιδιωτικών διαδικασιών παρέχει την ελευθερία και την ευελιξία της αλλαγής των ιδιωτικών πτυχών μιας εφαρμογής, χωρίς να επηρεάζεται το δημόσιο πρωτόκολλο των επιχειρήσεων.

Τα επιχειρηματικά μοντέλα πρέπει να περιγράφονται σαφώς με έναν ανεξάρτητο πλατφόρμας τρόπο και πρέπει να περικλείουν όλες τις πτυχές συμπεριφορών που έχουν δια-επιχειρηματική σημασία. Κάθε συμμετέχων μπορεί στη συνέχεια να καταλάβει και να σχεδιάσει τις διαδικασίες του σύμφωνα με το πρωτόκολλο, χωρίς τη συμμετοχή στη διαδικασία της ανθρώπινης επικοινωνίας και συμφωνίας, η οποία αυξάνει τον βαθμό δυσκολίας στη δημιουργία αυτοματοποιημένων δια-επιχειρηματικών διαδικασιών. Παρακάτω αναφέρουμε τις κυριότερες έννοιες που απαιτούνται για την περιγραφή των επιχειρηματικών πρωτοκόλλων και τη σχέση τους με εκείνες που απαιτούνται για να περιγραφεί μια εκτελέσιμη διαδικασία:

- τα Business πρωτόκολλα περιλαμβάνουν πάντοτε συμπεριφορές που εξαρτώνται από τα δεδομένα. Για παράδειγμα, ένα πρωτόκολλο αλυσίδας εφοδιασμού εξαρτάται από στοιχεία, όπως ο αριθμός των στοιχείων γραμμής μιας παραγγελίας, η συνολική αξία της παραγγελίας ή η προθεσμία παράδοσης. Ο καθορισμός του σκοπού των επιχειρήσεων σε αυτές τις περιπτώσεις απαιτεί τη χρήση συνθηκών και χρονικών περιορισμών.
- Η δυνατότητα του καθορισμού των κατ' εξαίρεση συνθηκών και τις συνέπειές τους, συμπεριλαμβανομένων των ακολουθιών ανάκτησης, είναι εξίσου σημαντική για τα πρωτόκολλα των επιχειρήσεων, όσο η δυνατότητα να καθοριστεί η συμπεριφορά του στην περίπτωση που όλα λειτουργήσουν σωστά.
- Οι μακροχρόνιες αλληλεπιδράσεις περιλαμβάνουν πολλαπλές, συχνά ένθετες μονάδες εργασίας, καθεμία από τις οποίες έχει τις δικές της απαιτήσεις σε δεδομένα. Τα επιχειρηματικά πρωτόκολλα απαιτούν, συχνά, το συντονισμό όλων των

συμμετεχόντων για το αποτέλεσμα (επιτυχία ή αποτυχία) των μονάδων της εργασίας σε διάφορα επίπεδα.

Αν θέλουμε να παρέχουμε ακριβείς και προβλέψιμες περιγραφές για την συμπεριφορά μιας υπηρεσίας σε ένα δια-επιχειρηματικό πρωτόκολλο, χρειαζόμαστε πλούσιο συμβολισμό με πολλά χαρακτηριστικά γνωρίσματα που θυμίζουν αρχαίο εκτελέσιμη γλώσσα. Η βασική διάκριση μεταξύ δημόσιων πρωτοκόλλων ανταλλαγής μηνυμάτων και εκτελέσιμων εσωτερικών διαδικασιών είναι ότι οι εσωτερικές διεργασίες χειρίζονται τα δεδομένα με πολύπλοκους ιδιωτικούς τρόπους που δεν χρειάζεται να περιγραφούν στα δημόσια πρωτόκολλα.

Σκεπτόμενοι τις πτυχές των επιχειρηματικών πρωτοκόλλων σχετικά με την επεξεργασία δεδομένων, είναι χρήσιμο να εξεταστεί η αναλογία με τα πρωτόκολλα δικτυακής επικοινωνίας. Τα πρωτόκολλα δικτύου καθορίζουν τη μορφή και το περιεχόμενο των φακέλων(envelope) που ρέουν στο δίκτυο(wire) και η συμπεριφορά του πρωτοκόλλου που περιγράφουν καθοδηγείται αποκλειστικά από τα δεδομένα σε αυτούς τους φακέλους. Με άλλα λόγια, υπάρχει σαφής φυσικός διαχωρισμός μεταξύ των δεδομένων του πρωτοκόλλου και το "ωφέλιμο φορτίο" δεδομένων. Ο διαχωρισμός είναι λιγότερο ξεκάθαρος στα πρωτόκολλα των επιχειρήσεων, επειδή τα σχετιζόμενα με το πρωτόκολλο δεδομένα τείνουν να είναι ενσωματωμένα σε άλλα δεδομένα της εφαρμογής.

### **2.3.2 Ενορχήστρωση και Χορογραφία υπηρεσιών**

Η αυτοματοποίηση των επιχειρησιακών διαδικασιών παρέχει τεράστια πλεονεκτήματα στις επιχειρήσεις. Η ανάγκη για αυτοματοποίηση απεικονίζει την ωριμότητα της επιχείρησης και την ικανότητά της να αναβαθμιστεί και να ανταποκριθεί στις μεταβαλλόμενες ανάγκες της αγοράς. Το πρόβλημα που αντιμετωπίζουν οι περισσότερες επιχειρήσεις σχετικά με τη ροή της επιχειρησιακής διαδικασίας είναι ότι οι διαδικασίες αποτελούνται από ένα πλήθος δραστηριοτήτων. Οι δραστηριότητες αυτές μπορεί να συμβαίνουν σε διαφορετικά τμήματα ή σε διαφορετικά συστήματα ηλεκτρονικών υπολογιστών και να χρησιμοποιούν έναν συνδυασμό σύγχρονων και ασύγχρονων διαδικασιών. Αν και μερικές διαδικασίες τρέχουν διαδοχικά, μερικές άλλες μπορεί να τρέχουν παράλληλα. Επομένως, είναι σημαντικό για τις επιχειρήσεις να διαχειριστούν επιτυχώς τις συντονισμένες δραστηριότητες, στη σειρά που απαιτείται, για την επίτευξη των στόχων τους.





Το πρότυπο που συζητείται παρακάτω - η Γλώσσα εκτέλεσης επιχειρησιακών διαδικασιών - Business Process Execution Language for Web Services (BPEL4WS) - έχει ως σκοπό να μειώσει την έμφυτη πολυπλοκότητα της σύνδεσης των υπηρεσιών Ιστού. Χωρίς αυτή, ένας οργανισμός αναγκάζεται να δημιουργήσει ιδιόκτητα επιχειρησιακά πρωτόκολλα, γεγονός που περιπλέκει την πραγματική συνεργασία μεταξύ των υπηρεσιών Ιστού. Δύο νέοι όροι που έχουν υιοθετηθεί για να περιγράψουν αυτήν την συνεργασία είναι η ενορχήστρωση και η χορογραφία.

### **Ενορχήστρωση(Orchestration)**

Αναφέρεται σε μια εκτελέσιμη επιχειρησιακή διαδικασία που μπορεί να αλληλεπιδράσει με εσωτερικές και εξωτερικές υπηρεσίες Ιστού. Η ενορχήστρωση περιγράφει το πώς οι υπηρεσίες Ιστού μπορούν να αλληλεπιδράσουν σε επίπεδο μηνυμάτων, συμπεριλαμβανομένης της επιχειρησιακής λογικής και της σειράς εκτέλεσης των αλληλεπιδράσεων. Αυτές οι αλληλεπιδράσεις μπορούν να καλύψουν τις εφαρμογές και τους οργανισμούς και να οδηγήσουν σε μια μακράς διάρκειας διαδικασία συναλλαγών. Με τη ενορχήστρωση, η διαδικασία ελέγχεται πάντα από την προοπτική του ενός από τα συμβαλλόμενα μέρη.

Η προσέγγιση της ενορχήστρωσης αναγνωρίζεται σε συστήματα, όπου η διαλειτουργικότητα των εφαρμογών επιτυγχάνεται βάσει μιας κεντρικά ελεγχόμενης ροής εργασιών (workflow). Όλες οι λεπτομέρειες της λογικής της ροής εργασιών περιλαμβάνεται στον ορισμό της διαδικασίας. Σε αυτό τον ορισμό αναφέρονται επίσης και οι συμμετέχοντες στην διαδικασία. Η διαδικασία από μόνη της ονομάζεται υπηρεσία διαδικασίας (*process service*) ενώ οι υπόλοιπες διαδικασίες που αλληλεπιδρούν με αυτή και καλούν ή καλούνται από αυτή ονομάζονται υπηρεσίες συνεργάτες (*partner services*).

Η ενορχήστρωση προσφέρει ένα περιβάλλον ενοποίησης για τις εφαρμογές ενός οργανισμού, ακόμα και αν αυτές βασίζονται σε διαφορετικές πλατφόρμες. Η προσέγγιση αυτή παρέχει τρία βασικά χαρακτηριστικά: υπάρχει δυνατότητα αλλαγής ή επέκτασης της λογικής της ροής εργασίας, διευκολύνεται η συγχώνευση των επιχειρηματικών διαδικασιών, ειδικά σε αρχιτεκτονικές που προσανατολίζονται σε υπηρεσίες (SOA), και δίνει τη δυνατότητα εξέλιξης σε επίπεδο ετερόκλητων επιχειρήσεων.

### Χορογραφία(Choreography)

Η τεχνική της χορογραφίας ανταποκρίνεται στην ιδέα της απευθείας οργάνωσης των διεργασιών των οργανισμών για την εκτέλεση των συναλλαγών τους. Υπό ιδανικές συνθήκες οι οργανισμοί θα μπορούσαν να συμφωνήσουν πάνω στη δομή των εσωτερικών διεργασιών τους προκειμένου να διαλειτουργήσουν και να επιτύχουν αυτή την επιδίωξη τους βασιζόμενες σε αυτοματοποιημένες λύσεις. Στα πλαίσια της προσέγγισης της χορογραφίας επιδιώκεται η ανταλλαγή μηνυμάτων, εφόσον έχει εγκατασταθεί ένα πλαίσιο συνεργασίας των διαφόρων υπηρεσιών, ενώ παράλληλα θεωρείται ότι δεν υπάρχει οντότητα που να έχει τον πλήρη έλεγχο της συνεργασίας αυτής. Πρόκειται ουσιαστικά για ένα καθολικό περιβάλλον διαλειτουργικών, ανεξάρτητων οντοτήτων.

Βασικές ιδιότητες της χορογραφίας είναι η επαναχρησιμοποίηση σε διαφορετικές επιχειρηματικές εργασίες, η δυνατότητα σύνθεσης των ξεχωριστών διεργασιών όσο και των ίδιων των χορογραφιών σε μεγαλύτερες οντότητες, αλλά και οι ξεχωριστές μονάδες λειτουργικότητας που συνθέτουν μια χορογραφία και μπορούν να επαναχρησιμοποιηθούν.

#### 2.3.3 BPEL4WS

Η γενική υιοθέτηση λύσεων αυτοματοποίησης επιχειρηματικών διαδικασιών απαιτεί ένα πρότυπο θεμελίωσης και μια εξειδικευμένη γλώσσα για τη σύνθεση των υπηρεσιών στις επιχειρηματικές διαδικασίες, που θα παρέχει τη δυνατότητα έκφρασης των διαδικασιών με τυποποιημένο τρόπο, χρησιμοποιώντας μια κοινώς αποδεκτή γλώσσα. Η BPEL4WS είναι μια τέτοια γλώσσα που τείνει να γίνει κυρίαρχο πρότυπο. Ο κύριος στόχος της BPEL4WS είναι να τυποποιηθεί η διαδικασία αυτοματοποίησης μεταξύ των υπηρεσιών web.

Εντός των επιχειρήσεων, η BPEL4WS χρησιμοποιείται για να τυποποιήσει την ολοκλήρωση των επιχειρησιακών εφαρμογών και επεκτείνει την ενσωμάτωση των πρότερα απομονωμένων συστημάτων. Μεταξύ των επιχειρήσεων, η BPEL4WS επιτρέπει την ευκολότερη και πιο αποτελεσματική ενσωμάτωση με τους επιχειρηματικούς εταίρους. Επιπλέον, υποκινεί τις επιχειρήσεις να καθορίσουν περαιτέρω τις επιχειρηματικές διαδικασίες τους, γεγονός που με τη σειρά του οδηγεί στην βελτιστοποίησή, στην αναδιοργάνωση και στην επιλογή των πλέον κατάλληλων διαδικασιών, με αποτέλεσμα την περαιτέρω βελτιστοποίηση της οργάνωσης.

Οι ορισμοί των επιχειρηματικών διαδικασιών που περιγράφονται στη BPEL4WS δεν επηρεάζουν τα υφιστάμενα συστήματα. Η BPEL4WS είναι η βασική τεχνολογία σε



περιβάλλοντα όπου οι λειτουργίες έχουν ήδη ή πρόκειται να εκτεθούν μέσω δικτυακών υπηρεσιών. Με την αύξηση στη χρήση της τεχνολογίας υπηρεσιών Ιστού, η σημασία της BPEL θα αυξηθεί περαιτέρω.

### Χαρακτηριστικά της BPEL4WS

Με την BPEL4WS μπορούμε να ορίσουμε απλές και σύνθετες επιχειρηματικές διαδικασίες. Σε κάποιο βαθμό, η BPEL4WS είναι παρόμοια με τις παραδοσιακές γλώσσες προγραμματισμού. Προσφέρει δομές, όπως βρόγχοι, διακλαδώσεις, μεταβλητές, αναθέσεις κλπ. που επιτρέποντας τον ορισμό των επιχειρηματικών διαδικασιών με αλγοριθμικό τρόπο. Ως εκ τούτου, από τη μία πλευρά προσφέρει δομές, οι οποίες καθιστούν τον ορισμό των διαδικασιών σχετικά απλό. Από την άλλη πλευρά, είναι λιγότερο πολύπλοκη από τις παραδοσιακές γλώσσες προγραμματισμού, γεγονός που απλουστεύει την εκμάθησή της.

Τα πιο σημαντικά BPEL4WS κατασκευάσματα έχουν σχέση με την επίκληση των web services. Η BPEL4WS καθιστά εύκολη την κλήση λειτουργιών των υπηρεσιών web σύγχρονα ή ασύγχρονα. Η κλήση αυτή μπορεί να γίνει παράλληλα ή σειριακά. Παρέχει πλούσιο λεξιλόγιο για το χειρισμό σφαλμάτων, το οποίο είναι πολύ σημαντικό, αφού οι αυτοδύναμες επιχειρηματικές διαδικασίες έχουν την ανάγκη αντίδρασης στις αποτυχίες με έξυπνο τρόπο. Η BPEL4WS παρέχει, επίσης, υποστήριξη για εκτέλεση μακροχρόνιων διαδικασιών και διαδικασιών επανόρθωσης, οι οποίες επιτρέπουν την μερική επαναφορά της εργασίας που εκτέλεσε μια διαδικασία, η οποία δεν ολοκληρώθηκε επιτυχώς. Παρακάτω αναφέρονται τα πιο σημαντικά χαρακτηριστικά που παρέχει η BPEL4WS:

- Περιγραφή της λογικής των επιχειρηματικών διαδικασιών μέσω της σύνθεσης υπηρεσιών.
- Σύνθεση μεγαλύτερων επιχειρηματικών διαδικασιών από μικρότερες διαδικασίες και υπηρεσίες.
- Χειρισμός σύγχρονης και ασύγχρονης (συχνά μακράς πορείας) κλήσης λειτουργιών των υπηρεσιών και διαχείριση callbacks που συμβαίνουν σε μεταγενέστερους χρόνους.
- Κλήση υπηρεσιών παράλληλα και σειριακά.
- Επιλεκτική επαναφορά δραστηριοτήτων σε περίπτωση αποτυχιών
- Διαχείριση πολλαπλών μακροχρόνιων δραστηριοτήτων συναλλαγής, οι οποίες μπορεί να είναι διακοπτόμενες (interruptible)

- Επαναφορά ή διακοπή αποτυχημένων δραστηριοτήτων για την ελαχιστοποίηση των εργασιών που πρέπει να επαναληφθούν.
- Ανακατεύθυνση εισερχόμενων μηνυμάτων στις κατάλληλες διαδικασίες και δραστηριότητες.
- Συσχέτιση αιτημάτων εντός και μεταξύ των επιχειρηματικών διαδικασιών.
- Προγραμματισμός ενεργειών με βάση την προθεσμία εκτέλεσης και ορισμός της σειράς εκτέλεσής τους.
- Εκτέλεση δραστηριοτήτων παράλληλα και δυνατότητα καθορισμού για το πώς οι παράλληλες ροές θα συγχωνεύονται, με βάση τις συνθήκες συγχρονισμού.
- Δόμηση επιχειρηματικών διαδικασιών σε διάφορα πεδία εφαρμογής
- Χειρισμός γεγονότων που σχετίζονται με μηνύματα(message related) ή με χρονικές στιγμές(time-related).

Η BPEL4WS χρησιμοποιεί, κατά μια έννοια, τις ιδιότητες του μηνύματος για τον εντοπισμό σχετικών με το πρωτόκολλο δεδομένων, τα οποία είναι ενσωματωμένα σε αυτά. Οι ιδιότητες μπορούν να θεωρηθούν ως "διαφανή" δεδομένα που σχετίζονται με τις δημόσιες πτυχές, σε αντίθεση με τα «αδιαφανή» δεδομένα που χρησιμοποιούν οι εσωτερικές / ιδιωτικές λειτουργίες. Τα διαφανή στοιχεία επηρεάζουν το δημόσιο πρωτόκολλο των επιχειρήσεων με άμεσο τρόπο, ενώ τα αδιαφανή δεδομένα είναι σημαντικά κυρίως για τα back-end συστήματα και επηρεάζουν το πρωτόκολλο της επιχείρησης μη ντετερμινιστικά, διότι ο τρόπος λήψης των αποφάσεων είναι αδιαφανής. Στην περίπτωση που η απόφαση πρέπει να αντανακλάται ως εναλλακτικές λύσεις στη συμπεριφορά του εξωτερικού επιχειρησιακού πρωτοκόλλου, η διαδικασία μπορεί να μοντελοποιηθεί με την εκχώρηση αδιαφανών τιμών στις ιδιότητες του μηνύματος. Θεωρούμε ως αρχή ότι τα δεδομένα που χρησιμοποιούνται για να επηρεάσουν τη συμπεριφορά ενός επιχειρηματικού πρωτοκόλλου πρέπει να είναι διαφανή και ως εκ τούτου θεωρούνται ιδιότητες. Η BPEL4WS επιτρέπει ρητά τη χρήση μη ντετερμινιστικών τιμών δεδομένων, ώστε να είναι δυνατό να συλλάβει την ουσία της δημόσιας συμπεριφοράς, κρύβοντας παράλληλα τις ιδιωτικές πτυχές.

Οι βασικές έννοιες της BPEL4WS μπορούν να εφαρμοστούν με δύο τρόπους. Μια BPEL4WS διαδικασία μπορεί να καθορίσει ρόλους ενός πρωτοκόλλου, χρησιμοποιώντας την έννοια της αφηρημένης διαδικασίας. Για παράδειγμα, σε ένα πρωτόκολλο αλυσίδας εφοδιασμού, ο αγοραστής και ο πωλητής αποτελούν δύο διακριτούς ρόλους, ο καθένας με τη δική του αφηρημένη διαδικασία. Η σχέση τους μοντελοποιείται συνήθως ως μια σύνδεση εταίρων(partner link). Οι αφηρημένες διεργασίες χρησιμοποιούν όλες τις έννοιες της



BPEL4WS, αλλά προσεγγίζουν την διαχείριση των δεδομένων κατά τρόπο που να αντικατοπτρίζει το επίπεδο αφαίρεσης που απαιτείται για να περιγράψει τις δημόσιες πτυχές του πρωτοκόλλου. Συγκεκριμένα, οι αφηρημένες διεργασίες χειρίζονται μόνο σχετικά με το πρωτόκολλο στοιχεία. Η BPEL4WS παρέχει έναν τρόπο για τον εντοπισμό αυτών των στοιχείων ως ιδιότητες του μηνύματος. Επιπλέον, οι αφηρημένες διεργασίες χρησιμοποιούν μη ντετερμινιστικές τιμές δεδομένων για να αποκρύψουν ιδιωτικές πτυχές της συμπεριφοράς.

Με τη χρήση της BPEL4WS είναι, επίσης, δυνατό να καθοριστεί μια εκτελέσιμη επιχειρηματική διαδικασία. Η λογική και η κατάσταση της διαδικασίας καθορίζουν τη φύση και την ακολουθία των αλληλεπιδράσεων των web services που διεξάγονται σε κάθε επιχειρηματικό εταίρο και, συνεπώς, το πρωτόκολλο αλληλεπίδρασης. Ενώ ο ορισμός της διαδικασίας δεν απαιτείται να είναι πλήρης από την άποψη της ιδιωτικής εφαρμογής (private implementation), η γλώσσα καθορίζει ουσιαστικά ένα φορητό σχήμα εκτέλεσης για τις επιχειρηματικές διαδικασίες που στηρίζονται αποκλειστικά στους πόρους των υπηρεσιών ιστού και τα XML δεδομένα. Επιπλέον, οι διαδικασίες εκτελούνται και να αλληλεπιδρούν με τους εταίρους με συνεκτικό τρόπο, ανεξάρτητα από την υποστήριξη της πλατφόρμας ή του μοντέλου προγραμματισμού που χρησιμοποιείται από την εφαρμογή του περιβάλλοντος φιλοξενίας.

Ακόμη και στις περιπτώσεις που η ιδιωτική εφαρμογή χρησιμοποιεί εξαρτώμενη από πλατφόρμα λειτουργικότητα, κάτι που συμβαίνει στις περισσότερες ρεαλιστικές περιπτώσεις, η συνέχεια του βασικού εννοιολογικού μοντέλου μεταξύ αφηρημένων και εκτελέσιμων διαδικασιών καθιστά δυνατή την εξαγωγή και την εισαγωγή των δημόσιων στοιχείων που περιλαμβάνονται στα πρωτόκολλα των επιχειρήσεων ως διαδικασίες ή πρότυπα, διατηρώντας παράλληλα το σκοπό και τη δομή των πρωτοκόλλων. Αυτό είναι αναμφισβήτητο η πιο ελκυστική προοπτική για τη χρήση της BPEL4WS, από την άποψη της απελευθέρωσης των δυνατοτήτων των Web Services, επειδή επιτρέπει την ανάπτυξη εργαλείων και τεχνολογιών που αυξάνουν κατά πολύ το επίπεδο του αυτοματισμού και κατά συνέπεια μειώνουν το κόστος για τη δημιουργία αυτοματοποιημένων δια-επιχειρησιακών διαδικασιών.

Η BPEL4WS βασίζεται σε διάφορες XML προδιαγραφές: WSDL 1.1, XML Schema 1.0 και XPath1.0. Τα WSDL μηνύματα και οι ορισμοί τύπων του XML Schema παρέχουν το μοντέλο δεδομένων που χρησιμοποιείται από τις BPEL4WS διαδικασίες. Το XPath παρέχει υποστήριξη για την επεξεργασία των δεδομένων. Όλοι οι εξωτερικοί πόροι και οι εταίροι απεικονίζονται



ως WSDL υπηρεσίες. Η BPEL4WS μπορεί να επεκταθεί για να υποστηρίξει μελλοντικές εκδόσεις των προτύπων αυτών.

## Συμπεράσματα

Υπάρχουν πολλά και διαφορετικά παραδείγματα σχεδιασμού για καταναμημένες λογικές λύσεις. Αυτό που διαχωρίζει την υπηρεσιοστρεφή αρχιτεκτονική είναι ο τρόπος με τον οποίο διεξάγει το διαχωρισμό του προβλήματος και το πώς διαμορφώνει τις επιμέρους μονάδες της λογικής λύσης. Για να καταλάβουμε τι ακριβώς σημαίνει αυτό, απαιτείται η κατανόηση των στρατηγικών στόχων της αρχιτεκτονικής αυτής σε συνδυασμό με τη γνώση των ακόλουθων αρχών σχεδιασμού:

- Τυποποιημένη σύμβαση παροχής υπηρεσιών(standardized service contract)
- Χαλαρή συνδεσιμότητα υπηρεσιών(service loose coupling)
- Αφαιρετικότητα υπηρεσιών(service abstraction)
- Επαναχρησιμοποίηση υπηρεσιών(service reusability)
- Αυτονομία υπηρεσιών(service autonomy)
- Υπηρεσίες χωρίς κατάσταση(service statelessness)
- Ανακαλυψιμότητα υπηρεσιών(service discoverability)
- Σύνθεση υπηρεσιών (service composability)
- Διαλειτουργικότητα (interoperability)

Η SOA και τα web services δεν αποτελούν πλέον αφηγημένες έννοιες ,αλλά πρόκειται για πραγματικές μεθόδους που εφαρμόζονται στην λύση συγχρόνων προβλημάτων της πληροφορικής.



## Ενότητα Β' Έρευνα και Ερευνητική Μεθοδολογία

Στόχοι και αποτελέσματα



*(Your Research Resource, από Peak Surveys)<sup>[E12]</sup>*

## Κεφάλαιο 3

# Έρευνα και Ερευνητική Μεθοδολογία

### Εισαγωγή

Η τεχνολογία αποτελεί ένα εξαιρετικά σύγχρονο μέσο που λειτουργεί υποστηρικτικά, προκειμένου να γίνει αποτελεσματικότερη η παροχή φροντίδας στους ασθενείς. Όπως είδαμε στα προηγούμενα κεφάλαια, οι Υπηρεσίες Διαδικτύου και η Υπηρεσιοστρεφής αρχιτεκτονική μπορούν να δώσουν τη λύση σε αρκετά τεχνολογικά ζητήματα και ζητήματα ασφάλειας δεδομένων που προκύπτουν από την εφαρμογή συστημάτων ηλεκτρονικού ιατρικού φακέλου.

Παρόλα αυτά υπάρχουν και ζητήματα που δεν μπορούν να προσεγγιστούν μέσω της τεχνικής δομής μιας αρχιτεκτονικής ή ενός πρωτοκόλλου επικοινωνίας. Στην κατηγορία αυτή ανήκουν η ενημέρωση και κατάρτιση των ανθρώπων που θα χρησιμοποιήσουν αυτά τα συστήματα, το κατά πόσο θα κατανοήσουν την χρησιμότητα και την αναγκαιότητά τους και, εν τέλει, η αποδοχή των εν λόγω συστημάτων.

Το κεφάλαιο αυτό αναφέρεται στα αποτελέσματα της έρευνας που διεξήχθη σχετικά με την χρήση του Ηλεκτρονικού Φακέλου Ασθενούς και του Ηλεκτρονικού Προσωπικού Ιατρικού Φακέλου και στην ερευνητική μεθοδολογία που ακολουθήθηκε.

### 3.1 Στόχος της έρευνας

Η έρευνα που διεξήχθη είχε ως σκοπό να προβάλει:

- το κατά πόσο ο κόσμος είναι ενημερωμένος σχετικά με τις νέες τεχνολογίες στον τομέα της ιατρικής περίθαλψης
- τις ανησυχίες του σχετικά με την εφαρμογή συστημάτων ηλεκτρονικού φακέλου ασθενούς
- το αν είναι διατεθειμένος να χρησιμοποιήσει και να υποστηρίξει τέτοια συστήματα.

### 3.2 Δείγμα της έρευνας

Η δειγματοληπτική έρευνα με ερωτηματολόγια, συνεντεύξεις και στατιστικές μεθόδους ανάλυσης είναι η πιο διαδεδομένη μορφή εμπειρικής έρευνας. Η τυποποίηση των στοιχείων που συλλέγονται, η δυνατότητα προσέγγισης μεγάλου μέρους του πληθυσμού, η επιδεκτικότητα των στοιχείων σε στατιστικές μεθόδους ανάλυσης για την ανάδειξη γενικών



τάσεων καθιστούν την δειγματοληπτική έρευνα την πλέον καθιερωμένη μέθοδο για την μελέτη των κοινωνικών απόψεων και δεδομένων.

Για την επιλογή αντιπροσωπευτικού δείγματος από το δειγματοληπτικό πλαίσιο προσφέρονται διαφορετικές μέθοδοι αλλά με κοινό χαρακτηριστικό: Όλες στηρίζονται στην θεωρία των πιθανοτήτων, κάθε άτομο έχει μια γνωστή μη μηδενική πιθανότητα να επιλεγεί στο δείγμα. Στην συγκεκριμένη έρευνα, χρησιμοποιήθηκε η απλή τυχαία δειγματοληψία η οποία είναι, ίσως, η πιο γνωστή μέθοδος για την επιλογή αντιπροσωπευτικού δείγματος. Χαρακτηριστικό της μεθόδου είναι ότι κάθε μονάδα του πληθυσμού έχει την ίδια πιθανότητα να επιλεγεί στο δείγμα, σε κάθε κλήρωση η πιθανότητα επιλογής είναι ίδια για όλες τις μονάδες. Η απλή τυχαία δειγματοληψία οδηγεί σε αντιπροσωπευτικό δείγμα του υπό έρευνα πληθυσμού και συνεπώς τα αποτελέσματα είναι γενικεύσιμα προς τον πληθυσμό.

### 3.3 Ερευνητική μεθοδολογία

Η μεθοδολογία της έρευνας περιλαμβάνει μια σειρά ερευνητικών εργαλείων, ποσοτικών και ποιοτικών. Οι ποιοτικές και ποσοτικές προσεγγίσεις έχουν πολλά κοινά: Και οι δύο προσπαθούν να συλλέξουν ακριβείς πληροφορίες για τις κοινωνικές απόψεις, έχουν όμως και διαφορές ως προς το είδος των πληροφοριών και τον τρόπο που τις συλλέγουν. Οι ποσοτικές μέθοδοι επικεντρώνονται σε αριθμητικά δεδομένα και σε στατιστικές συγκρίσεις, στην μέτρηση των θεωρητικών εννοιών μέσω εργαλείων, όπως είναι το τυποποιημένο ερωτηματολόγιο, προκειμένου να εξάγουν αιτιακές σχέσεις. Οι ποιοτικές μέθοδοι και τεχνικές αποτελούν αναγκαία εργαλεία στην ανάλυση ποσοτικών στοιχείων και στην διατύπωση νέων υποθέσεων. Αποτελούν τα βασικά εργαλεία για την συλλογή στοιχείων που δίνουν απάντηση σε ερωτήσεις που αφορούν το «πώς, γιατί, κάτω από ποιες συνθήκες, με ποιες επιπτώσεις». Η συνδυαστική και συμπληρωματική χρήση ποσοτικών και ποιοτικών μεθόδων αποτελεί την πλέον αποτελεσματική μεθοδολογική προσέγγιση για μια ολοκληρωμένη περιγραφή και εξήγηση των διαστάσεων της κοινωνικής πραγματικότητας.

Η διεξαγωγή της έρευνας έγινε σε δύο μέρη:

#### 3.3.1 Πρωτογενής έρευνα (Ποιοτική)

Ως ποιοτική έρευνα μπορούμε να ορίσουμε την διερεύνηση σε βάθος και την δυναμική αξιολόγηση στάσεων, απόψεων, κινήτρων, συναισθημάτων και αντιδράσεων απέναντι σε νέες έννοιες που μπορούν να αφορούν προϊόντα ή υπηρεσίες. Η ποιοτική έρευνα εντοπίζει

και εξηγεί τα βαθύτερα αίτια που καθορίζουν την συμπεριφορά των ατόμων, όπως επίσης τις βαθύτερες στάσεις και αντιλήψεις τους πάνω σε ένα θέμα.

Σε αυτό το στάδιο της έρευνας ακολουθήθηκαν οι παρακάτω τεχνικές συλλογής δεδομένων:

1. **Τεχνική ανάλυσης αρχείων:** Μελέτη και ανάλυση της βιβλιογραφίας και αρθρογραφίας σχετικά με τον ηλεκτρονικό ιατρικό φάκελο. Μελετήθηκαν κείμενα που έχουν σχέση με το θέμα της έρευνας, με σκοπό να αναγνωριστούν οι τάσεις και οι απόψεις που υπάρχουν για το συγκεκριμένο θέμα. Το αποτέλεσμα αυτής της μελέτης έχει αποτυπωθεί στα πρώτα κεφάλαια της διπλωματικής εργασίας.
2. **Τεχνική συνεντεύξεων:** Σκοπός μας σε αυτό το στάδιο ήταν η συλλογή απόψεων και η καταγραφή γνώσεων σχετικά με το υπό διερεύνηση θέμα. Ακολουθήθηκε η τεχνική της μη τυποποιημένης συνέντευξης με την μορφή άτυπης συζήτησης, με στόχο να παροτρυνθούν οι συμμετέχοντες στην έρευνα να εκφράσουν και να περιγράψουν την δική τους άποψη σχετικά με το αντικείμενο της έρευνας. Οι συνεντευξιζόμενοι ανήκαν στις παρακάτω κατηγορίες:
  - a. Ιατροί (τρεις στον αριθμό, με διαφορά στην ηλικιακή κατηγορία που ανήκουν)
  - b. Άτομα με εξοικείωση στις νέες τεχνολογίες (επαγγελματίες του χώρου της πληροφορικής, δύο στον αριθμό)
  - c. Άτομα που δεν ανήκουν σε καμία από τις παραπάνω κατηγορίες και θα μπορούσαν να αποτελέσουν το μέσο όρο (τρεις στον αριθμό, με διαφορά στην ηλικιακή κατηγορία που ανήκουν)

Τα αποτελέσματα της πρωτογενούς έρευνας χρησιμοποιήθηκαν στην δομή και διατύπωση του ερωτηματολογίου της δευτερογενούς έρευνας.

### 3.3.2 Δευτερογενής έρευνα (Ποσοτική)

Η δυνατότητα συλλογής στοιχείων από μεγάλο αριθμό ατόμων για τα ίδια θέματα και συνεπώς η συγκρισιμότητα, η δυνατότητα ποσοτικοποίησης και στατιστικής ανάλυσης των στοιχείων που συλλέγονται καθιστούν το τυποποιημένο ερωτηματολόγιο το κύριο εργαλείο της ποσοτικής έρευνας. Οι απαντήσεις των ερωτώμενων μετατρέπονται σε δείκτες των υπό έρευνα φαινομένων, με σκοπό τη στατιστική τους επεξεργασία και την κατάδειξη εμπειρικών γενικεύσεων.

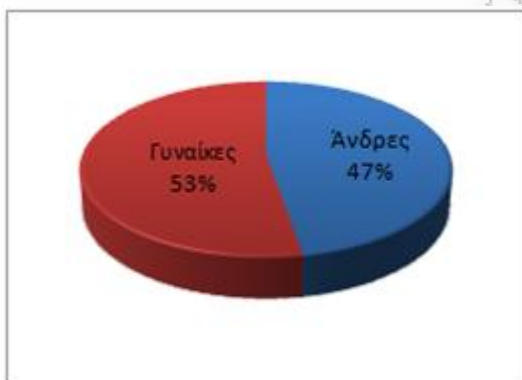
Το ερωτηματολόγιο συμπληρώθηκε τόσο μέσω διαδικτύου(στα πλαίσια μιας διαδικτυακής κοινότητας), όσο και σε χειρόγραφη μορφή(τυπωμένα ερωτηματολόγια). Πρόκειται για τυποποιημένο ερωτηματολόγιο, όπου οι ερωτήσεις είναι προκαθορισμένες, ακολουθούν μια συγκεκριμένη σειρά και έχουν κλειστή μορφή. Στόχος είναι οι διαφορές στις απαντήσεις να είναι συγκρίσιμες, κάτι το οποίο προϋποθέτει οι ερωτήσεις να ερμηνεύονται με τον ίδιο τρόπο από όλους τους ερωτώμενους.

### 3.4 Τα αποτελέσματα της ποσοτικής έρευνας

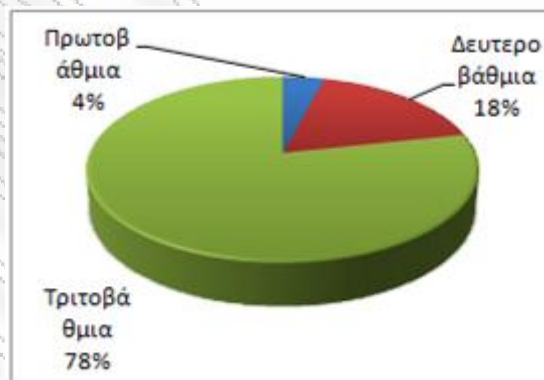
Το ερωτηματολόγιο που χρησιμοποιήθηκε παρουσιάζεται στο Παράρτημα Α.

Στην ποσοτική έρευνα συμμετείχαν συνολικά 74 άτομα, τα οποία μπορούν να κατηγοριοποιηθούν ως εξής:

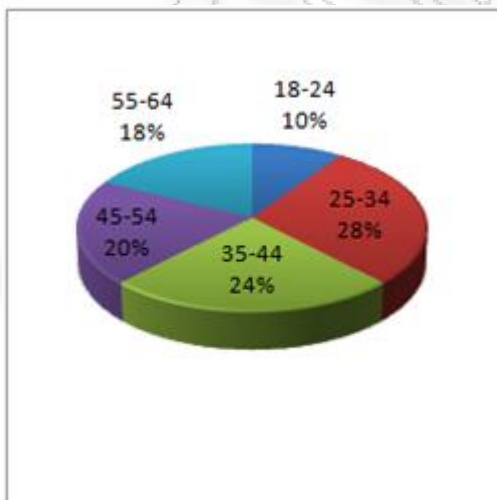
Ως προς το φύλο:



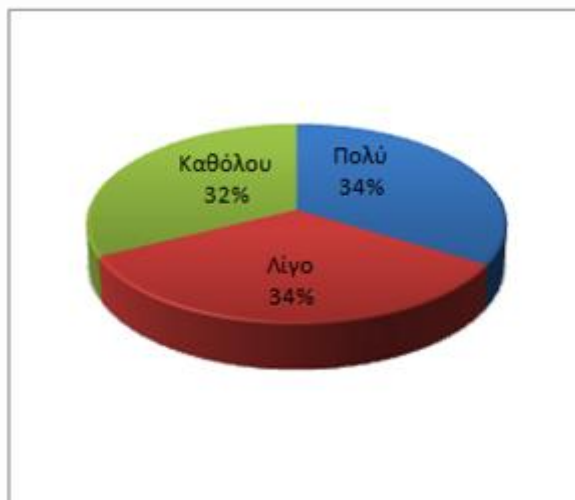
Ως προς την βαθμίδα εκπαίδευσης:



Ως προς την ηλικιακή κατηγορία:

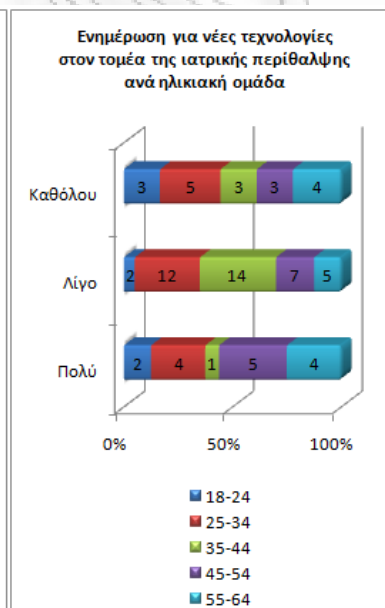
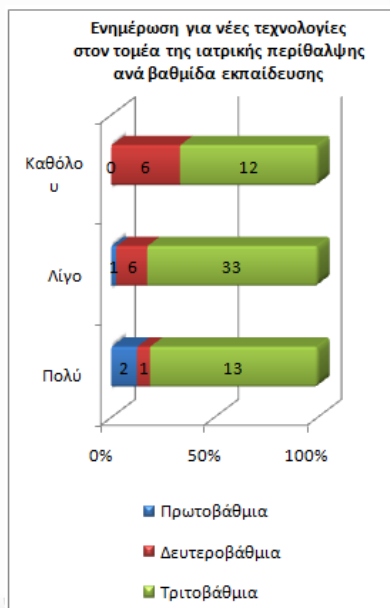


Ως προς την εξοικείωση με τις νέες τεχνολογίες :

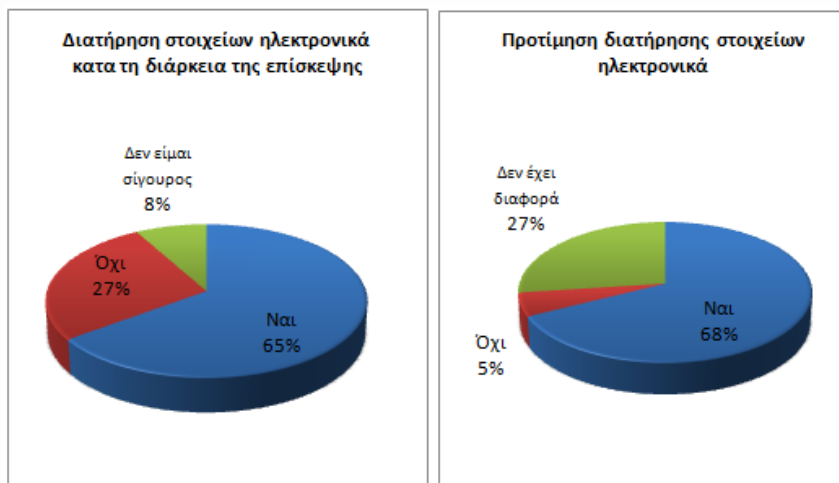


Παρακάτω θα αναφερθούμε στα σημαντικότερα αποτελέσματα της έρευνας:

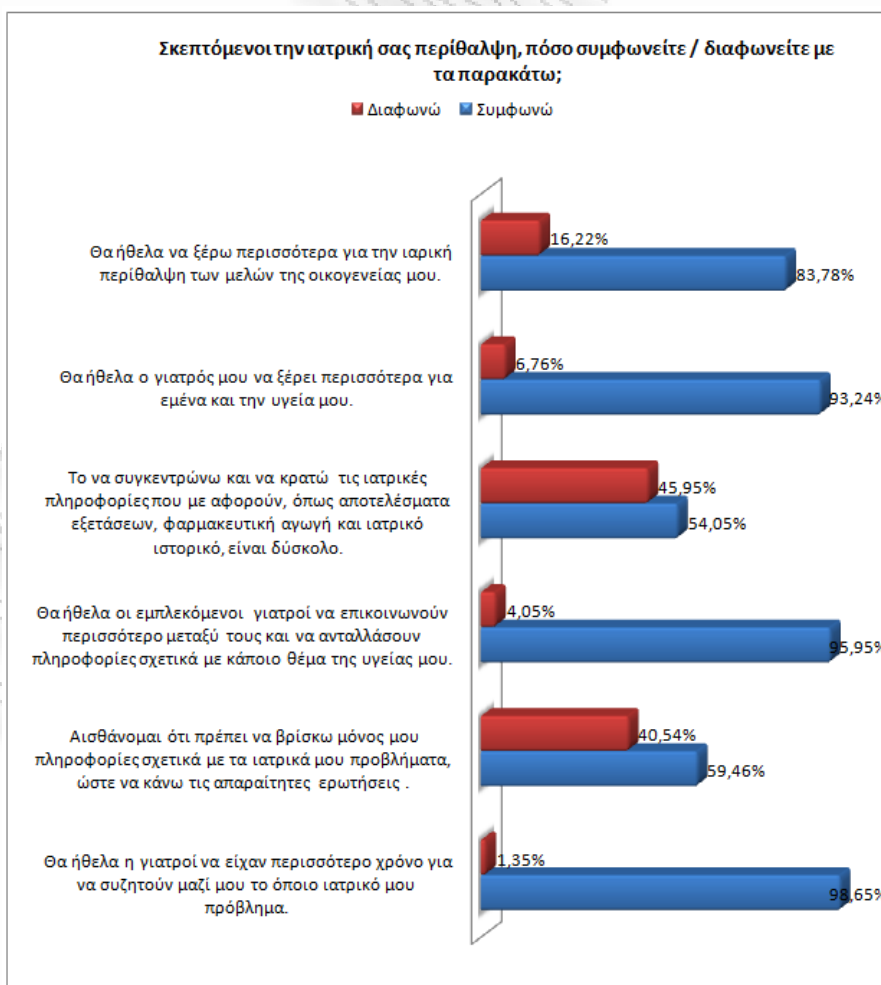
1. Σχετικά με την ενημέρωση για τις νέες τεχνολογίες στον τομέα της ιατρικής περίθαλψης, όπως ο ηλεκτρονικός φάκελος, οι ιστοσελίδες φύλαξης ιατρικών δεδομένων και άλλα, οι περισσότεροι απαντήσαντες είχαν μια σχετική ενημέρωση για τις τεχνολογικές εξελίξεις στον τομέα.



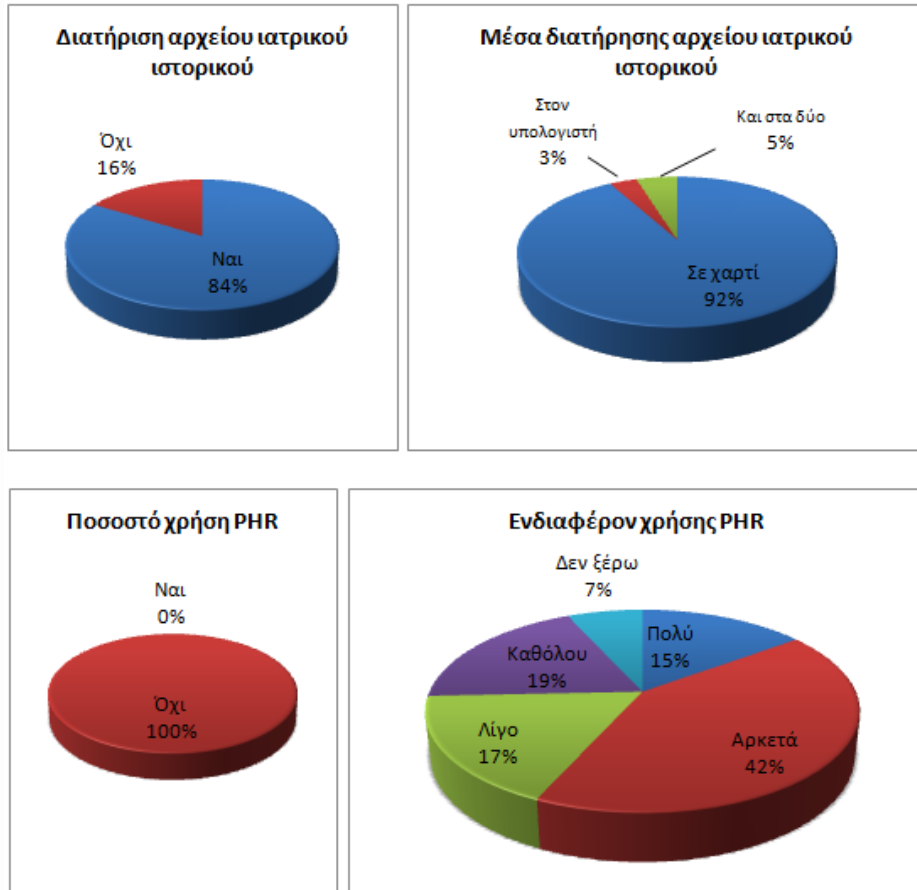
2. Οι περισσότεροι απαντήσαντες δήλωσαν ότι κατά την διάρκεια μιας επίσκεψής τους ο ιατρός διατηρεί κάποια από τα στοιχεία τους σε ηλεκτρονική μορφή και έδειξαν την προτίμησή τους σε αυτόν τον τρόπο αποθήκευσης δεδομένων.



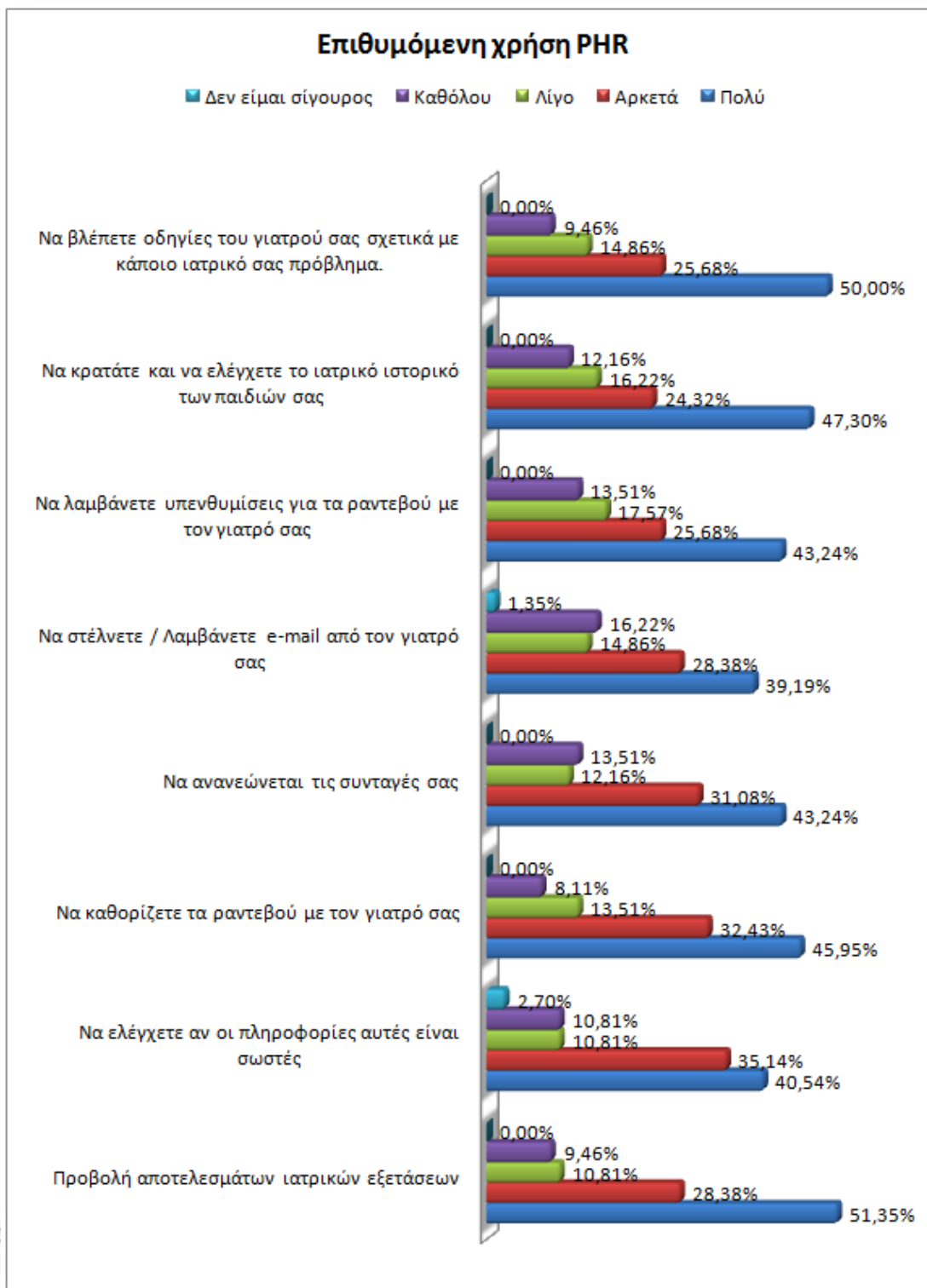
3. Οι περισσότεροι απαντήσαντες θα προτιμούσαν οι θεράποντες ιατροί τους να επικοινωνούν μεταξύ τους, να γνωρίζουν περισσότερα σχετικά με το πρόβλημα της υγείας που αντιμετωπίζουν και να τους διαθέτουν περισσότερο χρόνο, ενώ οι απόψεις δίστανται σχετικά με την δυσκολία αναζήτησης, συγκέντρωσης και διατήρησης ιατρικών δεδομένων και πληροφοριών.



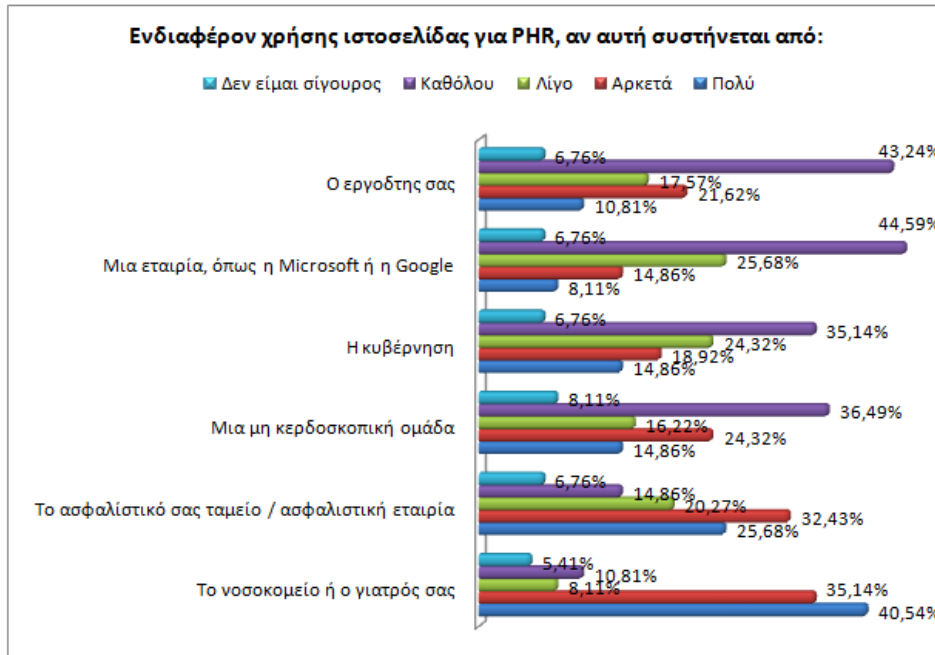
4. Το μεγαλύτερο ποσοστό των ερωτηθέντων διατηρούν αρχείο με το ιατρικό τους ιστορικό, με την συντριπτική πλειοψηφία να χρησιμοποιεί το χαρτί ως μέσο. Κανένας από τους συμμετέχοντες στην έρευνα δεν διατηρεί το ιστορικό του σε κάποια ιστοσελίδα, χωρίς όμως να απορρίπτουν αυτή τη εναλλακτική.



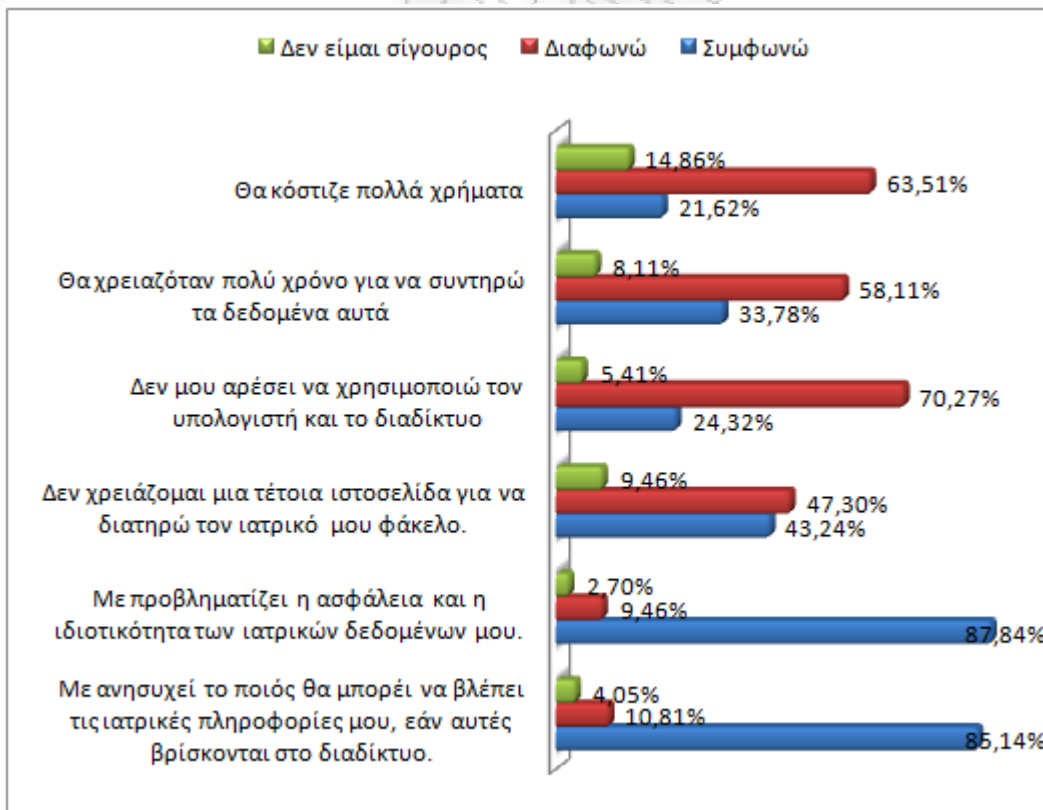
5. Σε σχέση με την επιθυμούμενη χρήση μιας ιστοσελίδας διατήρησης προσωπικών ιατρικών δεδομένων, το μεγαλύτερο ενδιαφέρον εντοπίζεται στην προβολή αποτελεσμάτων ιατρικών εξετάσεων, στον έλεγχο της ορθότητας των ιατρικών δεδομένων, στον καθορισμό ραντεβού και στην προβολή οδηγιών από γιατρούς, ενώ ακολουθούν με εξίσου μεγάλα ποσοστά η ανανέωση συνταγών και ο έλεγχος του ιστορικού των παιδιών.



6. Οι ερωτηθέντες θα σκεφτόντουσαν να χρησιμοποιήσουν μια ιστοσελίδα διατήρησης ιατρικών δεδομένων αν αυτή συστήνονταν από τον ιατρό ή το ασφαλιστικό ταμείο τους, ενώ δεν θα έδειχναν ενδιαφέρον αν αυτή προτεινόταν από τον εργοδότη τους, την κυβέρνηση ή κάποια εταιρία πληροφορικής.



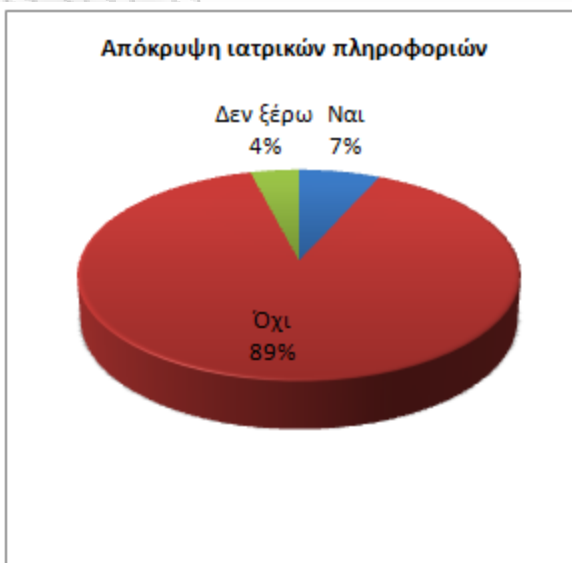
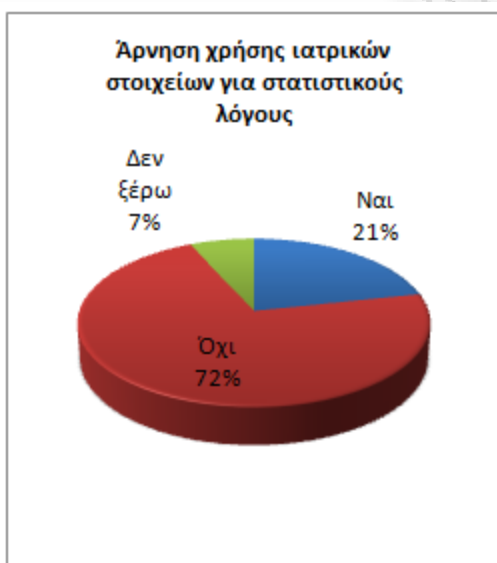
7. Ο κύριος προβληματισμός σχετικά με την χρήση μιας ιστοσελίδας καταχώρησης και διατήρησης ιατρικών δεδομένων εντοπίζεται στην ασφάλεια των δεδομένων και στον έλεγχο της πρόσβασης μη εξουσιοδοτημένων προσώπων σε αυτά.



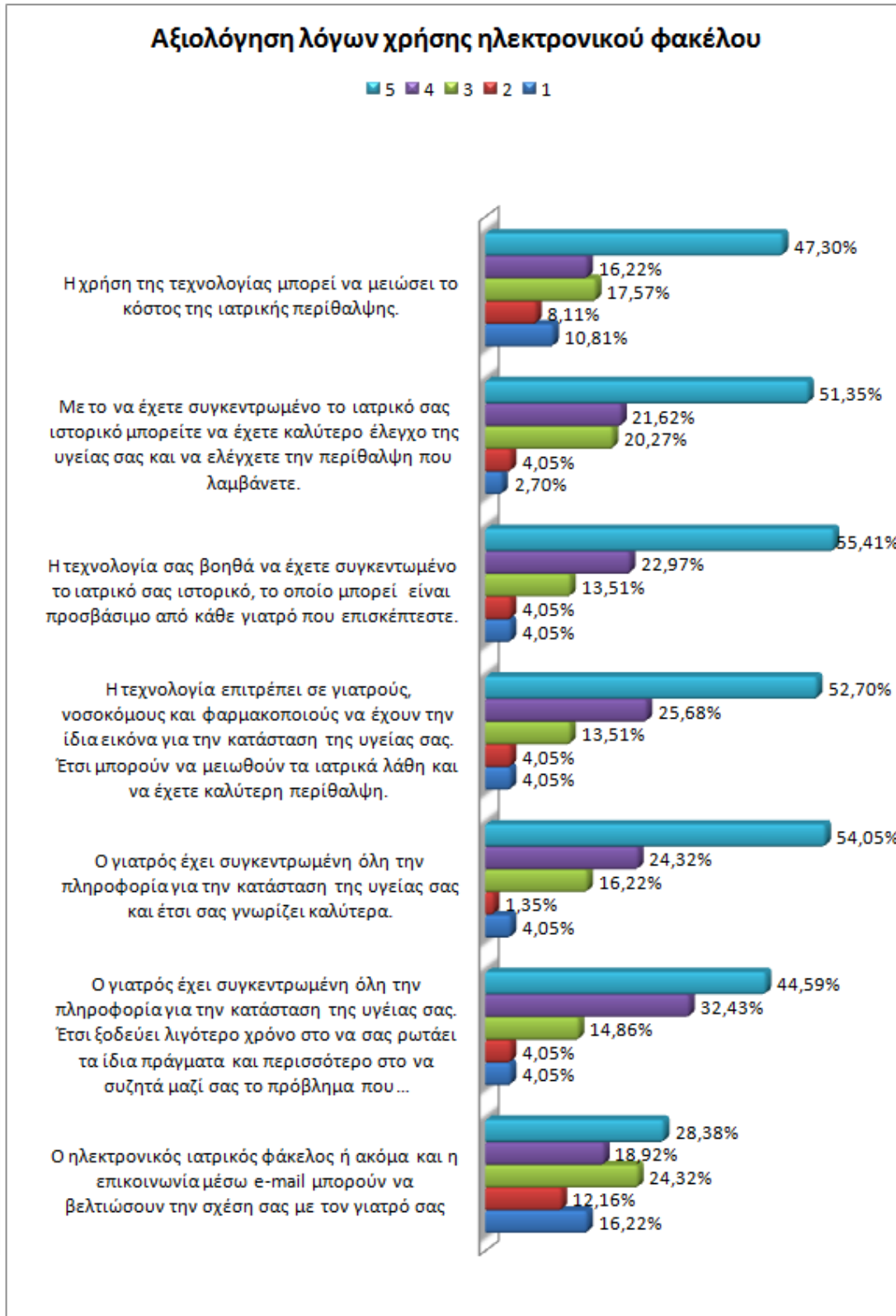




8. Σε σχέση με την διάθεση των ιατρικών τους δεδομένων για στατιστικούς και ερευνητικούς λόγους, οι περισσότεροι απαντήσαντες ήταν θετικοί, ενώ δηλώνουν ότι αυτός δεν θα ήταν λόγος απόκρυψης ιατρικών πληροφοριών από τον ιατρό τους.

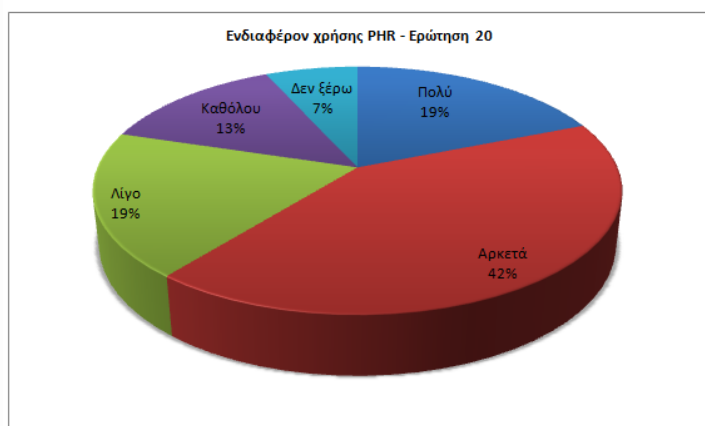
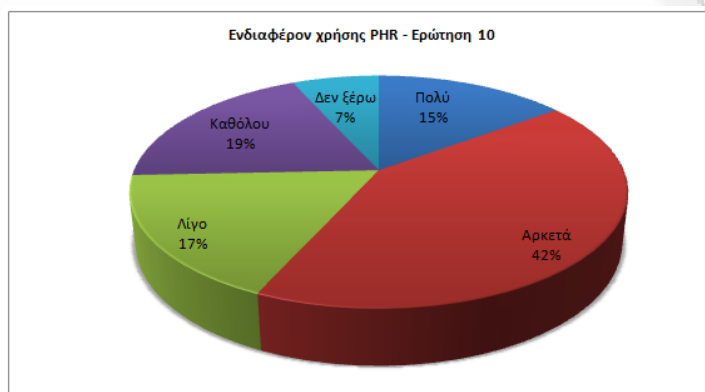


9. Οι περισσότεροι απαντήσαντες συμφωνούν ότι με την χρήση του ηλεκτρονικού φακέλου μπορούν να μειωθούν τα ιατρικά λάθη, ο ιατρός έχει καλύτερη και πιο ολοκληρωμένη εικόνα για την κατάσταση της υγείας τους και βοηθά στην συγκέντρωση και των έλεγχου των ιατρικών πληροφοριών από τους ίδιους.



10. Κατά την επανάληψη της ερώτησης για το ενδιαφέρον χρήσης κάποιας ιστοσελίδας διατήρησης του ιατρικού φακέλου στο τέλος του ερωτηματολογίου, αφού είχαν προηγηθεί οι λόγοι και τρόποι χρήσης της στις προηγούμενες ερωτήσεις,

παρατηρήσαμε μικρή αύξηση του ενδιαφέροντος προς τις σελίδες αυτές. Παρακάτω παραθέτουμε τα σχήματα με τα στατιστικά των αρχικών και τελικών απαντήσεων.



## Συμπεράσματα

Η ενημέρωση του ευρύτερου κοινού σχετικά με τις νέες τεχνολογίες στον τομέα της ιατρικής περίθαλψης, όπως ο ηλεκτρονικός φάκελος, οι ιστοσελίδες φύλαξης ιατρικών δεδομένων και άλλα, δεν είναι επαρκής για να υποστηρίξει την εισαγωγή και αποδοχή τέτοιων συστημάτων. Αν και αναγνωρίζουν τα οφέλη της χρήσης Ηλεκτρονικών Φακέλων Υγείας (προβολή αποτελεσμάτων ιατρικών εξετάσεων, έλεγχος της ορθότητας των ιατρικών δεδομένων, μείωση ιατρικών λαθών), υπάρχει έντονη ανησυχία σχετικά με την ασφάλεια των δεδομένων και τον έλεγχο της πρόσβασης μη εξουσιοδοτημένων προσώπων σε αυτά.

## Ενότητα Γ' Πληροφοριακό Σύστημα Υγείας

### Σχεδιασμός και Υλοποίηση



(Microsoft HealthVault Future, προσαρμόστηκε από XPLANE)<sup>[E13]</sup>

## Κεφάλαιο 4

# Ανάλυση του Συστήματος - Λειτουργικές Απαιτήσεις και Σχεδιασμός Αρχιτεκτονικής

### Εισαγωγή

Στις πρώτες ενότητες περιγράψαμε το θεωρητικό υπόβαθρο, στο οποίο βασίζεται ο Ηλεκτρονικός φάκελος Ασθενούς και η Service Oriented αρχιτεκτονική. Για την εφαρμογή αυτών, αποφασίστηκε η υλοποίηση ενός Πληροφοριακού Συστήματος Υγείας, το οποίο θα επικοινωνεί με ένα σύστημα Ηλεκτρονικού Φακέλου ασθενών. Σκοπός του πληροφοριακού συστήματος που αναπτύσσουμε είναι η αυτοματοποίηση της διαδικασίας διαχείρισης ασθενών σε μια οδοντιατρική κλινική.

Στο κεφάλαιο αυτό θα παρουσιάσουμε την ανάλυση και τον σχεδιασμό του Πληροφοριακού Συστήματος που αναπτύχθηκε. Θα περιγράψουμε το σύστημα που υλοποιήθηκε, την αρχιτεκτονική του και θα αναφερθούμε στις Υπηρεσίες που υλοποιήθηκαν. Τέλος, θα περιγράψουμε την βάση δεδομένων του συστήματος, μέσω Διαγραμμάτων Οντοτήτων - Συσχετίσεων και Σχεσιακών Μοντέλων.

### 4.1 Περιγραφή του προς υλοποίηση συστήματος

Το σύστημα που θα υλοποιήσουμε αποτελείται από τρεις συνεργαζόμενες εφαρμογές:

- 1) Εφαρμογή γραμματείας
- 2) Εφαρμογή θεράποντος ιατρού
- 3) Ηλεκτρονικός φάκελος ασθενούς

Στη συνέχεια θα περιγράψουμε συνοπτικά τις επιχειρησιακές διαδικασίες που υποστηρίζουν οι εφαρμογές αυτές.

#### Εφαρμογή γραμματείας

Η γραμματεία της κλινικής έχει έναν καθαρά διαχειριστικό ρόλο. Χρησιμοποιεί μια desktop εφαρμογή για να εκτελέσει τις παρακάτω διαδικασίες:

- Επεξεργασία ιατρών κλινικής: Η γραμματεία αναλαμβάνει να συνδέσει έναν ιατρό με την κλινική, κατά την έναρξη της συνεργασίας τους. Όσο ο ιατρός συνεργάζεται με την

κλινική, η γραμματεία μπορεί να επεξεργάζεται τα στοιχεία του ιατρού και να καθορίζει τα ραντεβού του. Αν η συνεργασία ιατρού-κλινικής λήξη, η γραμματεία αναλαμβάνει την αποσύνδεση του ιατρού από την κλινική.

- Επεξεργασία ασθενών κλινικής. Η γραμματεία αναλαμβάνει την καταχώρηση των στοιχείων των ασθενών, την διαχείριση των ραντεβού τους και τον χειρισμό των οικονομικών δεδομένων τους.

#### Εφαρμογή Θεράποντος Ιατρού

Κάθε ιατρός της κλινικής χρησιμοποιεί μια desktop εφαρμογή για να εκτελέσει τις παρακάτω διαδικασίες:

- Διαχείριση ραντεβού. Ο ιατρός μπορεί να δει τα ραντεβού του (σε επίπεδο ημέρας, εβδομάδας και μήνα)
- Διαχείριση ασθενών. Ο ιατρός μπορεί να δει τα στοιχεία των ασθενών του, να ορίσει και να επεξεργαστεί τα σχέδια θεραπείας τους. Επίσης, ο ιατρός μπορεί να δει τον ηλεκτρονικό φάκελο των ασθενών του.

#### Εφαρμογή Ηλεκτρονικού φακέλου

Πρόκειται για μια web εφαρμογή, την οποία χρησιμοποιεί ο ασθενής για να δει τα ραντεβού και τα στοιχεία του φακέλου του. Στην εφαρμογή αυτή ο ασθενής μπορεί να τροποποιήσει μόνο τα δημογραφικά του στοιχεία.

#### Βασικοί δρώντες (actors) του συστήματος

Με βάση την περιγραφή της λειτουργικότητας του συστήματος, μπορούμε να αναγνωρίσουμε τους παρακάτω χρήστες στο σύστημα:

- Γραμματεία
- Θεράπων ιατρός
- Ασθενής

## 4.2 Επιχειρησιακές διαδικασίες και λειτουργικές προδιαγραφές

Η καταγραφή των επιχειρησιακών διαδικασιών, που καλείται να καλύψει το σύστημά μας, είναι πολύ σημαντική για τον πλήρη σχεδιασμό της αρχιτεκτονικής του. Από την ανάλυση και την μοντελοποίηση των διαδικασιών αυτών προκύπτουν οι λειτουργικές προδιαγραφές του συστήματος. Λειτουργικές προδιαγραφές είναι οι δηλώσεις των υπηρεσιών που πρέπει να παρέχει το σύστημα, πως το σύστημα θα πρέπει να αντιδράσει με την εισαγωγή συγκεκριμένων δεδομένων και πως το σύστημα θα πρέπει να συμπεριφέρεται σε συγκεκριμένες καταστάσεις. Σε μερικές περιπτώσεις οι λειτουργικές απαιτήσεις μπορεί επίσης να εκφράζουν σαφώς τι δεν θα πρέπει να κάνει το σύστημα. Με βάση τις επιχειρησιακές διαδικασίες μπορούμε να αναγνωρίσουμε τις παρακάτω λειτουργικές προδιαγραφές:

A/A	ΟΜΑΔΑ ΧΡΗΣΤΩΝ	ΠΕΡΙΓΡΑΦΗ ΑΠΑΙΤΗΣΗΣ	ΠΡΟΔΙΑΓΡΑΦΗ	ΕΠΙΠΕΔΟ ΑΝΑΓΚΑΙΟΤΗΤΑΣ
1	Θεράπων ιατρός, Γραμματεία	Θα πρέπει να μπορούν να εισέλθουν στο σύστημα με το λογαριασμό τους, χρησιμοποιώντας δηλαδή το username και το password τους.	Σύνδεση του χρήστη (είσοδος- login )	Απαραίτητη
2	Γραμματεία	Θα πρέπει να μπορεί να δει όλα τα ραντεβού όλων των ιατρών της κλινικής	Προβολή ραντεβού ιατρού	Απαραίτητη
3	Γραμματεία	Θα πρέπει να μπορεί να προσθέσει ραντεβού σε επιλεγμένο ιατρό.	Προσθήκη ραντεβού	Απαραίτητη
4	Γραμματεία	Θα πρέπει να μπορεί να επεξεργαστεί τα στοιχεία επιλεγμένου ραντεβού	Επεξεργασία ραντεβού	Απαραίτητη
5	Γραμματεία	Θα πρέπει να μπορεί να συνδέσει έναν ιατρό με την κλινική	Προσθήκη ιατρού	Απαραίτητη
6	Γραμματεία	Θα πρέπει να μπορεί να επεξεργαστεί τα στοιχεία ενός ιατρού της κλινικής	Επεξεργασία ιατρού	Απαραίτητη
7	Γραμματεία	Θα πρέπει να μπορεί να αποσυνδέσει έναν ιατρό από την κλινική μετά τη λήξη της συνεργασίας τους	Διαγραφή ιατρού	Απαραίτητη
8	Γραμματεία	Θα πρέπει να μπορεί να αναζητήσει έναν ιατρό της κλινικής	Αναζήτηση ιατρού	Απαραίτητη
9	Γραμματεία	Θα πρέπει να μπορεί να προσθέσει έναν ασθενή στην κλινική	Προσθήκη ασθενούς	Απαραίτητη

A/A	ΟΜΑΔΑ ΧΡΗΣΤΩΝ	ΠΕΡΙΓΡΑΦΗ ΑΠΑΙΤΗΣΗΣ	ΠΡΟΔΙΑΓΡΑΦΗ	ΕΠΙΠΕΔΟ ΑΝΑΓΚΑΙΟΤΗΤΑΣ
10	Γραμματεία	Θα πρέπει να μπορεί να επεξεργαστεί τα δημογραφικά στοιχεία ενός ασθενούς της κλινικής	Επεξεργασία ασθενούς	Απαραίτητη
11	Γραμματεία	Θα πρέπει να μπορεί να αναζητήσει έναν ασθενή της κλινικής	Αναζήτηση ασθενούς	Απαραίτητη
12	Γραμματεία	Θα πρέπει να μπορεί να δει τα ραντεβού ενός ασθενούς της κλινικής	Προβολή ραντεβού ασθενούς	Απαραίτητη
13	Γραμματεία	Θα πρέπει να μπορεί να δει τα χρέη και τις πληρωμές ενός ασθενούς	Προβολή οικονομικών στοιχείων ασθενούς	Απαραίτητη
14	Γραμματεία	Θα πρέπει να μπορεί να καταχωρήσει τις πληρωμές του ασθενούς	Εισαγωγή πληρωμής	Απαραίτητη
15	Γραμματεία	Θα πρέπει να μπορεί να δει βασικά στοιχεία για τα πλάνα θεραπείας του ασθενούς	Προβολή πλάνων θεραπείας	Προαιρετική
16	Θεράπων ιατρός	Θα πρέπει να μπορεί να δει τα ραντεβού του	Προβολή ραντεβού ιατρού	Απαραίτητη
17	Θεράπων ιατρός	Θα πρέπει να μπορεί να αναζητήσει τους ασθενείς του	Αναζήτηση ασθενούς	Απαραίτητη
18	Θεράπων ιατρός	Θα πρέπει να μπορεί να δει τα ραντεβού που έχει με τον συγκεκριμένο ασθενή	Προβολή ραντεβού ασθενούς	Προαιρετική
19	Θεράπων ιατρός	Θα πρέπει να μπορεί να δει τα σχέδια θεραπείας του ασθενούς	Προβολή σχεδίων θεραπείας	Απαραίτητη
20	Θεράπων ιατρός	Θα πρέπει να μπορεί να προσθέσει ένα σχέδιο θεραπείας σε επιλεγμένο ασθενή	Προσθήκη σχεδίου θεραπείας	Απαραίτητη
21	Θεράπων ιατρός	Θα πρέπει να μπορεί να επεξεργαστεί ένα σχέδιο θεραπείας ενός ασθενούς	Επεξεργασία σχεδίου θεραπείας	Απαραίτητη
22	Θεράπων ιατρός	Θα πρέπει να μπορεί να διαγράψει ένα λανθασμένο σχέδιο θεραπείας, αν αυτό δεν έχει αρχίσει να εκτελείται.	Διαγραφή σχεδίου θεραπείας	Προαιρετική
23	Θεράπων ιατρός	Θα πρέπει να μπορεί να καταχωρήσει μια συνταγή για τον ασθενή	Καταχώρηση συνταγής	Απαραίτητη
24	Θεράπων	Θα πρέπει να μπορεί να επεξεργαστεί	Επεξεργασία	Απαραίτητη

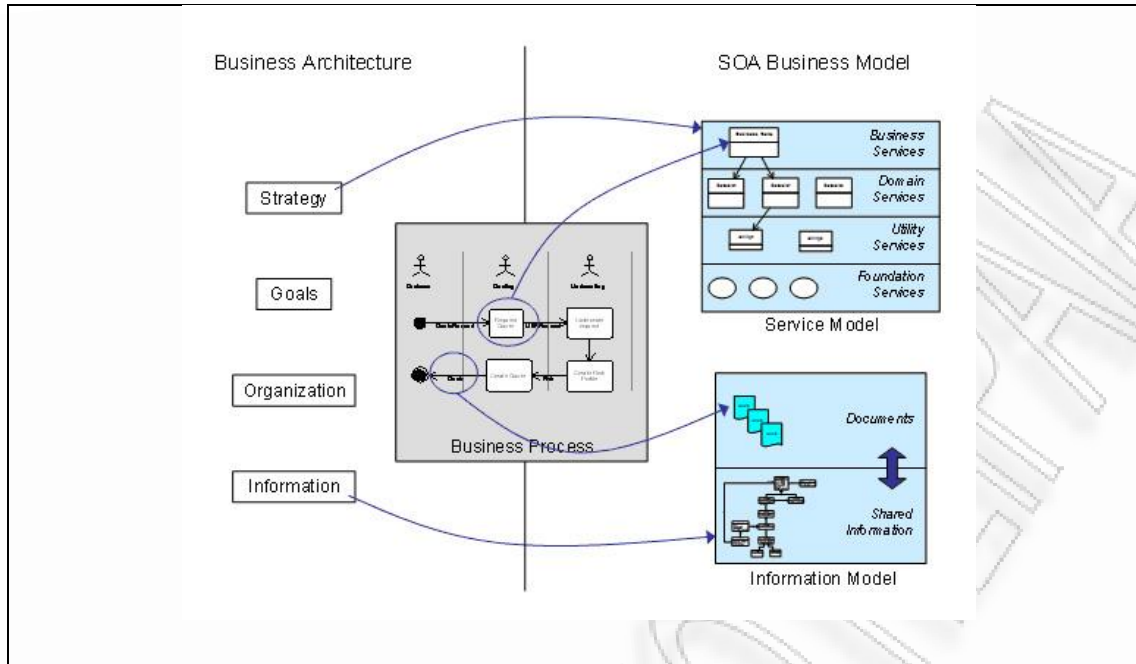


A/A	ΟΜΑΔΑ ΧΡΗΣΤΩΝ	ΠΕΡΙΓΡΑΦΗ ΑΠΑΙΤΗΣΗΣ	ΠΡΟΔΙΑΓΡΑΦΗ	ΕΠΙΠΕΔΟ ΑΝΑΓΚΑΙΟΤΗΤΑΣ
	ιατρός	μια συνταγή ενός ασθενούς	συνταγής	
25	Θεράπων ιατρός	Θα πρέπει να μπορεί να δει τον ιατρικό φάκελο του ασθενούς	Προβολή ιατρικού φακέλου	Απαραίτητη
26	Ασθενής	Θα πρέπει να μπορεί να εισέλθει στο σύστημα Ηλεκτρονικού Φακέλου με το λογαριασμό του, χρησιμοποιώντας δηλαδή το username και το password του.	Σύνδεση του χρήστη (είσοδος-login)	Απαραίτητη
27	Ασθενής	Θα πρέπει να μπορεί να δει τα πλήρη στοιχεία του Ηλεκτρονικού του φακέλου	Προβολή ιατρικού φακέλου	Απαραίτητη
28	Ασθενής	Θα πρέπει να μπορεί να τροποποιήσει τα δημογραφικά του στοιχεία	Επεξεργασία Δημογραφικών στοιχείων	Απαραίτητη
29	Ασθενής	Θα πρέπει να μπορεί να δημιουργήσει τον Ηλεκτρονικό Ιατρικό του φάκελο	Δημιουργία Ηλεκτρονικού Ιατρικού φακέλου	Προαιρετική

### 4.3 Σχεδιασμός Αρχιτεκτονικής

Όπως ήδη αναφέρθηκε η SOA δεν ορίζει αυστηρούς δεσμούς επικοινωνίας ανάμεσα στα εμπλεκόμενα μέρη (components). Στις κλασικές πολλαπλών επιπέδων (multi-tier) αρχιτεκτονικές τα επιμέρους επίπεδα επικοινωνούν με σύνθετους και αυστηρά καθορισμένους τρόπους, ενώ πολλές φορές η ύπαρξη ενός επιπέδου προϋποθέτει την ύπαρξη κάποιου άλλου. Στη SOA τα επιμέρους μέρη είναι ανεξάρτητα μεταξύ τους και παρέχουν υπηρεσίες προς όλους τους ενδιαφερόμενους. Οι διεπαφές επικοινωνίας και η σύνθεση των υπηρεσιών για τον ορισμό μιας ολοκληρωμένης διαδικασίας δεν απαιτούν αυστηρές συνδέσεις μεταξύ τους. Κάθε Web είναι αυτόνομο και μπορεί να συμβάλει στον ορισμό παραπάνω από μιας διαδικασίας (process definition).

Για τον σχεδιασμό της αρχιτεκτονικής του συστήματος χρησιμοποιήθηκε η «Από Πάνω προς τα Κάτω» (Top Down) μεθοδολογία. Ξεκινώντας από το επίπεδο των επιχειρησιακών διαδικασιών και των λειτουργικών προδιαγραφών καταλήξαμε στις επιμέρους σύνθετες και απλούστερες υπηρεσίες που έπρεπε να δημιουργηθούν και στην ενορχήστρωση αυτών.

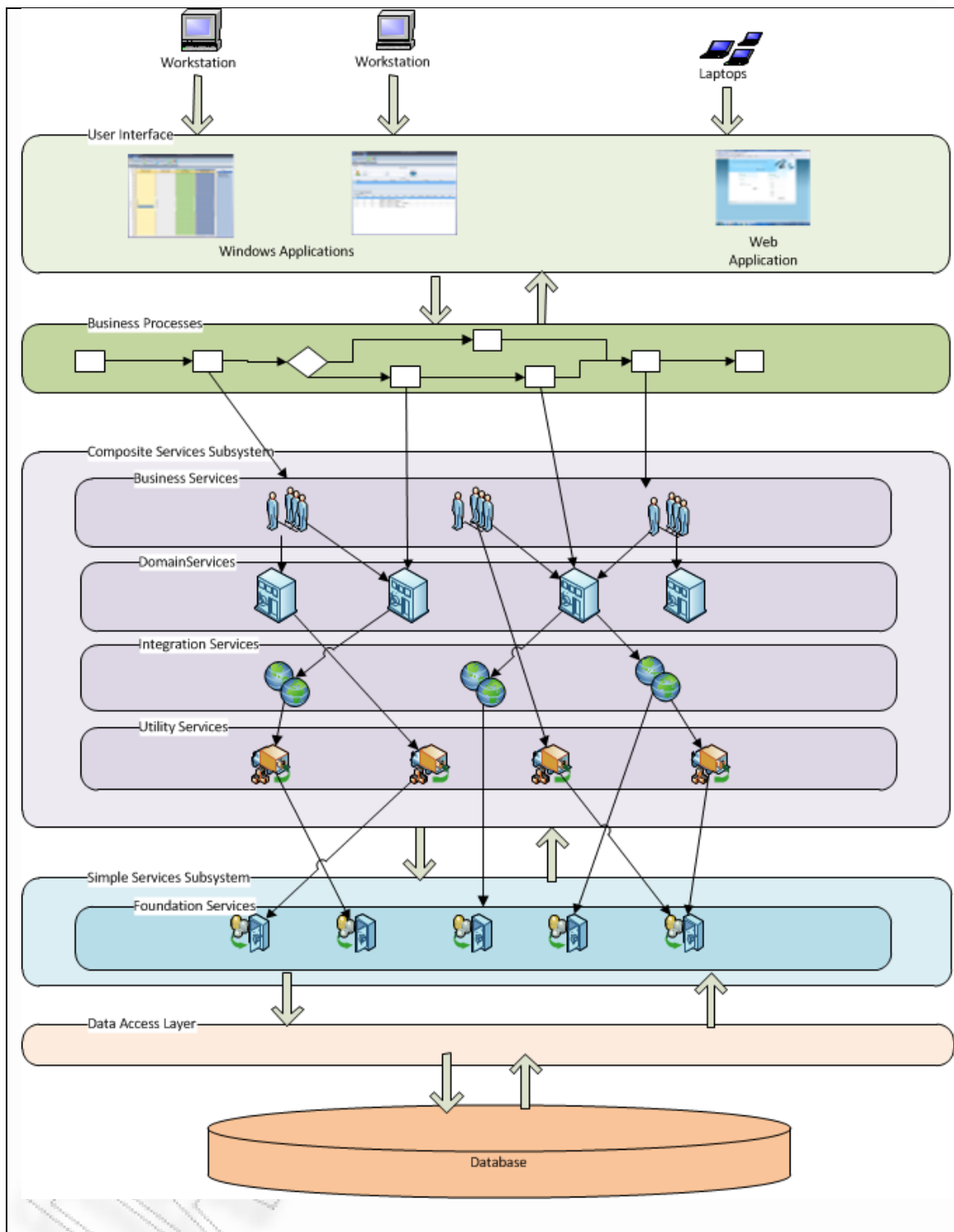


Εικόνα 12: Top Down Σχεδιασμός Αρχιτεκτονικής [E14]

Το σύστημα που υλοποιήθηκε αποτελείται από πέντε κύρια υποσυστήματα:

1. Υποσύστημα Διεπαφής Χρήστη
2. Υποσύστημα Επιχειρηματικών διαδικασιών
3. Υποσύστημα Σύνθετων Υπηρεσιών
4. Υποσύστημα Απλών Υπηρεσιών
5. Υποσύστημα Βάσης Δεδομένων

Η αλληλεπίδραση των συστημάτων φαίνεται στην παρακάτω εικόνα.



Εικόνα 13: Η αρχιτεκτονική του συστήματος

#### 4.3.1 Υποσύστημα Διεπαφής Χρήστη

Είναι το υποσύστημα της εφαρμογής που αφορά το γραφικό της περιβάλλον και αποτελεί την διαπαφή του συστήματος με τους τελικούς χρήστες. Επιτρέπει, δηλαδή, στους χρήστες να πραγματοποιήσουν διαδικασίες login, εγγραφής, αναζήτησης πληροφοριών κ.α., όπως αυτά

αναφέρθηκαν παραπάνω. Η κάθε λειτουργία αυτού του υποσυστήματος αλληλεπιδρά με το Υποσύστημα των Coarse-Grained Υπηρεσιών.

Για την δημιουργία του User-Interface subsystem, χρησιμοποιήθηκαν τεχνολογίες του .NET framework και συγκεκριμένα Windows Forms και ASP.NET, με χρήση της γλώσσας C#. Το User Interface των εφαρμογών παρουσιάζεται αναλυτικότερα στο Παράρτημα Β.

#### 4.3.2 Υποσύστημα Επιχειρηματικών Διαδικασιών

Στο υποσύστημα αυτό υπάρχουν οι επιχειρησιακές διαδικασίες, που καλύπτει το σύστημά μας. Οι διαδικασίες αυτές παρουσιάστηκαν, μέσω των λειτουργικών προδιαγραφών που απορρέουν από αυτές, στην ενότητα 4.2. Οι επιχειρησιακές διαδικασίες αναλαμβάνουν την χορογραφία των Υπηρεσιών των δύο αμέσως επόμενων υποσυστημάτων (υποσύστημα Σύνθετων Υπηρεσιών και Υποσύστημα Απλών Υπηρεσιών). Αν και υπάρχουν μοντέλα και τυποποιήσεις που επιτρέπουν την δομημένη υλοποίηση των επιχειρησιακών διαδικασιών (BPEL, workflows, WS-CDL), στην παρούσα εργασία οι επιχειρησιακές διαδικασίες δεν έχουν υλοποιηθεί με κάποιο συγκεκριμένο εργαλείο. Η χορογραφία των υπηρεσιών γίνεται μέσω του κώδικα υλοποίησης της διεπαφής του συστήματος.

#### 4.3.3 Υποσύστημα Σύνθετων Υπηρεσιών

Το υποσύστημα των Σύνθετων Υπηρεσιών επεξεργάζεται όλες τις διαδικασίες και επιτρέπει σε εξωτερικά components να χρησιμοποιήσουν τις υπηρεσίες του συστήματος. Αποτελεί το orchestration component του συστήματος. Λαμβάνει είσοδο από εξωτερικά αιτούμενα μέρη (external requesting components) και εκτελεί το process καλώντας πολλά άλλα components, ώστε να επιστρέψει την απάντηση στο αρχικό component που πραγματοποίησε την κλήση του.

Στο υποσύστημα αυτό συναντάμε τέσσερα είδη υπηρεσιών:

- **Επιχειρησιακές Υπηρεσίες(Business Services):** Είναι οι υπηρεσίες με τον υψηλότερο βαθμό λειτουργικότητας. Οι υπηρεσίες αυτές συνθέτουν την λειτουργικότητα τους από άλλες υπηρεσίες του επιπέδου αυτού, με χαμηλότερη λειτουργικότητα. Οι Επιχειρησιακές Υπηρεσίες του συστήματός μας είναι, ουσιαστικά, ολόκληρες ροές εργασίας(workflows)που έχουν δημοσιευτεί σαν υπηρεσίες (published as services).
- **Τοπικές Υπηρεσίες(Domain Services).** Είναι υπηρεσίες με μεσαίου βαθμού λειτουργικότητα, οι οποίες αφορούν διάφορα domains του συστήματος και

χρησιμοποιούνται εντός των περιοχών αυτών. Για παράδειγμα, έχουμε τοπικές υπηρεσίες που αφορούν την Κλινική, τον Γιατρό, τον Ασθενή και άλλα.

- **Υπηρεσίες ολοκλήρωσης(Integration Services).** Είναι υπηρεσίες με μεσαίου και χαμηλού βαθμού λειτουργικότητα που φροντίζουν για την συνεπή προβολή δεδομένων στο πληροφοριακό Σύστημα Υγείας και στο Σύστημα Ηλεκτρονικού Φακέλου. Ουσιαστικά συλλέγουν δεδομένα που είναι κοινά και στα δύο συστήματα, ώστε η πληροφορία που παρουσιάζεται σε καθένα από αυτά να είναι πλήρης.
- **Υπηρεσίες Υποστήριξης (Utility Services).** Είναι υπηρεσίες με χαμηλού βαθμού λειτουργικότητα και παρέχουν βασική λειτουργικότητα που είναι σημαντική για αρκετές διαδικασίες.

#### 4.3.4 Υποσύστημα Απλών Υπηρεσιών

Το υποσύστημα αυτό περιλαμβάνει τις Βασικές Υπηρεσίες (Foundation Services) του συστήματος. Τα Fine grained Services μέσω του Data Access Layer επικοινωνούν με τη βάση δεδομένων. Περιλαμβάνει μικρές και απλές μεθόδους που υλοποιούν διαδικασίες της βάσης δεδομένων (transactions, call procedures κ.α.). Πρόκειται για μονάδες κώδικα που εκτελούν συγκεκριμένες μεθόδους, ενώ επιτρέπουν σε εξωτερικά συστήματα να τις χρησιμοποιούν. Θωρακίζουν με ασφάλεια τα στοιχεία εκείνα που θα έθεταν σε κίνδυνο τη βάση δεδομένων. Έτσι χρησιμοποιώντας στοιχεία αυτού του υποσυστήματος κάθε κομμάτι του Coarse subsystem μπορεί επίσης να πραγματοποιεί CRUD (Create, Read, Update, and Delete) λειτουργίες στη βάση.

#### 4.3.5 Υποσύστημα Βάσης Δεδομένων

Τα διαγράμματα και τα μοντέλα που υλοποιούν τη βάση δεδομένων θα παρουσιαστούν αναλυτικά παρακάτω.

### 4.4 Οι Υπηρεσίες του συστήματος

Στην ενότητα αυτή θα παρουσιάσουμε αναλυτικές όλες τις υπηρεσίες που υλοποιήθηκαν (Coarse και Fine grained) προκειμένου ολοκληρωθούν και να αυτοματοποιηθούν οι επιχειρηματικές διαδικασίες. Κατά την ανάλυση των υπηρεσιών, και όπου αυτό είναι απαραίτητο, θα γίνεται μια σύντομη αναφορά στις ενέργειες που έχει εκτελέσει ο χρήστης μέσω του User Interface, οι οποίες έχουν οδηγήσει στην κλήση της Υπηρεσίας. Το User Interface θα παρουσιαστεί αναλυτικά σε επόμενο κεφάλαιο.

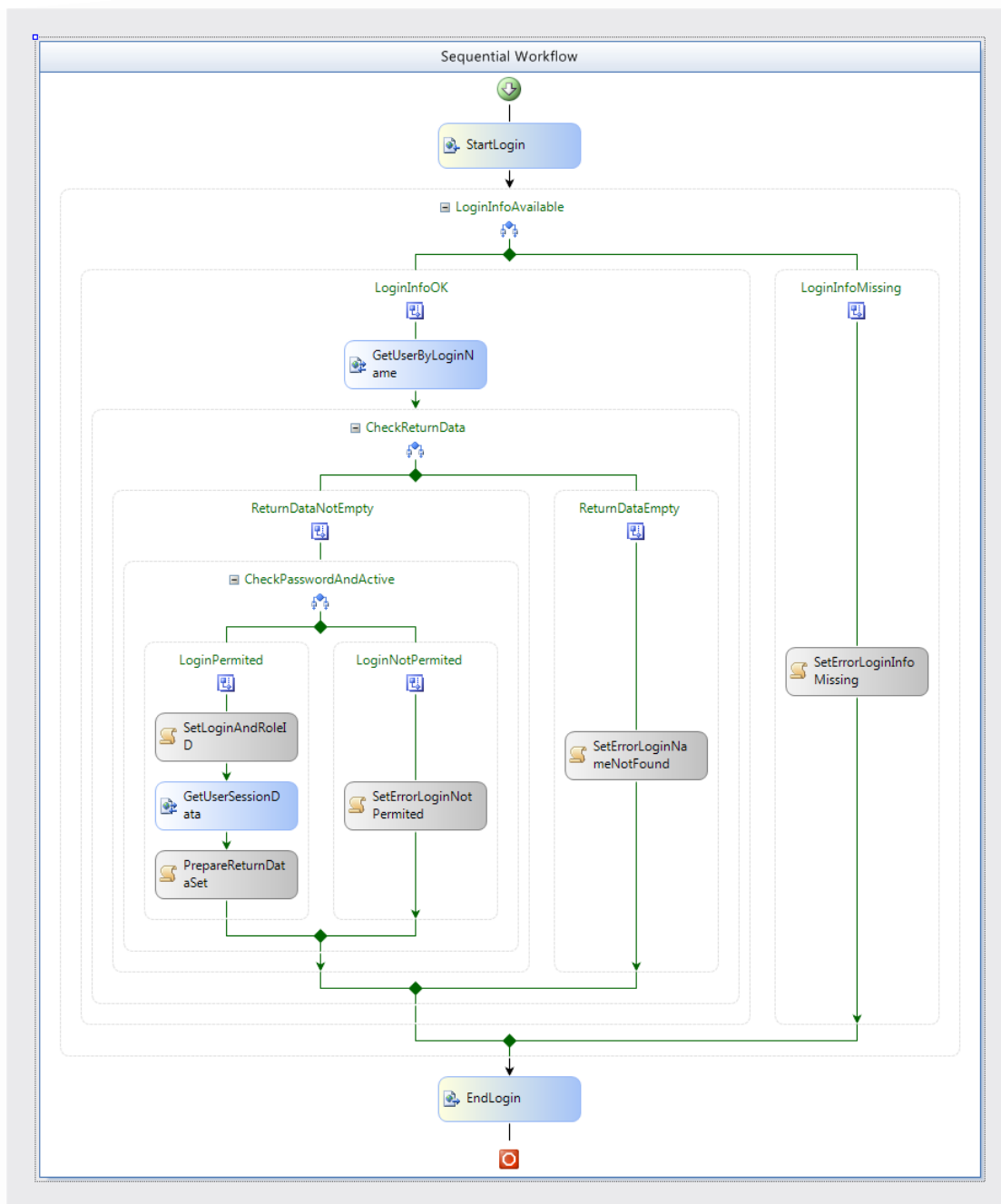
#### 4.4.1 Επιχειρησιακές Υπηρεσίες(Business Services)

Όπως αναφέραμε παραπάνω, οι Επιχειρησιακές Υπηρεσίες του συστήματός μας είναι, ουσιαστικά, ολόκληρες ροές εργασίας(workflows)που έχουν δημοσιευτεί σαν υπηρεσίες (published as services).

##### Εισαγωγή χρήστη στο σύστημα (Login\_WebService)

Η υπηρεσία αυτή καλείται όταν ο χρήστης μπαίνει στην εφαρμογή. Σε επίπεδο UI, ο χρήστης έχει εισάγει στην φόρμα που του εμφανίζεται το «όνομα Χρήστη» και τον «Κωδικό Πρόσβασης» και έχει πατήσει το κουμπί «Login». Τότε καλείται η Υπηρεσία «Login\_WebService», η οποία εκτελεί τα παρακάτω βήματα(παρουσιάζονται στην εικόνα 8.3):

1. Ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα στοιχεία. Αν ο έλεγχος είναι επιτυχής συνεχίζει στα υπόλοιπα βήματα, διαφορετικά εμφανίζει μήνυμα λάθους.
2. Καλεί την Υπηρεσία Υποστήριξης «UsersWS», προκειμένου να εξάγει τα στοιχεία του χρήστη από την βάση, βασιζόμενη στο Όνομα Χρήστη που έχει εισαχθεί. Αν ο χρήστης βρεθεί στην βάση συνεχίζει στα επόμενα βήματα, διαφορετικά επιστρέφει μήνυμα λάθους.
3. Συγκρίνει τον κωδικό χρήστη που έχει εισάγει ο χρήστης στην φόρμα με τον κωδικό που είναι αποθηκευμένος στην βάση. Ελέγχει επίσης αν ο χρήστης είναι ενεργός ή όχι. Αν και οι δύο έλεγχοι είναι επιτυχής, τότε συνεχίζει στο επόμενο βήμα. Αν κάποιος έλεγχος αποτύχει επιστρέφει μήνυμα λάθους.
4. Καλεί την Υπηρεσία Υποστήριξης «UsersWS» για να πάρει κάποια επιπλέον στοιχεία για τον συγκεκριμένο χρήστη (όπως το ονοματεπώνυμο του) και επιστρέφει τα στοιχεία αυτά.



Εικόνα 14: Business Service – Login\_WebService

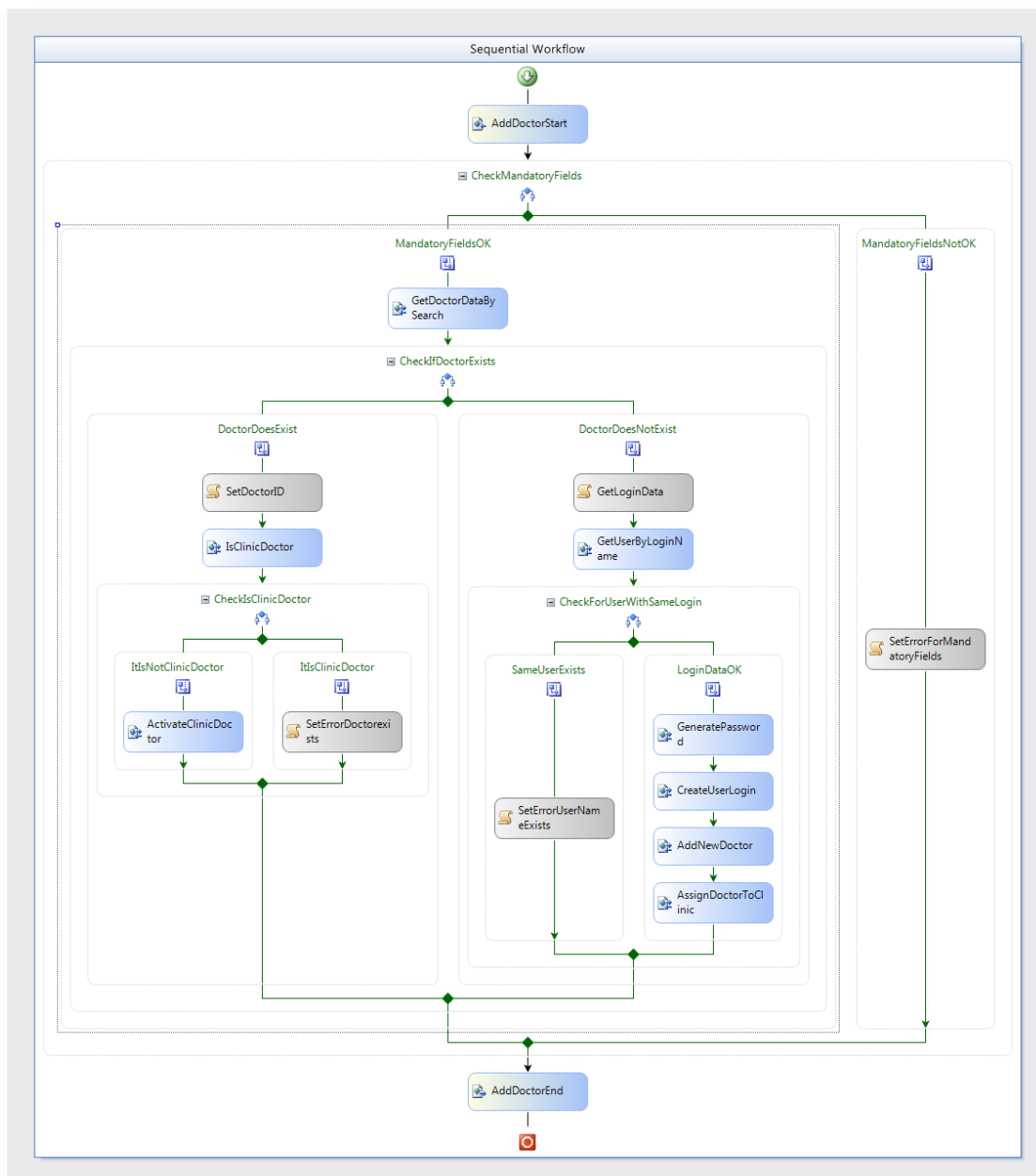
### Εισαγωγή Ιατρού(AddDoctor\_WebService)

Ο χρήστης έχει εισάγει τα στοιχεία του νέου Ιατρού στην κατάλληλη φόρμα και έχει επιλέξει να αποθηκεύσει τον νέο Ιατρό. Τότε καλείται η Υπηρεσία «AddDoctor\_WebService», η οποία εκτελεί τα παρακάτω βήματα(παρουσιάζονται στην εικόνα 14):

1. Ελέγχει αν όλα τα υποχρεωτικά πεδία έχουν συμπληρωθεί. Αν έχουν συμπληρωθεί προχωράει στο επόμενο βήμα, διαφορετικά επιστρέφει μήνυμα λάθους.

2. Καλεί την Τοπική Υπηρεσία «CISDoctorWS», με την οποία αναζητά στην βάση αν υπάρχει άλλος Ιατρός καταχωρημένος με τα ίδια στοιχεία.
3. Αν υπάρχει Ιατρός με τα ίδια στοιχεία, τότε:
  - a. Καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να ελέγξει αν ο γιατρός αυτός έχει ήδη συνδεθεί με την συγκεκριμένη κλινική.
  - b. Αν ο γιατρός δεν έχει συνδεθεί με την κλινική, καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να συνδέσει τον γιατρό με την κλινική. Διαφορετικά επιστρέφει μήνυμα λάθους.
4. Αν δεν υπάρχει Ιατρός με τα ίδια στοιχεία, τότε:
  - a. Καλεί την Υπηρεσία Υποστήριξης «UsersWS» με σκοπό να ελέγξει αν υπάρχει άλλος χρήστης με το ίδιο «Όνομα Χρήστη» με αυτό που έχει εισαχθεί στην φόρμα.
  - b. Αν υπάρχει άλλος χρήστης καταχωρημένος, τότε επιστρέφει μήνυμα λάθους. Διαφορετικά συνεχίζει στα επόμενα βήματα.
  - c. Καλεί την Υπηρεσία Υποστήριξης «UsersWS» για να δημιουργήσει τον κωδικό πρόσβασης, με τον οποίο θα μπορεί ο νέος ιατρός να συνδέεται στο σύστημα.
  - d. Καλεί την Υπηρεσία Υποστήριξης «UsersWS» για να καταχωρήσει τα στοιχεία του ονόματος και του κωδικού πρόσβασης στη βάση.
  - e. Καλεί την Τοπική Υπηρεσία «CISDoctorWS» για να αποθηκεύσει τα υπόλοιπα στοιχεία του ιατρού στην βάση.
  - f. Καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να συνδέσει τον γιατρό με την συγκεκριμένη κλινική.





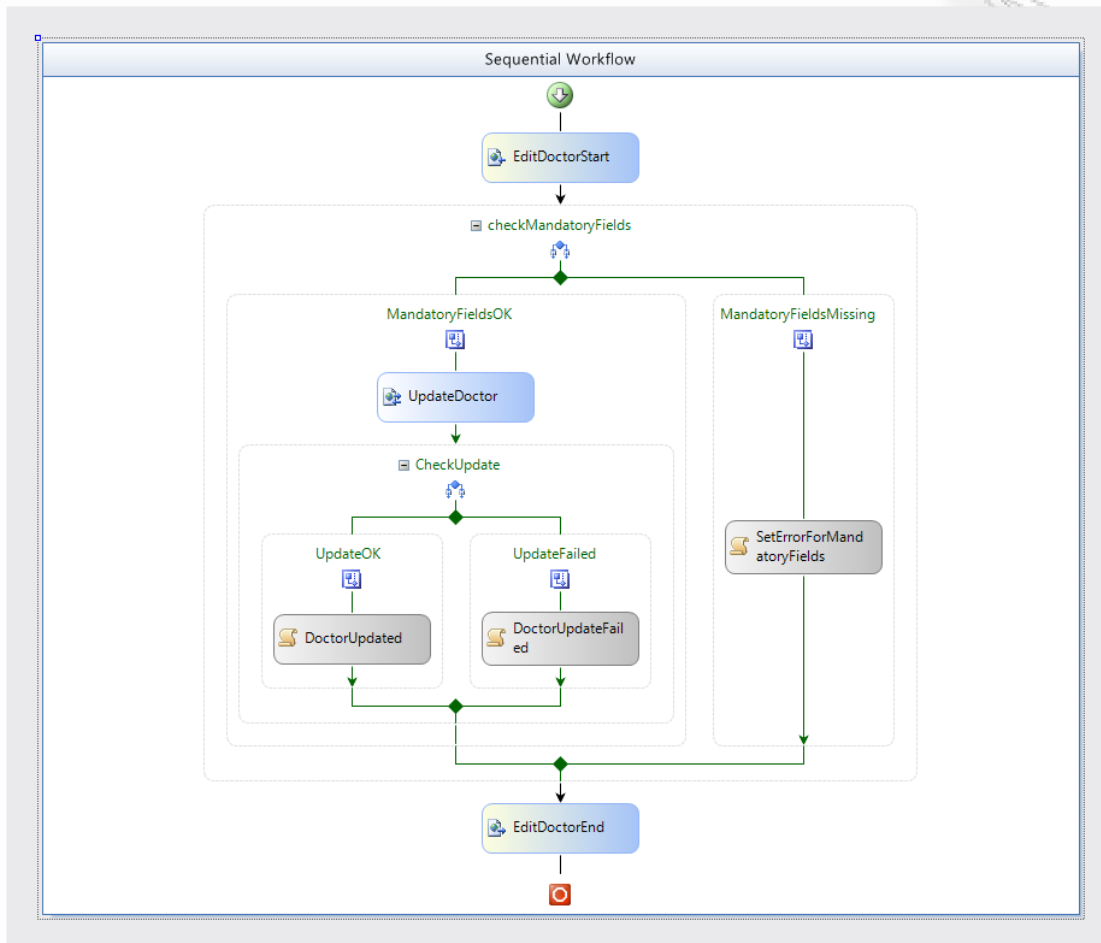
Εικόνα 15: Business Service – AddDoctor\_WebService

### Επεξεργασία Ιατρού (WFDoctorEdit\_WebService)

Ο χρήστης έχει τροποποιήσει τα στοιχεία του Ιατρού στην κατάλληλη φόρμα και έχει επιλέξει να αποθηκεύσει τις αλλαγές. Τότε καλείται η Υπηρεσία «WFDoctorEdit\_WebService», η οποία εκτελεί τα παρακάτω βήματα(παρουσιάζονται στην εικόνα 15):

1. Ελέγχει αν όλα τα υποχρεωτικά πεδία έχουν συμπληρωθεί. Αν έχουν συμπληρωθεί προχωράει στο επόμενο βήμα, διαφορετικά επιστρέφει μήνυμα λάθους.
2. Καλεί την Τοπική Υπηρεσία «CISDoctorWS» για να αποθηκεύσει τα τροποποιημένα στοιχεία του ιατρού στην βάση.

3. Αν η ενημέρωση είναι επιτυχής, επιστρέφει μήνυμα επιτυχούς ενημέρωσης, διαφορετικά επιστρέφει μήνυμα λάθους.

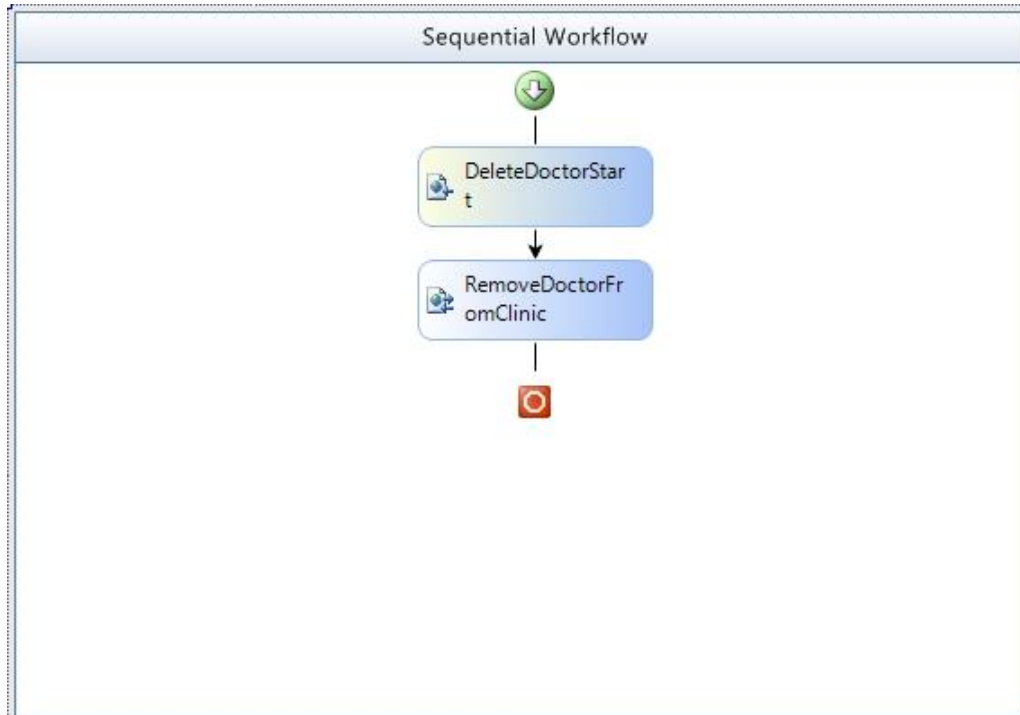


Εικόνα 16: Business Service – WFDactorEdit\_WebService

### Διαγραφή Ιατρού(WFDactorDelete\_WebService)

Ο χρήστης έχει αναζητήσει και έχει επιλέξει τον Ιατρό που θέλει να διαγράψει. Στην συνέχεια πατάει το κουμπί «Delete Doctor». Τότε καλείται η Υπηρεσία «WFDactorDelete\_WebService», η οποία εκτελεί τα παρακάτω βήματα(παρουσιάζονται στην εικόνα 16):

1. Καλεί την Τοπική Υπηρεσία «CISDoctorWS» για να διαγράψει τα στοιχεία του ιατρού από την βάση.
2. Καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να αποσυνδέσει τον γιατρό από την συγκεκριμένη κλινική.

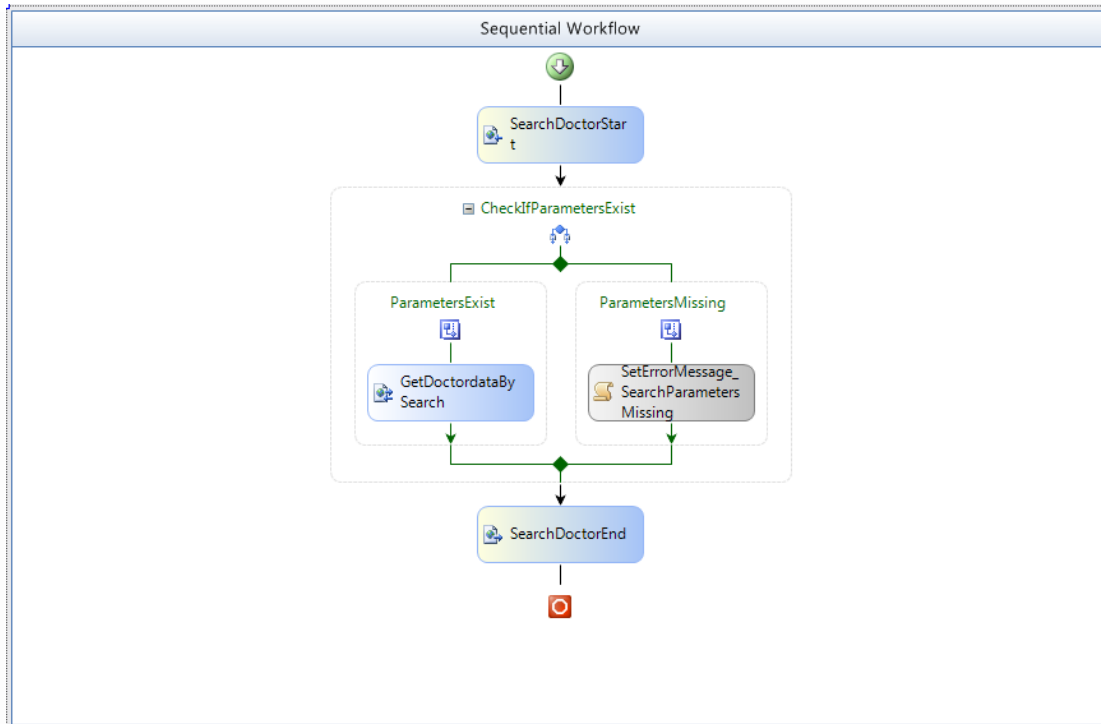


Εικόνα 17: Business Service – WFDoctorDelete\_WebService

### Αναζήτηση ιατρού(SearchDoctor\_WebService)

Ο χρήστης έχει εισάγει τιμές στα κριτήρια αναζήτησης και έχει πατήσει το κουμπί «Search». Τότε καλείται η Υπηρεσία «SearchDoctor\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 17):

1. Αν έχει συμπληρωθεί τουλάχιστον ένα κριτήριο αναζήτησης, τότε συνεχίζει στο επόμενο βήμα. Διαφορετικά επιστρέφει μήνυμα λάθους.
2. Καλεί την Τοπική Υπηρεσία «CISDoctorWS» για να ανακτήσει τα στοιχεία του ιατρού από την βάση.
3. Επιστρέφει τα στοιχεία αυτά στο component που την κάλεσε.



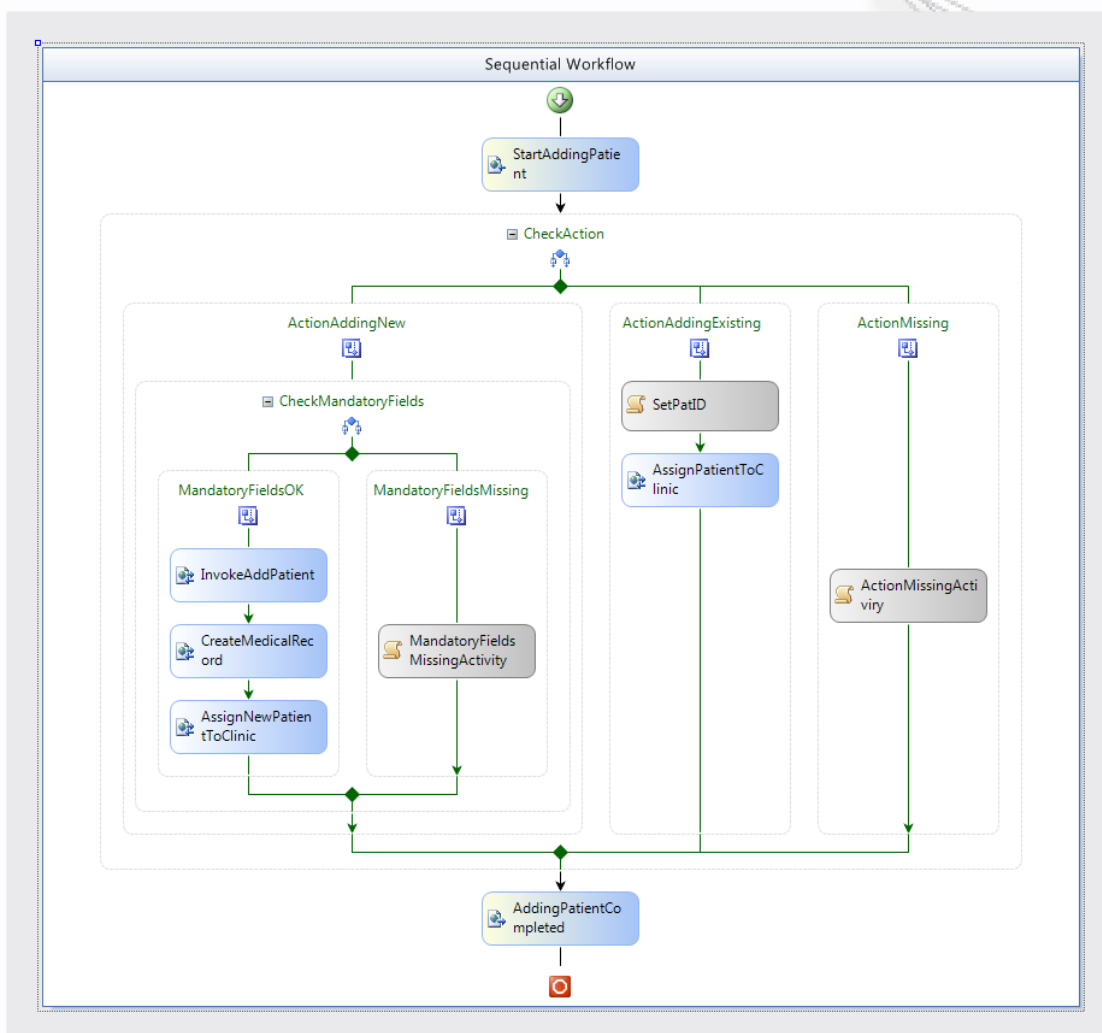
Εικόνα 18: Business Service – SearchDoctor\_WebService

### Προσθήκη Ασθενούς(AddPatient\_WebService)

Ο χρήστης έχει ελέγξει αν υπάρχουν ήδη τα στοιχεία του ασθενούς στην βάση, οπότε τα στοιχεία έχουν εμφανιστεί στην φόρμα ή έχει εισάγει τα στοιχεία του νέου ασθενούς στην κατάλληλη φόρμα και έχει επιλέξει να αποθηκεύσει τον νέο ασθενή. Τότε καλείται η Υπηρεσία «AddPatient\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 18):

1. Αν πρόκειται για προσθήκη νέου ασθενούς στην κλινική:
  - a. Ελέγχει αν όλα τα υποχρεωτικά πεδία έχουν συμπληρωθεί. Αν έχουν συμπληρωθεί προχωράει στο επόμενο βήμα, διαφορετικά επιστρέφει μήνυμα λάθους.
  - b. Καλεί την Τοπική Υπηρεσία «CISPatientWS» για να προσθέσει τα στοιχεία του νέου ασθενούς στην βάση.
  - c. Καλεί την Τοπική Υπηρεσία «CISPatientWS» για να εκκινήσει την διαδικασία δημιουργίας Ηλεκτρονικού Φακέλου για τον νέο ασθενή.
  - d. Καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να συνδέσει τον ασθενή με την συγκεκριμένη κλινική.
2. Αν πρόκειται για την σύνδεση υπάρχοντος ασθενούς με την κλινική:

- α. Καλεί την Τοπική Υπηρεσία «CISClinicsWS» για να συνδέσει τον ασθενή με την συγκεκριμένη κλινική.

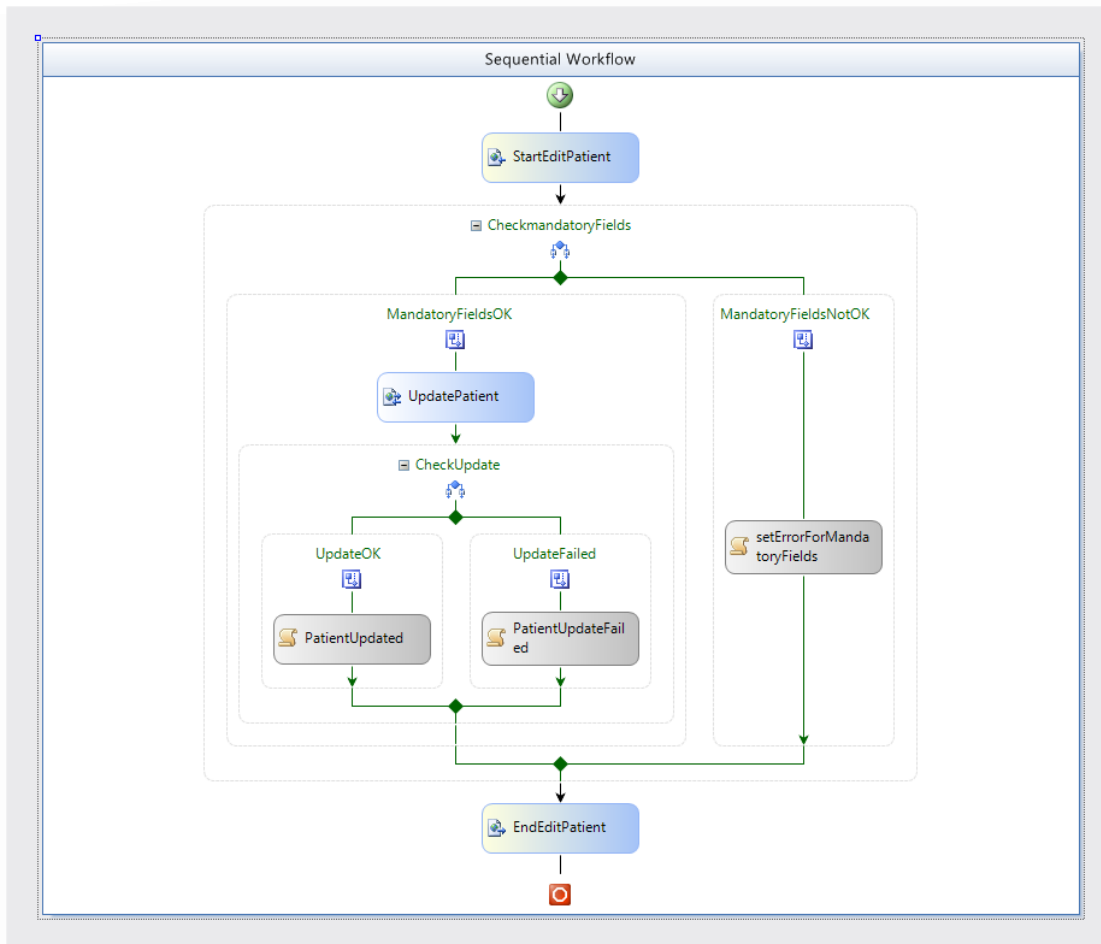


Εικόνα 19: Business Service – AddPatient\_WebService

### Τροποποίηση Ασθενούς(EditPatient\_WebService)

Ο χρήστης έχει τροποποιήσει τα στοιχεία του ασθενούς στην κατάλληλη φόρμα και έχει επιλέξει να αποθηκεύσει τις αλλαγές. Τότε καλείται η Υπηρεσία «EditPatient\_WebService», η οποία εκτελεί τα παρακάτω βήματα(παρουσιάζονται στην εικόνα 19):

1. Ελέγχει αν όλα τα υποχρεωτικά πεδία έχουν συμπληρωθεί. Αν έχουν συμπληρωθεί προχωράει στο επόμενο βήμα, διαφορετικά επιστρέφει μήνυμα λάθους.
2. Καλεί την Τοπική Υπηρεσία «CISPatientWS» για να αποθηκεύσει τα τροποποιημένα στοιχεία του ιατρού στην βάση.
3. Αν η ενημέρωση είναι επιτυχής, επιστρέφει μήνυμα επιτυχούς ενημέρωσης, διαφορετικά επιστρέφει μήνυμα λάθους.



Εικόνα 20: Business Service – EditPatient\_WebService

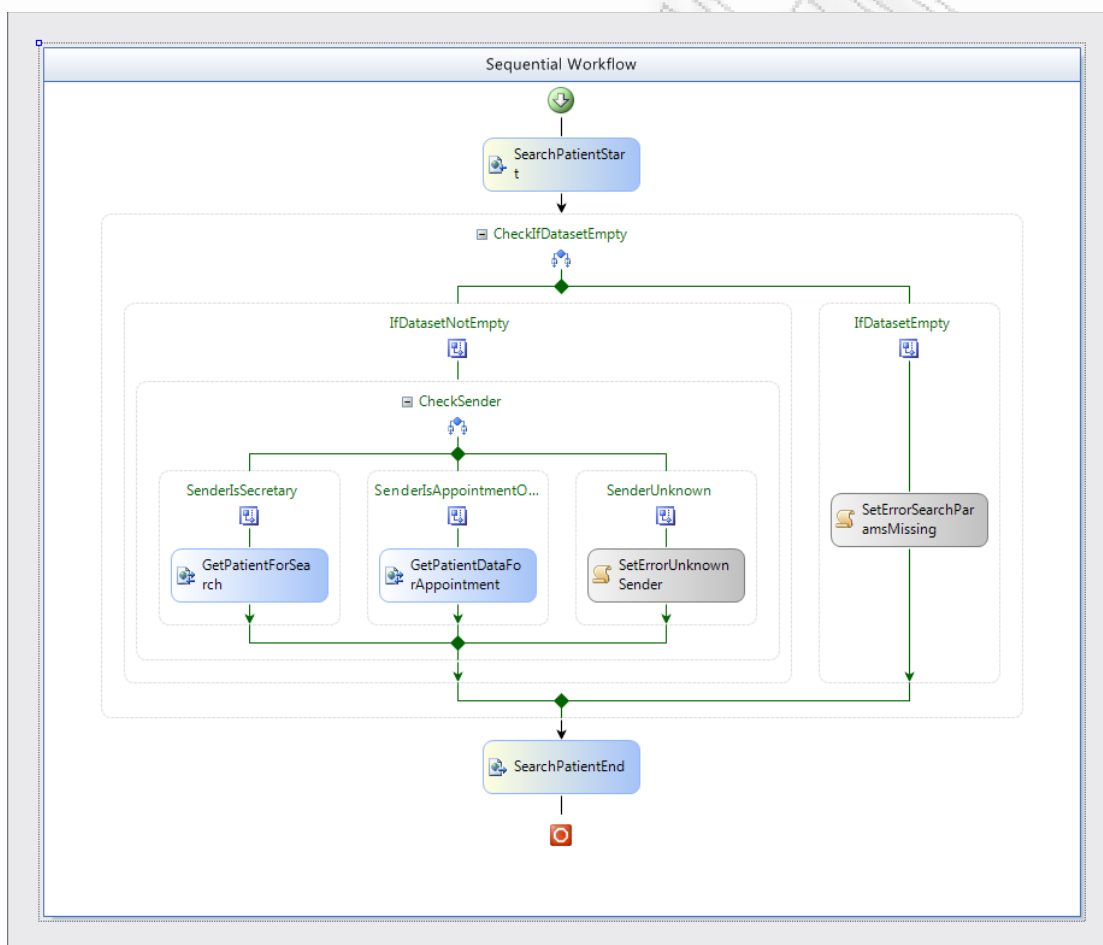
### Αναζήτηση Ασθενούς (SearchPatient\_WebService)

Η Υπηρεσία αυτή καλείται από δύο διαφορετικά components του συστήματος, οπότε έχουμε δύο διαφορετικά σενάρια που οδηγούν στην εκτέλεσή της:

- Αναζήτηση από την εφαρμογή της γραμματείας: Ο χρήστης έχει εισάγει τιμές στα κριτήρια αναζήτησης και έχει πατήσει το κουμπί «Search».
- Αναζήτηση από την εφαρμογή του ιατρού: Ο χρήστης έχει εισάγει τιμές στα κριτήρια αναζήτησης και έχει πατήσει το κουμπί «Search».
- Αναζήτηση ασθενούς κατά την καταχώρηση ραντεβού. Ο χρήστης έχει επιλέξει να καταχωρήσει ένα ραντεβού με ένα συγκεκριμένο γιατρό. Ο χρήστης έχει εισάγει τιμές στα κριτήρια αναζήτησης της φόρμας που του έχει εμφανιστεί και έχει πατήσει το κουμπί «Search».

Σε όλες τις παραπάνω περιπτώσεις καλείται η Υπηρεσία «SearchPatient\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 20):

1. Αν έχει συμπληρωθεί τουλάχιστον ένα κριτήριο αναζήτησης, τότε συνεχίζει στο επόμενο βήμα. Διαφορετικά επιστρέφει μήνυμα λάθους.
2. Αν η αναζήτηση εκτελείται από την εφαρμογή της γραμματείας, καλεί την Τοπική Υπηρεσία «CISPatientWS» για να ανακτήσει τα στοιχεία του ασθενούς από την βάση. Αν η αναζήτηση εκτελείται από την εφαρμογή του ιατρού ή από τα ραντεβού, καλεί την Τοπική Υπηρεσία «CISPatientAppointmentsWS» για να ανακτήσει τα στοιχεία του ασθενούς από την βάση.
3. Επιστρέφει τα στοιχεία αυτά στο component που την κάλεσε.



Εικόνα 21: Business Service – SearchPatient\_WebService

### Επιστροφή Στοιχείων Ασθενούς

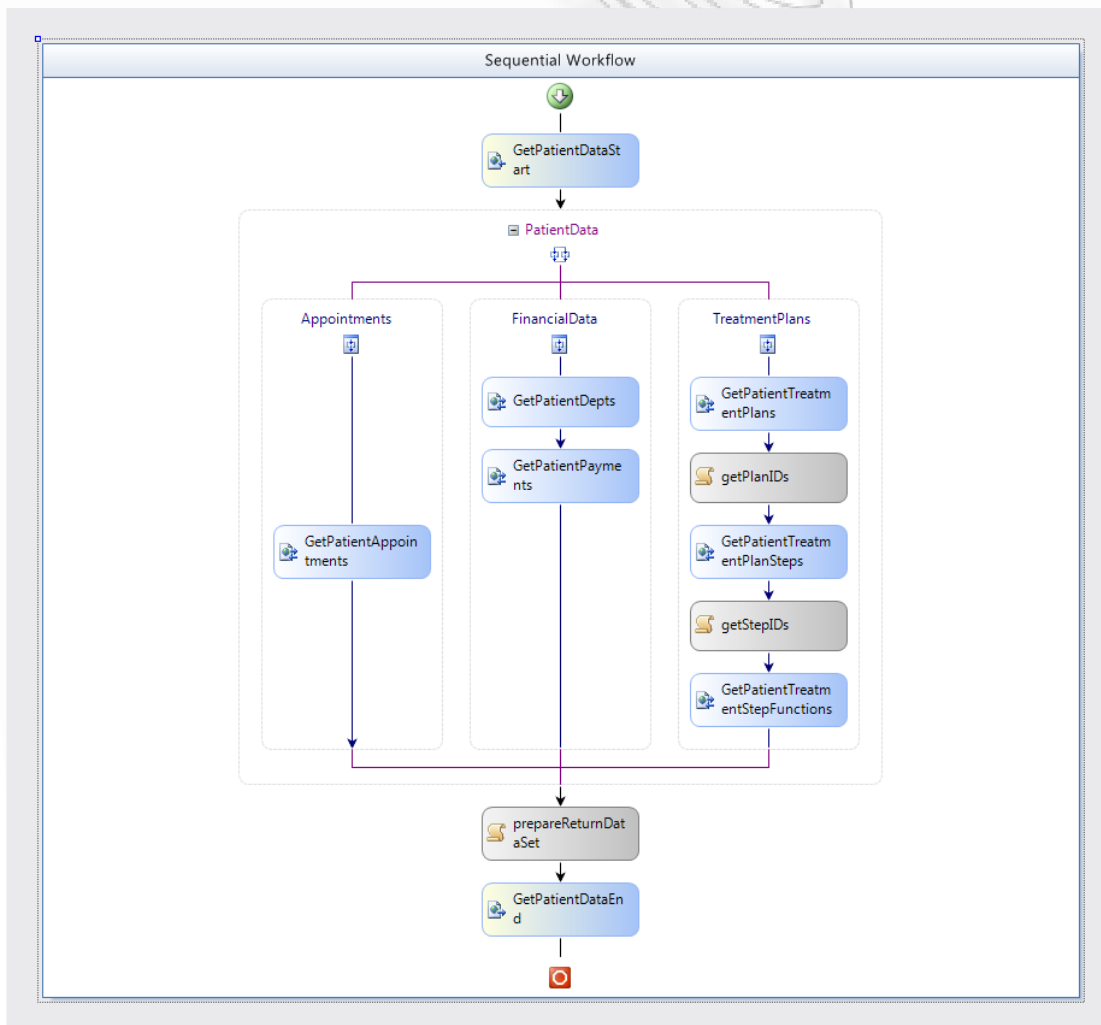
Διαφορετικά στοιχεία του ασθενούς εμφανίζονται σε διαφορετικούς ρόλους χρηστών και σε διαφορετικά σημεία στην εφαρμογή. Υπάρχουν τέσσερις Υπηρεσίες που επιστρέφουν στοιχεία του ασθενούς και παρουσιάζονται παρακάτω.

### WFPatientDataForSecretary\_WebService

Η υπηρεσία αυτή καλείται από την εφαρμογή της γραμματείας. Ο χρήστης έχει επιλέξει έναν από τους ασθενείς που του επέστρεψαν τα κριτήρια αναζήτησης. Τότε καλείται η Υπηρεσία «WFPatientDataForSecretary\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 21):

Τα παρακάτω βήματα εκτελούνται παράλληλα:

1. Καλεί την Τοπική Υπηρεσία «CISPatientAppointmentsWS», η οποία επιστρέφει τα αποθηκευμένα ραντεβού του ασθενούς.
2. Καλεί την Τοπική Υπηρεσία «CISPatientFinancialdataWS», η οποία επιστρέφει τα οικονομικά στοιχεία του ασθενούς.
3. Καλεί την Τοπική Υπηρεσία «CISPatientTreatmentPlansWS», η οποία επιστρέφει τα στοιχεία για τα σχέδια θεραπείας του ασθενούς.



Εικόνα 22: Business Service – WFPatientDataForSecretary\_WebService

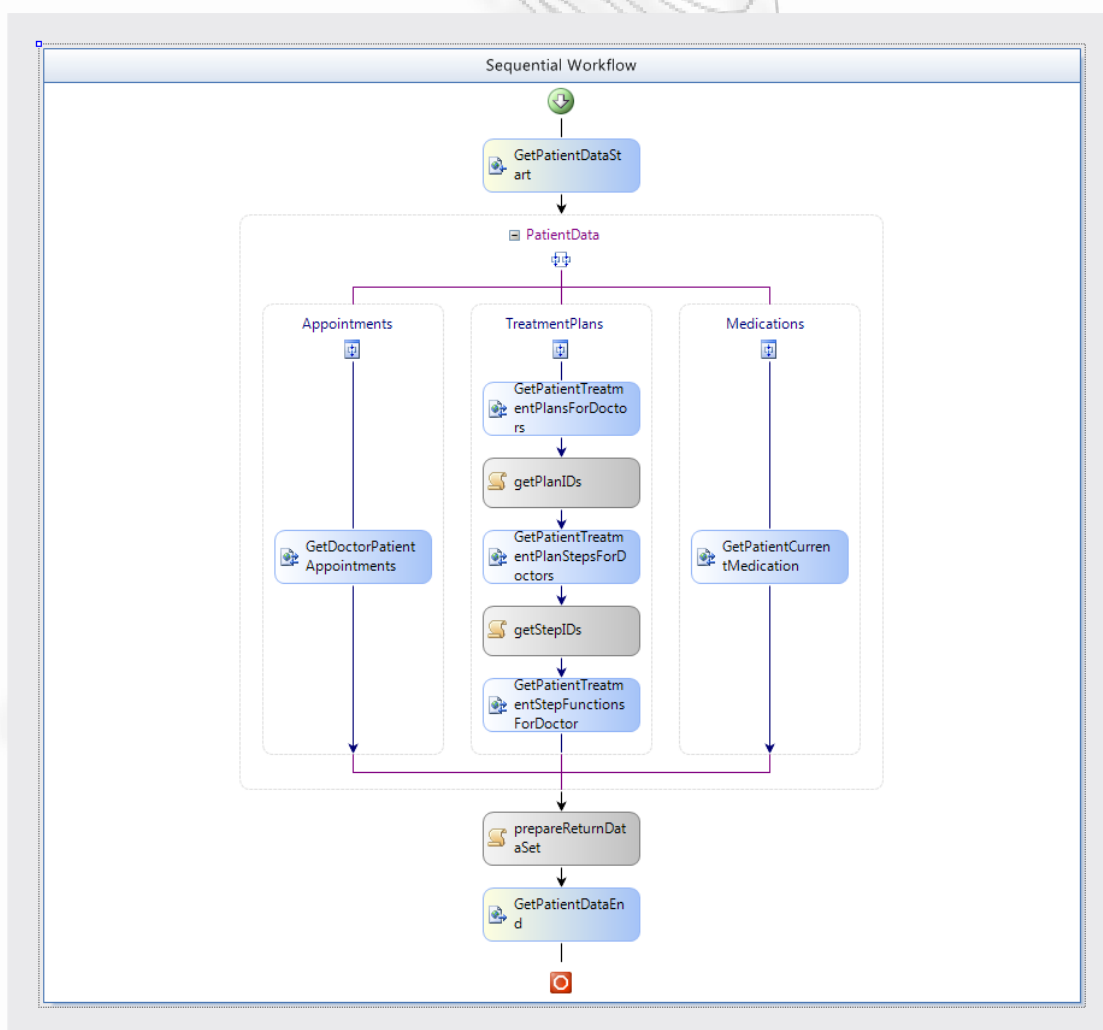


### WFPatientDataForDoctor\_WebService

Η υπηρεσία αυτή καλείται από την εφαρμογή του ιατρού. Ο χρήστης έχει επιλέξει έναν από τους ασθενείς που του επέστρεψαν τα κριτήρια αναζήτησης. Τότε καλείται η Υπηρεσία «WFPatientDataForDoctor\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 22):

Τα παρακάτω βήματα εκτελούνται παράλληλα:

1. Καλεί την Τοπική Υπηρεσία «CISDoctorWS», η οποία επιστρέφει τα αποθηκευμένα ραντεβού του ασθενούς.
2. Καλεί την Τοπική Υπηρεσία «CISPatientTreatmentPlansWS», η οποία επιστρέφει τα στοιχεία για τα σχέδια θεραπείας του ασθενούς.
3. Καλεί την Τοπική Υπηρεσία «CISPatientMedicationsWS», η οποία επιστρέφει τα στοιχεία για την φαρμακευτική αγωγή του ασθενούς.



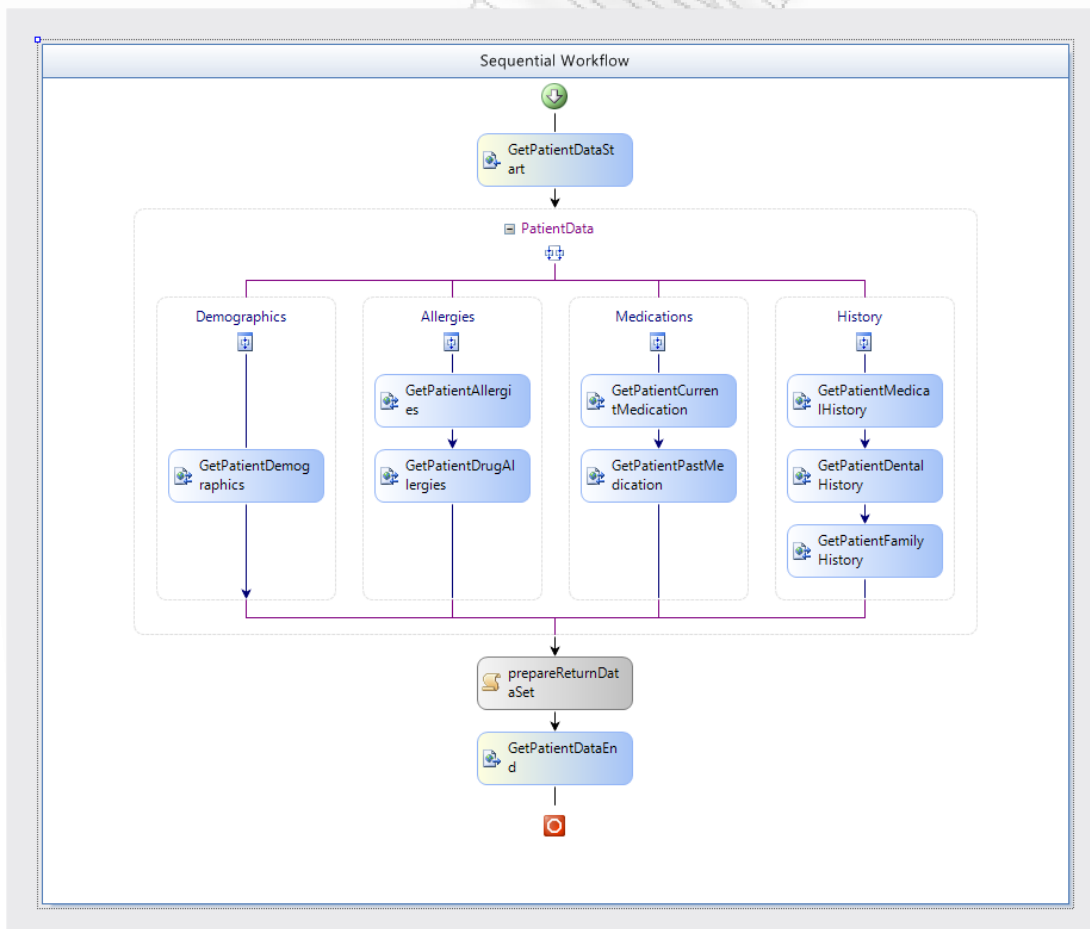
Εικόνα 23: Business Service – WFPatientDataForDoctor\_WebService

### WFPatientDataForMedicalRecord\_WebService

Η υπηρεσία αυτή καλείται από την εφαρμογή του ιατρού. Ο χρήστης έχει επιλέξει ήδη έναν από τους ασθενείς του και θέλει να δει τα στοιχεία του Ιατρικού του Φακέλου. Τότε καλείται η Υπηρεσία «WFPatientDataForMedicalRecord\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 23):

Τα παρακάτω βήματα εκτελούνται παράλληλα:

1. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει τα πλήρη αποθηκευμένα δημογραφικά στοιχεία του ασθενούς.
2. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει στοιχεία για τις πιθανές αλλεργίες του ασθενούς.
3. Καλεί την Τοπική Υπηρεσία «CISPatientMedicationsWS», η οποία επιστρέφει τα στοιχεία για την φαρμακευτική αγωγή του ασθενούς.
4. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει στοιχεία σχετικά με το ιατρικό, οδοντιατρικό και οικογενειακό ιστορικό του ασθενούς.



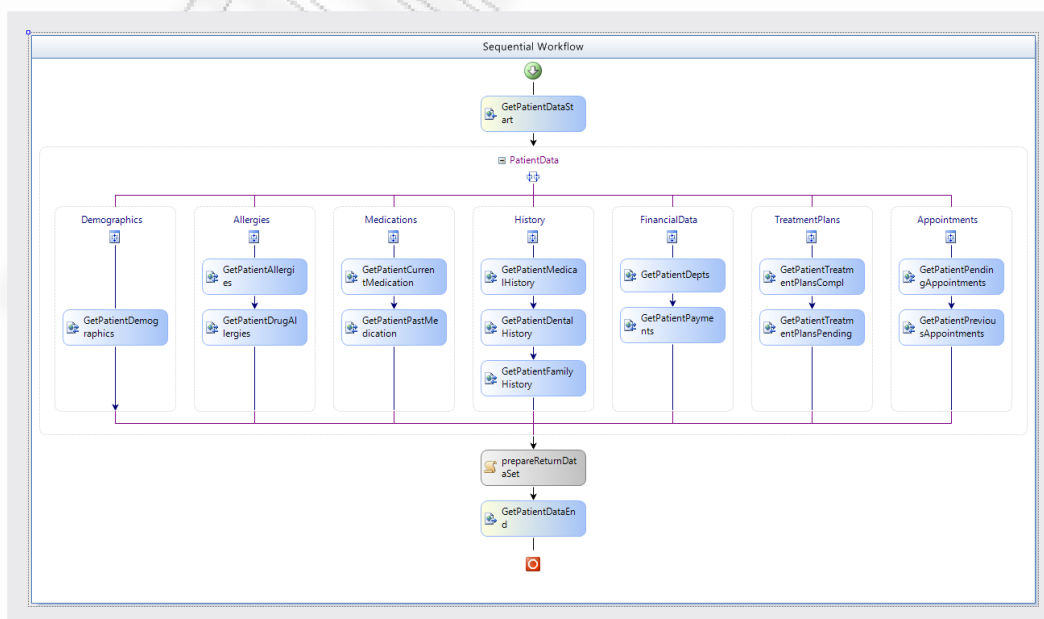
Εικόνα 24: Business Service – WFPatientDataForMedicalRecord\_WebService

## WFPatientDataForPHR\_WebService

Η υπηρεσία αυτή καλείται από την εφαρμογή του Ηλεκτρονικού Φακέλου Ασθενούς. Ο χρήστης έχει εισαχθεί στο σύστημα και η εφαρμογή πρέπει να του εμφανίσει όλα τα στοιχεία του ιατρικού του φακέλου. Τότε καλείται η Υπηρεσία «WFPatientDataForPHR\_WebService», η οποία εκτελεί τα παρακάτω βήματα (παρουσιάζονται στην εικόνα 24):

Τα παρακάτω βήματα εκτελούνται παράλληλα:

1. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει τα πλήρη αποθηκευμένα δημογραφικά στοιχεία του ασθενούς.
2. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει στοιχεία για τις πιθανές αλλεργίες του ασθενούς.
3. Καλεί την Τοπική Υπηρεσία «CISPatientMedicationsWS», η οποία επιστρέφει τα στοιχεία για την φαρμακευτική αγωγή του ασθενούς.
4. Καλεί την Τοπική Υπηρεσία «CISPatientHistoryWS», η οποία επιστρέφει στοιχεία σχετικά με το ιατρικό, οδοντιατρικό και οικογενειακό ιστορικό του ασθενούς.
5. Καλεί την Τοπική Υπηρεσία «CISPatientFinancialDataWS», η οποία επιστρέφει τα οικονομικά στοιχεία του ασθενούς
6. Καλεί την Τοπική Υπηρεσία «CISPatientTreatmentPlansWS», η οποία επιστρέφει τα στοιχεία για τα σχέδια θεραπείας του ασθενούς.
7. Καλεί την Τοπική Υπηρεσία «CISPatientAppointmentsWS», η οποία επιστρέφει τα αποθηκευμένα ραντεβού του ασθενούς.



Εικόνα 25: Business Service – WFPatientDataForPHR\_WebService

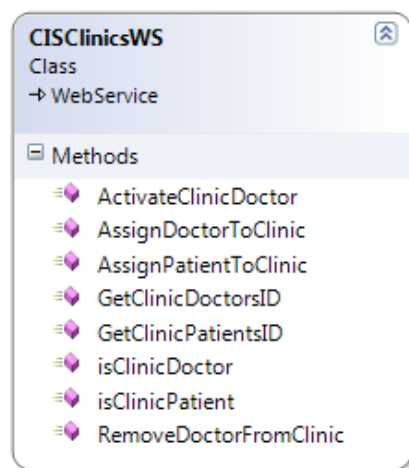
#### 4.4.2 Τοπικές Υπηρεσίες(Domain Services)

Παρακάτω παρουσιάζονται οι Τοπικές Υπηρεσίες του συστήματος και οι μέθοδοι που έχει διαθέσιμες η κάθε υπηρεσία. Οι περισσότερες Τοπικές Υπηρεσίες καλούν Υπηρεσίες Ολοκλήρωσης, Υποστήριξης ή Βασικές Υπηρεσίες, προκειμένου να επιστρέψουν την κατάλληλη πληροφορία στο component που τις κάλεσε.

##### Domain Κλινικής

##### Τοπική Υπηρεσία «CISClinicsWS»

Η λειτουργικότητα της υπηρεσίας σχετίζεται με τον έλεγχο για το αν ένας ιατρός ή ένας ασθενής ανήκουν στην κλινική και με την σύνδεση και αποσύνδεσή τους με την κλινική. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα(εικόνα 25). Η Υπηρεσία αυτή καλεί, με τη σειρά της, την Βασική Υπηρεσία «ExecuteQRYWS» για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση.

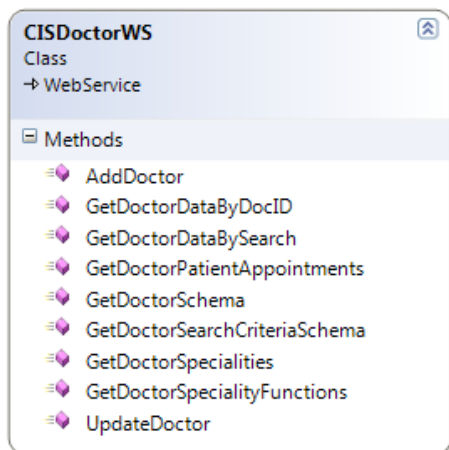


Εικόνα 26: Domain Service – CISClinicsWS

##### Domain Ιατρού

##### Τοπική υπηρεσία «CISDoctorWS»

Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν έναν ιατρό στο σύστημα, όπως η προσθήκη του στο σύστημα, η ενημέρωση των στοιχείων του, η αναζήτησή του και άλλα. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 26). Η Υπηρεσία αυτή καλεί, με τη σειρά της, την Βασική Υπηρεσία «ExecuteQRYWS» για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση.



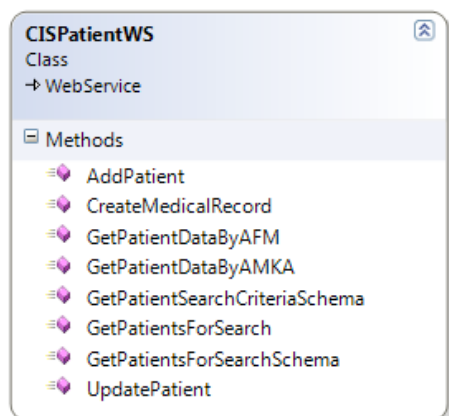
Εικόνα 27: Domain Service – CISDoctorWS

### **Domain Ασθενούς**

Στο Domain του ασθενούς έχουμε περισσότερες υπηρεσίες, ανάλογα με την λειτουργικότητα και τις διαδικασίες που πρέπει να καλυφθούν.

### **Τοπική Υπηρεσία «CISPatientWS»**

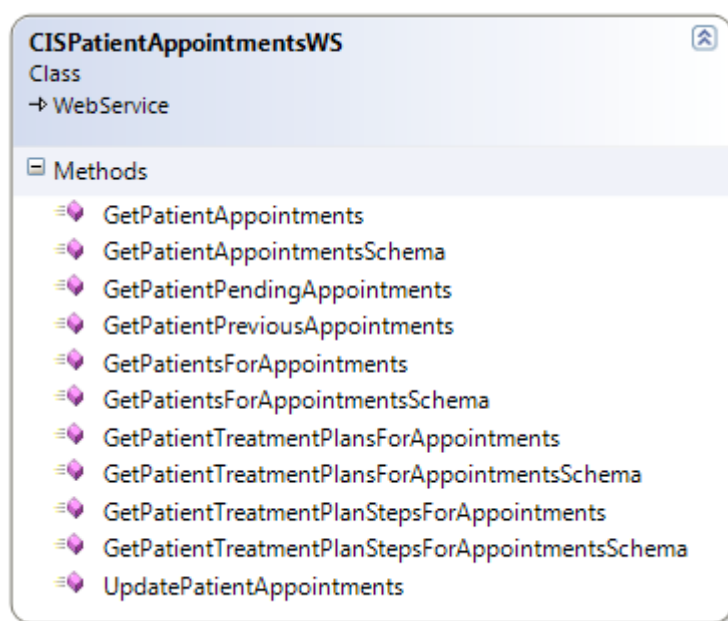
Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν έναν ασθενή στο σύστημα, όπως η προσθήκη του στο σύστημα, η ενημέρωση των στοιχείων του, η αναζήτησή του και άλλα. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 27). Η Υπηρεσία αυτή καλεί, με τη σειρά της, την Βασική Υπηρεσία «ExecuteQRYWS», για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση, αλλά και την Υπηρεσία Ολοκλήρωσης «PatientDataWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



Εικόνα 28: Domain Service – CISPatientWS

### Τοπική Υπηρεσία «CISPatientAppointmentsWS»

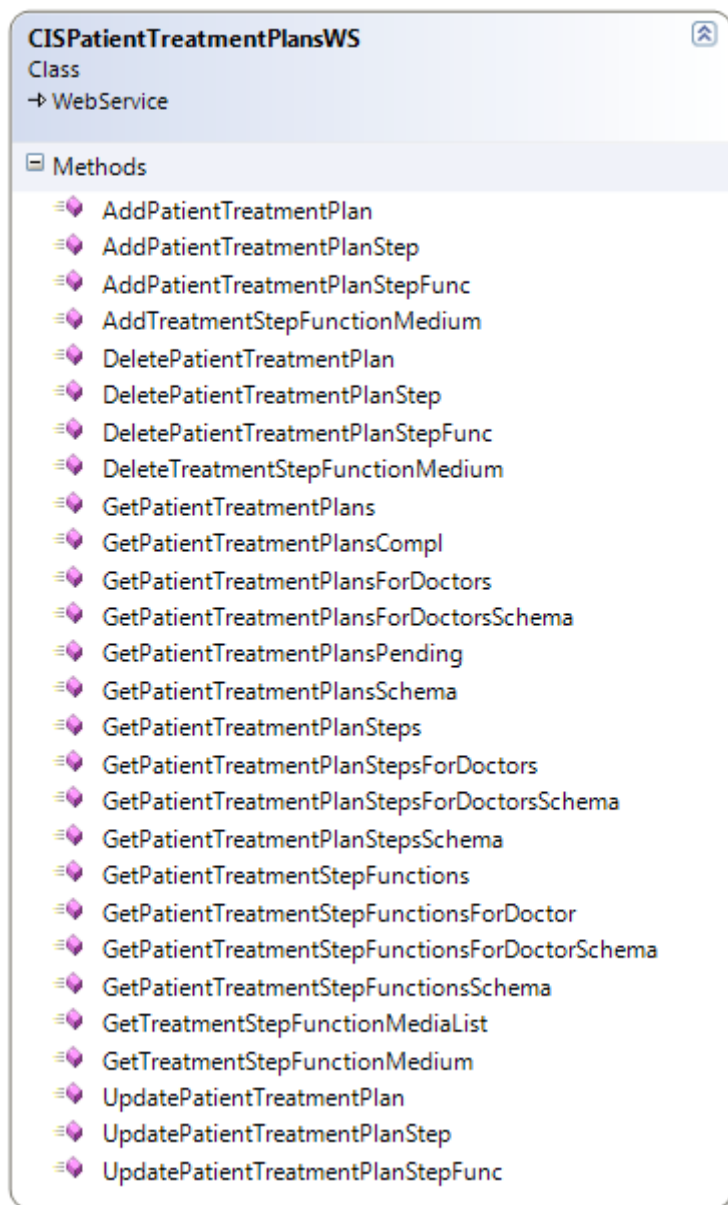
Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν τα ραντεβού ενός ασθενούς. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 28). Η Υπηρεσία αυτή καλεί, με τη σειρά της, τις Βασικές Υπηρεσίες «ExecuteQRYWS» και «RunProcedureWS», για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση, αλλά και την Υπηρεσία Ολοκλήρωσης «PatientAppointmentsWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



Εικόνα 29: Domain Service – CISPatientAppointmentsWS

### Τοπική Υπηρεσία «CISPatientTreatmentPlansWS»

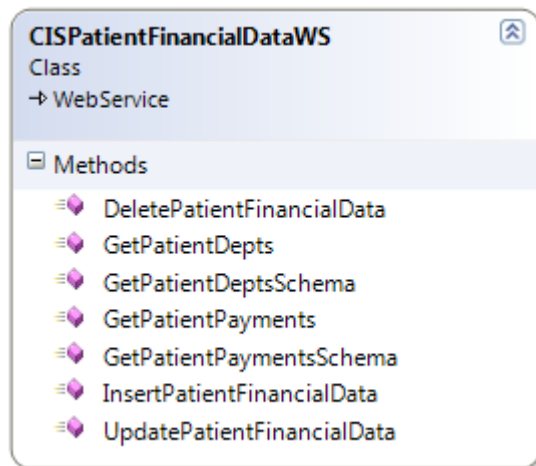
Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν τα σχέδια θεραπείας ενός ασθενούς. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 29). Η Υπηρεσία αυτή καλεί, με τη σειρά της, τις Βασικές Υπηρεσίες «ExecuteQRYWS» και «RunProcedureWS», για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση, αλλά και την Υπηρεσία Ολοκλήρωσης «PatientTreatmentPlansWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



Εικόνα 30: Domain Service – CISPatientTreatmentPlansWS

### Τοπική Υπηρεσία «CISPatientFinancialDataWS»

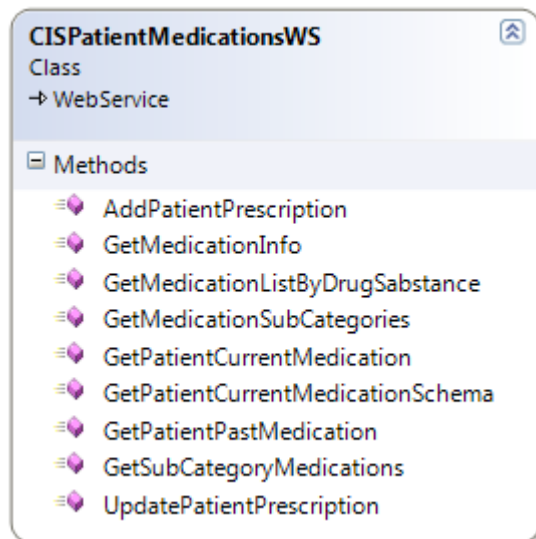
Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν τα οικονομικά στοιχεία (πληρωμές και χρέη) ενός ασθενούς. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 30). Η Υπηρεσία αυτή καλεί, με τη σειρά της, την Βασική Υπηρεσία «ExecuteQRYWS», για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση, αλλά και την Υπηρεσία Ολοκλήρωσης «PatientFinancialDataWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



Εικόνα 31: Domain Service – CISPatientFinancialDataWS

### Τοπική Υπηρεσία «CISPatientMedicationsWS»

Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν την φαρμακευτική αγωγή ενός ασθενούς. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 31). Η Υπηρεσία αυτή καλεί τις Βασικές Υπηρεσίες «ExecuteQRYWS» και «RunProcedureWS», για να αποθηκεύσει ή να ανακτήσει στοιχεία από την βάση, αλλά και την Υπηρεσία Ολοκλήρωσης «PatientMedicationsWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



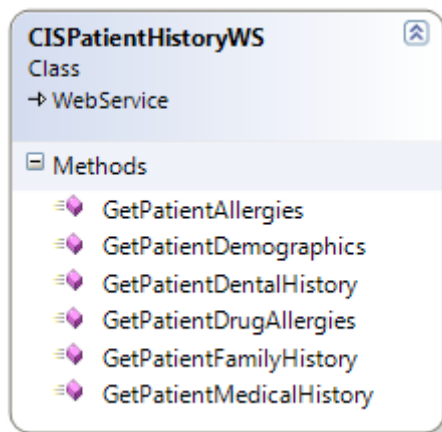
Εικόνα 32: Domain Service – CISPatientMedicationsWS

### Τοπική Υπηρεσία «CISPatientHistoryWS»

Η λειτουργικότητα της υπηρεσίας σχετίζεται με ενέργειες που αφορούν το ιατρικό, ιστορικό ενός ασθενούς. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 32). Η



Υπηρεσία αυτή καλεί τις Υπηρεσίες Ολοκλήρωσης «PatientAllergiesWS», «PatientHistoryWS» και «PatientDataWS», για να συλλέξει πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.



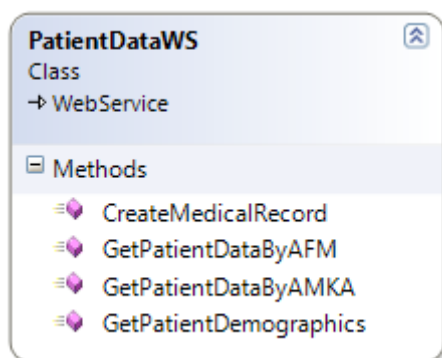
Εικόνα 33: Domain Service – CISPatientHistoryWS

#### 4.4.3 Υπηρεσίες Ολοκλήρωσης(Integration Services)

Παρακάτω παρουσιάζονται οι Υπηρεσίες Ολοκλήρωσης του συστήματος και οι μέθοδοι που έχει διαθέσιμες η κάθε υπηρεσία. Οι Υπηρεσίες Ολοκλήρωσης καλούν Βασικές Υπηρεσίες, προκειμένου να επιστρέψουν την κατάλληλη πληροφορία στο component που τις κάλεσε. Χρησιμοποιούνται για να συλλέξουν πλήρη στοιχεία για τον ασθενή και από τα δύο συστήματα.

##### Υπηρεσία Ολοκλήρωσης «PatientDataWS»

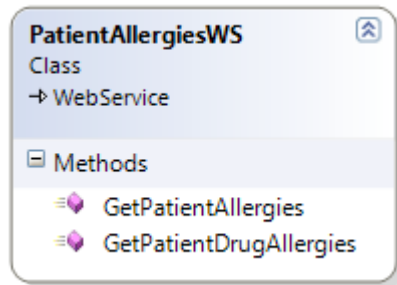
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά με τα δημογραφικά στοιχεία του ασθενούς, τα οποία επιστρέφει κατά την αναζήτησή του από τα συστήματα. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξαγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 33).



Εικόνα 34: Integration Service – PatientDataWS

### Υπηρεσία Ολοκλήρωσης «PatientAllergiesWS»

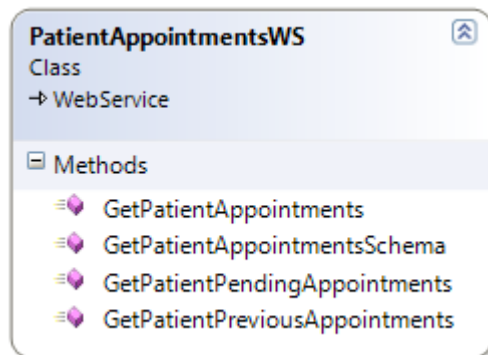
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά τις αλλεργίες του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 34).



Εικόνα 35: Integration Service – PatientAllergiesWS

### Υπηρεσία Ολοκλήρωσης «PatientAppointmentsWS»

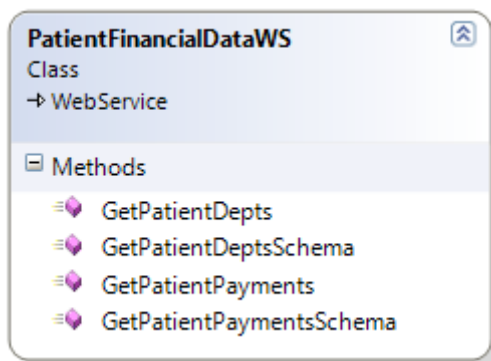
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά τα ραντεβού του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 35).



Εικόνα 36: Integration Service – PatientAppointmentsWS

### Υπηρεσία Ολοκλήρωσης «PatientFinancialDataWS»

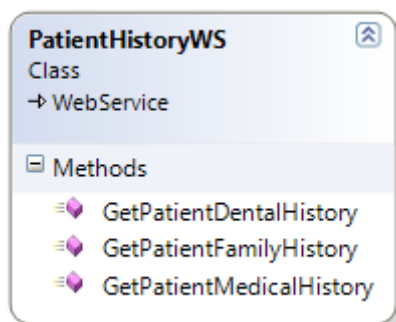
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά τα οικονομικά στοιχεία του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 36).



Εικόνα 37: Integration Service – PatientFinancialDataWS

### Υπηρεσία Ολοκλήρωσης «PatientHistoryWS»

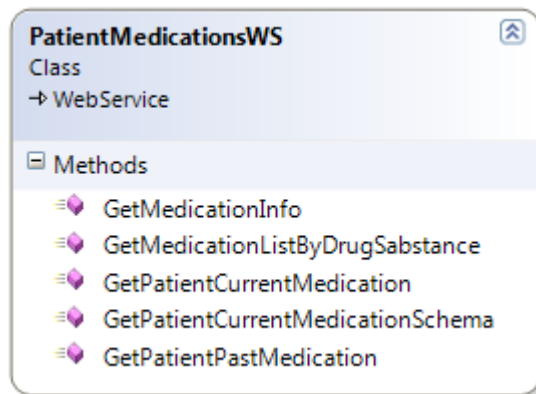
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά με το ιατρικό, οδοντιατρικό και οικογενειακό ιστορικό του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 37).



Εικόνα 38: Integration Service – PatientHistoryWS

### Υπηρεσία Ολοκλήρωσης « PatientMedicationsWS»

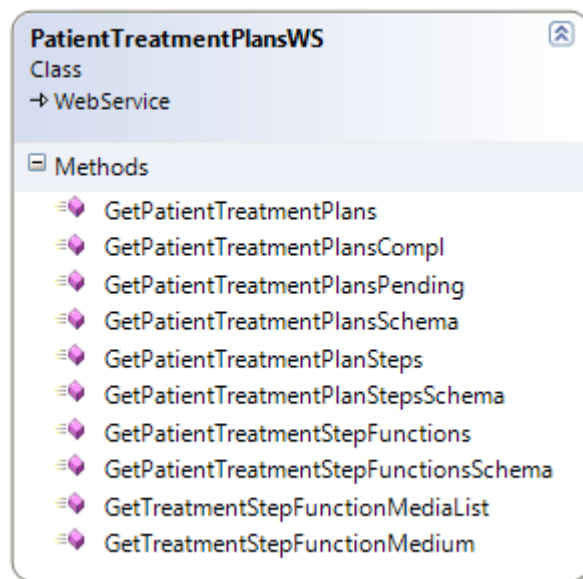
Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά με την φαρμακευτική αγωγή του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 38).



Εικόνα 39: Integration Service – PatientMedicationsWS

#### Υπηρεσία Ολοκλήρωσης « PatientTreatmentPlansWS»

Η Υπηρεσία αυτή συλλέγει δεδομένα σχετικά με τα σχέδια θεραπείας του ασθενούς. Χρησιμοποιεί την βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει δεδομένα από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 39).



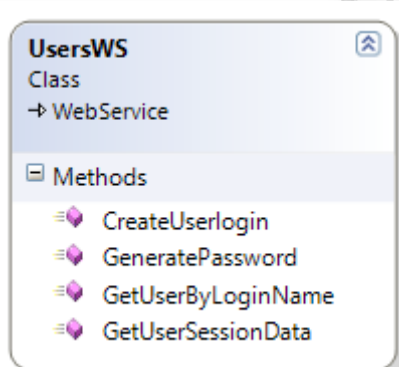
Εικόνα 40: Integration Service – PatientTreatmentPlansWS

#### 4.4.4 Υπηρεσίες Υποστήριξης (Utility Services)

Παρακάτω παρουσιάζονται οι Υπηρεσίες Υποστήριξης του συστήματος και οι μέθοδοι που έχει διαθέσιμες η κάθε υπηρεσία. Οι Υπηρεσίες Υποστήριξης καλούν Βασικές Υπηρεσίες, προκειμένου να επιστρέψουν την κατάλληλη πληροφορία στο component που τις κάλεσε και παρέχουν βασική λειτουργικότητα που είναι σημαντική για αρκετές διαδικασίες.

### Υπηρεσία Υποστήριξης «UsersWS»

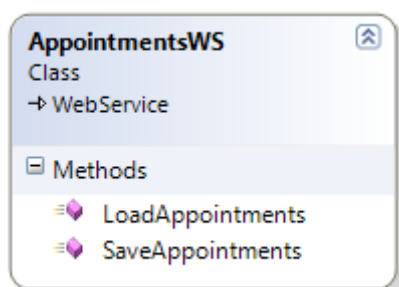
Η Υπηρεσία αυτή ασχολείται με τα Logins των χρηστών στο σύστημα. Ανακτά πληροφορίες για τους χρήστες κατά την σύνδεσή του ή δημιουργεί το όνομα χρήστη και το κωδικό, με τα οποία θα συνδέονται. Χρησιμοποιεί τις Βασικές Υπηρεσίες «ExecuteQRYWS» και «RunProcedureWS», για να εξάγει δεδομένα από την βάση ή για να αποθηκεύσει δεδομένα σε αυτή. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 40).



Εικόνα 41: Utility Service – UsersWS

### Υπηρεσία Υποστήριξης «AppointmentsWS»

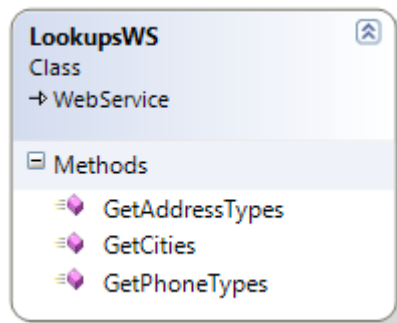
Η Υπηρεσία αυτή ασχολείται με την ανάκτηση και την αποθήκευση ενός ραντεβού. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 41).



Εικόνα 42: Utility Service – AppointmentsWS

### Υπηρεσία Υποστήριξης «LookupsWS»

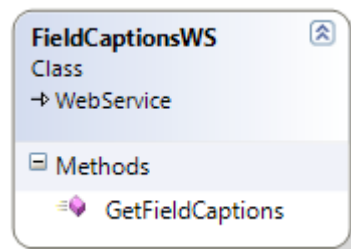
Η Υπηρεσία αυτή ασχολείται με την ανάκτηση δεδομένων από την βάση που αποτελούν Λίστες Τιμών . Χρησιμοποιεί την Βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει τα δεδομένα αυτά από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 8.42).



Εικόνα 43: Utility Service – LookupsWS

#### Υπηρεσία Υποστήριξης «FieldCaptionsWS»

Η Υπηρεσία αυτή επιστρέφει το όνομα με το οποίο θα παρουσιάζεται κάθε πεδίο της βάσης στην εφαρμογή. Χρησιμοποιεί την Βασική Υπηρεσία «ExecuteQRYWS», για να εξάγει τα δεδομένα αυτά από την βάση. Οι μέθοδοι που διαθέτει παρουσιάζονται στη παρακάτω εικόνα (εικόνα 43).



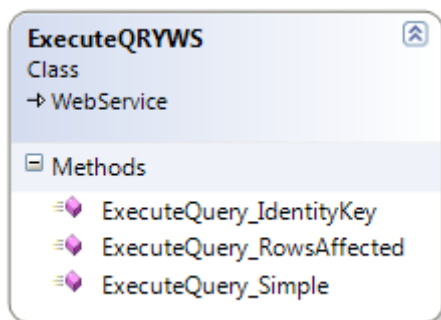
Εικόνα 44: Utility Service – FieldCaptionsWS

#### 4.4.5 Βασικές Υπηρεσίες (Foundation Services)

Παρακάτω παρουσιάζονται οι Βασικές Υπηρεσίες του συστήματος και οι μέθοδοι που έχει διαθέσιμες η κάθε υπηρεσία. Οι Βασικές Υπηρεσίες επικοινωνούν με την βάση μέσω του Data Access Layer, προκειμένου να επιστρέψουν την κατάλληλη πληροφορία στο component που τις κάλεσε. Ουσιαστικά εκτελούν CRUD (Create, Read, Update, and Delete) λειτουργίες στη βάση.

##### Βασική Υπηρεσία «ExecuteQRYWS»

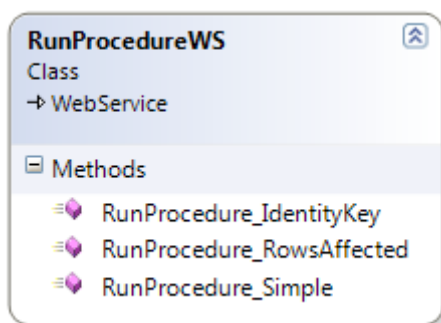
Δεδομένου ενός query, η υπηρεσία αυτή επιστρέφει από την βάση ένα dataset, το πόσες εγγραφές ενημερώθηκαν ή το identity key της εγγραφής που προστέθηκε. Οι μέθοδοι που παρέχει είναι οι εξής:



Εικόνα 45: Foundation Service – ExecuteQRYWS

#### Βασική Υπηρεσία « RunProcedureWS»

Δεδομένου του ονόματος μιας store procedure και των παραμέτρων της, η υπηρεσία αυτή επιστρέφει από την βάση ένα dataset, το πόσες εγγραφές ενημερώθηκαν ή το identity key της εγγραφής που προστέθηκε. Οι μέθοδοι που παρέχει είναι οι εξής:

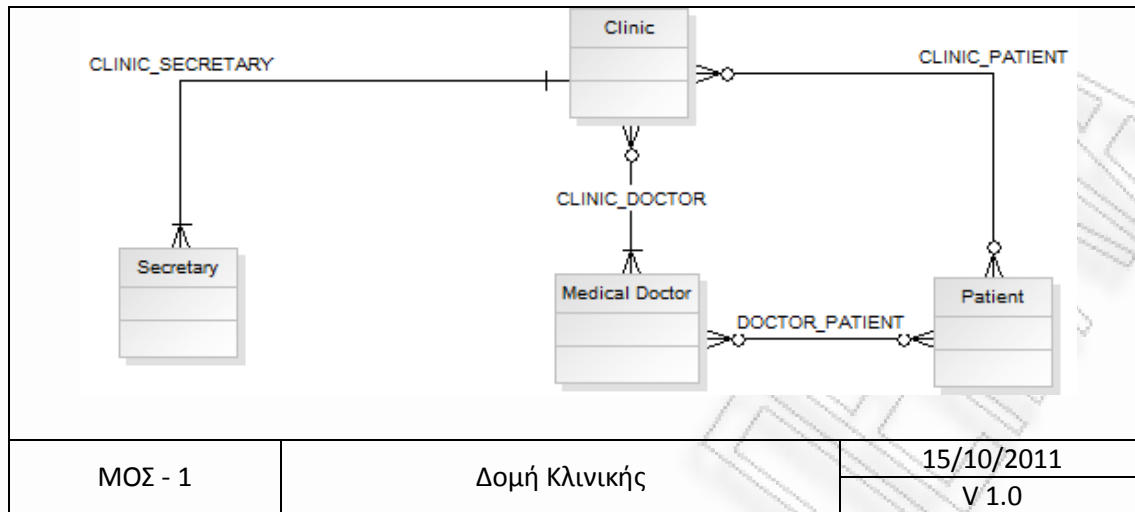


Εικόνα 46: Foundation Service – RunProcedureWS

## 4.5 Η Βάση Δεδομένων του συστήματος

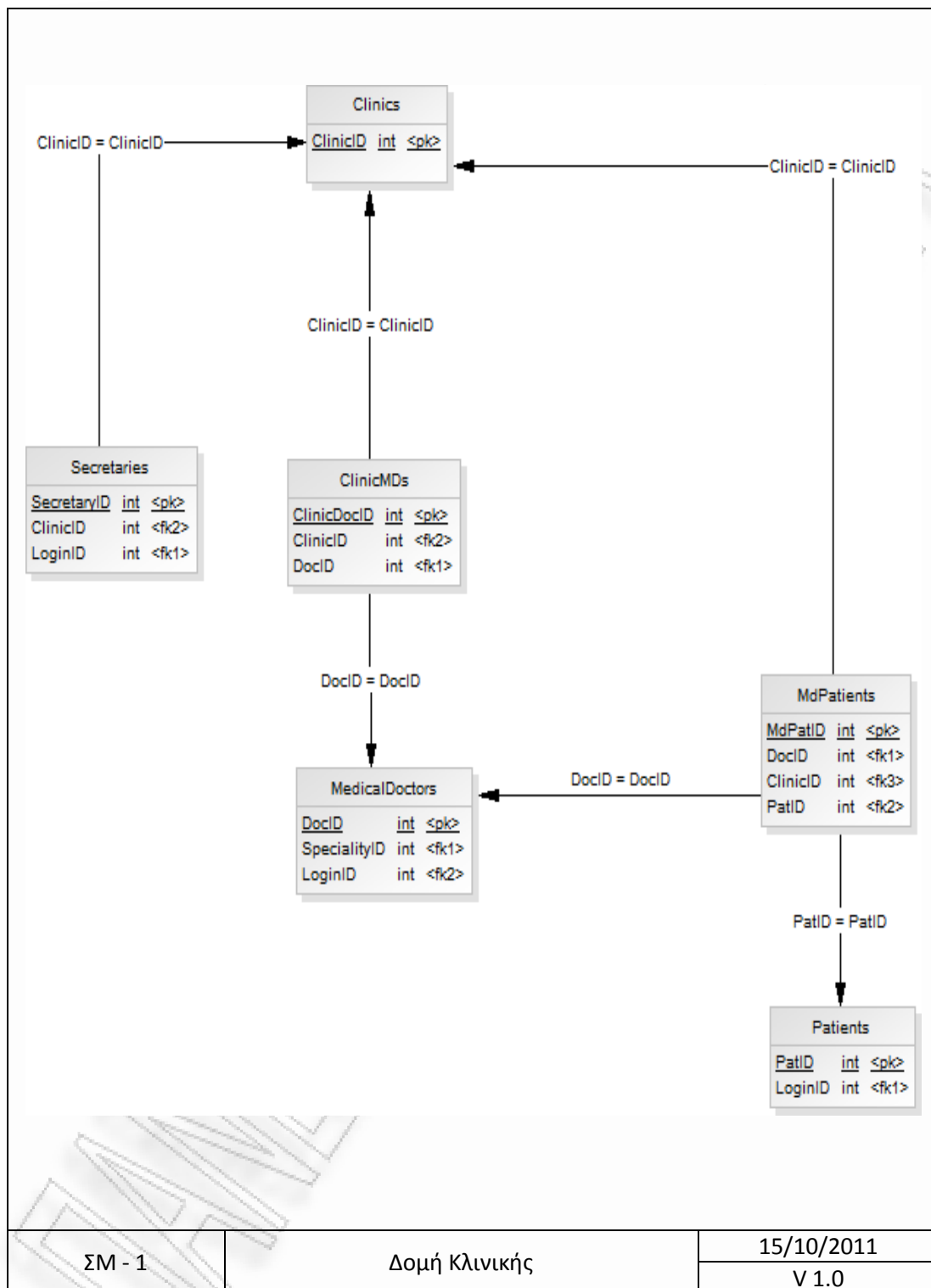
Σύμφωνα με την εκμείυση και ανάλυση απαιτήσεων ορίσαμε τις οντότητες του συστήματός μας και τις συσχετίσεις μεταξύ τους. Στην ενότητα αυτή παρουσιάζονται τα Διαγράμματα Οντοτήτων - Συσχετίσεων που περιγράφουν τη λειτουργία του συστήματός και μας οδήγησαν στην σχεδίαση της βάσης δεδομένων του περιγεγραμμένου συστήματος. Για κάθε διάγραμμα Οντοτήτων-Συσχετίσεων παρουσιάζεται και το αντίστοιχο Σχεσιακό Μοντέλο.

## Δομή Κλινικής

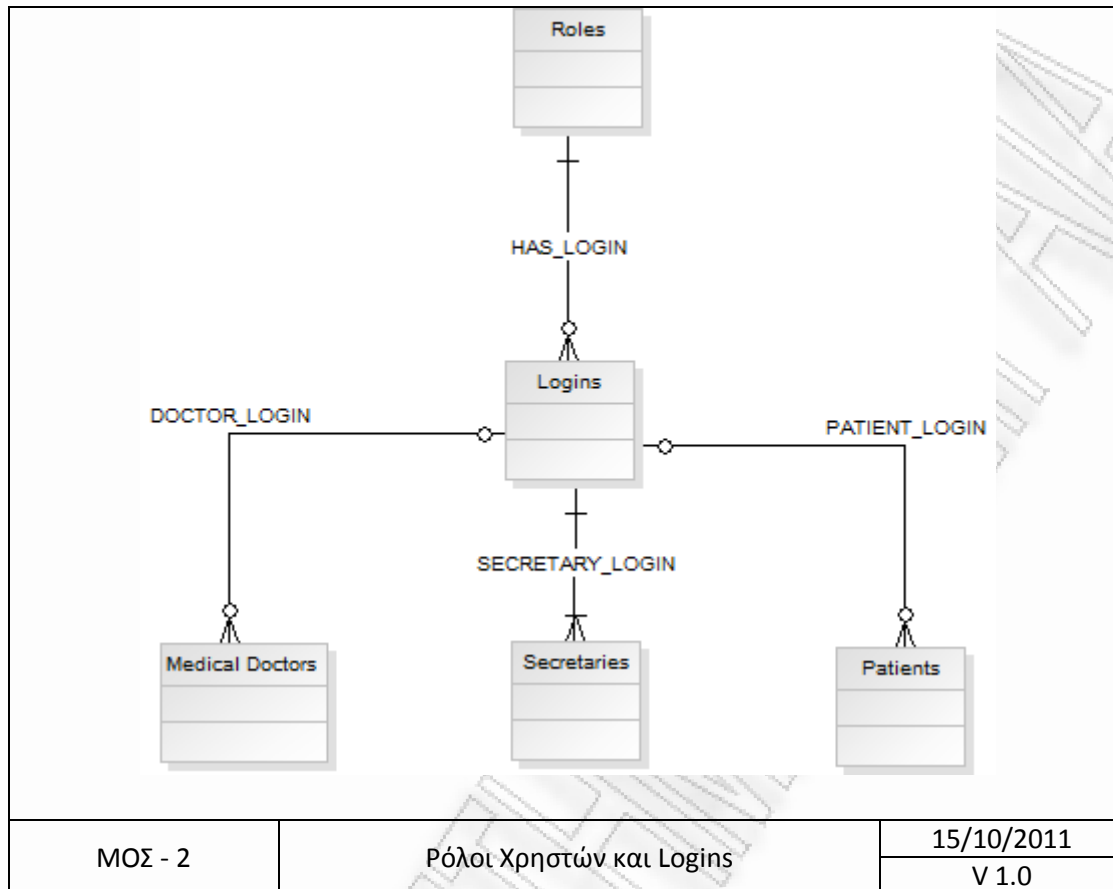


- Μια Κλινική έχει μία ή περισσότερες Γραμματείες.
- Μια Γραμματεία ανήκει σε μια Κλινική.
- Μια Κλινική έχει έναν η περισσότερους Ιατρούς.
- Ένας Ιατρός μπορεί να εργάζεται σε πολλές Κλινικές.
- Μια Κλινική έχει πολλούς Ασθενής
- Ένας Ασθενής επισκέπτεται πολλές Κλινικές.
- Ένας Ασθενής επισκέπτεται πολλούς Ιατρούς μιας Κλινικής και ένα Ιατρός εξετάσει πολλούς Ασθενείς σε μια Κλινική

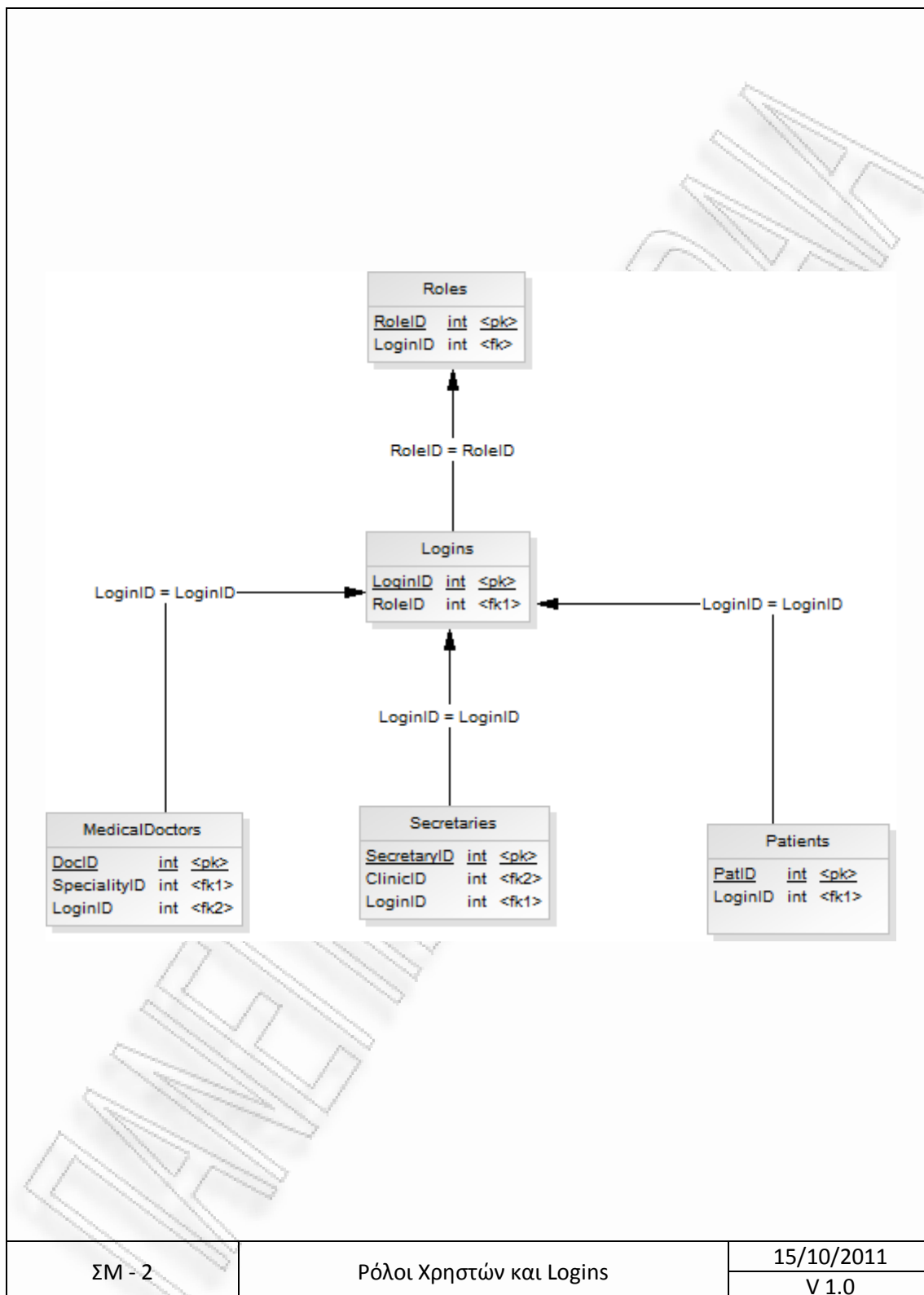




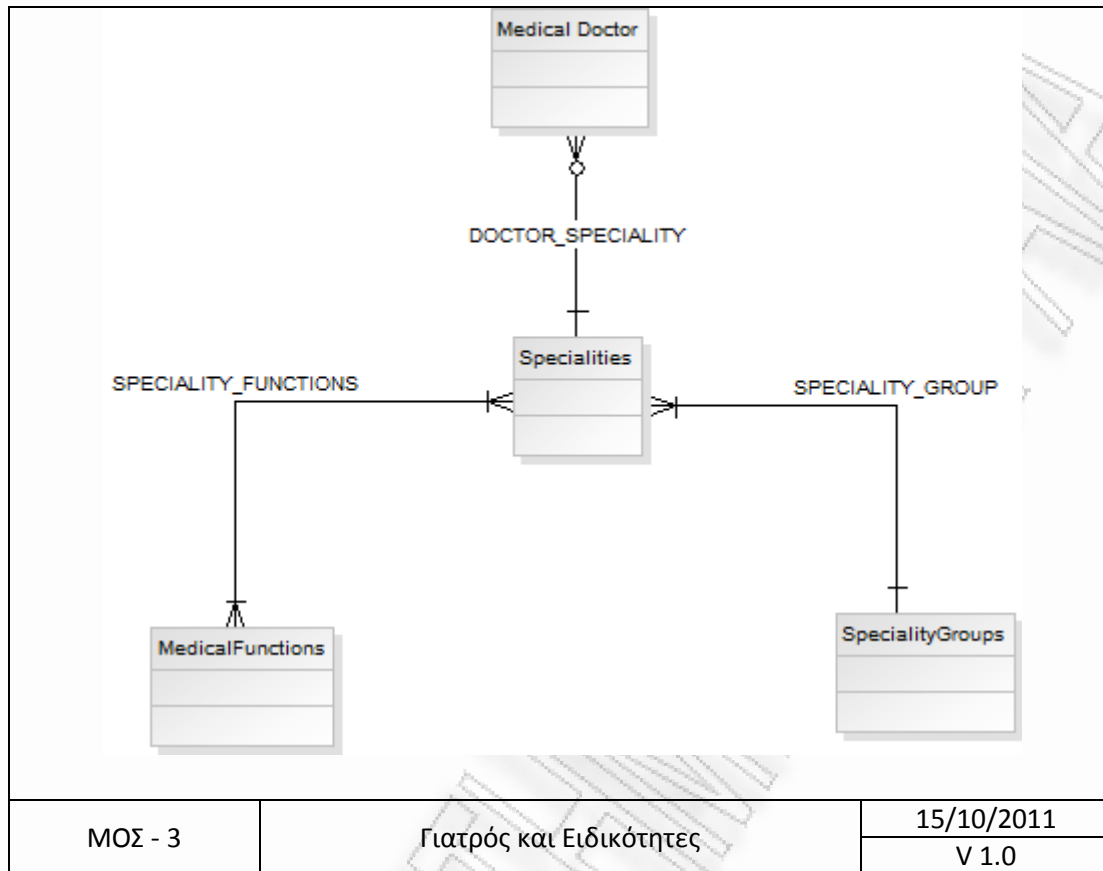
## Ρόλοι Χρηστών και Logins



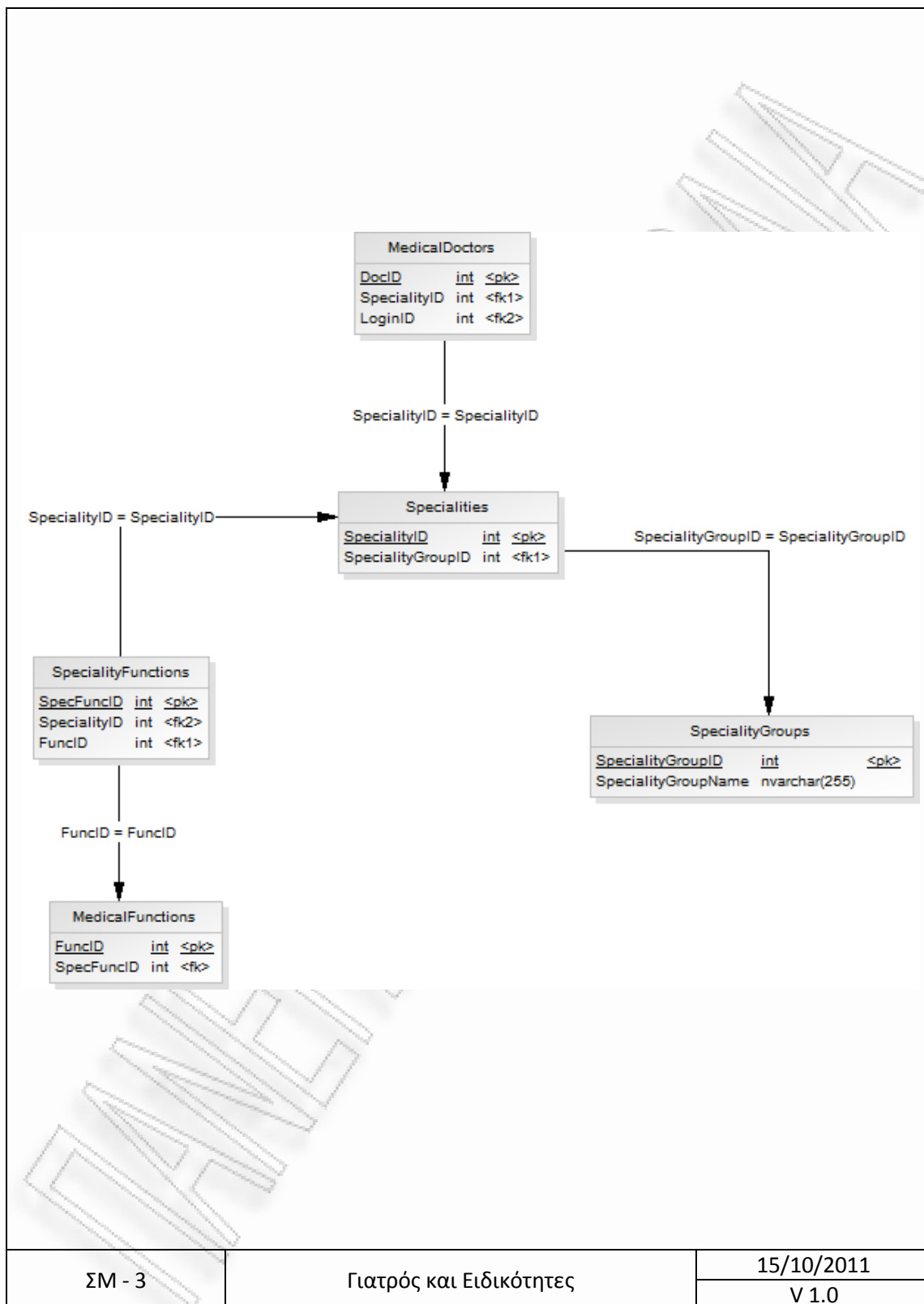
- Κάθε Γιατρός έχει ένα username και password για να κάνει Login στο σύστημα.
- Login στο σύστημα κάνουν πολλοί Γιατροί.
- Κάθε Γραμματεία έχει ένα username και password για να κάνει Login στο σύστημα
- Login στο σύστημα κάνουν πολλές Γραμματείες
- Κάθε Ασθενής έχει ένα username και password για να κάνει Login στο σύστημα
- Login στο σύστημα κάνουν πολλοί Ασθενείς.
- Κάθε Login συνδέεται με ένα Ρόλο (γιατρός, γραμματεία, ασθενής)
- Σε κάθε Ρόλο αντιστοιχούν πολλά Logins.



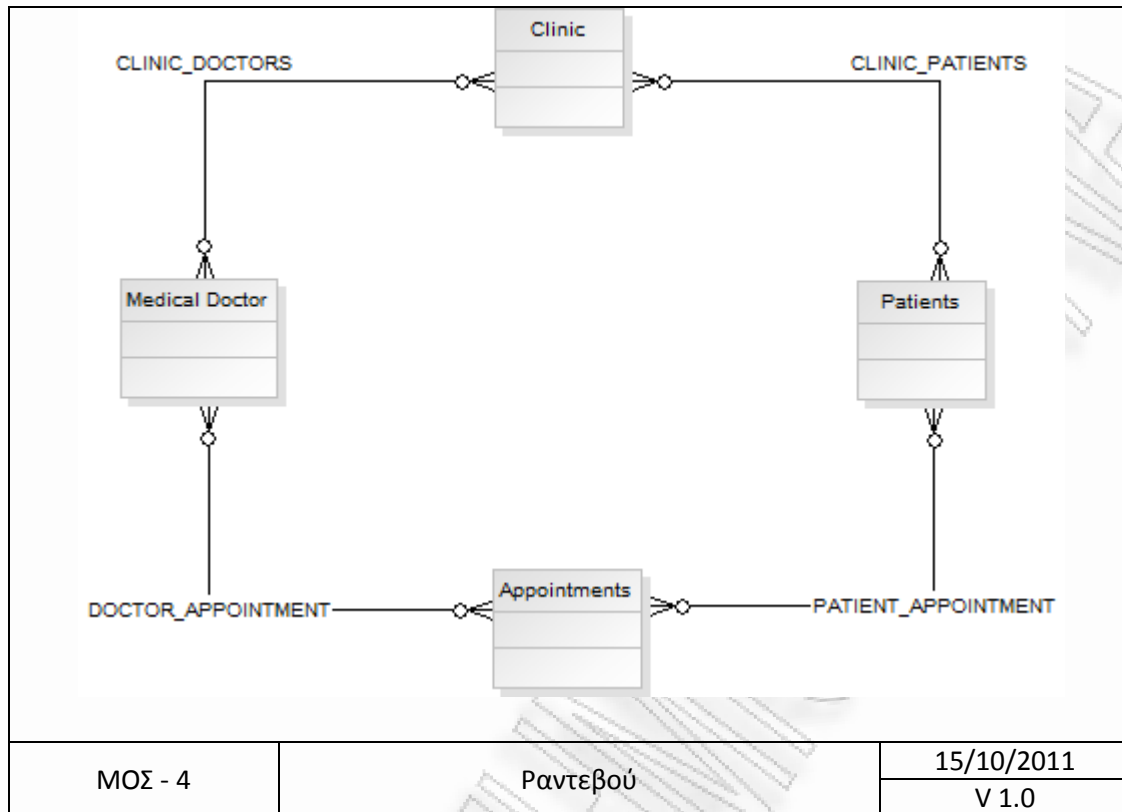
### Γιατρός και Ειδικότητες



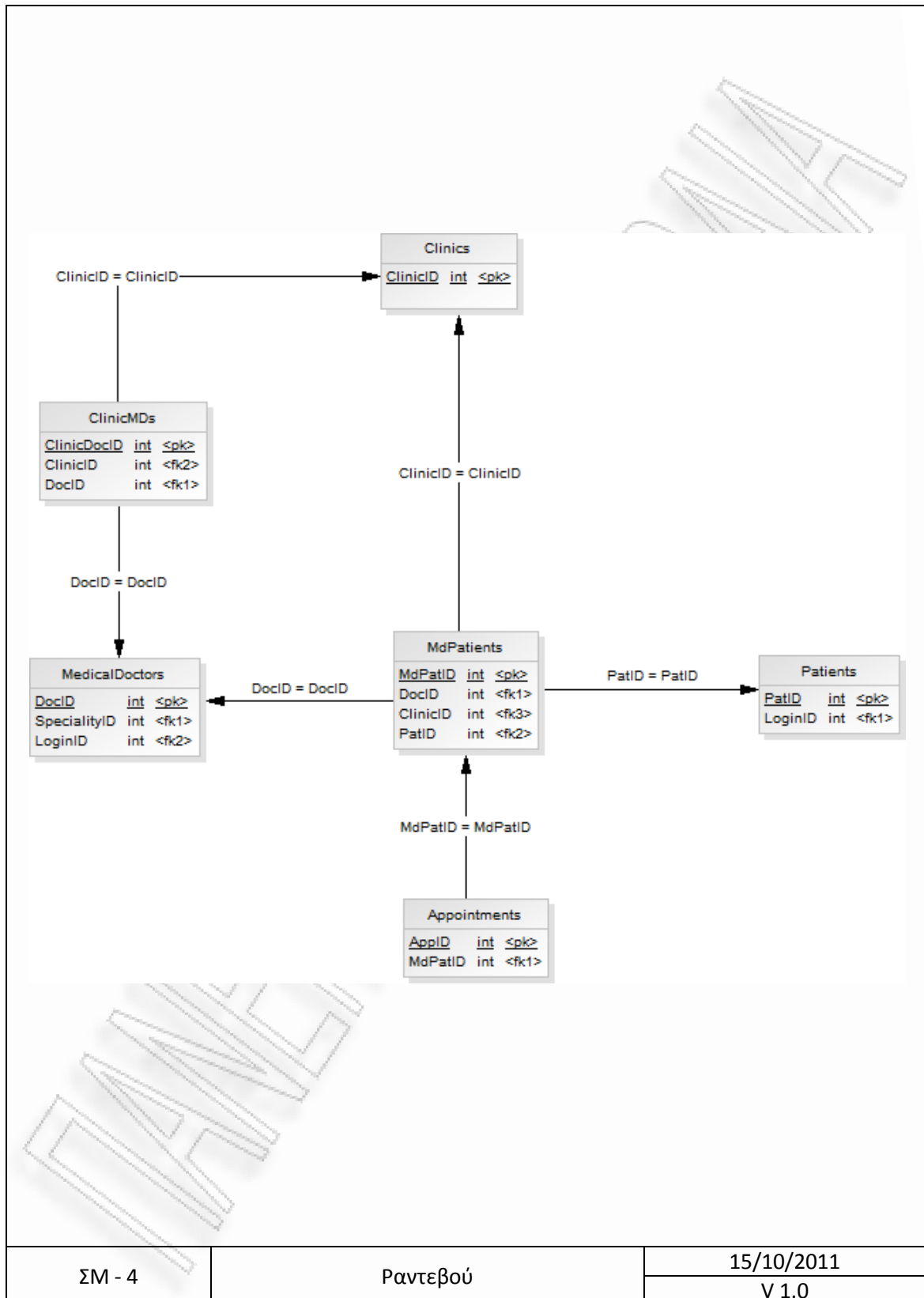
- Κάθε Γιατρός έχει μία Ειδικότητα.
- Μια Ειδικότητα μπορούν να την έχουν πολλοί Γιατροί.
- Μια Ειδικότητα ανήκει σε μια Κατηγορία Ειδικοτήτων
- Μια Κατηγορία Ειδικοτήτων περιλαμβάνει πολλές Ειδικότητες.
- Ο Γιατρός μιας Ειδικότητας μπορεί να εκτελέσει πολλές Ιατρικές Πράξεις και μια Ιατρική Πράξη μπορεί να εκτελεστεί από Γιατρούς πολλών Ειδικοτήτων
- Ένας Γιατρός μπορεί να εκτελέσει μόνο τις Ιατρικές Πράξεις που σχετίζονται με την Ειδικότητά του.



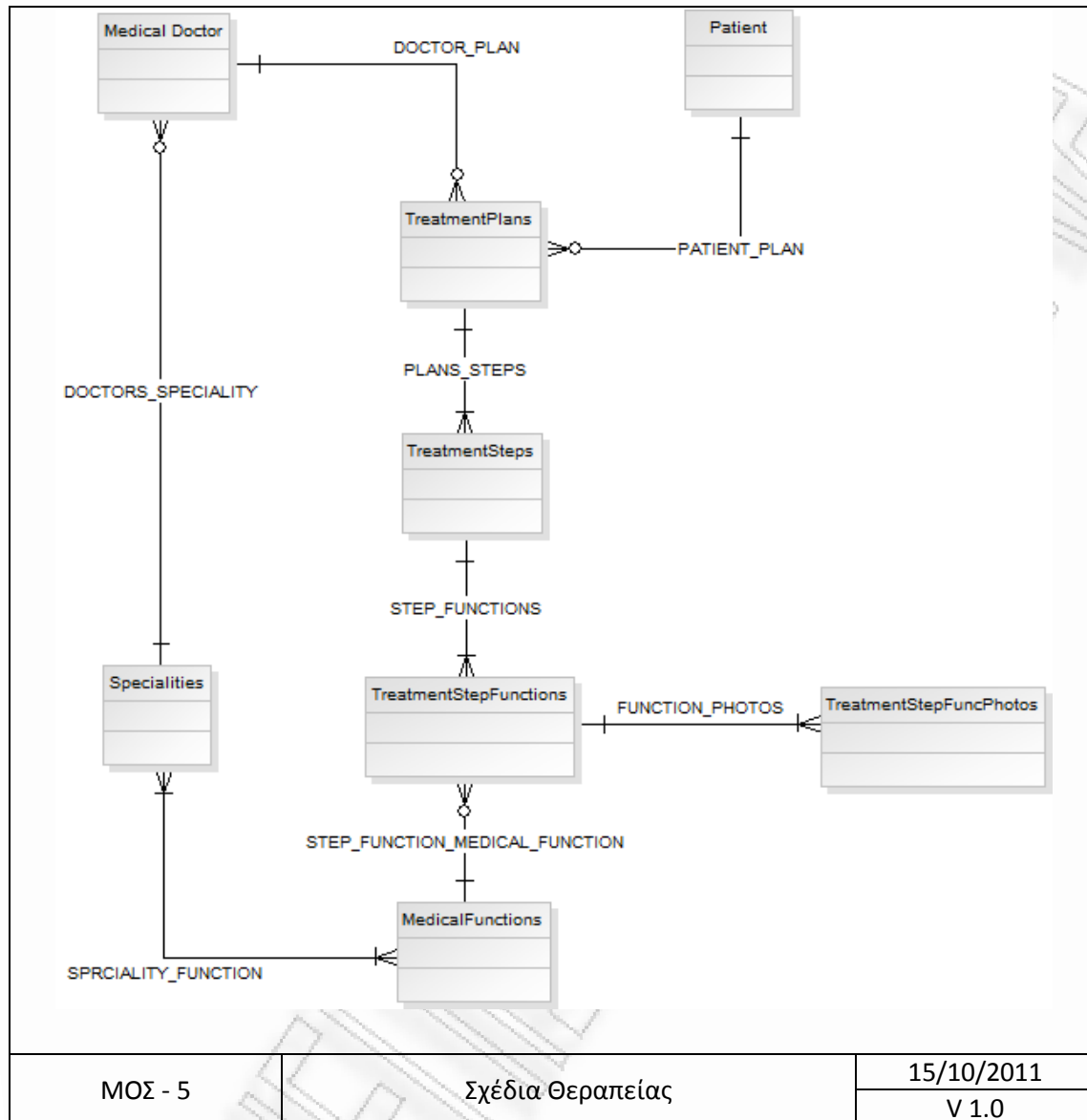
## Ραντεβού



- Ένας Γιατρός έχει Ραντεβού σε πολλές Κλινικές και σε μια Κλινική έχουν Ραντεβού πολλοί Γιατροί.
- Έναν Γιατρός έχει πολλά Ραντεβού με έναν Ασθενή και ένα Ασθενής έχει Ραντεβού με πολλούς Γιατρούς.
- Σε μια Κλινική έχουν Ραντεβού πολλοί Ασθενείς και ένα Ασθενής έχει Ραντεβού σε πολλές Κλινικές.
- Ένα Ραντεβού αφορά ένα Γιατρό και έναν Ασθενή σε μια Κλινική



## Σχέδια Θεραπείας

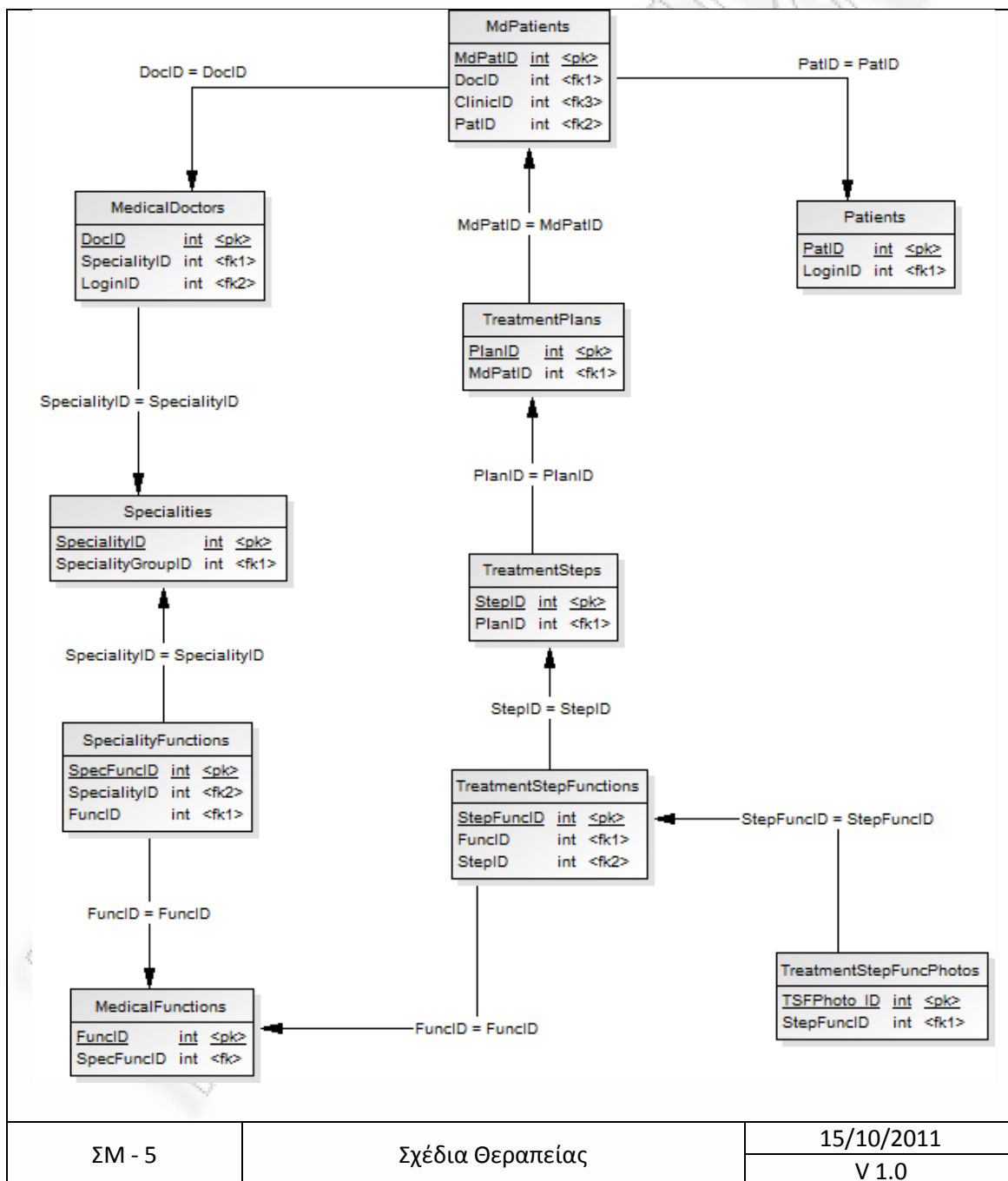


- Ένας Γιατρός ετοιμάζει πολλά Σχέδια Θεραπείας και κάθε Σχέδιο Θεραπείας ανήκει σε έναν Γιατρό.
- Ένας Ασθενής έχει πολλά Σχέδια Θεραπείας και ένα Σχέδιο Θεραπείας αφορά έναν Ασθενή.
- Ένα Σχέδιο Θεραπείας έχει πολλά Βήματα Θεραπείας και ένα Βήμα Θεραπείας ανήκει σε ένα Σχέδιο Θεραπείας.
- Ένα Βήμα Θεραπείας περιλαμβάνει πολλές Πράξεις Θεραπείας και μια Πράξη Θεραπείας εκτελείται σε ένα Βήμα Θεραπείας.

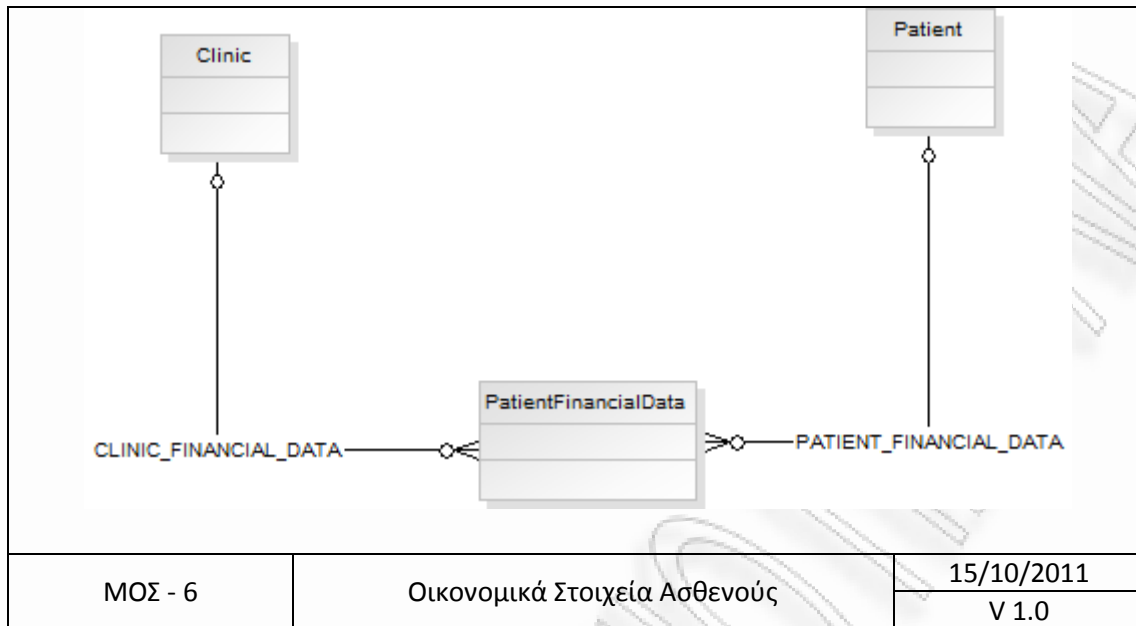




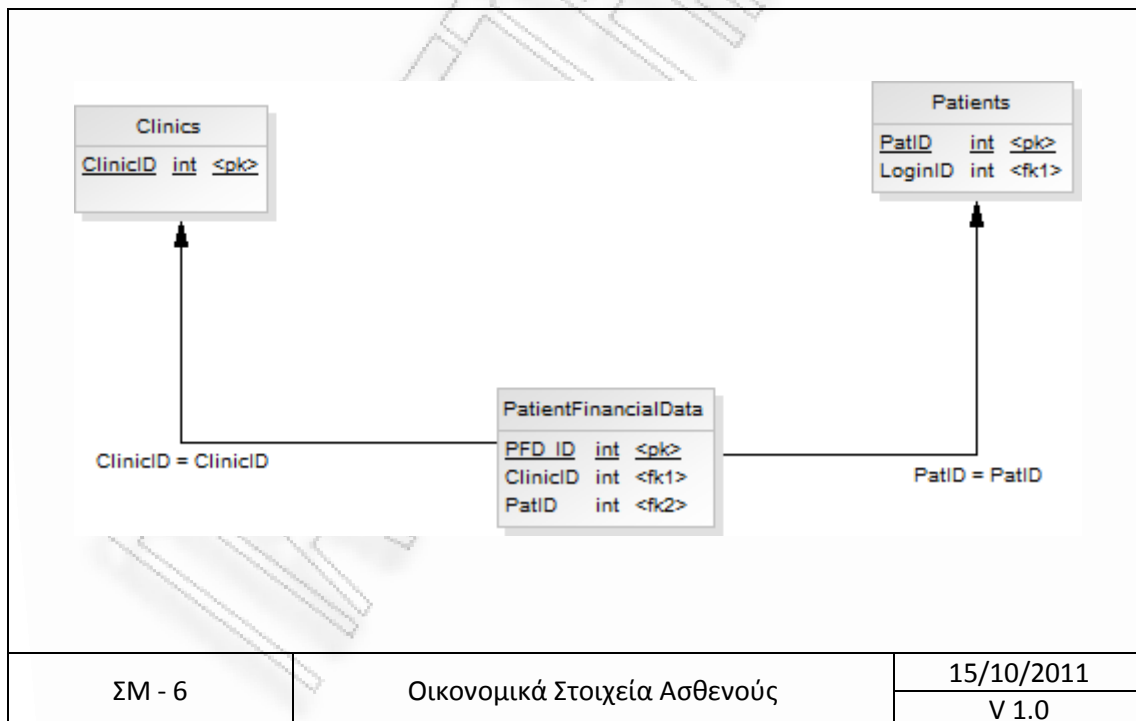
- Κάθε Πράξη Θεραπείας είναι μια Ιατρική Πράξη και μια Ιατρική Πράξη μπορεί να εκτελεστεί σε πολλές Πράξεις Θεραπείας
- Με κάθε Πράξη Θεραπείας μπορούν να συνδεθούν πολλές Φωτογραφίες και μια Φωτογραφία αφορά μια Πράξη Θεραπείας.
- Ένας Γιατρός μπορεί να εκτελέσει μόνο Ιατρικές Πράξεις που σχετίζονται με την Ειδικότητά του σε ένα Σχέδιο Θεραπείας.



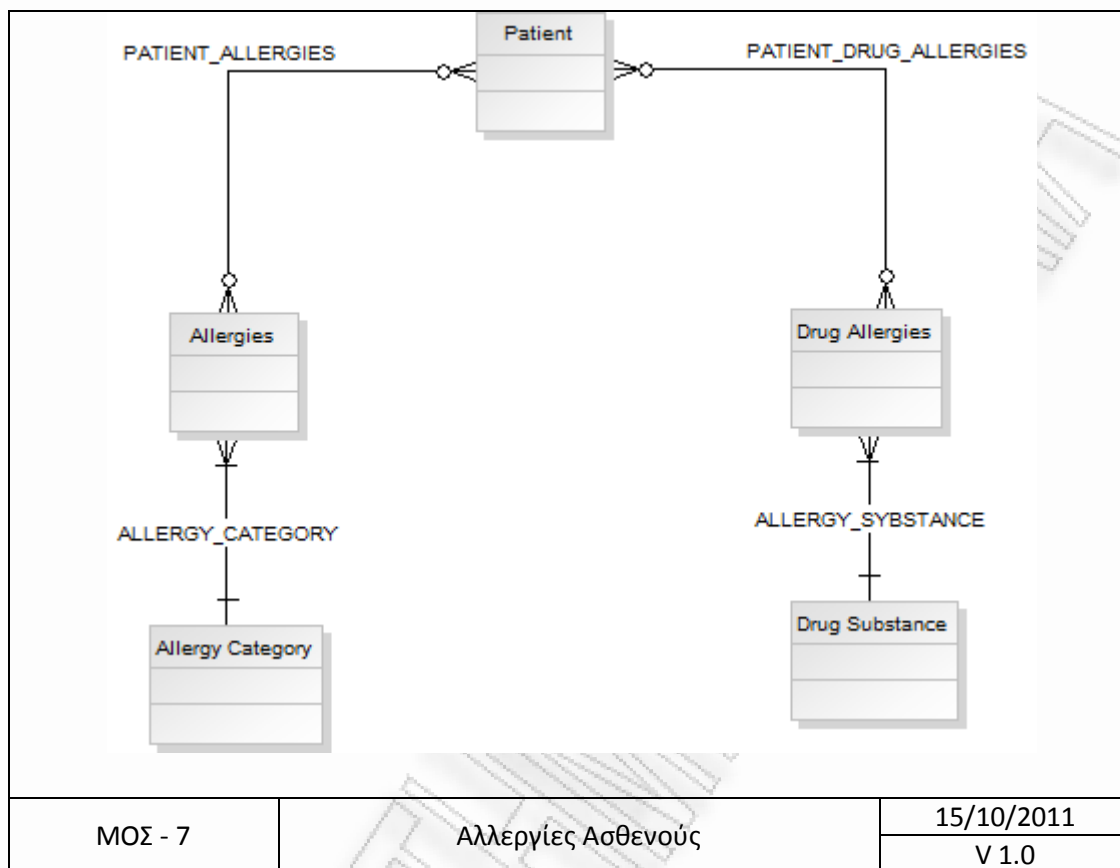
### Οικονομικά Στοιχεία Ασθενούς



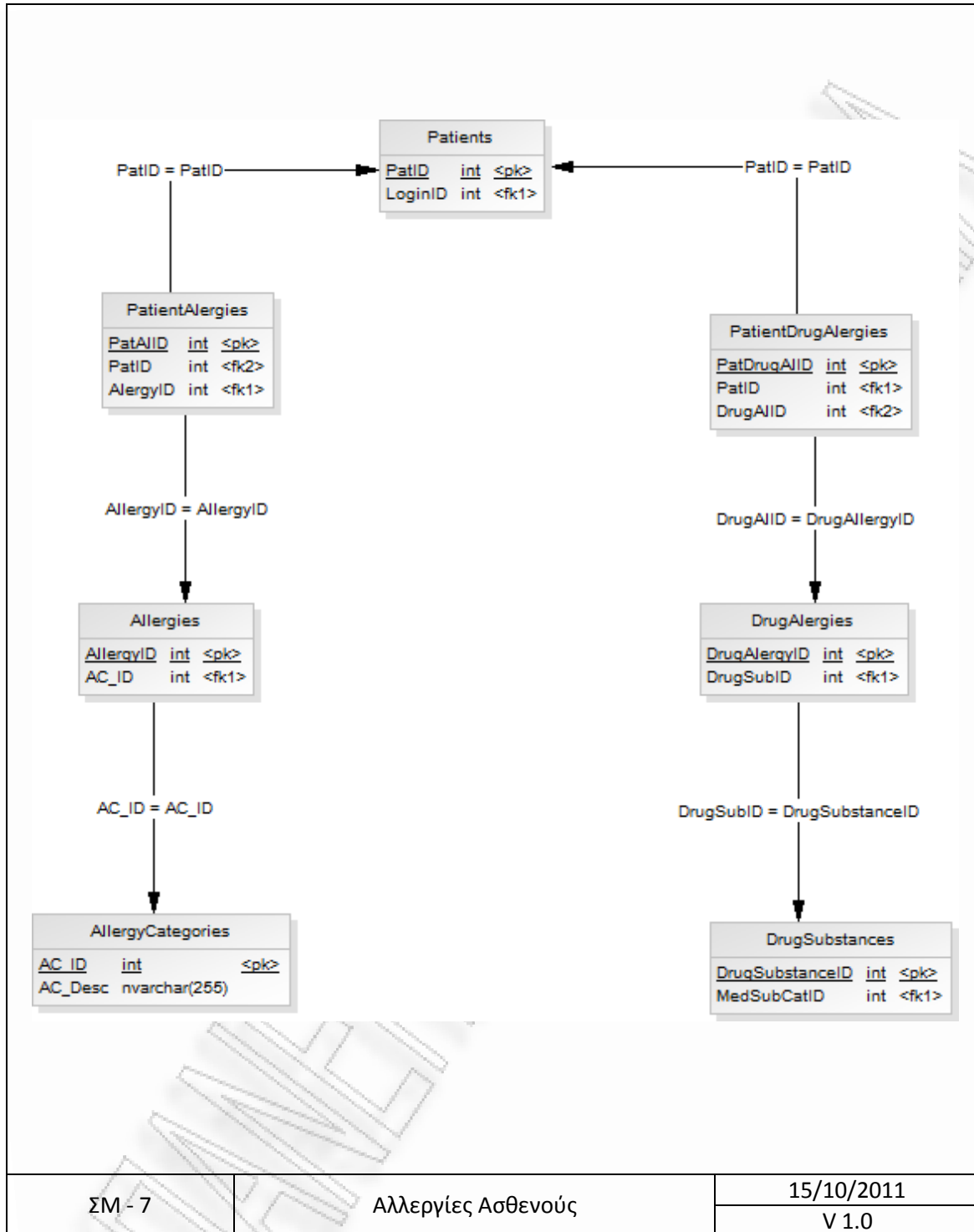
- Μια Κλινική διατηρεί Οικονομικά Στοιχεία για πολλούς Ασθενείς.
- Για έναν Ασθενή διατηρούνται Οικονομικά Στοιχεία σε πολλές Κλινικές.



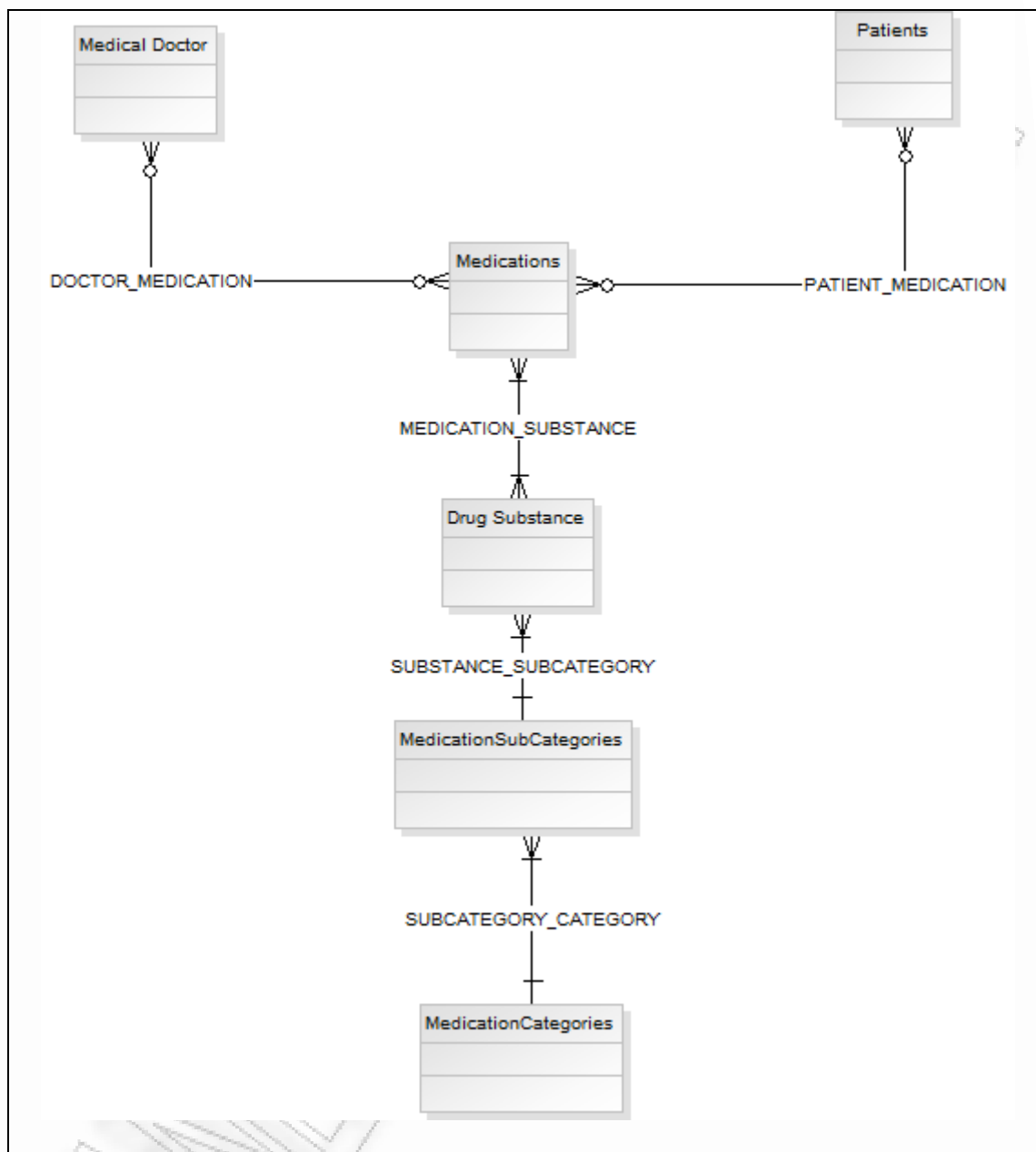
## Αλλεργίες Ασθενούς



- Ένας Ασθενής έχει πολλές Αλλεργίες
- Μια Αλλεργία εμφανίζεται σε πολλούς Ασθενείς
- Μια Αλλεργία ανήκει σε μια Κατηγορία Αλλεργιών
- Μια Κατηγορία Αλλεργιών περιλαμβάνει πολλές Αλλεργίες.
- Ένας Ασθενής έχει πολλές Φαρμακευτικές Αλλεργίες
- Μια Φαρμακευτική Αλλεργία εμφανίζεται σε πολλούς Ασθενείς
- Μια Φαρμακευτική Αλλεργία σχετίζεται με μια Φαρμακευτική Ουσία
- Μια Φαρμακευτική Ουσία μπορεί να προκαλέσει πολλές φαρμακευτικές Αλλεργίες



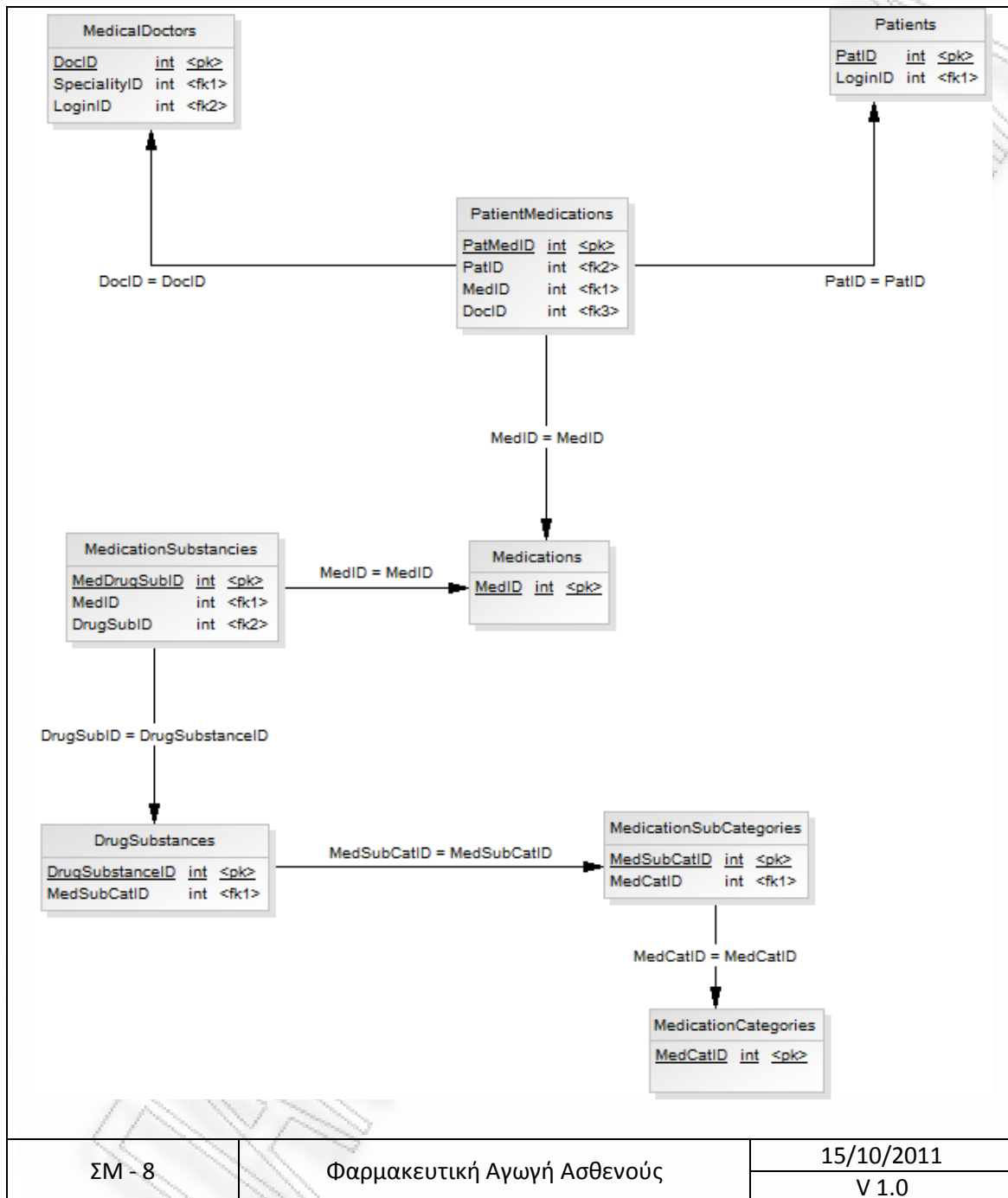
## Φαρμακευτική Αγωγή Ασθενούς



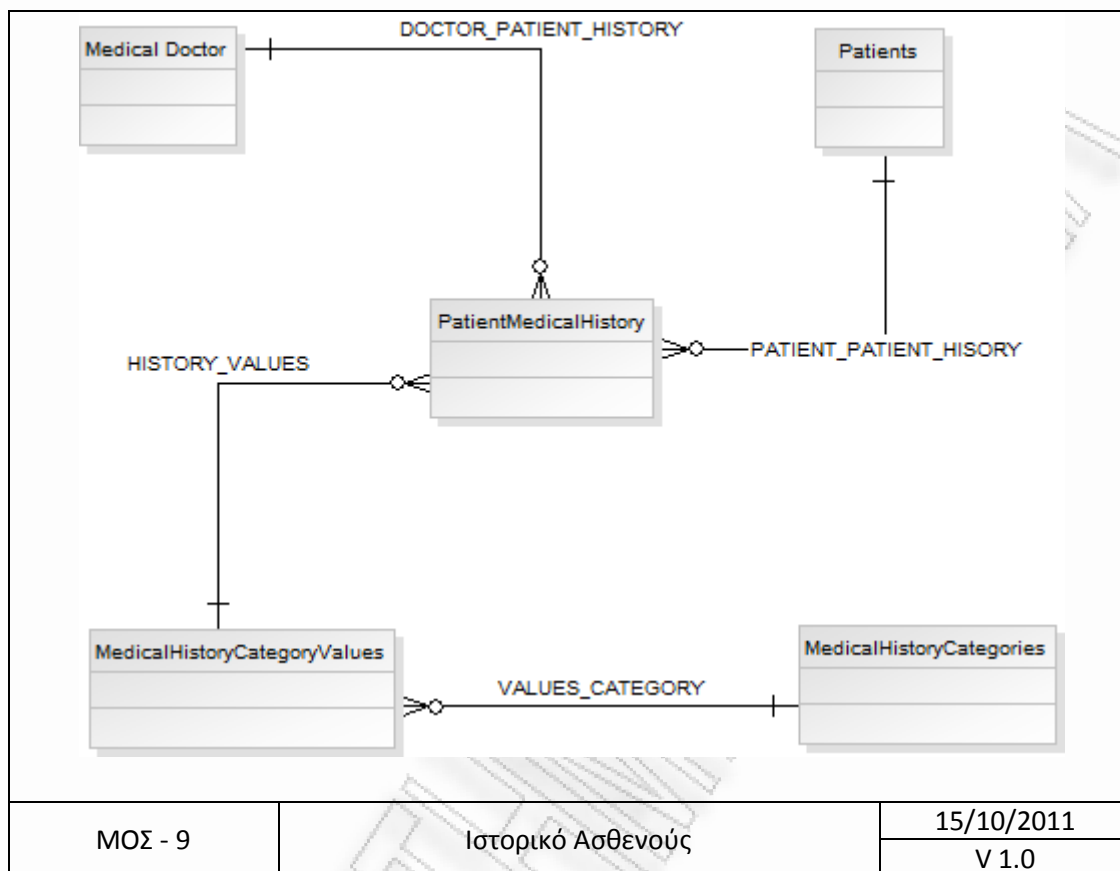
ΜΟΣ - 8	Φαρμακευτική Αγωγή Ασθενούς	15/10/2011
		V 1.0

- Ένας Γιατρός χορηγεί Φάρμακα σε πολλούς Ασθενείς.
- Ένας Ασθενής παίρνει Φάρμακα που του έχουν συνταγογραφήσει πολλοί Γιατροί.
- Ένα Φάρμακο περιέχει πολλές Φαρμακευτικές Ουσίες.
- Μια Φαρμακευτική Ουσία συναντάται σε πολλά Φάρμακα.
- Μια Φαρμακευτική Ουσία ανήκει σε μια Υποκατηγορία Φαρμάκων.
- Μια Υποκατηγορία Φαρμάκων περιλαμβάνει πολλές Φαρμακευτικές Ουσίες.

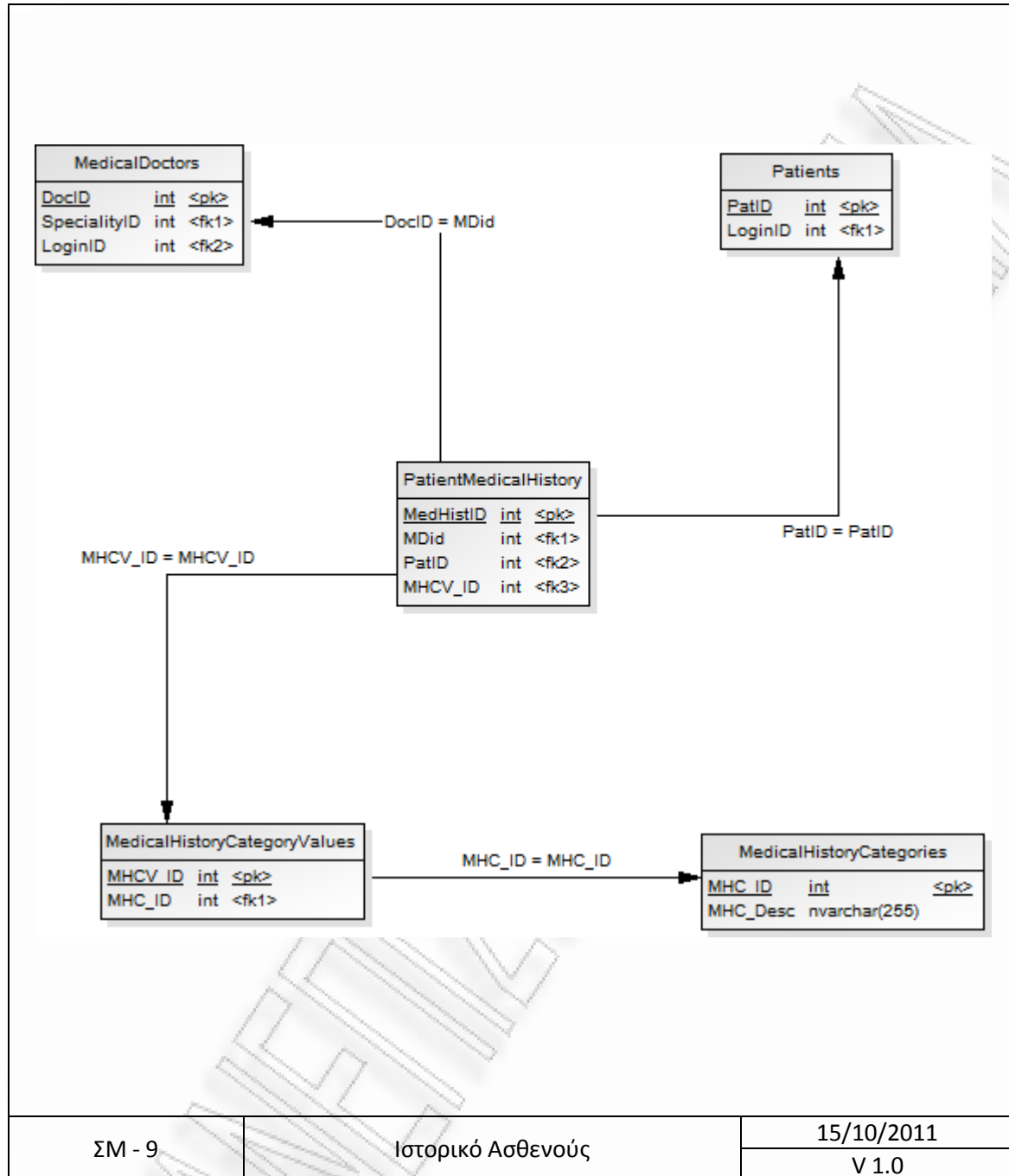
- Μια Υποκατηγορία φαρμάκων ανήκει σε μια Κατηγορία Φαρμάκων
- Μια Κατηγορία Φαρμάκων περιλαμβάνει πολλές Υποκατηγορίες Φαρμάκων



## Ιστορικό Ασθενούς



- Ένας Γιατρός ενημερώνει το Ιατρικό Ιστορικό πολλών Ασθενών
- Το Ιατρικό Ιστορικό ενός Ασθενούς ενημερώνεται από πολλούς Γιατρούς.
- Κάθε Ιατρικό Ιστορικό περιλαμβάνει μια από τις Τιμές μιας Κατηγορίας Ιατρικού Ιστορικού
- Μια Τιμή μιας Κατηγορίας Ιατρικού Ιστορικού εμφανίζεται στο ιατρικό ιστορικό πολλών Ασθενών.
- Μια Τιμή Κατηγορίας Ιατρικού Ιστορικού ανήκει σε μια Κατηγορία Ιατρικού Ιστορικού
- Μια Κατηγορία Ιατρικού Ιστορικού περιλαμβάνει πολλές Τιμές Κατηγορίας Ιατρικού Ιστορικού.







## Κεφάλαιο 6 Πρότυπα, Εργαλεία, Τεχνολογίες και Γλώσσες Υλοποίησης

Στο κεφάλαιο αυτό θα αναφερθούμε συνοπτικά στα εργαλεία και τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος.

### Microsoft .NET Framework

Το Microsoft .NET Framework είναι μια τεχνολογία λογισμικού που είναι διαθέσιμη με διάφορα λειτουργικά συστήματα της Microsoft. Περιλαμβάνει:

- μια μεγάλη βιβλιοθήκη έτοιμων λύσεων στα κοινά προβλήματα προγραμματισμού,
- ένα περιβάλλον εκτέλεσης (runtime)
- μια εικονική μηχανή (virtual machine), που διαχειρίζεται την εκτέλεση των προγραμμάτων, τα οποία γράφονται συγκεκριμένα για το πλαίσιο αυτό, και
- ένα σύνολο εργαλείων για τη διαμόρφωση (configuration) και το χτίσιμο (building) των εφαρμογών.

Το .NET Framework έχει σχεδιαστεί για να εκπληρώσει τους ακόλουθους στόχους:

- Να παρέχει ένα συνεπές αντικειμενοστρεφές περιβάλλον προγραμματισμού:
  - είτε ο κώδικας αποθηκεύεται και εκτελείται τοπικά,
  - είτε εκτελείται τοπικά αλλά διανέμεται στο Διαδίκτυο (Internet-distributed),
  - είτε εκτελείται απομακρυσμένα (remotely).
- Να παρέχει ένα περιβάλλον εκτέλεσης κώδικα, που ελαχιστοποιεί την διανομή λογισμικού (software deployment) και τα προβλήματα των εκδόσεων (versioning conflicts).
- Να παρέχει ένα περιβάλλον εκτέλεσης κώδικα, που προωθεί την ασφαλή εκτέλεση του κώδικα, συμπεριλαμβανομένου του κώδικα που δημιουργείται από ένα άγνωστο ή ημι-εμπιστευμένο τρίτο πρόσωπο.
- Να παρέχει ένα περιβάλλον εκτέλεσης κώδικα, που περιορίζει τα προβλήματα απόδοσης (performance) των διερμηνευόμενων (scripted) ή ερμηνευμένων (interpreted) περιβαλλόντων.



- Να δώσει στον υπεύθυνο για την ανάπτυξη την ευκαιρία να ασχοληθεί με μια μεγάλη ποικιλία εφαρμογών, όπως οι βασισμένες στα WINDOWS εφαρμογές και οι βασισμένες στο διαδίκτυο εφαρμογές.
- Να εξασφαλίσει ότι ο κώδικας που έχει γραφτεί, βασισμένος στο .NET Framework, μπορεί να ενοποιηθεί με οποιοδήποτε άλλο κώδικα.

Το .NET Framework αποτελείται από δύο κύρια συστατικά: το περιβάλλον εκτέλεσης κοινής γλώσσας (Common Language Runtime - CLR) και τη βιβλιοθήκη κλάσεων (Class Library). Το CLR λειτουργεί ως ένας πράκτορας που διαχειρίζεται τον κώδικα στο χρόνο εκτέλεσης, παρέχει τις υπηρεσίες πυρήνα, όπως η διαχείριση μνήμης και η διαχείριση νημάτων, καθώς επίσης επιβάλλοντας την ασφάλεια των βιβλιοθηκών, των μεθόδων και των κλάσεων, και άλλες μορφές ακρίβειας κώδικα, που προωθούν την ασφάλεια και την ευρωστία. Στην πραγματικότητα, η έννοια της διαχείρισης κώδικα είναι μια θεμελιώδης αρχή του χρόνου εκτέλεσης. Η βιβλιοθήκη κλάσεων, το άλλο κύριο συστατικό του .NET Framework, είναι μια περιεκτική, αντικειμενοστρεφής συλλογή των επαναχρησιμοποιήσιμων τύπων που μπορούν να χρησιμοποιηθούν για την ανάπτυξη, οι οποίες κυμαίνονται από τις παραδοσιακές εφαρμογές γραμμής εντολών ή γραφικής διεπαφής με το χρήστη (Graphical User Interface - GUI), ως τις εφαρμογές βασισμένες στις πιο πρόσφατες καινοτομίες που παρέχονται από την ASP.NET, όπως οι Web Forms και τα XML Web Services. Το CLR είναι υποσύνολο του CLI (Common Language Infrastructure), και είναι πρότυπο ECMA – 335.

Το .NET Framework στην έκδοση 4 αποτέλεσε την πλατφόρμα ανάπτυξης του συστήματος, μέσω των Windows Forms και της ASP.NET που αποτελούν υποσύνολό του.

### **Microsoft Visual Studio**

Το Microsoft Visual Studio είναι το κύριο ενσωματωμένο περιβάλλον ανάπτυξης (Integrated Development Environment - IDE) που έχει δημιουργήσει η Microsoft. Μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών κονσόλας ή γραφικής διεπαφής χρήστη, όπως επίσης Windows εφαρμογών, ιστοχώρων, εφαρμογών διαδικτύου και υπηρεσιών διαδικτύου.

Το Visual Studio περιλαμβάνει έναν συντάκτη κώδικα που υποστηρίζει το IntelliSense καθώς επίσης και αναπαραγωγή κώδικα. Ο ενσωματωμένος διορθωτής (debugger) λειτουργεί και ως διορθωτής επιπέδου πηγαίου κώδικα και ως διορθωτής επιπέδου μηχανής. Άλλα ενσωματωμένα εργαλεία που περιλαμβάνονται είναι ο σχεδιαστής φορμών για την δημιουργία GUI εφαρμογών, ο σχεδιαστής εφαρμογών Ιστού, ο σχεδιαστής κλάσεων και ο



σχεδιαστής σχημάτων βάσεων δεδομένων. Επιτρέπει την προσθήκη plug-ins για την ενίσχυση της λειτουργίας σχεδόν σε κάθε επίπεδο, συμπεριλαμβανομένης της προσθήκης υποστήριξης για συστήματα ελέγχου πηγής (όπως το Supervision και το Team Foundation Server), την προσθήκη νέων συνόλων εργαλείων, όπως οι συντάκτες και οι οπτικοί σχεδιαστές για τις εξαρτώμενες από το πεδίο γλώσσες ή εργαλεία για άλλες πτυχές του κύκλου ζωής της ανάπτυξης λογισμικού (όπως ο Team Explorer).

Το Visual Studio υποστηρίζει διάφορες γλώσσες με τη βοήθεια των γλωσσικών υπηρεσιών, οι οποίες επιτρέπουν σε οποιαδήποτε γλώσσα προγραμματισμού να υποστηριχθεί (σε διαφορετικό βαθμό) από το συντάκτη και το διορθωτή κώδικα. Οι ενσωματωμένες γλώσσες περιλαμβάνουν τη C, τη C++, τη Visual Basic .NET και τη Visual C#. Η υποστήριξη για άλλες γλώσσες, όπως η Chrome, η F#, η Python, και η Ruby, μεταξύ άλλων, μπορεί να υπάρξει μέσω γλωσσικών υπηρεσιών, που πρέπει να εγκατασταθούν ξεχωριστά. Υποστηρίζει, επίσης, την XML/XSLT, την HTML/XHTML, τη JavaScript και τα CSS.

Το Microsoft Visual Studio στην έκδοση 2010 αποτέλεσε το εργαλείο ανάπτυξης όλων των εφαρμογών.

### **Microsoft SQL Server**

Ο Microsoft SQL Server είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) που παράγεται από τη Microsoft. Η γλώσσα διατύπωσης επερωτήσεων του είναι η Transact-SQL (T-SQL), μια εφαρμογή του ANSI/ISO προτύπου με όνομα Δομημένη Γλώσσα Επερωτήσεων (Structured Query Language - SQL) που χρησιμοποιείται και από τη Microsoft και τη Sybase.

Μερικοί από τους λόγους χρήσης του MS-SQL Server είναι το ενσωματωμένο security μέσω του Windows Active Directory, η δυνατότητα που παρέχει για αυτοματοποιημένα backups (Full και Incremental), οι δυνατότητες που παρέχει για scalability και redundancy μέσω clustering και η δυνατότητα διαχωρισμού και οι αυξημένες δυνατότητες παραλληλίας εκτέλεσης των ζητούμενων ερωτημάτων .

Ο Microsoft SQL Server 2008 αποτέλεσε το σύστημα διαχείρισης της βάσης του συστήματος.

### **Windows Forms**

Ως Windows Forms αναφέρεται η Γραφική Προγραμματιστική Διεπαφή η οποία παρέχεται από το .NET framework για την δημιουργία της διεπαφής της εφαρμογής με το χρήστη.

Σημαντικότερο πλεονέκτημα των Windows Forms είναι η πληθώρα 3rd party controls για την γρήγορη ανάπτυξη ενός ιδιαίτερα φιλικού και καλαίσθητου User Interface. Για τη ανάπτυξη των windows εφαρμογών του συστήματος χρησιμοποιήθηκαν controls των DevExpress και DotNetBar.

### **ASP.NET**

Η ASP.NET είναι ένα πλαίσιο εφαρμογών Ιστού (web application framework), που αναπτύσσεται και συντηρείται από τη Microsoft, την οποία οι προγραμματιστές μπορούν να χρησιμοποιήσουν για να δημιουργήσουν δυναμικούς ιστοχώρους, εφαρμογές Ιστού και υπηρεσίες Ιστού. Παρουσιάστηκε, αρχικά, τον Ιανουάριο του 2002 με την έκδοση 1.0 του .NET Framework και είναι ο διάδοχος των Σελίδων Ενεργού Εξυπηρετητή (Active Server Pages - ASP) της Microsoft. Η ASP.NET στηρίζεται στο χρόνο εκτέλεσης κοινής γλώσσας (CLR), επιτρέποντας στους προγραμματιστές να γράψουν ASP.NET κώδικα χρησιμοποιώντας οποιαδήποτε υποστηρίξιμη .NET γλώσσα.

Η ASP.NET στην έκδοση 3.5 αποτέλεσε την πλατφόρμα κατασκευής του Ηλεκτρονικού Φακέλου Ασθενούς.

### **Windows Workflow Foundation**

Το Windows Workflow Foundation (WF) είναι μια τεχνολογία της Microsoft που παρέχει ένα API, μια μηχανή εκτέλεσης ροών εργασίες, καθώς και ένα σχεδιαστικό περιβάλλον για την υλοποίηση διεργασιών που εκτελούνται ως ροές εργασίας στα πλαίσια .NET εφαρμογών. Η τρέχουσα έκδοση του WF κυκλοφόρησε ως μέρος του .NET Framework 4 και αναφέρεται ως (WF4).

Ένα work flow αποτελείται από μία ακολουθία διακριτών προγραμματιστικών βημάτων, όπου το καθένα είναι μοντελοποιημένο ως μία ενέργεια. Το .NET Framework παρέχει μια βιβλιοθήκη ενεργειών, οι οποίες παραμετροποιημένες καταλλήλως συντελούν στην εκτέλεση της επιθυμητής λειτουργικότητας. Επίσης, παρέχεται η δυνατότητα στον προγραμματιστή να δημιουργήσει νέες ενέργειες και να τις εντάξει ως δομικά στοιχεία στο work flow του.

Με την ενθυλάκωση (encapsulation) λειτουργιών σε μία ενέργεια, δίνεται η δυνατότητα παραγωγής ευκολότερου και καλύτερα διαχειρίσιμου κώδικα. Άλλο ένα σημαντικό στοιχείο των work flows είναι η ευκολία που παρέχεται στη διαχείριση διαδικασιών, των οποίων η



ολοκλήρωσή απαιτεί μεγάλες αναμονές, μέσω της διατήρησης καταστάσεων (state) που παρέχει.

Το Windows Workflow Foundation χρησιμοποιήθηκε για την κατασκευή των Επιχειρησιακών Υπηρεσιών του συστήματος.

### **C# (C-Sharp)**

Η C# είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού που αναπτύχθηκε από τη Microsoft ως τμήμα του .NET Framework και που εγκρίθηκε αργότερα ως πρότυπο από την ECMA (ECMA - 334) και τον ISO (ISO/IEC 23270). Ο Anders Hejlsberg, σχεδιαστής της Delphi, καθοδηγεί την ανάπτυξη της C#, η οποία έχει αντικειμενοστρεφή σύνταξη βασισμένη στη C++ ενώ περιλαμβάνει επιρροές από διάφορες άλλες γλώσσες προγραμματισμού (ειδικότερα τη Delphi και τη Java), με μια ιδιαίτερη έμφαση στην απλοποίηση.

Η C# αποτέλεσε την γλώσσα υλοποίησης όλων των εφαρμογών του συστήματος.

## Κεφάλαιο 7

### Συμπεράσματα

Η Ιατρική Επιστήμη εξελίσσεται και προσαρμόζεται διαρκώς στις ανάγκες της κοινωνίας. Η εξέλιξη της τεχνολογίας σε συνδυασμό με την έρευνα και τη μελέτη έχουν συμβάλει καθοριστικά στην ανάπτυξη αυτή. Κύριο στοιχείο και οδηγός αυτής της εξέλιξης στον τομέα της υγείας αποτελεί η καταγραφή των γεγονότων και η μετέπειτα ανάλυση αυτών, για να εξαχθούν συμπεράσματα που θα βοηθήσουν στην βελτίωση των συνθηκών της υγειονομικής περίθαλψης.

Η καταγραφή των περιστατικών υγείας είναι μια διαδικασία, η οποία δεν έχει νόημα να γίνεται συνολικά για ένα πληθυσμό, αλλά για κάθε άτομο χωριστά και θα πρέπει να είναι δυναμική και διαρκής. Μόνο έτσι μπορούν να εξαχθούν σωστά συμπεράσματα και να δοθούν λύσεις στα μεμονωμένα ιατρικά προβλήματα που παρουσιάζει ο καθένας μας.

Στις μέρες μας οι πολίτες έχουν γίνει πιο απαιτητικοί σε σχέση με παλιότερα σε ό,τι αφορά τις υπηρεσίες που τους προσφέρονται και ειδικά σε ένα χώρο τόσο ευαίσθητο όσο αυτός της υγείας. Έχουν την απαίτηση από τον γιατρό τους να είναι πλήρως ενήμερος για την κατάσταση της υγείας τους και να μπορεί να ενημερώσει και τους ίδιους αποτελεσματικά, ώστε να αισθάνονται σίγουροι και ασφαλείς.

Ο Ηλεκτρονικός Φάκελος Υγείας, λόγω της πληρότητας που τον χαρακτηρίζει, είναι χρήσιμος, όχι μόνο στον ασθενή πληθυσμό, αλλά και σε όλους όσους επιθυμούν να έχουν τα ιατρικά τους δεδομένα συγκεντρωμένα. Ένας πλήρης ιατρικός φάκελος αποτελεί από μόνος του ένα στοιχείο ασφάλειας προς το πρόσωπο το οποίο αφορά, αφού του παρέχει τη δυνατότητα να παρακολουθεί με σωστό τρόπο την πορεία της υγείας του.

Τα ιατρικά αρχεία, ωστόσο, ενός ασθενούς αποτελούν ιδιαίτερα ευαίσθητα προσωπικά δεδομένα. Αυτό σημαίνει ότι όποιος έχει πρόσβαση σε αυτά πρέπει να είναι άτομο, το οποίο δεν θα τα χρησιμοποιήσει με λάθος τρόπο και για δικό του όφελος. Η ασφάλεια των ιατρικών δεδομένων είναι ένα σημαντικότερο θέμα που πρέπει να συζητηθεί, αν και η τεχνολογία έχει δώσει αρκετές ουσιαστικές και αποτελεσματικές λύσεις.

#### Η συμβολή της SOA

Οι οργανισμοί υγειονομικής περίθαλψης καλούνται να διαχειριστούν ένα αυξανόμενο πλήθος δεδομένων και συστημάτων. Το κόστος δημιουργίας, με την ενσωμάτωση και συντήρηση



αυτών των συστημάτων αυξάνονται, όπως και τα αιτήματα των χρηστών του κάθε συστήματος. Οι οργανισμοί πρέπει να αντιμετωπίσουν τις εξελισσόμενες κλινικές απαιτήσεις καθώς και την υποστήριξη του κύκλου των εσόδων και των επιχειρηματικών λειτουργιών της διοίκησης. Επιπλέον, αυξάνονται οι απαιτήσεις για τη διαλειτουργικότητα με άλλους οργανισμούς, για την υποστήριξη της περίθαλψης σε περιφερειακό επίπεδο. Η υπηρεσιοστρεφής αρχιτεκτονική παρέχει το σχεδιασμό και τις αρχές διαχείρισης του συστήματος, έτσι ώστε να υποστηρίζουν την επαναχρησιμοποίηση και την κοινή χρήση των πόρων του συστήματος σε όλο το φάσμα της υγειονομικής περίθαλψης. Επιπλέον, δεν απαιτεί την αναδιοργάνωση των υφιστάμενων συστημάτων. Με τη SOA οι υπάρχουσες διαδικασίες μπορεί να συνδυαστούν με νέες δυνατότητες για την οικοδόμηση μιας βιβλιοθήκης υπηρεσιών που χρησιμοποιούνται ως μέρος των επιχειρηματικών λύσεων. Χρησιμοποιώντας κοινές υπηρεσίες που είναι ευθυγραμμισμένες με τις επιχειρηματικές διαδικασίες, η SOA ενισχύει τη διαλειτουργικότητα, ενώ μειώνει την ανάγκη για συγχρονισμό των δεδομένων μεταξύ των απομονωμένων συστημάτων. Οι υπηρεσίες μπορούν να διατίθενται, ανεξάρτητα από τη θέση τους, ώστε να δημιουργήσουν λύσεις που φτάνουν πέρα από το κάθε τμήμα και την οργάνωση της υγειονομικής περίθαλψης.

Η SOA ορίζει μια υπηρεσία ως ανεξάρτητη μονάδα εργασίας που είναι αυτοδύναμη και έχει καλά καθορισμένες και κατανοητές δυνατότητες. Μια μονάδα εργασίας μπορεί να είναι μια ολόκληρη διαδικασία, μια μέθοδος ενίσχυσης μιας διαδικασίας, ή ένα βήμα από μια επιχειρηματική διαδικασία. Οι υπηρεσίες υποστηρίζουν άμεσα τις επιχειρηματικές διαδικασίες και τις εννοχρηστώνουν ως λύση του συστήματος.

Κάθε λειτουργία του συστήματος μπορεί να διαχωριστεί σε επιμέρους τμήματα, ώστε να αυξήσει τη δυνατότητα επαναχρησιμοποίησης των υπηρεσιών. Για παράδειγμα, η λειτουργία «Καταχώρηση Ασθενούς» μπορεί να χωριστεί στις εργασίες «Εύρεση και προβολή φακέλου ασθενούς», «δημιουργία και ενημέρωση φακέλου ασθενούς», «έλεγχος της ασφάλισης του ασθενούς», «ενημέρωση ιατρικού ιστορικού» και άλλες επιχειρηματικές δραστηριότητες που εκτελούνται κατά τη διάρκεια της διαδικασίας καταχώρησης. Αυτός ο κατακερματισμός της διαδικασίας επιτρέπει σε άλλες υπηρεσίες και εφαρμογές να χρησιμοποιούν τμήματα της συγκεκριμένης διαδικασίας. Για παράδειγμα, η εργασία «Εύρεση και προβολή φακέλου ασθενούς» μπορεί να χρησιμοποιηθεί από το μεγαλύτερο μέρος του οργανισμού, ενώ η εργασία «δημιουργία και ενημέρωση φακέλου ασθενούς» μπορεί να χρησιμοποιηθεί μόνο από το ιατρικό προσωπικό. Η SOA παρέχει ένα περιβάλλον στο οποίο οι λειτουργίες μπορούν να τυποποιηθούν και να χρησιμοποιηθούν σε πολλά συστήματα και διαδικασίες.

### Μελλοντικές επεκτάσεις του υλοποιημένου συστήματος

Η παρούσα εργασία καλύπτει ένα μικρό μέρος των λειτουργικών απαιτήσεων ενός Ολοκληρωμένου Πληροφοριακού Συστήματος Υγείας. Γενικεύοντας τις παραδοχές που παρουσιάστηκαν κατά τον σχεδιασμό του συστήματος που υλοποιήθηκε, θα μπορούσαμε να ορίσουμε τις παρακάτω βασικές επεκτάσεις:

- Δυνατότητα ορισμού περισσότερων της μίας κλινικής( ή μονάδας υγείας γενικότερα)
- Δυνατότητα υποστήριξης όλων των ιατρικών ειδικοτήτων, πέραν της Οδοντιατρικής
- Δημιουργία εφαρμογής διαχείρισης των προμηθευτών και των συνεργαζόμενων με τον ιατρικό τομέα ειδικοτήτων (π.χ. τεχνίτες, συντηρητές ιατρικών μηχανημάτων και άλλα)
- Πλήρης ανεξαρτητοποίηση του συστήματος Ηλεκτρονικού Φακέλου
- Ανάπτυξη διαδικασιών στο σύστημα του Ηλεκτρονικού Φακέλου, που θα αυξάνουν την διαδραστικότητα του συστήματος με το Πληροφοριακό Σύστημα Υγείας (π.χ. δυνατότητα αίτησης για ανανέωση συνταγής, δυνατότητα ορισμού ραντεβού από τον ασθενή, αναζήτηση μονάδων υγείας και ιατρών και άλλα )
- Δυνατότητα ορισμού του είδους των ιατρικών δεδομένων του ασθενούς που μπορεί να προβληθεί σε άλλα πρόσωπα (π.χ. ποια στοιχεία του μπορεί να δει η γραμματεία κάθε κλινικής, περιορισμός των δεδομένων ανά ιατρική ειδικότητα κ.α.)
- Αύξηση της ασφάλειας των ιατρικών δεδομένων με τη χρήση, για παράδειγμα, έξυπνων καρτών, Barcode ή κωδικών πρόσβασης.

### Μελλοντικές επεκτάσεις του θεωρητικού πλαισίου

Όπως έχουμε ήδη αναφέρει, ο όγκος της πληροφορίας που σχετίζεται με τον τομέα της υγείας είναι τεράστιος και ολοένα αυξανόμενος. Είναι, επομένως, σαφές ότι για να υποστηριχθεί η λειτουργία ενός ολοκληρωμένου πληροφοριακού συστήματος υγείας σχεδιασμένο με βάση την υπηρεσιοστρεφή αρχιτεκτονική, απαιτεί μια μεγάλη επένδυση σε τεχνικό εξοπλισμό, ο οποίος θα υποστηρίζει την αποθήκευση των δεδομένων και την εκτέλεση των υπηρεσιών και θα εξασφαλίσει την συνεχή λειτουργία του συστήματος.

Επιπλέον, για να είναι προσβάσιμος ο Ηλεκτρονικός Φάκελος ενός ασθενούς από όλες τις μονάδες Υγείας, αλλά και από τον ίδιο τον ασθενή, όπου και αν αυτός βρίσκεται, οδηγούμαστε μοιραία στη χρήση του διαδικτύου.





Μια νέα προσέγγιση στον χώρο των καταναμημένων συστημάτων που χρησιμοποιεί και κάποιες υφιστάμενες τεχνολογίες αποτελεί το Cloud Computing. Σκοπός του είναι η παροχή πόρων με την μορφή υπηρεσίας στους χρήστες των συστημάτων. Τέτοιοι πόροι είναι η υπολογιστική ισχύς (CPU), η αποθηκευτική δυνατότητα (storage) κ.ά. Η έννοια κάθε άλλο παρά καινούρια είναι. Παλαιότερα ήταν γνωστό ως utility computing, grid computing κ.λπ. αλλά όπως όλες οι τεχνολογίες χρειάζονται το χρόνο τους για να ωριμάσουν και να γίνουν και οικονομικά ελκυστικές, ώστε να μην απασχολούν μόνο τους early adopters. Πλέον το cloud computing είναι πιο ώριμο από ποτέ να πραγματοποιήσει αυτό που υπόσχεται: να δώσει τους υπολογιστικούς πόρους με τη μορφή υπηρεσίας, με την ίδια λογική που διέπει άλλους τομείς, όπως την παροχή ηλεκτρικού ρεύματος. Η αγορά είναι πλέον ώριμη, αλλά ταυτοχρόνως με διαρκή εξέλιξη.

Έννοιες όπως οι Infrastructure as a Service(IaaS), Platform as a Service(PaaS) και Software as a Service(SaaS), έχουν αρχίσει να συζητούνται και να μελετώνται στον τομέα της πληροφορικής. Εντούτοις, η μεταφορά εφαρμογών και δεδομένων στο cloud και ιδιαίτερα αυτών που σχετίζονται με ευαίσθητα προσωπικά δεδομένα, όπως τα απόρρητα ιατρικά στοιχεία όλων των πολιτών, είναι αμφιλεγόμενη καθώς υπάρχουν αρκετές ανησυχίες σχετικά με την ασφάλεια που μπορεί να παρασχεθεί.

## Παράρτημα Α

### Ηλεκτρονικός Ιατρικός Φάκελος Ασθενούς – Ερωτηματολόγιο

1. Πόσο έχετε ακούσει για τις παρακάτω νέες τεχνολογίες στον τομέα της ιατρικής περίθαλψης;

	Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν είμαι σίγουρος
a. Χρήση ηλεκτρονικών ιατρικών φακέλων από γιατρούς	26	21	15	11	1
b. Εργαλεία που χρησιμοποιούνται στο σπίτι, όπως π.χ. εργαλεία μέτρησης της πίεσης του αίματος ή της γλυκόζης, τα οποία μπορούν να στείλουν τα αποτελέσματα των μετρήσεων σε γιατρούς μέσω του διαδικτύου.	22	12	20	18	2
c. Προγράμματα και εφαρμογές κινητών τηλεφώνων που χρησιμοποιούνται για την φύλαξη ή την παρακολούθηση ιατρικών πληροφοριών.	5	10	16	40	3
d. Ιστοσελίδες, όπου μπορεί κάποιος να φυλάσσει και να ενημερώνει το ιατρικό του ιστορικό, όπως αποτελέσματα εξετάσεων, φαρμακευτική αγωγή, επισκέψεις σε γιατρούς κ.α.	13	11	17	29	4

2. Έχετε ακούσει για προγράμματα της κυβέρνησης που έχουν ως σκοπό να προτρέψουν ιατρούς και νοσοκομεία, ώστε να χρησιμοποιούν περισσότερο την τεχνολογία;

Ναι	Όχι	Δεν είμαι σίγουρος
38	29	7

3. Διατηρεί ο γιατρός σας ηλεκτρονικούς ιατρικούς φακέλους;

Ναι	Όχι	Δεν είμαι σίγουρος
31	23	20

4. Έχετε ποτέ αισθανθεί αμηχανία ή απογοήτευση κατά την διάρκεια μιας επίσκεψής σας στο γιατρό, επειδή δεν γνωρίζετε τι γράφει στην ιατρική σας καρτέλα;

Ναι	Όχι	Δεν είμαι σίγουρος
16	52	6

5. a. Κατά την διάρκεια μιας επίσκεψής σας, ο γιατρός σας κρατάει πληροφορίες σε κάποιον υπολογιστή;

Ναι	Όχι	Δεν είμαι σίγουρος
48	20	6

**Εάν έχετε απαντήσει ΝΑΙ στην παραπάνω ερώτηση(ερώτηση 5α), απαντήστε στα παρακάτω. Διαφορετικά συνεχίστε στην ερώτηση 6 :** Κατά την διάρκεια της επίσκεψης έχετε σκεφτεί κάτι από τα παρακάτω, όταν ο γιατρός σας κρατάει πληροφορίες στον υπολογιστή;

**b. Ο γιατρός μου δίνει λιγότερη σημασία, εάν γράφει τις πληροφορίες σε χαρτί.**



Ναι Όχι Δεν είμαι σίγουρος  
6 38 8

c. Ο γιατρός έχει πολλές (ιατρικές) πληροφορίες για μένα στον υπολογιστή.

Ναι Όχι Δεν είμαι σίγουρος  
15 18 9

6. Κατά την διάρκεια μιας επίσκεψής σας, θα προτιμούσατε ο γιατρός σας να κρατάει πληροφορίες σε υπολογιστή ή να τις γράφει σε χαρτί;

Ναι Όχι Δεν έχει καμία διαφορά για μένα  
50 4 20

7. Σκεπτόμενοι την ιατρική σας περίθαλψη, πόσο συμφωνείτε / διαφωνείτε με τα παρακάτω;

	Συμφωνώ κάθετα	Συμφωνώ	Διαφωνώ	Διαφωνώ κάθετα	Δεν είμαι σίγουρος
a. Θα ήθελα η γιατροί να είχαν περισσότερο χρόνο για να συζητούν μαζί μου το όποιο ιατρικό μου πρόβλημα.	35	38	1	0	0
b. Αισθάνομαι ότι πρέπει να βρίσκω μόνος μου πληροφορίες σχετικά με τα ιατρικά μου προβλήματα, ώστε να κάνω τις απαραίτητες ερωτήσεις .	5	39	23	3	4
c. Θα ήθελα οι εμπλεκόμενοι γιατροί να επικοινωνούν περισσότερο μεταξύ τους και να ανταλλάσσουν πληροφορίες σχετικά με κάποιο θέμα της υγείας μου.	29	42	0	1	2
d. Το να συγκεντρώνω και να κρατώ τις ιατρικές πληροφορίες που με αφορούν, όπως αποτελέσματα εξετάσεων, φαρμακευτική αγωγή και ιατρικό ιστορικό, είναι δύσκολο.	11	29	24	6	4
e. Θα ήθελα ο γιατρός μου να ξέρει περισσότερα για εμένα και την υγεία μου.	25	44	4	0	1
f. Θα ήθελα να ξέρω περισσότερα για την ιατρική περίθαλψη των μελών της οικογενείας μου.	18	44	8	1	3

8. a. Κρατάτε εσείς ή κάποιο άλλο μέλος της οικογένειάς σας αρχεία ή αντίγραφα του δικού σας ιατρικού ιστορικού (εξετάσεις, φαρμακευτική αγωγή, άλλες ιατρικές πληροφορίες)

Ναι Όχι Δεν είμαι σίγουρος  
62 12 0

**b. Εάν απαντήσατε ΝΑΙ στην παραπάνω ερώτηση:** Που κρατάτε αυτά τα αρχεία;

Σε χαρτιά Στον υπολογιστή Και στα δύο  
57 2 3

9. Έχετε κάνει κάτι από τα παρακάτω χρησιμοποιώντας το διαδίκτυο ή το e-mail σας;



	Ναι	Όχι	Δεν είμαι σίγουρος
a. Έρευνα στο διαδίκτυο για πληροφορίες σχετικά με κάποιο ιατρικό πρόβλημα (δικό σας ή όχι)	62	11	1
b. Χρήση του διαδικτύου/e-mail για να πάρετε τα αποτελέσματα κάποιων εξετάσεων	11	63	0
c. Αποστολή ή λήψη e-mail σε/από τον γιατρό σας	16	57	1
d. Ηλεκτρονική ανανέωση κάποιας ιατρικής συνταγής	2	71	1
e. Εισαγωγή πληροφοριών σε κάποια ιστοσελίδα σχετικά με το τι τρώτε, τι βάρος έχετε, πόσο συχνά γυμνάζεστε κλπ.	27	45	2
f. Εισαγωγή πληροφοριών σε κάποια ιστοσελίδα σχετικά με κάποιο ιατρικό σας πρόβλημα	22	51	1
g. Αναζήτηση πληροφοριών για κάποιον γιατρό	34	40	0
<b>h. <u>Αν απαντήσατε ΝΑΙ στα e ή f παραπάνω:</u> Εισάγατε κάποια προσωπικά σας στοιχεία στην ιστοσελίδα, όπως όνομα, διεύθυνση ή αριθμός μητρώου ;</b>			
Ναι	Όχι	Δεν ξέρω	
5	34	1	

**10. Κάποιες εταιρίες, όπως η Microsoft και η Google, προσφέρουν δικτυακούς τόπους (ιστοσελίδες), όπου μπορείτε να κρατάτε και να ενημερώνετε το ιατρικό σας ιστορικό(εξετάσεις, φάρμακα, επισκέψεις σε γιατρούς κ.α.) μέσω διαδικτύου. Για τον σκοπό αυτό σας δίνετε ένας κωδικός πρόσβασης, ώστε να μπορείτε μόνο εσείς να έχετε πρόσβαση στις πληροφορίες αυτές.**

**a. Έχετε ποτέ χρησιμοποιήσει κάποια τέτοια ιστοσελίδα για να κρατήσετε το ιατρικό σας ιστορικό;**

Ναι	Όχι	Δεν ξέρω
0	74	0

**b. Έχετε ποτέ χρησιμοποιήσει κάποια τέτοια ιστοσελίδα για να κρατήσετε το ιατρικό ιστορικό συγγενικού σας προσώπου;**

Ναι	Όχι	Δεν ξέρω
0	74	0

**c. Εάν έχετε απαντήσει ΝΑΙ στην ερώτηση 10, ποιός σας σύστησε αυτή την ιστοσελίδα;**

- Ο γιατρός μου
- Το ασφαλιστικό μου ταμείο/ ασφαλιστική εταιρία
- Το αφεντικό μου
- Φίλος/Συγγενής
- Άλλο (παρακαλώ αναφέρετε): .....

**d. Εάν έχετε απαντήσει ΟΧΙ στην ερώτηση 10, πόσο θα σας ενδιέφερε να χρησιμοποιήσετε μια τέτοια ιστοσελίδα για να κρατήσετε το δικό σας ιατρικό ιστορικό;**

Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν ξέρω
11	31	13	14	5



**ε. Για να κρατήσετε το ιατρικό ιστορικό κάποιου συγγενικού σας προσώπου;**

Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν ξέρω
4	28	20	19	3

11. Παρακάτω αναφέρονται κάποιοι τρόποι που μπορεί κάποιος να χρησιμοποιήσει τις ιατρικές του πληροφορίες ηλεκτρονικά. Αν είχατε τη δυνατότητα να κρατάτε τις ιατρικές σας πληροφορίες ηλεκτρονικά σε κάποια ιστοσελίδα, πόσο θα σας ενδιέφερε να κάνετε τα παρακάτω;

	Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν είμαι σίγουρος
a. Να βλέπετε τα αποτελέσματα ιατρικών εξετάσεων	38	21	8	7	0
b. Να ελέγχετε αν οι πληροφορίες αυτές είναι σωστές	30	26	8	8	2
c. Να καθορίζετε τα ραντεβού με τον γιατρό σας	34	24	10	6	0
d. Να ανανεώνεται τις συνταγές σας	32	23	9	10	0
e. Να στέλνετε / Λαμβάνετε e-mail από τον γιατρό σας	29	21	11	12	1
f. Να λαμβάνετε υπενθυμίσεις για τα ραντεβού με τον γιατρό σας	32	19	13	10	0
g. Να κρατάτε και να ελέγχετε το ιατρικό ιστορικό των παιδιών σας	35	18	12	9	0
h. Να βλέπετε οδηγίες του γιατρού σας σχετικά με κάποιο ιατρικό σας πρόβλημα.	37	19	11	7	0

12. Σε περίπτωση που διατηρείτε ιατρικά στοιχεία σε κάποια ιστοσελίδα, απαντήστε στις παρακάτω ερωτήσεις. Διαφορετικά συνεχίστε στην ερώτηση 13. Το να βλέπετε τις ιατρικές σας πληροφορίες στο διαδίκτυο:

	Ναι	Όχι	Δεν είμαι σίγουρος
a. Σας κάνει να πιστεύετε ότι γνωρίζετε περισσότερα πράγματα για την υγεία σας;	0	0	0
b. Σας κάνει να πιστεύετε ότι γνωρίζετε περισσότερα πράγματα σχετικά με την ιατρική περίθαλψη που σας παρέχεται;	0	0	0
c. Σας έχει βοηθήσει στο να βελτιώσετε την κατάσταση της υγείας σας;	0	0	0
d. Σας έχει προτρέψει να κάνετε ερωτήσεις στον γιατρό σας, που δεν τις κάνατε στο παρελθόν;	0	0	0
e. Σας κάνει να αισθάνεστε ότι είστε περισσότερο δεμένοι με τον γιατρό σας;	0	0	0
f. Σας αποτρέπει από το να αλλάξετε γιατρό;	0	0	0
g. Σας έχει βοηθήσει στο να βρείτε κάποια πληροφορία που λείπει ή που είναι λάθος;	0	0	0

13. Πολλές διαφορετικές ομάδες μπορούν να παρέχουν τέτοιου είδους ιστοσελίδες, όπου μπορείτε να διατηρήσετε το ιατρικό σας ιστορικό. Πόσο θα σας ενδιέφερε να χρησιμοποιήσετε μια τέτοια ιστοσελίδα, αν αυτή σας την συνέστηνε:



	Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν είμαι σίγουρος
a. Το νοσοκομείο ή ο γιατρός σας	30	26	6	8	4
b. Το ασφαλιστικό σας ταμείο / ασφαλιστική εταιρία	19	24	15	11	5
c. Μια μη κερδοσκοπική ομάδα	11	18	12	27	6
d. Η κυβέρνηση	11	14	18	26	5
e. Μια εταιρία, όπως η Microsoft ή η Google	6	11	19	33	5
f. Ο εργοδότης σας	8	16	13	32	5

14. Παρακάτω παρουσιάζονται κάποιες σκέψεις σχετικά με τις ιστοσελίδες καταχώρησης και διατήρησης ιατρικών δεδομένων. Πόσο συμφωνείτε ή διαφωνείτε με την κάθε μία;

	Συμφωνώ κάθετα	Συμφωνώ	Διαφωνώ	Διαφωνώ κάθετα	Δεν είμαι σίγουρος
a. Με ανησυχεί το ποιός θα μπορεί να βλέπει τις ιατρικές πληροφορίες μου, εάν αυτές βρίσκονται στο διαδίκτυο.	31	32	7	1	3
b. Με προβληματίζει η ασφάλεια και η ιδιοτικότητα των ιατρικών δεδομένων μου.	37	28	6	1	2
c. Δεν χρειάζομαι μια τέτοια ιστοσελίδα για να διατηρώ τον ιατρικό μου φάκελο.	13	19	30	5	7
d. Δεν μου αρέσει να χρησιμοποιώ τον υπολογιστή και το διαδίκτυο	5	13	24	28	4
e. Θα χρειαζόταν πολύ χρόνο για να συντηρώ τα δεδομένα αυτά	6	19	31	12	6
f. Θα κόστιζε πολλά χρήματα	6	10	29	18	11

15. Παρακάτω παρουσιάζονται κάποια πράγματα που θα μπορούσαν να σας επηρεάσουν στο να εγγραφείτε ή όχι σε μια ιστοσελίδα καταχώρησης και διατήρησης ιατρικών δεδομένων. Θα εγγραφόσασταν σε μια τέτοια ιστοσελίδα, εάν:

	Ναι	Όχι	Δεν είμαι σίγουρος
a. Σας έλεγε ο γιατρός σας ότι είναι ασφαλής;	38	19	17
b. Την χρησιμοποιούσε κάποιο μέλος της οικογένειάς σας ή κάποιος φίλος σας;	30	27	17
c. Αν στην ιστοσελίδα αναφερόταν ότι τα δεδομένα σας είναι ασφαλή;	26	28	20
d. Αν εμπιστευόσασταν την εταιρία/ τον οργανισμό που διατηρεί την ιστοσελίδα;	37	27	10
e. Αν στην ιστοσελίδα αναφερόταν ξεκάθαρα η πολιτική ασφάλειας και προστασίας των προσωπικών σας δεδομένων;	37	21	16
f. Αν σας έλεγαν ότι θα ενημερώνεστε στην περίπτωση που κάποιος άλλος έχει πρόσβαση στις πληροφορίες σας;	31	33	10
g. Αν υπήρχαν σοβαρές ποινές προς τις ιστοσελίδες αυτές, σε περίπτωση που αποκάλυπταν κάποια πληροφορία σας σε τρίτους;	40	17	17



16. Αν ο γιατρός σας διατηρούσε ηλεκτρονικούς φακέλους για τους ασθενείς του και είχε τη δυνατότητα να αποστείλει πληροφορίες σχετικά με την υγεία σας και την περίθαλψη που σας παρέχεται σε ερευνητές, στατιστολόγους, ασφαλιστικές εταιρίες και άλλους (οι πληροφορίες αυτές είναι ανώνυμες και ΔΕΝ περιλαμβάνουν το όνομά σας, την διεύθυνσή σας και τον αριθμό μητρώου σας),

a. Θα διαφωνούσατε με το να δοθούν αυτές οι πληροφορίες;

Ναι	Όχι	Δεν ξέρω
16	53	5

b. Υπάρχουν πληροφορίες για την υγεία σας που δεν θα δίνετε στον γιατρό σας;

Ναι	Όχι	Δεν ξέρω
5	66	3

17. Κάποιοι λένε ότι η χρήση της τεχνολογίας, όπως ο ηλεκτρονικός ιατρικός φάκελος, τα διάφορα εργαλεία και οι ιστοσελίδες καταχώρησης ιατρικών δεδομένων μπορούν να βοηθήσουν τόσο τους ασθενείς όσο και το σύστημα υγείας γενικότερα. Από το 1(λίγο) μέχρι το 5 (πολύ) αξιολογήστε το κατά πόσο πιστεύετε ότι οι παρακάτω είναι λόγοι για να χρησιμοποιηθούν τέτοιες τεχνολογίες.

	1	2	3	4	5
a. Ο ηλεκτρονικός ιατρικός φάκελος ή ακόμα και η επικοινωνία μέσω e-mail μπορούν να βελτιώσουν την σχέση σας με τον γιατρό σας	12	9	18	14	21
b. Ο γιατρός έχει συγκεντρωμένη όλη την πληροφορία για την κατάσταση της υγείας σας. Έτσι ξοδεύει λιγότερο χρόνο στο να σας ρωτάει τα ίδια πράγματα και περισσότερο στο να συζητά μαζί σας το πρόβλημα που αντιμετωπίζετε.	3	3	11	24	33
c. Ο γιατρός έχει συγκεντρωμένη όλη την πληροφορία για την κατάσταση της υγείας σας και έτσι σας γνωρίζει καλύτερα.	3	1	12	18	40
d. Η τεχνολογία επιτρέπει σε γιατρούς, νόσοκώμους και φαρμακοποιούς να έχουν την ίδια εικόνα για την κατάσταση της υγείας σας. Έτσι μπορούν να μειωθούν τα ιατρικά λάθη και να έχετε καλύτερη περίθαλψη.	3	3	10	19	39
e. Η τεχνολογία σας βοηθά να έχετε συγκεντρωμένο το ιατρικό σας ιστορικό, το οποίο μπορεί είναι προσβάσιμο από κάθε γιατρό που επισκέπτεστε. Έτσι δεν χρειάζεται να επαναλαμβάνετε το ιστορικό σας σε κάθε γιατρό, ούτε να έχετε μαζί σας τα αποτελέσματα παλαιότερων εξετάσεων από άλλους γιατρούς σε κάθε επίσκεψη.	3	3	10	17	41
f. Με το να έχετε συγκεντρωμένο το ιατρικό σας ιστορικό μπορείτε να έχετε καλύτερο έλεγχο της υγείας σας και να ελέγχετε την περίθαλψη που λαμβάνετε.	2	3	15	16	38
g. Η χρήση της τεχνολογίας μπορεί να μειώσει το κόστος της ιατρικής περίθαλψης.	8	6	13	12	35

18. Πόσο σας προβληματίζει/ανησυχεί η ασφάλεια και η ιδιωτικότητα των προσωπικών ιατρικών δεδομένων σας;

Πολύ	Αρκετά	Λίγο	Καθόλου	Δεν είμαι σίγουρος
38	23	9	4	0

19. Όπως και με τις ηλεκτρονικές τραπεζικές συναλλαγές (online banking) υπάρχουν ανησυχίες σχετικά με την ασφάλεια των δεδομένων. Παρόλα αυτά δεν θα πρέπει οι ανησυχίες αυτές να μας εμποδίζουν στο να ενημερωνόμαστε για το πως μπορεί η τεχνολογία να βελτιώσει την ιατρική περίθαλψη. Πόσο συμφωνείτε με την παραπάνω παρατήρηση;



Συμφωνώ κάθετα 14 Συμφωνώ 50 Διαφωνώ 3 Διαφωνώ κάθετα 0 Δεν είμαι σίγουρος 7

20. Τώρα που έχετε δει το πως μπορεί να χρησιμοποιηθεί ο ηλεκτρονικός φάκελος ασθενούς και οι ιστοσελίδες καταχώρησής του, πόσο θα σας ενδιέφερε να χρησιμοποιήσετε μια τέτοια ιστοσελίδα;

Πολύ 14 Αρκετά 31 Λίγο 14 Καθόλου 10 Δεν είμαι σίγουρος 5

#### ΓΕΝΙΚΑ ΣΤΟΙΧΕΙΑ

21.Φύλο: Άνδρας 35 Γυναίκα 39

22.Ηλικία: 18-24 7 25-34 21 35-44 18 45-54 15 55-64 13 65+ 0

23.Εκπαίδευση: Πρωτοβάθμια 3 Δευτεροβάθμια 13 Τριτοβάθμια 58

24.Πόσο συχνά κάνετε τα παρακάτω:

	Σχεδόν κάθε μέρα	Μια φορά την εβδομάδα	Μια φορά το μήνα	Λιγότερο συχνά	Καθόλου
a. Αγορές από το διαδίκτυο	2	4	11	21	36
b. Παρακολούθηση των τραπεζικών σας λογαριασμών στο διαδίκτυο	7	9	4	4	50
c. Χρήση e-mail	41	9	1	4	19
d. Χρήση chat	23	11	2	3	35
e. Περιήγηση και ενημέρωση μέσω του διαδικτύου	46	6	6	4	12
f. Παρακολούθηση βίντεο στο διαδίκτυο	33	8	5	9	19

25.Έχετε δημιουργήσει ποτέ προσωπικό προφίλ σε κάποια διαδικτυακή κοινότητα, όπως το Face book;

Ναι 37 Όχι 36 Δεν είμαι σίγουρος 1

26.Χρησιμοποιείτε εφαρμογές στο κινητό σας τηλέφωνο που συνδέονται στο διαδίκτυο;

Ναι 25 Όχι 45 Δεν είμαι σίγουρος 4



## Παράρτημα Β Παράδειγμα χρήσης του συστήματος

Στο Παράρτημα αυτό παρουσιάζεται ένα παράδειγμα χρήσης του συστήματος που αναπτύχθηκε. Η παρουσίαση χωρίζεται σε τρία τμήματα, ένα για κάθε εφαρμογή του συστήματος.

### Β.1 Εφαρμογή γραμματείας

Ο χρήστης εισάγει τα στοιχεία του στην φόρμα σύνδεσης που εμφανίζεται:

Clinic Management Information System

LOGIN

User Name

Password

Cancel Login

Το σύστημα ελέγχει τα δεδομένα του χρήστη και τον ανακατευθύνει στην αρχική οθόνη της εφαρμογής.

Medical Clinic Management System

Appointments Doctors Patients

Add Patient Edit Patient Patients Refresh Exit

Doctors	Appointments
Καλογερόπουλος Παναγιώτης Τρίτη, 8 Νοεμβρίου	Μονεμάς - First Appointment Εισαγωγή - Urgent
Παπαδόπουλος Νικόλαος Τρίτη, 8 Νοεμβρίου	Σιδανίου - First
Μαρκόπουλος Αθανάσιος Τρίτη, 8 Νοεμβρίου	Γιάννου - Urgent
Αθανασίου Μάριος Τρίτη, 8 Νοεμβρίου	Φανιάς - Urgent
Παλασιόγλου Κωνσταντίνος Τρίτη, 8 Νοεμβρίου	Νέου - Appointment
Γεωργίου Αθανάσιος Τρίτη, 8 Νοεμβρίου	Φανιάς - First Appointment

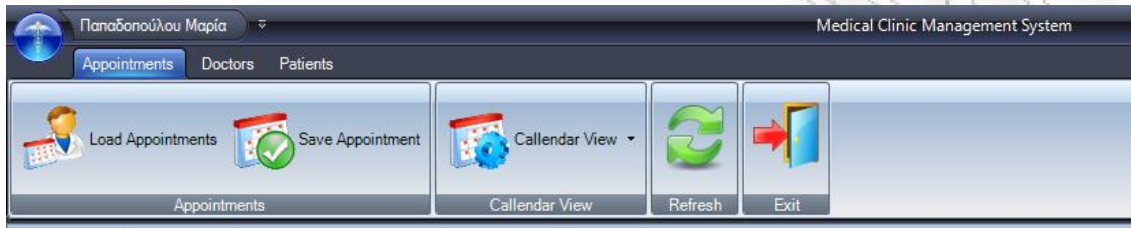
Resources

- Αθανασίου Μάριος
- Γεωργίου Αθανάσιος
- Καλογερόπουλος Παναγιώτης
- Μαρκόπουλος Αθανάσιος
- Παλασιόγλου Κωνσταντίνος
- Παπαδόπουλος Νικόλαος

Η γραμματεία από αυτή την εφαρμογή μπορεί να διαχειριστεί:

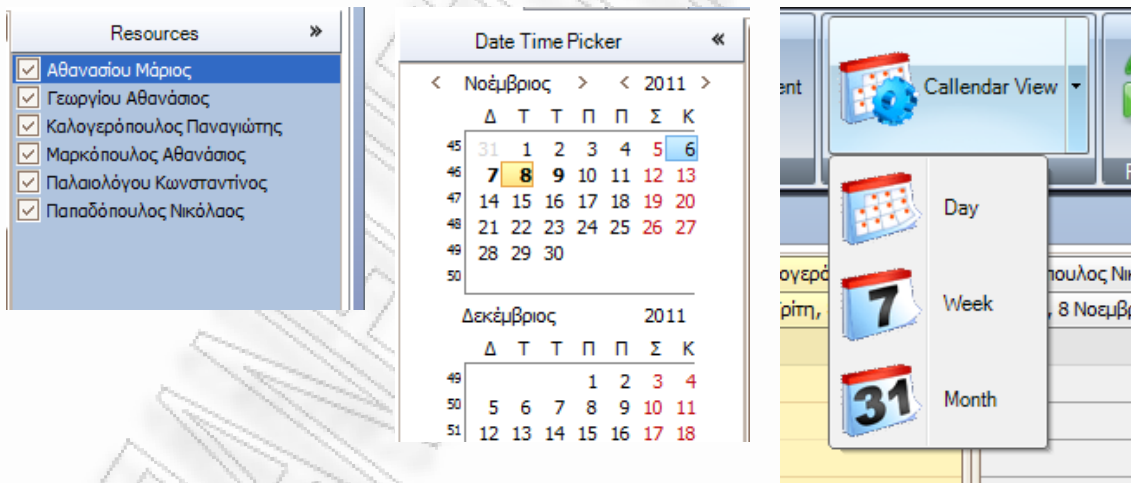
- τα ραντεβού
- τους γιατρούς τις κλινικής και
- τους ασθενείς της κλινικής

Για κάθε ένα από τα παραπάνω υπάρχει ξεχωριστή καρτέλα στο menu της εφαρμογής.



### Καρτέλα «Appointments»

Η γραμματεία έχει την δυνατότητα να δει τα ραντεβού όλων των ιατρών ή αυτών που θα επιλέξει από το menu Resources. Επίσης μπορεί να επιλέξει την ημερομηνία για την οποία θέλει να δει τα ραντεβού ή να επιλέξει αν θα τα βλέπει σε επίπεδο ημέρας, εβδομάδας ή μήνα.



Πατώντας στην στήλη ενός ιατρού και σε μια συγκεκριμένα ώρα από το ημερολόγιο, μπορεί να καταχωρήσει ραντεβού για τον συγκεκριμένο γιατρό.

Στην φόρμα που εμφανίζεται μπορεί να:

- Επιλέξει ασθενή
- Να ορίσει την ώρα έναρξης και λήξης του ραντεβού

- Να ορίσει το Θέμα του ραντεβού και να βάλει σχόλια.
- Να ορίσει αν το ραντεβού γίνεται στα πλαίσια κάποιου σχεδίου θεραπείας ή αν είναι ανεξάρτητο αυτού.

MyAppointmentEditForm

Subject

Start: 17/10/2011 12:00:00 μμ

End: 17/10/2011 12:30:00 μμ

Patient: [Select Patient]

According to Treatment Plan  
 Urgent

Doctor: Παπαδόπουλος Νικόλαος

Status: Busy

Comments

Cancel Save

Για να επιλέξει ασθενή πατά το κουμπί «Select patient». Μπορεί να αναζητήσει ασθενή με βάση το όνομα, το επώνυμο, το ΑΜΚΑ και το ΑΦΜ του.

SelectPatientForm

Search Patient

Patient

Last Name: Μανιφόβας AMKA: [ ]  
First Name: [ ] AFM: [ ]

	First Name	Last Name	Father's Name	AFM	AMKA	Registration Number
▶	Στάθης	Μανιφόβας	Χρήστος	6385114225	1254896354	123456

Plan Comments	Doctor
▶ Αποκατάσταση λει...	Καλογερόπουλος Παναγι.

Step Order	Step Comments
▶ 1	Αρχική εξέταση/Ενημέρωση και ...
2	Αφαίρεση της ακίνητης αποκατ...
3	Αποτρύγωση και ριζική απόξεση ...
4	Χειρουργική αποκάλυψη κλινικής...
5	Ενδοδοντική θεραπεία στα # 43...
6	Επανελέγχος-Αξιολόγηση.

Cancel Select Patient

Αφού συμπληρώσει και τα υπόλοιπα στοιχεία της φόρμας, σώζει το ραντεβού.

MyAppointmentEditForm

Subject: Μανιφόβας - New App

Start: 8/11/2011 2:30:00 μμ

End: 8/11/2011 3:30:00 μμ

Patient: Μανιφόβας Στένης

According to Treatment Plan

Urgent

Doctor: Παλαολόγου Κωνσταντίνος

Status: Busy

Comments: Check up

Cancel Save

Το νέο ραντεβού εμφανίζεται στο ημερολόγιο του γιατρού.

Medical Clinic Management System

Appointments Doctors Patients

Load Appointments Save Appointment Calendar View Refresh Exit

Appointments Doctors Patients

Παλαολόγου Κωνσταντίνος  
Τμήμ. 8 Ημερ.Βίου

Resources

- Αθανάσιου Μίρας
- Γεωργίου Αθανάσιος
- Κολογρούτσου Παναγιώτης
- Μαρκέπουλας Αθανάσιος
- Παλαολόγου Κωνσταντίνος
- Παναγιώτου Κωνσταντίνος

4:00

5:00

6:00

7:00

8:00

9:00

10:00

11:00

12:00

1:00

2:00

3:00

4:00

5:00

6:00

7:00

8:00

Μανιφόβας - New App  
Check up

Το ραντεβού αυτό μπορεί να τροποποιηθεί κάνοντας διπλό κλικ πάνω του ή να διαγραφεί.

### Καρτέλα «Doctors»

Η γραμματεία έχει τη δυνατότητα να προσθέσει γιατρούς στην κλινική, να τροποποιήσει τα στοιχεία τους και να τους αποσυνδέσει από την κλινική. Αυτό το κάνει από την καρτέλα των γιατρών.



Για να προσθέσει έναν γιατρό πατάει το κουμπί «Add Doctor». Στη φόρμα που εμφανίζεται συμπληρώνει τα στοιχεία του γιατρού:

Για να τροποποιήσει τα στοιχεία ενός ιατρού, πρέπει πρώτα να τον αναζητήσει και να τον επιλέξει. Η γραμματεία μπορεί να αναζητήσει έναν γιατρό με βάση το όνομά του, το επώνυμό του, την ειδικότητα ή το ΑΦΜ του.

Αφού συμπληρώσει ένα ή περισσότερα από αυτά τα στοιχεία και πατήσει το κουμπί «Search», το σύστημα επιστρέφει όλους τους ιατρούς που πληρούν αυτά τα κριτήρια αναζήτησης:

Speciality	First Name	Last Name	Father's Name	ID Number	ID Type	AFM	Date Of Birth	Phone 1	Phone 1 Type	Phone 2	Phone 2 Type
ΧΕΡΟΥΡΓΙΚΗ ΣΤ.	Παναγιώτης	Καλογερόπουλος	Νικόλαος	IT35672	ΑΔΤ	18947627	1/1/1975	2101234567	Ονείας	6921345678	Κινητό
ΧΕΡΟΥΡΓΙΚΗ ΣΤ...	Νικόλαος	Παπαδόπουλος	Γιώργος	AI58634	ΑΔΤ	76486684	12/6/1975	2722109847	Ονείας	6973847562	Κινητό

Για τον επιλεγμένο ιατρό, η γραμματεία μπορεί να δει στην οθόνη και τα ραντεβού του.

Medical Clinic Management System

Appointments Doctors Patients

Search for Doctor

Doctor

Last Name:  Speciality: ΧΕΡΟΥΡΓΙΚΗ ΣΤΟΜΑΤΟΣ

First Name:  AFM:

Search Results

Speciality	First Name	Last Name	Father's Name	ID Number	ID Type	AFM	Date Of Birth	Phone 1	Phone 1 Type	Phone 2	Phone 2 Type
ΧΕΡΟΥΡΓΙΚΗ ΣΤ.	Παναγιώτης	Καλογερόπουλος	Νικόλαος	IT35672	ΑΔΤ	18947627	1/1/1975	2101234567	Ονείας	6921345678	Κινητό
ΧΕΡΟΥΡΓΙΚΗ ΣΤ...	Νικόλαος	Παπαδόπουλος	Γιώργος	AI58634	ΑΔΤ	76486684	12/6/1975	2722109847	Ονείας	6973847562	Κινητό

Date picker: 8 Νοεμβρίου 2011

Τρίτη, 8 Νοεμβρίου

9:00

10:00

11:00

12:00

1:00

2:00

3:00

4:00

5:00

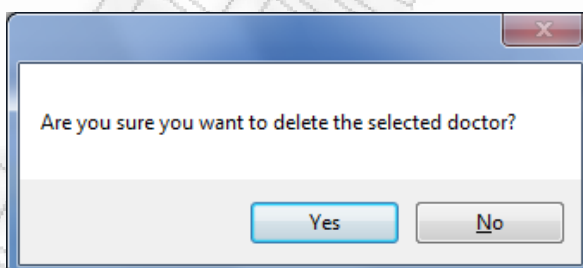
6:00

7:00

Μοναδικός - First Appointment  
General Check up

Πατώντας το κουμπί «Edit Doctor», εμφανίζεται η φόρμα με τα στοιχεία του γιατρού. Η γραμματεία μπορεί να τα τροποποιήσει και να αποθηκεύσει τα νέα στοιχεία.

Πατώντας το κουμπί «Delete Doctor» η γραμματεία μπορεί να αποσυνδέσει τον γιατρό από την κλινική. Το σύστημα εμφανίζει μήνυμα επιβεβαίωσης και στη συνέχεια προχωρά στην διαγραφή του γιατρού.



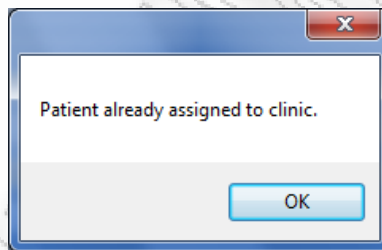
### Καρτέλα «Patients»

Η γραμματεία έχει τη δυνατότητα να προσθέσει ασθενείς στην κλινική και να τροποποιήσει τα στοιχεία τους. Αυτό το κάνει από την καρτέλα των ασθενών.



Για να προσθέσει έναν ασθενή πατάει το κουμπί «Add Patient». Στη φόρμα που εμφανίζεται συμπληρώνει τα στοιχεία του ασθενούς:

Αρχικά η γραμματέας πρέπει να συμπληρώσει το ΑΜΚΑ του ασθενούς και να εκτελέσει αναζήτηση. Αν ο ασθενής υπάρχει στην βάση του συστήματος, τότε εμφανίζεται ανάλογο μήνυμα.

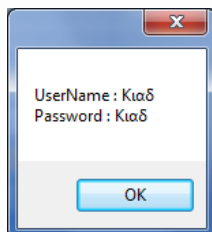


Διαφορετικά η γραμματέας πρέπει να συμπληρώσει τα στοιχεία του ασθενούς.





Όταν ολοκληρωθεί η διαδικασία της αποθήκευσης, το σύστημα επιστρέφει στην γραμματεία το username και το password του ασθενούς. Τα στοιχεία αυτά η γραμματεία τα δίνει στον ασθενή, για να μπορεί να δει τον ηλεκτρονικό ιατρικό του φάκελο.



Για να τροποποιήσει τα στοιχεία ενός ασθενούς, πρέπει πρώτα να τον αναζητήσει και να τον επιλέξει. Η γραμματεία μπορεί να αναζητήσει έναν ασθενή με βάση το όνομά του, το επώνυμό του, το ΑΜΚΑ ή το ΑΦΜ του.

Αφού συμπληρώσει ένα ή περισσότερα από αυτά τα στοιχεία και πατήσει το κουμπί «Search», το σύστημα επιστρέφει όλους τους ασθενείς που πληρούν αυτά τα κριτήρια αναζήτησης:

First Name	Last Name	Father's Name	AFM	ID Number	ID Type	AMKA	Registration Number	Date Of Birth	Place Of Birth	Phone 1	Phone 1 Type
Χριστίνα	Κιαδάρη	Κωνσταντίνος	1236547512	ΑΔΤ	ΚΤ 12568	0305200015		1/1/1900		2721065325	Οικίας

Πατώντας το κουμπί «Edit Patient», εμφανίζεται η φόρμα με τα στοιχεία του ασθενούς. Η γραμματεία μπορεί να τα τροποποιήσει και να αποθηκεύσει τα νέα στοιχεία.

Για τον επιλεγμένο ασθενή η γραμματεία μπορεί να δει επίσης τα ραντεβού και τα οικονομικά στοιχεία του.

### Ραντεβού ασθενούς

First Name	Last Name	Father's Name	AFM	ID Number	ID Type	AMKA	Registration Number	Date Of Birth	Place Of Birth	Phone 1	Phone 1 Type
Στέφανος	Μανιφόρος	Χρήστος	6985114225	ΑΙ 25418	ΑΔΤ	1254896354	123456	12/9/1976	Αθήνα	21045954716	Οικίας

Doctor	App. Start	App. End	Subject	Comments	Completed	Plan Comments	Step Comments	Step Order	Step Price
Καλογερούλου Πανα...	8/11/2011 10:30 πμ	8/11/2011 11:30 πμ	Μανιφόρος - First A...	General Check up	<input type="checkbox"/>				
Παλαιολόγου Κωνσταν...	8/11/2011 2:30 μμ	8/11/2011 3:30 μμ	Μανιφόρος - New A...	Check up	<input type="checkbox"/>				
Καλογερούλου Πανα...	23/11/2011 10:30 πμ	23/11/2011 11:00 πμ	Μανιφόρος - Step 1	Μανιφόρος - Step 1	<input type="checkbox"/>	Ανοκατάσταση λει...	Αρχική εξέταση Επν...	1	50.0000

Η γραμματεία μπορεί να επιλέξει ένα ραντεβού και να το χαρακτηρίσει «Ολοκληρωμένο», μετά την ολοκλήρωση του ραντεβού. Αυτό θα έχει ως αποτέλεσμα να καταχωρηθούν νέα οικονομικά στοιχεία για τον ασθενή (το ποσό που πρέπει να πληρώσει για το ραντεβού που μόλις ολοκληρώθηκε).

### Οικονομικά στοιχεία ασθενούς

Στην καρτέλα αυτή η γραμματεία βλέπει τα ποσά που πρέπει να πληρώσει ο ασθενής και τις πληρωμές που έχει κάνει.

First Name	Last Name	Father's Name	AFM	ID Number	ID Type	AMKA	Registration Number	Date Of Birth	Place Of Birth	Phone 1	Phone 1 Type
Στάθης	Μανιράβας	Χρήστος	6985114225	Ai 25418	ΑΔΤ	1254896354	123456	12/9/1976	Αθήνα	21045854716	Οικίας

Amount	Date	Subject	App. Start	App. End	DoctorName	Amount	Date	Comments
50,00	6/11/2011	Μανιράβας - First Ap...	8/11/2011 10:30 πμ	8/11/2011 11:30 πμ	Καλογερόπουλος Πα...	50,00	6/11/2011	Appointment: 8/11/2011
50,00	6/11/2011	Μανιράβας - New App	9/11/2011 2:30 μμ	9/11/2011 3:30 μμ	Παλαιολόγου Κωνστ...			

Τα χρέη του ασθενούς καταχωρούνται αυτόματα με την ολοκλήρωση ενός ραντεβού. Τις πληρωμές τις καταχωρεί η γραμματεία πατώντας το κουμπί «Add». Στη φόρμα που εμφανίζεται συμπληρώνει το ποσό της πληρωμής και για ποιο λόγο γίνεται η πληρωμή αυτή.

Payments

Amount: 50,00 €

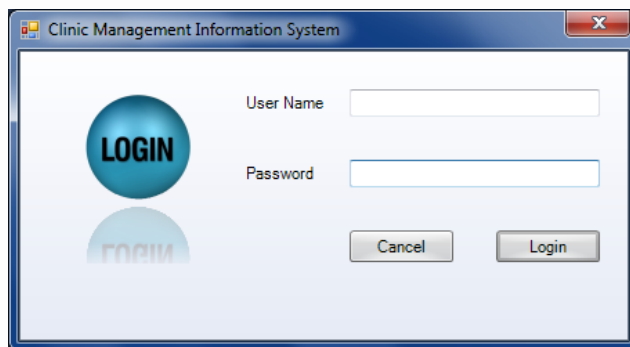
Comments: Appointment: 8/11/2011

Cancel Save

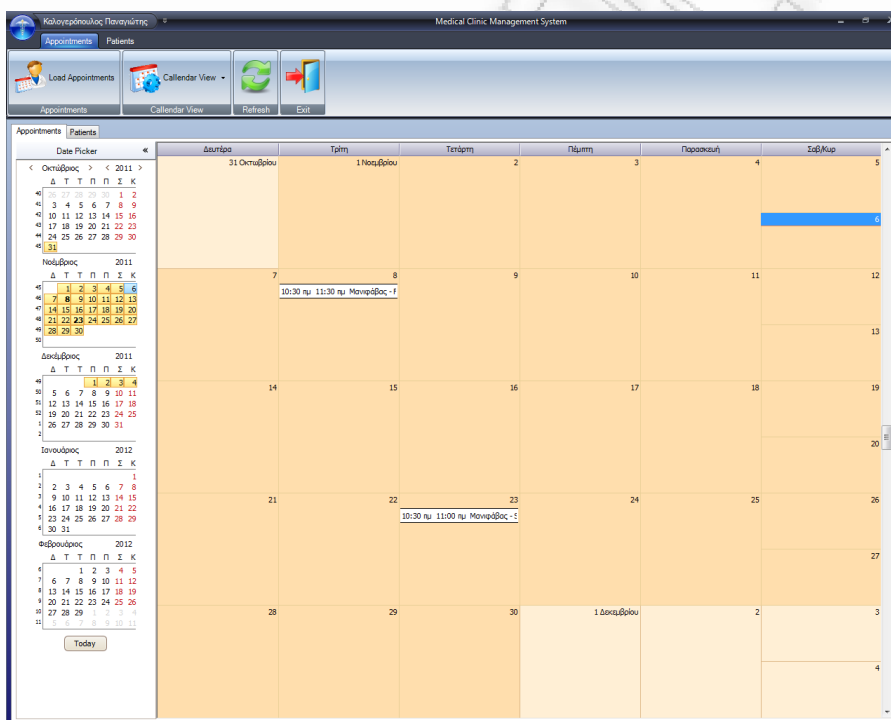
Οι πληρωμές μπορούν να τροποποιηθούν ή και να διαγραφούν.

### B.2 Εφαρμογή γιατρού

Ο χρήστης εισάγει τα στοιχεία του στην φόρμα σύνδεσης που εμφανίζεται:



Το σύστημα ελέγχει τα δεδομένα του χρήστη και τον ανακατευθύνει στην αρχική οθόνη της εφαρμογής.



Ο γιατρός μπορεί από αυτή την εφαρμογή:

- Να δει τα ραντεβού του
- Να διαχειριστεί τους ασθενείς του

Για κάθε ένα από τα παραπάνω υπάρχει ξεχωριστή καρτέλα στο menu της εφαρμογής.



### Καρτέλα «Appointments»

Από την καρτέλα αυτή ο γιατρός μπορεί να δει τα ραντεβού του. Δεν μπορεί να τροποποιήσει αυτά τα ραντεβού. Μπορεί να επιλέξει την ημέρα για την οποία θέλει να προβάλλει τα ραντεβού του και να επιλέξει αν θα τα βλέπει σε επίπεδο ημέρας, εβδομάδας ή μήνα.

The screenshot displays the 'Appointments' interface. On the left, there is a 'Date Picker' showing a calendar for October 2011. The main area is a calendar grid with columns for days of the week (Δευτέρα, Τρίτη, Τετάρτη, Πέμπτη, Παρασκευή, Σαβ/Κυρ) and rows for dates. A specific appointment is highlighted on October 6th at 10:30 pm. The interface also includes a 'Patients' tab and a 'Today' button.

### Καρτέλα «Patients»

Από την καρτέλα αυτή ο γιατρός μπορεί να διαχειριστεί τους ασθενείς του. Για να διαχειριστεί έναν ασθενή ο γιατρός, θα πρέπει πρώτα να τον αναζητήσει. Η αναζήτηση μπορεί να γίνει με βάση το όνομά του, το επώνυμό του, το ΑΜΚΑ ή το ΑΦΜ του.

Αφού συμπληρώσει ένα ή περισσότερα από αυτά τα στοιχεία και πατήσει το κουμπί «Search», το σύστημα επιστρέφει όλους τους ασθενείς που πληρούν αυτά τα κριτήρια αναζήτησης:

The screenshot shows the 'Patients' search interface. It features a search form with the following fields: Last Name (Μανιράβας), First Name, AMKA, and AFM. A search icon is visible. Below the form, the search results are displayed in a table:

First Name	Last Name	Father's Name	AFM	AMKA	Registration Number
Στάθης	Μανιράβας	Χριστός	6985114225	1254896354	123456

Για τον επιλεγμένο ασθενή ο γιατρός μπορεί:

- Να δει τα ραντεβού που έχει με τον συγκεκριμένο ασθενή
- Να συνταγογραφήσει φάρμακα στον ασθενή
- Να ορίσει τα σχέδια θεραπείας του ασθενούς

### Ραντεβού

Ο γιατρός μπορεί να δει τα ραντεβού που έχει με τον συγκεκριμένο ασθενή. Τα ραντεβού αυτά δεν μπορεί να τα τροποποιήσει.

App. Start	App. End	Subject	Comments	Completed	Plan Comments	Step Comments	Step Order	Step Price
8/11/2011 10:30 πμ	8/11/2011 11:30 πμ	Μονιφόρος - First Appo.	General Check up					
23/11/2011 10:30 πμ	23/11/2011 11:00 πμ	Μονιφόρος - Step 1	Μονιφόρος - Step 1		Αποκατάσταση λειτου...	Αρχική εξέταση/Ενημέρ...	1	50,0000

### Φαρμακευτική Αγωγή

Ο γιατρός μπορεί να δει την τρέχουσα φαρμακευτική αγωγή του ασθενούς και να συνταγογραφήσει νέα φάρμακα.

Medication	Pharmaceutical Form	Substance Constration	Start Medication	Finish Medication	Duration(Days)	Doctor	Comment
Depon	Δισκία αναβρόζοντα	500 mg	8/11/2011 10:24 μμ	15/11/2011 10:24 μμ	7	Καλογερόπουλος Παναγιώ	2 την ημέρα

Για να συνταγογραφήσει ένα νέο φάρμακο, πατάει το κουμπί «Add». Στη φόρμα που εμφανίζεται ο γιατρός πρέπει να επιλέξει την κατηγορία του φαρμάκου, το φάρμακο, την ημέρα έναρξης της χρήσης του φαρμάκου και την διάρκεια που θα παίρνει ο ασθενής το φάρμακο. Επίσης εισάγει σχόλια για αυτή την φαρμακευτική αγωγή.

**Patient Prescription**

Medication Category: Αναλγητικά - Αντιπυρετικά

Medication: Depon - Δισκία αναβρόζοντα - 500 mg

Start Medication: Τρίτη, 8 Νοεμβρίου 2011

Duration (days): 7

Comments: 2 την ημέρα

Buttons: Cancel, Save

Ο γιατρός μπορεί επίσης να τροποποιήσει την φαρμακευτική αγωγή, πατώντας το κουμπί «Edit».

## Σχέδια Θεραπείας

Ο γιατρός μπορεί να δει τα σχέδια θεραπείας που έχει ο ασθενής και να προσθέσει νέα σχέδια θεραπείας.

Complaint Of	Clinical Examination	Plan Comments	Created	Updated	Plan Completed
Για να φτιάξω τα δόντια μου και να...	Κακότεχνες προσθετικές αποκατα...	Αποκατάσταση λειτουργίας και αισ...	6/11/2011 1:55 μμ	6/11/2011 3:38 μμ	

Step Order	Step Comments	Appointment	Completed
1	Αρχική εξέταση/Ενημέρωση...	23/11/2011 10:30 ημ.	<input type="checkbox"/>
2	Αφαίρεση της ακίνητης αποκατ...		<input type="checkbox"/>
3	Αποτρύγωση και ριζική απόξεση ...		<input type="checkbox"/>
4	Χειρουργική αποκάλυψη κλινικής ...		<input type="checkbox"/>
5	Ενδοδοντική θεραπεία στα # 43...		<input type="checkbox"/>
6	Επανελέγχος-Αξιολόγηση.		<input type="checkbox"/>

Med. Function	Function Comments	Additional Comments	Duration(Days)	Price
Ενημέρωση ασθεν...		Ενημέρωση ασθεν...	00:15:00	10.0000
Ακτινογραφίες		Ακτινογραφίες	00:30:00	40.0000

Για να προσθέσει ένα σχέδιο θεραπείας, ο γιατρός πατάει το κουμπί «Add». Στην φόρμα που εμφανίζεται ο γιατρός εισάγει τα βασικά σχόλια για το πλάνο, τα βήματα του πλάνου και το ποιες ιατρικές ενέργειες θα εκτελούνται σε κάθε βήμα. Ο ιατρός μπορεί να συμπληρώσει σταδιακά τα στοιχεία αυτά, αποθηκεύοντας το πλάνο και στη συνέχεια τροποποιώντας αυτό για να προσθέσει επιπλέον βήματα και ιατρικές ενέργειες.

Complaint Of: Για να φτιάξω τα δόντια μου και να μπορώ να χαμογελάω καλύτερα

Clinical Examination: Κακότεχνες προσθετικές αποκαταστάσεις, Εναποθέσεις πλάκας και τρυγίας, Ελλειμματική φατνιακή ακρολοφία (ύψος) στις οπίσθιες περιοχές της κάτω γνάθου

Plan Comments: Αποκατάσταση λειτουργίας και αισθητικής

Plan Steps

Step Order	Step Comments	Appointment	Completed
1	Αρχική εξέταση/Ενημέρωση και ε...	23/11/2011 10:30 ημ	<input type="checkbox"/>
2	Αφαίρεση της ακίνητης αποκατ...		<input type="checkbox"/>
3	Αποτρύγωση και ριζική απόξεση ...		<input type="checkbox"/>
4	Χειρουργική αποκάλυψη κλινικής ...		<input type="checkbox"/>
5	Ενδοδοντική θεραπεία στα # 43...		<input type="checkbox"/>
6	Επανελέγχος-Αξιολόγηση.		<input type="checkbox"/>

Step Functions

Med. Function	Function Comments	Additional Comments	Duration(Days)	Price
Ενημέρωση ασθενούς		Ενημέρωση ασθενούς	00:15:00	10.0000
Ακτινογραφίες		Ακτινογραφίες	00:30:00	40.0000

Buttons: Save, Cancel

Για να προσθέσει ένα βήμα στο πλάνο πατάει το κουμπί «Add». Στη φόρμα που εμφανίζεται, ο γιατρός συμπληρώνει τα σχόλια για το τι θα εκτελεστεί στο συγκεκριμένο βήμα.

Step Comments

Αφαίρεση της ακίνητης αποκατάστασης στα #16-(15)-(14)-13-(12)-11-21-22 , #44-43, #33-34 και κατασκευή των αντίστοιχων μεταβατικών αποκαταστάσεων

Save Cancel

Ο γιατρός μπορεί επίσης να τροποποιήσει και να διαγράψει βήματα στα πλαίσια ενός πλάνου, πατώντας τα κουμπιά «Edit» και «Delete» αντίστοιχα. Για να δει τα στοιχεία που έχει συμπληρώσει σε ένα βήμα, χωρίς να τα τροποποιήσει, μπορεί να πατήσει τα κουμπί «Details».

Για να προσθέσει μια ιατρική ενέργεια στο επιλεγμένο βήμα πατάει το κουμπί «Add». Στη φόρμα που εμφανίζεται, ο γιατρός συμπληρώνει το είδος της ενέργειας, την τιμή της, το χρόνο που θα χρειαστεί για την εκτέλεσή της και σχετικά σχόλια.

Function Αφαίρεση ακίνητης αποκατάστασης

Price 100,00 Duration 01:00

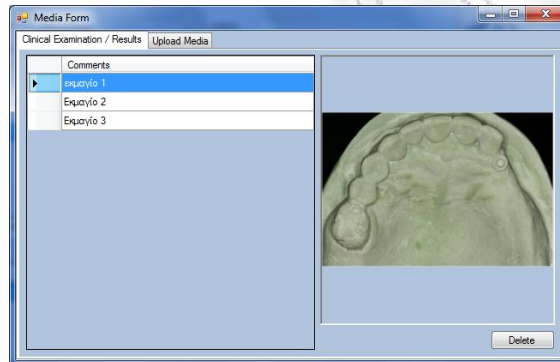
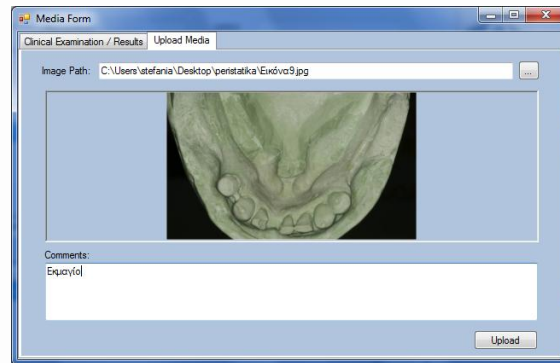
Comments

στα #16-(15)-(14)-13-(12)-11-21-22 , #44-43, #33-34

Save Cancel

Ο γιατρός μπορεί επίσης να τροποποιήσει και να διαγράψει ενέργειες στα πλαίσια ενός επιλεγμένου βήματος, πατώντας τα κουμπιά «Edit» και «Delete» αντίστοιχα. Για να δει τα στοιχεία που έχει συμπληρώσει για μια ενέργεια, χωρίς να τα τροποποιήσει, μπορεί να πατήσει τα κουμπί «Details».

Ο γιατρός έχει επίσης τη δυνατότητα να προσθέσει στοιχεία πολυμέσων (π.χ. φωτογραφίες) που προέκυψαν από την εκτέλεση μιας ιατρικής ενέργειας. Για να το κάνει αυτό μπορεί να πατήσει το κουμπί «Add/Edit Media» της αρχικής οθόνης. Στη φόρμα που εμφανίζεται ο γιατρός επιλέγει την φωτογραφία που θέλει να προσθέσει και εισάγει ένα σχόλιο για αυτή.



### Ιατρικός Φάκελος

Ο γιατρός μπορεί να δει τον ιατρικό φάκελο του επιλεγμένου ασθενούς, πατώντας το κουμπί «View Medical Record» που βρίσκεται στο menu. Στη φόρμα που εμφανίζεται ο γιατρός μπορεί να δει:

- Τα δημογραφικά στοιχεία του ασθενούς
- Τις αλλεργίες του ασθενούς (γενικές αλλεργίες ή αλλεργίες σε φάρμακα)
- Την τρέχουσα και τις παλαιότερες φαρμακευτικές αγωγές που έχει λάβει ο ασθενής
- Το γενικό ιατρικό ιστορικό του ασθενούς
- Το γενικό οδοντιατρικό ιστορικό του ασθενούς
- Το οικογενειακό ιατρικό ιστορικό του ασθενούς.

Τα στοιχεία αυτά ο γιατρός μπορεί μόνο να τα δει και χι να τα τροποποιήσει.



The screenshot shows a web application window titled "Patient Medical Record". On the left, there is a vertical navigation menu with tabs: "Demographics", "Allergies", "Medications", "Medical History", "Dental History", and "Family History". The "Demographics" tab is selected. The main area contains a form with the following fields:

Last Name	: Μανιφάβας	ID Type	: ΑΔΤ
First Name	: Στάθης	ID Number	: ΑΙ 25418
Father's Name	: Χρήστος	AFM	: 6985114225
Date Of Birth	: 12/9/1976 12:00:00 πμ	Registration Number	: 123456
Place Of Birth	: Αθήνα	AMKA	: 1254896354
Occupation	: Προγραμματιστής/τρια	Marital Status	: Married
Address 1	: Ηρώων Πολυτεχνείου 30	Phone 1	: 21045854716
Address 1 Type	: Οικίας	Phone 1 Type	: Οικίας
City 1	: Αθήνα	Phone 2	: 69725632144
Address 2	:	Phone 2 Type	: Κινητό
Address 2 Type	:	Email	: smanifavas@mail.com
City 2	:		

### B.3 Εφαρμογή ηλεκτρονικού ιατρικού φακέλου

Ο ασθενής εισάγει τα στοιχεία του στην αρχική σελίδα και πατάει το κουμπί «Login»

The screenshot shows the login and sign up page of a medical clinic. At the top, there is a banner with the text "WE ARE PROUD TO PROVIDE QUALITY HEALTH CARE TO PATIENTS" and a quote from "Στέφανος Μανιφάβας". Below the banner, there are two main sections: "Log In" and "Sign Up".

**Log In:**

- User Name:
- Password:
- Remember me next time.
- 

**Sign Up:**

- User Name:
- Password:
- Confirm Password:
- E-mail:
- Security Question:
- Security Answer:
- 

At the bottom, there is a footer with "medical clinic" and "© All Rights Reserved".

Το σύστημα ανακατευθύνει τον ασθενή στην σελίδα με τα στοιχεία του ιατρικού του φακέλου.

Stefania Maniavas

Demographics Medication Appointments Treatment Plans Allergies Dental Record Medical Record Family Record Financial Data

Patient Demographics

First Name:	Στάθης	Last Name:	Μανιφάβας
Father Name:	Χρήστος	Sex:	Male: <input type="checkbox"/> Female: <input checked="" type="checkbox"/>
Date Of Birth:	12/9/1976 12:00:00 ημ	Place Of Birth:	Αθήνα
ID Type:	ΑΔΤ	ID Number:	ΑΙ 25418
AFM:	6985114225	AMKA:	1254896354
Registration Number:	123456	E-Mail:	smanifavas@mail.com
Occupation:	Προγραμματιστής/τρια	Marital Status:	
Address 1:	Ηρώων Πολυτεχνείου 30	Address 2:	
Address 1 Type:	Οικίας	Address 2 Type:	
City 1:	Αθήνα	City 2:	
Phone 1:	21045854716	Phone 2:	69725632144
Phone 1 Type:	Οικίας	Phone 2 Type:	Κινητό

[Update](#) [Cancel](#)

medical clinic © All Rights Reserved

Ο ασθενής μπορεί να δει:

- Τα δημογραφικά του στοιχεία. Τα στοιχεία αυτά μπορεί να τα τροποποιήσει.
- Την τρέχουσα και τις παλαιότερες φαρμακευτικές αγωγές που έχει ακολουθήσει
- Τα ραντεβού του
- Τα σχέδια θεραπείας του
- Τις αλλεργίες του
- Το γενικό ιατρικό ιστορικό του
- Το γενικό οδοντιατρικό ιστορικό του
- Το οικογενειακό ιατρικό ιστορικό του
- Τα οικονομικά στοιχεία του

## Παράρτημα Γ

### Παρουσίαση μιας ολοκληρωμένης επιχειρηματικής διαδικασίας

Στο Παράρτημα αυτό θα παρουσιάσουμε τον κώδικα και την πλήρη διαδικασία υλοποίησης της επιχειρηματικής διαδικασίας «Εισαγωγή στο Σύστημα». Με παρόμοιο τρόπο έχουν υλοποιηθεί όλες οι επιχειρηματικές διαδικασίες του συστήματος.

Κατά την έναρξη εκτέλεσης της εφαρμογής εκτελείται ο παρακάτω κώδικας:

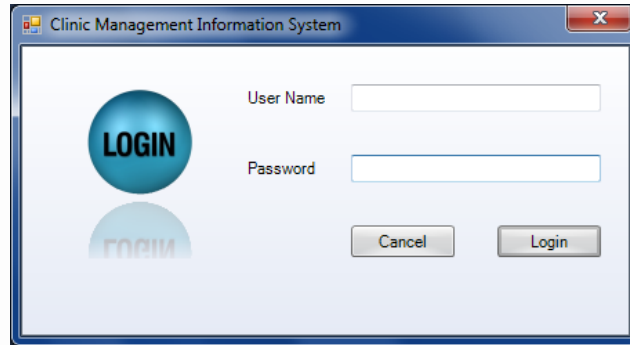
```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    using (LoginForm frmLogin = new LoginForm())
    {
        if (frmLogin.ShowDialog() == DialogResult.OK)
        {
            DataSet sessionData = frmLogin.sessionData;
            frmLogin.Close();

            switch ((int)sessionData.Tables["sessionData"].Rows[0]["RoleID"])
            {
                case 1:
                    Application.Run(new SecretaryForm(sessionData));
                    break;
                case 3:
                    Application.Run(new DoctorForm(sessionData));
                    break;
                default:
                    MessageBox.Show("This user is not permitted to use this
application!");
                    break;
            }
        }
    }
}
```

Ο κώδικας αυτός διαχειρίζεται την εμφάνιση των κατάλληλων οθονών στον χρήστη.

Ο παραπάνω κώδικας εκτελεί τα εξής:

1. Αρχικά εμφανίζει την φόρμα εισαγωγής στο σύστημα (Login Form)
2. Όταν η εργασία στη φόρμα ολοκληρωθούν επιτυχώς, ενημερώνει το DataSet sessionData με τα στοιχεία που επέστρεψε η φόρμα.
3. Ανάλογα με τον ρόλο του χρήστη που εκτέλεσε το Login (η πληροφορία για το ρόλο υπάρχει μέσα στα sessionData), εμφανίζει την οθόνη της γραμματείας ή την οθόνη του ιατρού, διαφορετικά εμφανίζει το μήνυμα λάθους.



Στην φόρμα αυτή ο χρήστης εισάγει το username και το password του και πατάει το κουμπί Login. Με το πάτημα του κουμπιού εκτελείται η παρακάτω ρουτίνα:

```
private void LoginBtn_Click(object sender, System.EventArgs e)
{
    progressBar.Visible = true;
    progressBar.Enabled = true;
    Application.DoEvents();

    using (Login_WebService mySvc = new Login_WebService())
    {
        try
        {
            string error = string.Empty;
            DataSet result =
mySvc.InitializeLoginData(this.UsernameTB.Text.Trim(), this.PasswordTB.Text, out
error);

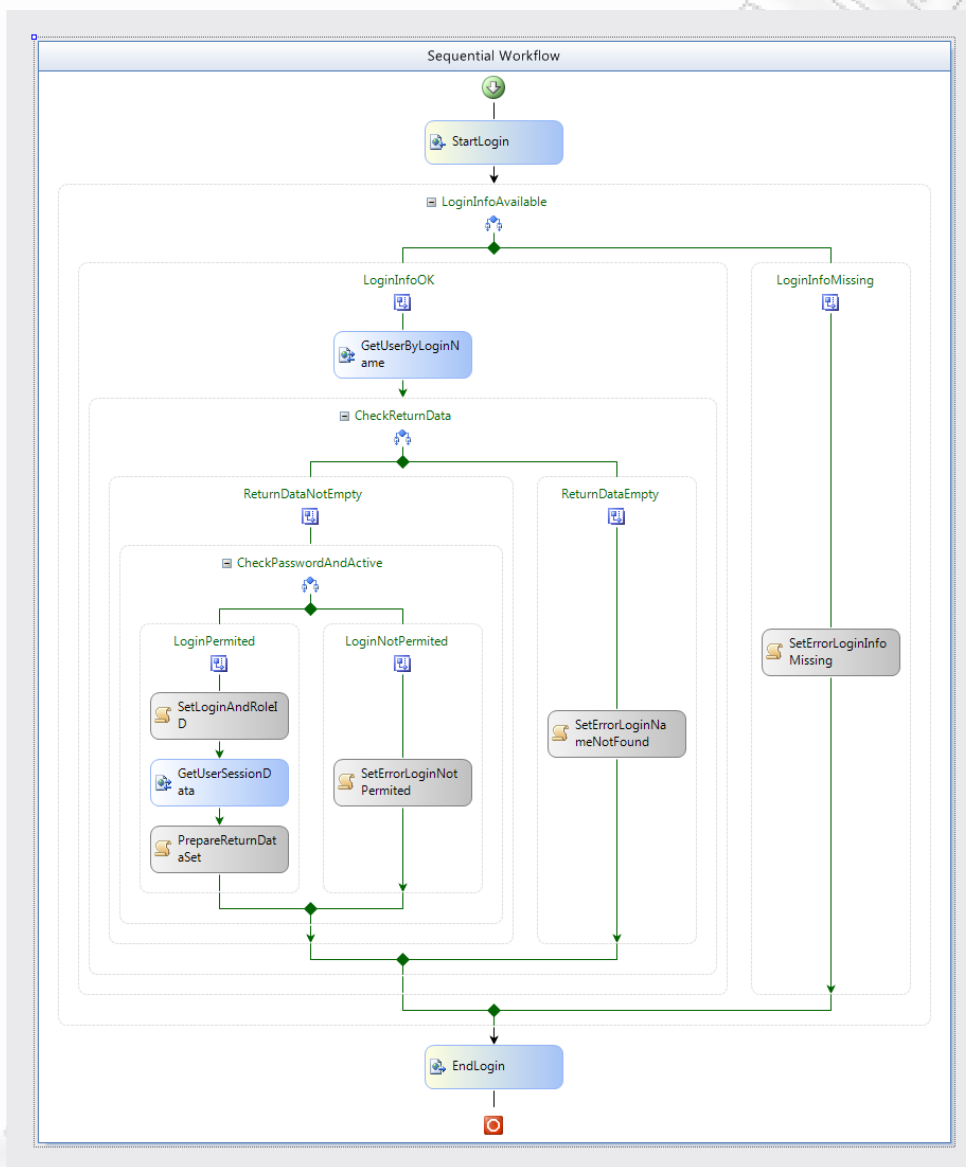
            if (error != null && error != string.Empty)
            {
                MessageBox.Show(error);
                progressBar.Visible = false;
                progressBar.Enabled = false;
            }
            else
            {
                sessionData = result;
                progressBar.Enabled = false;
                this.DialogResult = DialogResult.OK;
            }
        }
        catch (Exception ex)
        {
            progressBar.Enabled = false;
            MessageBox.Show(ex.ToString());
        }
    }
}
```

Ο παραπάνω κώδικας εκτελεί τα εξής:

1. Καλεί το Business Service “Login\_WebService”, με παραμέτρους τα στοιχεία που εισήγαγε ο χρήστης
2. Αν το service επιστρέψει λογικό σφάλμα, τότε το εμφανίζει στην οθόνη, διαφορετικά παίρνει τα session data που έχουν επιστραφεί και κλείνει την φόρμα

3. Αν επιστρέψει μη διαχειρίσιμο σφάλμα(Exception), το εμφανίζει στην οθόνη

Η υλοποίηση της υπηρεσίας “Login\_WebService” έχει γίνει με τη χρήση ενός workflow. Με την κλήση της υπηρεσίας ξεκινά η εκτέλεση του workflow:



Όπως φαίνεται και από το παραπάνω σχήμα, εκτελούνται τα εξής:

1. Ελέγχει αν ο χρήστης έχει συμπληρώσει όλα τα απαραίτητα στοιχεία. Αν ο έλεγχος είναι επιτυχής συνεχίζει στα υπόλοιπα βήματα, διαφορετικά εμφανίζει μήνυμα λάθους.
2. Καλεί την Υπηρεσία Υποστήριξης «UsersWS», προκειμένου να εξαγάγει τα στοιχεία του χρήστη από την βάση, βασιζόμενη στο Όνομα Χρήστη που έχει εισαχθεί. Αν ο χρήστης βρεθεί στην βάση συνεχίζει στα επόμενα βήματα, διαφορετικά επιστρέφει μήνυμα λάθους.

3. Συγκρίνει τον κωδικό χρήστη που έχει εισάγει ο χρήστης στην φόρμα με τον κωδικό που είναι αποθηκευμένος στην βάση. Ελέγχει επίσης αν ο χρήστης είναι ενεργός ή όχι. Αν και οι δύο έλεγχοι είναι επιτυχείς, τότε συνεχίζει στο επόμενο βήμα. Αν κάποιος έλεγχος αποτύχει επιστρέφει μήνυμα λάθους.
4. Καλεί την Υπηρεσία Υποστήριξης «UsersWS» για να πάρει κάποια επιπλέον στοιχεία για τον συγκεκριμένο χρήστη (όπως το ονοματεπώνυμο του) και επιστρέφει τα στοιχεία αυτά.

Ο κώδικας υλοποίησης του workflow παρουσιάζεται παρακάτω:

```
public sealed partial class Login : SequentialWorkflowActivity
{
    // Input params
    public static DependencyProperty LoginNameProperty =
DependencyProperty.Register("LoginName", typeof(System.String), typeof(Login));
    public static DependencyProperty PasswordProperty =
DependencyProperty.Register("Password", typeof(System.String), typeof(Login));

    // Intermediate params - Serialized DataSets
    public static DependencyProperty RetrievedDataProperty =
DependencyProperty.Register("RetrievedData", typeof(System.Data.DataSet),
typeof(Login));
    public static DependencyProperty RetrievedDataSetProperty =
DependencyProperty.Register("RetrievedDataSet", typeof(System.Data.DataSet),
typeof(Login));

    public static DependencyProperty LoginIDProperty =
DependencyProperty.Register("LoginID", typeof(System.Int32), typeof(Login));
    public static DependencyProperty RoleIDProperty =
DependencyProperty.Register("RoleID", typeof(System.Int32), typeof(Login));

    // Output Params
    public static DependencyProperty ErrorProperty =
DependencyProperty.Register("Error", typeof(System.String), typeof(Login));
    public static DependencyProperty ReturnDataSetProperty =
DependencyProperty.Register("ReturnDataSet", typeof(System.Data.DataSet),
typeof(Login));

    [DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
    [BrowsableAttribute(true)]
    [CategoryAttribute("Parameters")]
    public int LoginID
    {
        get
        {
            return ((int)(base.GetValue(Login.LoginIDProperty)));
        }
        set
        {
            base.SetValue(Login.LoginIDProperty, value);
        }
    }

    [DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
```



```
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public int RoleID
{
    get
    {
        return ((int)(base.GetValue(Login.RoleIDProperty)));
    }
    set
    {
        base.SetValue(Login.RoleIDProperty, value);
    }
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public DataSet ReturnDataSet
{
    get
    {
        return ((DataSet)(base.GetValue(Login.ReturnDataSetProperty)));
    }
    set
    {
        base.SetValue(Login.ReturnDataSetProperty, value);
    }
}

public Login()
{
    InitializeComponent();
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public string LoginName
{
    get
    {
        return ((string)(base.GetValue(Login.LoginNameProperty)));
    }
    set
    {
        base.SetValue(Login.LoginNameProperty, value);
    }
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public string Password
{
    get
    {
        return ((string)(base.GetValue(Login.PasswordProperty)));
    }
    set
    {
        base.SetValue(Login.PasswordProperty, value);
    }
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
```

```
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public string Error
{
    get
    {
        return ((string)(base.GetValue(Login.ErrorProperty)));
    }
    set
    {
        base.SetValue(Login.ErrorProperty, value);
    }
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public DataSet RetrievedData
{
    get
    {
        return (DataSet)(base.GetValue(Login.RetrievedDataProperty));
    }
    set
    {
        base.SetValue(Login.RetrievedDataProperty, value);
    }
}

[DesignerSerializationVisibilityAttribute(DesignerSerializationVisibility.Visible)]
[BrowsableAttribute(true)]
[CategoryAttribute("Parameters")]
public DataSet RetrievedDataSet
{
    get
    {
        return (DataSet)(base.GetValue(Login.RetrievedDataSetProperty));
    }
    set
    {
        base.SetValue(Login.RetrievedDataSetProperty, value);
    }
}

private void SetErrorLoginInfoMissing_ExecuteCode(object sender, EventArgs e)
{
    Error = "Login Info missing!";
}

private void SetErrorLoginNameNotFound_ExecuteCode(object sender, EventArgs e)
{
    Error = "Invalid login name!";
}

private void SetErrorLoginNotPermitted_ExecuteCode(object sender, EventArgs e)
{
    Error = "Invalid password or user inactive!";
}

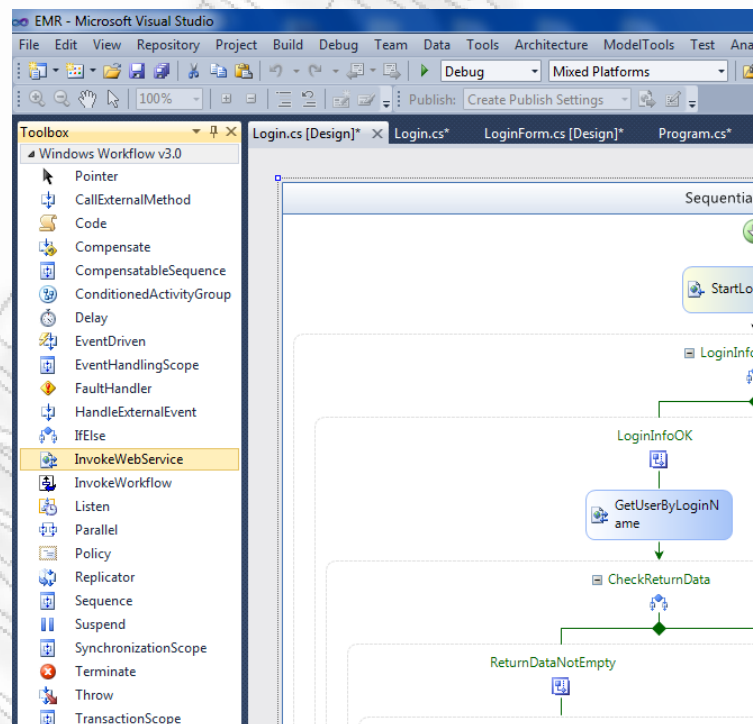
private void PrepareReturnDataSet_ExecuteCode(object sender, EventArgs e)
{
    DataTable dt = new DataTable("sessionData");
    if (this.RetrievedDataSet != null)
    {
```



```
dt = this.RetrievedDataSet.Tables[0].Copy();
ReturnDataSet.Tables.Add(dt);
}
else
{
    Error = "No session data found.";
}
}

private void SetLoginAndRoleID_ExecuteCode(object sender, EventArgs e)
{
    LoginID = (int)this.ReturnDataSet.Tables[0].Rows[0]["LoginID"];
    RoleID = (int)this.ReturnDataSet.Tables[0].Rows[0]["RoleID"];
}
}
```

Όπως φαίνεται από τον κώδικα υλοποίησης, το μόνο που χρειάζεται να ορίσουμε είναι οι παράμετροι που διαχειρίζεται το workflow και ο κώδικας που θα εκτελέσουν τα code activities. Η παραμετροποίηση της κλήσης της υπηρεσίας «UsersWS»(η οποία επαναχρησιμοποιείται μέσα στο ίδιο workflow), γίνεται στον design editor του workflow. Για να καλέσουμε μια υπηρεσία χρησιμοποιούμε το InvokeWebService activity που είναι διαθέσιμο στο toolbox.



Στην συνέχεια ορίζουμε τις ιδιότητες αυτού του activity:

<p style="text-align: center;">Sequential Workflow</p> <pre> graph TD     StartLogin[StartLogin] --&gt; LoginInfoAvailable[LoginInfoAvailable]     </pre>	<p>Properties</p> <p><b>StartLogin</b> System.Workflow.Activities.WebServiceInputActivity</p> <table border="1"> <tr><td>(Name)</td><td>StartLogin</td></tr> <tr><td>Description</td><td></td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>InputReceived</td><td></td></tr> <tr><td>InterfaceType</td><td>EMR.LoginWS.ILogin</td></tr> <tr><td>IsActivating</td><td>True</td></tr> <tr><td>LoginName</td><td>Activity=Login, Path=LoginName</td></tr> <tr><td>MethodName</td><td>InitializeLoginData</td></tr> <tr><td>Password</td><td>Activity=Login, Path=Password</td></tr> <tr><td>Roles</td><td></td></tr> </table>	(Name)	StartLogin	Description		Enabled	True	InputReceived		InterfaceType	EMR.LoginWS.ILogin	IsActivating	True	LoginName	Activity=Login, Path=LoginName	MethodName	InitializeLoginData	Password	Activity=Login, Path=Password	Roles					
(Name)	StartLogin																								
Description																									
Enabled	True																								
InputReceived																									
InterfaceType	EMR.LoginWS.ILogin																								
IsActivating	True																								
LoginName	Activity=Login, Path=LoginName																								
MethodName	InitializeLoginData																								
Password	Activity=Login, Path=Password																								
Roles																									
<pre> graph TD     LoginInfoOK[LoginInfoOK] --&gt; GetUserByLoginName[GetUserByLoginName]     GetUserByLoginName --&gt; CheckReturnData[CheckReturnData]     </pre>	<p>Properties</p> <p><b>GetUserByLoginName</b> System.Workflow.Activities.InvokeWebServiceActivity</p> <table border="1"> <tr><td>(Name)</td><td>GetUserByLoginName</td></tr> <tr><td>(ReturnValue)</td><td>Activity=Login, Path=ReturnDataSet</td></tr> <tr><td>Description</td><td></td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>Invoked</td><td></td></tr> <tr><td>Invoking</td><td></td></tr> <tr><td>MethodName</td><td>GetUserByLoginName</td></tr> <tr><td>ProxyClass</td><td>EMR.BusinessServices.US_UsersWS.UsersWS</td></tr> <tr><td>SessionId</td><td></td></tr> <tr><td>URL</td><td>http://localhost:6741/UsersWS.asmx</td></tr> <tr><td>userLoginName</td><td>Activity=Login, Path=LoginName</td></tr> </table>	(Name)	GetUserByLoginName	(ReturnValue)	Activity=Login, Path=ReturnDataSet	Description		Enabled	True	Invoked		Invoking		MethodName	GetUserByLoginName	ProxyClass	EMR.BusinessServices.US_UsersWS.UsersWS	SessionId		URL	http://localhost:6741/UsersWS.asmx	userLoginName	Activity=Login, Path=LoginName		
(Name)	GetUserByLoginName																								
(ReturnValue)	Activity=Login, Path=ReturnDataSet																								
Description																									
Enabled	True																								
Invoked																									
Invoking																									
MethodName	GetUserByLoginName																								
ProxyClass	EMR.BusinessServices.US_UsersWS.UsersWS																								
SessionId																									
URL	http://localhost:6741/UsersWS.asmx																								
userLoginName	Activity=Login, Path=LoginName																								
<pre> graph TD     GetUserSessionData[GetUserSessionData] --&gt; PrepareReturnData[PrepareReturnData]     </pre>	<p>Properties</p> <p><b>GetUserSessionData</b> System.Workflow.Activities.InvokeWebServiceActivity</p> <table border="1"> <tr><td>(Name)</td><td>GetUserSessionData</td></tr> <tr><td>(ReturnValue)</td><td>Activity=Login, Path=RetrievedDataSet</td></tr> <tr><td>Description</td><td></td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>Invoked</td><td></td></tr> <tr><td>Invoking</td><td></td></tr> <tr><td>loginID</td><td>Activity=Login, Path=LoginID</td></tr> <tr><td>MethodName</td><td>GetUserSessionData</td></tr> <tr><td>ProxyClass</td><td>EMR.BusinessServices.US_UsersWS.UsersWS</td></tr> <tr><td>roleID</td><td>Activity=Login, Path=RoleID</td></tr> <tr><td>SessionId</td><td></td></tr> <tr><td>URL</td><td>http://localhost:6741/UsersWS.asmx</td></tr> </table>	(Name)	GetUserSessionData	(ReturnValue)	Activity=Login, Path=RetrievedDataSet	Description		Enabled	True	Invoked		Invoking		loginID	Activity=Login, Path=LoginID	MethodName	GetUserSessionData	ProxyClass	EMR.BusinessServices.US_UsersWS.UsersWS	roleID	Activity=Login, Path=RoleID	SessionId		URL	http://localhost:6741/UsersWS.asmx
(Name)	GetUserSessionData																								
(ReturnValue)	Activity=Login, Path=RetrievedDataSet																								
Description																									
Enabled	True																								
Invoked																									
Invoking																									
loginID	Activity=Login, Path=LoginID																								
MethodName	GetUserSessionData																								
ProxyClass	EMR.BusinessServices.US_UsersWS.UsersWS																								
roleID	Activity=Login, Path=RoleID																								
SessionId																									
URL	http://localhost:6741/UsersWS.asmx																								

Properties

**EndLogin** System.Workflow.Activities.WebServiceOutputActivity

(Name)	<b>EndLogin</b>
(ReturnValue)	Activity=Login, Path=ReturnDataSet
Description	
Enabled	True
Error	Activity=Login, Path=Error
InputActivityName	<b>StartLogin</b>
SendingOutput	

Έτσι ορίζουμε ποια μέθοδο του service θα καλέσει, τι παραμέτρους θα πάρει ως είσοδο και σε ποια παράμετρο θα καταχωρηθεί η έξοδος που επιστρέφει η μέθοδος που καλέσαμε.

Μέσω του design editor του workflow ορίζονται και τα διάφορα If-Else activities που χρησιμοποιούνται για να καθορίσουν τα διαφορετικά μονοπάτια που υπάρχουν στο workflow:

CheckReturnData

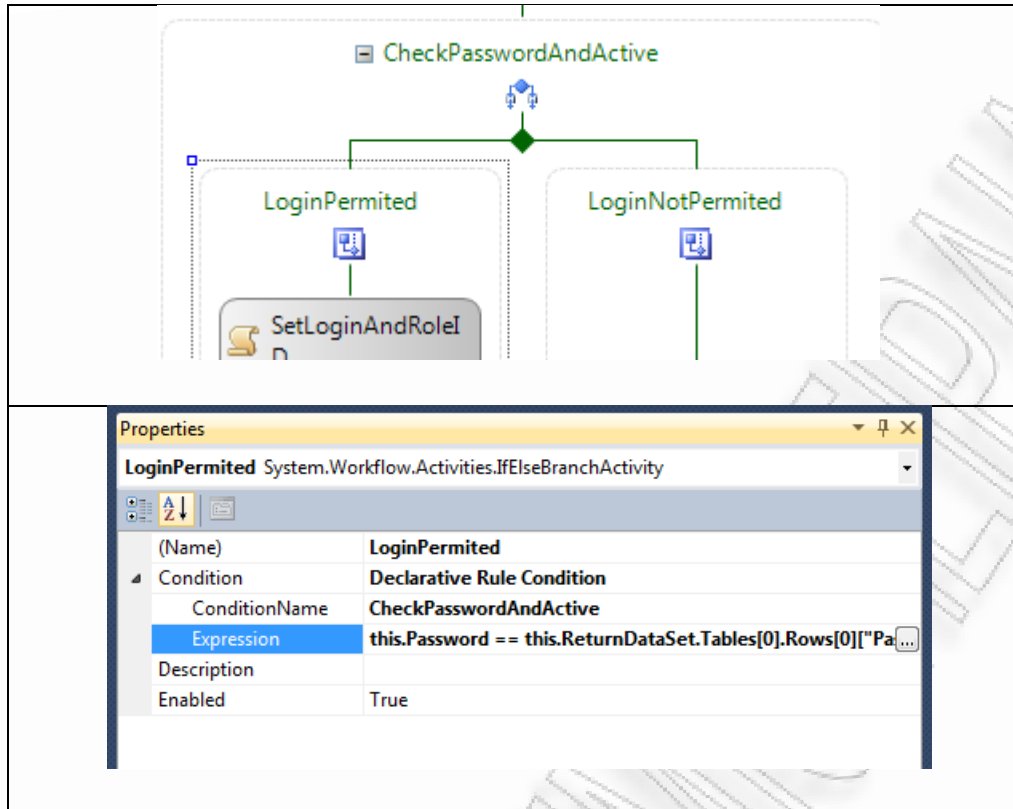
ReturnDataNotEmpty

ReturnDataEmpty

Properties

**ReturnDataNotEmpty** System.Workflow.Activities.IfElseBranchActivity

(Name)	<b>ReturnDataNotEmpty</b>
Condition	Declarative Rule Condition
ConditionName	CheckReturnDataEmpty
Expression	this.ReturnDataSet != null && this.ReturnDataSet.Tables.Cou...
Description	
Enabled	True



Η Υπηρεσία Υποστήριξης «UsersWS» καλείται σε δύο σημεία μέσα στο workflow. Στην πρώτη κλήση της χρησιμοποιείται η μέθοδός της «GetUserByLoginName», η οποία εκτελεί τα εξής:

```
[WebMethod]
public DataSet GetUserByLoginName(string userLoginName)
{
    if (userLoginName != string.Empty)
    {
        try
        {
            DBObject dbo = new DBObject();
            System.Data.SqlClient.SqlParameter[] parameters = new
                System.Data.SqlClient.SqlParameter[]
            {
                new System.Data.SqlClient.SqlParameter("LoginName",
                    userLoginName)
            };
            string sql = "SELECT * FROM Logins WHERE LoginName = @LoginName";
            using (FS_ExecuteQRYSRV srv = new FS_ExecuteQRYSRV())
            {
                return srv.ExecuteQuery_Simple(sql, parameters, "Logins");
            }
        }
        catch (Exception ex)
        {
            System.Diagnostics.Debug.WriteLine("US_UsersWS.GetUserByLoginName: "
                + ex.ToString());
            throw ex;
        }
    }
}
```

```
else
{
    System.Diagnostics.Debug.WriteLine("US_UsersWS.GetUserByLoginName:
        incorrect parameters (userLoginName)");
    return null;
}
}
```

Η μέθοδος αυτή ορίζει τις παραμέτρους που θα χρησιμοποιηθούν από την Βασική Υπηρεσία FS\_ExecuteQRYSRV και συγκεκριμένα την μέθοδο ExecuteQuery\_Simple, η οποία θα κληθεί για να εξάγει τα δεδομένα από τη βάση. Σε περίπτωση σφάλματος, αυτό θα επιστρέφεται στο component που κάλεσε την υπηρεσία.

Στην δεύτερη κλήση της χρησιμοποιείται η μέθοδός της «GetUserSessionData», η οποία εκτελεί τα εξής:

```
[WebMethod]
public DataSet GetUserSessionData(int loginID, int roleID)
{
    if (loginID > 0 && roleID > 0)
    {
        string sqlFields = "SELECT Logins.*, ";
        string sqlTables = "FROM Logins ";
        switch (roleID)
        {
            case 1:
                sqlFields += "Secretaries.FirstName, Secretaries.LastName,
                    Secretaries.ClinicID ";
                sqlTables += "INNER JOIN Secretaries ON Secretaries.LoginID =
                    Logins.LoginID ";
                break;
            case 2:
                sqlFields += "Patients.PatID, Patients.FirstName, Patients.LastName
                    ";
                sqlTables += "INNER JOIN Patients ON Patients.LoginID =
                    Logins.LoginID ";
                break;
            case 3:
                sqlFields += "MedicalDoctors.DocID, MedicalDoctors.FirstName,
                    MedicalDoctors.LastName, MedicalDoctors.SpecialityID ";
                sqlTables += "INNER JOIN MedicalDoctors ON MedicalDoctors.LoginID =
                    Logins.LoginID ";
                break;
            case 4:
                sqlFields += "'Admin' as FirstName, 'Admin' as LastName ";
                break;
        }
        string sqlCmd = sqlFields + sqlTables + "WHERE Logins.LoginID = @LoginID";
        try
        {
            DBObject dbo = new DBObject();
            SqlParameter[] parameters = new SqlParameter[]
            {
                new SqlParameter("LoginID", loginID)
            };
            using (FS_ExecuteQRYSRV srv = new FS_ExecuteQRYSRV())
            {
                return dbo.ExecuteQuery(sqlCmd, parameters, "sessionData");
            }
        }
    }
}
```

```
        catch (Exception ex)
        {
            System.Diagnostics.Debug.WriteLine("US_UsersWS.GetUserSessionData: " +
                                                ex.ToString());

            throw (ex);
        }
    }
else
{
    System.Diagnostics.Debug.WriteLine("US_UsersWS.GetUserSessionData: wrong
parameters sent(loginID, roleID)");
    return null;
}
}
```

Και η μέθοδος αυτή ορίζει τις παραμέτρους που θα χρησιμοποιηθούν από την Βασική Υπηρεσία FS\_ExecuteQRYSRV και συγκεκριμένα την μέθοδο ExecuteQuery\_Simple, η οποία θα κληθεί για να εξάγει τα δεδομένα από τη βάση. Σε περίπτωση σφάλματος, αυτό θα επιστρέφεται στο component που κάλεσε την υπηρεσία.

Με τη σειρά της η υπηρεσία FS\_ExecuteQRYSRV, η οποία χρησιμοποιείται και από τις δύο μεθόδους της Υπηρεσίας Υποστήριξης «UsersWS» εκτελεί τα παρακάτω:

```
[WebMethod]
public DataSet ExecuteQuery_Simple(string sql, DataSet parameters, string
tableName)
{
    SqlParameter[] param = null;
    if (parameters != null && parameters.Tables["Parameters"].Rows.Count > 0)
    {
        int count = parameters.Tables["Parameters"].Rows.Count;
        param = new SqlParameter[count];
        int i = 0;
        foreach (DataRow dr in parameters.Tables["Parameters"].Rows)
        {
            param[i] = new SqlParameter(dr["ParameterName"].ToString(),
                                        dr["ParameterValue"]);
            i++;
        }
    }
    try
    {
        return EMR.DAL.DBObject.ExecuteQuery(sql, parameters, tableName);
    }
    catch (Exception ex)
    {
        System.Diagnostics.Debug.WriteLine("Execute query: " + ex.ToString());
        throw ex;
    }
}
```

Η μέθοδος αυτή επικοινωνεί με το Data Access Layer του συστήματος, προκειμένου να εξάγει τα κατάλληλα δεδομένα από την βάση. Ο κώδικας που εκτελείται είναι ο εξής:



```
public DataSet ExecuteQuery(string sql, SqlParameter[] parameters, string
tableName)
{
    using (SqlCommand command = new SqlCommand(sql))
    {
        if (parameters != null)
            command.Parameters.AddRange(parameters);
        else
            command.Parameters.Clear();
        command.CommandType = CommandType.Text;
        DataSet dataset = null;
        try
        {
            if (transaction != null)
                dataset = DBSingleDatabase.db.ExecuteDataSet(command,
                    transaction);
            else
                dataset = DBSingleDatabase.db.ExecuteDataSet(command);
        }
        catch (Exception ex)
        {
            Debug.WriteLine(ex);
        }
        if (dataset != null && dataset.Tables.Count > 0)
        {
            dataset.Tables[0].TableName = tableName;
        }
        return dataset;
    }
}
```

To Data Access Layer χρησιμοποιεί μεθόδους και στοιχεία της Enterprise Library.

## Βιβλιογραφία και Αναφορές

1. AEMY, «Ηλεκτρονική Υγεία - Ηλεκτρονικός Ιατρικός Φάκελος», <http://www.aemy.gr/web/guest/74>
2. Al Saganich, «Java and Web Services Primer», ONJava.com, Ιούλιος 2001; <http://www.onjava.com/lpt/a/1025>, (24/9/2008).
3. Amatayakul M., «Electronic Health Records: A Practical Guide for Professionals and Organizations», American Health Information Management Association, 2004, [http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1\\_015872.pdf](http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1_015872.pdf)
4. Andrew Troelsen, «Pro C# 2008 and the .NET 3.5 Platform», Fourth Edition, Apress, 2007.
5. Arsanjani A., «Service-oriented modeling and architecture», Developer Works, Νοέμβριος 2004, <http://www.ibm.com/developerworks/library/ws-soa-design1/>
6. Arsanjani A., Zhang Λ., Ellis M., Allam A., Channabasavaiah K., « Design an SOA solution using a reference architecture», Developer Works, Μάρτιος 2007, <http://www.ibm.com/developerworks/library/ar-archtemp/>
7. Chappell D., «Introducing Windows Workflow Foundation», Microsoft Corporation, 2007
8. CHIME, UCL Centre for Health Informatics & Multiprofessional Education, <http://www.ucl.ac.uk/chime/>
9. Chitnis M., Tiwari P., Ananthamurthy L., «Introducing Web Services Part 2: Architecture», Νοέμβριος 2002, <http://www.developer.com/services/article.php/1495091>
10. Dick R., Steen E., Detmer D., « The Computer-Based Patient Record», The national Academies Press, 1997, [http://www.nap.edu/openbook.php?record\\_id=5306&page=179](http://www.nap.edu/openbook.php?record_id=5306&page=179)
11. ebPML.org, «Web Service architecture», 2003, <http://www.ebpml.org/webservices.htm>
12. Erl T., «SOA Design paterns», <http://www.soapatterns.org/>
13. Ferguson D., Stockton M., «SOA programming model for implementing Web Services», Developer Works, Ιούνιος 2005, <http://www.ibm.com/developerworks /webservices /library/ws-soa-progmodel/>





14. Garets D., Davis M., «Electronic Medical Records vs. Electronic Health Records: Yes, There Is a Difference», HIMSS Analytics, Ιανουάριος 2006, [http://www.himssanalytics.org/docs/WP\\_EMR\\_EHR.pdf](http://www.himssanalytics.org/docs/WP_EMR_EHR.pdf)
15. IBM, "New to SOA and Web services", 2007; <http://www-128.ibm.com/developerworks/webservices/newto/websvc.html>, (24/9/2008).
16. IBM, «Service oriented Architecture», <http://www.texnologia.gr/index.asp?mod=articles&id=1670>
17. Johns M., «Health information management technology», American Health Information Management Association, [http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1\\_015899.pdf](http://library.ahima.org/xpedio/groups/public/documents/ahima/bok1_015899.pdf)
18. Kanneganti R., Chodavarapu P., «SOA Security in action», <http://www.manning.com/kanneganti/>
19. Mandar Chitnis, Pravin Tiwari and Lakshmi Ananthamurthy, «Introduction to Web Services – Part 2: Architecture», Developer.com, 2002; <http://www.developer.com/services/article.php/1495091>, (24/9/2008).
20. Matthew MacDonald and Mario Szpuszta, «Pro ASP.NET 3.5 in C# 2008», Second Edition, Apress, 2007.
21. Microsoft Developer Network, «.NET Framework Conceptual Overview», Νοέμβριος 2007; <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>, (25/9/2008).
22. Microsoft Health Vault, <http://www.microsoft.com/en-us/healthvault/>
23. Microsoft HealthVault Future, XPLANE, [http://etimpera.com/?page\\_id=21](http://etimpera.com/?page_id=21)
24. Newcomer E. και Lomow G, "Understanding SOA with Web Services", Addison-Wesley Professional, Μάρτιος 2005, p. 314-420.
25. Options O.E, «Web Services: Έτοιμες υπηρεσίες στο Internet και πως μπορούν να χρησιμοποιηθούν από τις ΜΜΕ», Μάρτιος 2003; [http://www.go-online.gr/ebusiness/specials/article.html?article\\_id=213](http://www.go-online.gr/ebusiness/specials/article.html?article_id=213), (24/9/2008).
26. Papazoglou M., Heuvel W., «Service oriented architectures: approaches, technologies and research issues», The VLDB Journal, 2007, 16:389–415
27. Pearson, «SOA: Principles of Service Design», Prentice Hall
28. Psychology Wiki, «Electronic Health Record», [http://psychology.wikia.com/wiki/Electronic\\_Health\\_Record](http://psychology.wikia.com/wiki/Electronic_Health_Record)

29. Psychology Wiki, «electronic medical record», [http://psychology.wikia.com/wiki/Electronic\\_medical\\_record](http://psychology.wikia.com/wiki/Electronic_medical_record)
30. Psychology Wiki, «medical record», [http://psychology.wikia.com/wiki/Medical\\_records](http://psychology.wikia.com/wiki/Medical_records)
31. Pulier E., Taylor H., «Understanding Enterprise SOA», Μάιος 2006, Developers.com, <http://www.developer.com/>
32. Ravi Trivedi, «Web Services Tutorial: Understanding XML and XML Schema — Part 1», Developer.com, Απρίλιος 2003; <http://www.developer.com/services/print.php/2195981>, (24/9/2008).
33. Roger Wolter, «XML Web Services Basics», Microsoft Developer Network, Δεκέμβριος 2001; <http://msdn.microsoft.com/en-us/library/ms996507.aspx>, (24/9/2008).
34. Rosen M., Business Architecture and SOA, August 2006, <http://www.bpminstitute.org/articles/article/article/business-architecture-and-soa.html>
35. Thys M., Web Services implementation into the SOA Stack, February 2010, <http://thysmichels.com/2010/02/07/web-service-implementation-into-the-soa-stack/>
36. Waagemann P., «EHR vs. CPR vs. EMR», Healthcare Informatics online, 2003, [http://www.providersedge.com/ehdocs/ehr\\_articles/EHR\\_vs\\_CPR\\_vs\\_EMR.pdf](http://www.providersedge.com/ehdocs/ehr_articles/EHR_vs_CPR_vs_EMR.pdf)
37. Web Services Technology, <http://www.yaldex.com/>
38. Whipple E., Web Services Discovery, 2004, <http://conferences.embarcadero.com/article/32287>
39. Wikipedia, «.NET Framework», Αύγουστος 2008; [http://en.wikipedia.org/wiki/Microsoft\\_.NET#Microsoft\\_.NET](http://en.wikipedia.org/wiki/Microsoft_.NET#Microsoft_.NET), (25/9/2008).
40. Wikipedia, «ASP.NET», Σεπτέμβριος 2008; <http://en.wikipedia.org/wiki/Asp.net>, (25/9/2008).
41. Wikipedia, «C Sharp (Programming Language)», Σεπτέμβριος 2008; [http://en.wikipedia.org/wiki/C\\_Sharp\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/C_Sharp_(programming_language)), (25/9/2008).
42. Wikipedia, «Cascading Style Sheets», Σεπτέμβριος 2008; [http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets), (25/9/2008).
43. Wikipedia, «Google health», [http://en.wikipedia.org/wiki/Google\\_Health](http://en.wikipedia.org/wiki/Google_Health)
44. Wikipedia, «Medical Record», [http://en.wikipedia.org/wiki/Medical\\_record](http://en.wikipedia.org/wiki/Medical_record)
45. Wikipedia, «Microsoft SQL Server», Σεπτέμβριος 2008; [http://en.wikipedia.org/wiki/SQL\\_Server\\_2005](http://en.wikipedia.org/wiki/SQL_Server_2005), (25/9/2008).



46. Wikipedia, «Microsoft Visual Studio», Σεπτέμβριος 2008; [http://en.wikipedia.org/wiki/Visual\\_studio](http://en.wikipedia.org/wiki/Visual_studio), (25/9/2008).
47. Wikipedia, «Windows Communication Foundation», Σεπτέμβριος 2008; [http://en.wikipedia.org/wiki/Windows\\_Communication\\_Foundation](http://en.wikipedia.org/wiki/Windows_Communication_Foundation), (25/9/2008).
48. Your Research Resource, Peak Surveys, <http://www.peaksurveys.com/>
49. Zimmermann O., Krogdahl P., Gee C., « Elements of Service-Oriented Analysis and Design», Developer Works, Ιούνιος 2004, <http://www.ibm.com/developerworks/webservices/library/ws-soad1/>
50. Θεμιστοκλέους Μ., Μαντζάνα Β., «Υπηρεσίες Παγκόσμιου Ιστού και Υπηρεσιοστρεφής Αρχιτεκτονική» αυτό-έκδοση, 2010
51. Λιάσκος Ι., Πληροφορική της Υγείας: Πληροφοριακά Συστήματα Νοσοκομείων, 2007
52. Μαντζάνα Β., «Πληροφοριακά Συστήματα υγείας», 2007
53. Μπότσης Τ. , Χαλκιώτης Σ. , «Πληροφορική Υγείας», Δίαυλος, 2005
54. Παρατηρητήριο για την ψηφιακή Ελλάδα, «Μελέτη για τη χρήση τεχνολογιών πληροφορικής και επικοινωνιών στον τομέα υγείας και πρόνοιας», <http://www.observatory.gr/page/default.asp?la=1&id=2101&pk=400&return=183>

## Πηγές Εικόνων

- E1. Business Intelligence Technologies, SOA (Service Oriented Architecture) for Health Informatics, Φεβρουάριος 2011; <http://www.bitanz.com/>
- E2. Μπότσης Τ. , Χαλκιώτης Σ. , «Πληροφορική Υγείας», Δίαυλος, 2005
- E3. Λιάσκος Ι., Πληροφορική της Υγείας: Πληροφοριακά Συστήματα Νοσοκομείων, 2007
- E4. Μαντζάνα Β., «Πληροφοριακά Συστήματα υγείας», 2007
- E5. Προκήρυξη έργου «Ολοκληρωμένο Πληροφοριακό Σύστημα Υγείας» της ΔΥΠΕ Ανατολικής Μακεδονίας Θράκης, 2005, [www.syzefxis.gov](http://www.syzefxis.gov)
- E6. ebPML.org, «Web Service architecture», 2003, <http://www.ebpml.org/webservices.htm>
- E7. Whipple E., Web Services Discovery, 2004, <http://conferences.embarcadero.com/article/32287>
- E8. Chitnis M., Tiwari P., Ananthamurthy L., «Introducing Web Services Part 2: Architecture», Νοέμβριος 2002, <http://www.developer.com/services/article.php/1495091>
- E9. Web Services Technology, <http://www.yaldex.com/>
- E10. Pearson, «SOA: Principles of Service Design», Prentice Hall



- E11. Thys M., Web Services implementation into the SOA Stack, February 2010, <http://thysmichels.com/2010/02/07/web-service-implementation-into-the-soa-stack/>
- E12. Your Research Resource, Peak Surveys, <http://www.peaksurveys.com/>
- E13. Microsoft HealthVault Future, XPLANE, [http://etimpera.com/?page\\_id=21](http://etimpera.com/?page_id=21)
- E14. Rosen M., Business Architecture and SOA, August 2006, <http://www.bpminstitute.org/articles/article/article/business-architecture-and-soa.html>

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ