



Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Συστημάτων Μετάδοσης Πληροφορίας και  
Τεχνολογίας Υλικών

## **Ολοκληρωμένο σύστημα διαχείρισης εθελοντικής αιμοδοσίας στο υπολογιστικό νέφος**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**ΡΕΒΕΚΚΑ ΠΑΛΑΙΟΛΟΓΟΥ**

**ΦΩΤΟΠΟΥΛΟΣ ΗΛΙΑΣ**

**Επιβλέπων :** Διονύσιος-Δημήτριος Κουτσούρης

Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2015





Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Συστημάτων Μετάδοσης Πληροφορίας και  
Τεχνολογίας Υλικών

## Ολοκληρωμένο σύστημα διαχείρισης εθελοντικής αιμοδοσίας στο υπολογιστικό νέφος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΡΕΒΕΚΚΑ ΠΑΛΑΙΟΛΟΓΟΥ**

**ΦΩΤΟΠΟΥΛΟΣ ΗΛΙΑΣ**

**Επιβλέπων :** Διονύσιος-Δημήτριος Κουτσούρης

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 20η Οκτωβρίου 2015.

.....  
Διονύσιος-Δημήτριος Κουτσούρης  
Καθηγητής Ε.Μ.Π.

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

.....  
Ματσόπουλος Γεώργιος  
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2015

.....  
**ΡΕΒΕΚΚΑ ΠΑΛΑΙΟΛΟΓΟΥ**

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

.....  
**ΦΩΤΟΠΟΥΛΟΣ ΗΛΙΑΣ**

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

Copyright © Ρεβέκκα Παλαιολόγου, Φωτόπουλος Ηλίας, 2015.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

*Nulla dies  
sine linea.*

APELLES



## Ευχαριστίες

Με τη λήξη αυτού του ακαδημαϊκού κύκλου, θέλουμε να ευχαριστήσουμε όλους όσους στάθηκαν στο πλευρό μας και μας στήριξαν κατά τη διάρκεια της εκπόνησης της διπλωματικής μας εργασίας αλλά και καθ' όλη τη διάρκεια των σπουδών μας .

Θα θέλαμε αρχικά να ευχαριστήσουμε τον επιβλέποντα καθηγητή, κ. Δημήτριο Κουτσούρη, που μας έδωσε τη δυνατότητα να ασχοληθούμε με ένα τόσο ενδιαφέρον και σημαντικό θέμα.

Επίσης θα θέλαμε να ευχαριστήσουμε τον κ. Ιωάννη Κουρή για τη συνεχή και αποτελεσματική βοήθειά του, καθώς και τη συνολική υποστήριξή του καθ' όλη τη διάρκεια εκπόνησης της διπλωματικής μας εργασίας.

Τέλος, οφείλουμε ένα μεγάλο ευχαριστώ στις οικογένειές μας, στους γονείς μας Βασίλη-Κέλλυ και Κωνσταντίνο-Βασιλική αντίστοιχα, καθώς και τους φίλους μας, για την αμέριστη συμπαράσταση, υποστήριξή και συνολική τους βοήθεια σε όλα τα επίπεδα.

Ρεβέκκα Παλαιολόγου

Φωτόπουλος Ηλίας,

Αθήνα, 20η Οκτωβρίου 2015





## Περίληψη

Τα τελευταία χρόνια έχει παρατηρηθεί μια ραγδαία αύξηση στις ανάγκες για αίμα και για τα προϊόντα του αίματος γενικότερα, η οποία αναμένεται να συνεχίσει να αυξάνεται στα επόμενα χρόνια. Για να καλυφθούν οι όλο και μεγαλύτερες ανάγκες είναι μονόδρομος η κινητοποίηση, η ευαισθητοποίηση και η καλύτερη οργάνωση των εθελοντών έτσι ώστε να επιτευχθεί μεγαλύτερος αριθμός εθελοντικών αιμοδοσιών και κατ' επέκταση αυτάρκεια. Η παρούσα διπλωματική αφορά τον σχεδιασμό και υλοποίηση ενός ολοκληρωμένου συστήματος αιμοδοσίας το οποίο έχει ως στόχο να βοηθήσει σημαντικά στην εξεύρεση νέων εθελοντών, αλλά και ταυτόχρονα να διατηρήσει τους υπάρχοντες αυξάνοντας την συστηματικότητα τους και τον αριθμό των δωρεών που πραγματοποιούν. Μεγάλη βαρύτητα δόθηκε στον σχεδιασμό της αρχιτεκτονικής του συστήματος και διασύνδεσης του με τα διάφορα υποσύστημα τα του, αξιοποιώντας τις κρατούσες τεχνολογικές προσεγγίσεις περί ανάπτυξης συστημάτων πολλαπλών επιπέδων. Επίσης έχει γίνει εκτενής μελέτη και χρήση της τεχνολογίας υπολογιστικού νέφους λόγω της μεγάλης ελαστικότητας, της επεκτασιμότητας και της οικονομίας κλίμακας που προσφέρει. Συγκεκριμένα το σύστημα αποτελείται από τα παρακάτω διασυνδεδεμένα υποσυστήματα: i) Εφαρμογή έξυπνου κινητού τηλεφώνου (smartphone) προοριζόμενη για χρήση από τους εθελοντές αιμοδότες ii) Διαδικτυακή εφαρμογή (cloud portal) για διαχείριση των αιτημάτων αιμοδοσίας από τα κέντρα αιμοδοσίας iii) Υποσύστημα ελέγχου του ιατρικού ιστορικού του υποψηφίου εθελοντή αιμοδότη μέσω ανταλλαγής δεδομένων με χρήση πρωτοκόλλων HL7 και CDA Documents και απόφασης της καταλληλότητας του ή μη. Για το υποσύστημα ανταλλαγής δεδομένων μελετήθηκαν τα ζητήματα διαλειτουργικότητας με τρίτα συστήματα και προτάθηκαν μηχανισμοί διαλειτουργικότητας που εξυπηρετούν τις ανάγκες ανταλλαγής δεδομένων μεταξύ συστημάτων.

## Λέξεις κλειδιά

εθελοντική αιμοδοσία, πληροφοριακό σύστημα αιμοδοσίας, μητρώο αιμοδοτών, αιμοεπαγρύπνηση, εφαρμογή κινητού, υπολογιστικό νέφος, συστήματα υποστήριξης κλινικών αποφάσεων



## **Abstract**

Over the last few years we have witnessed a surge in demand of blood and generally of blood products. We expect the paucity of blood products to be further exacerbated by the ageing of the population. Efforts should be made in order to establish and maintain sufficient numbers of regular, volunteer blood donors to ensure an adequate and safe blood supply. In this thesis we aim to synergistically exploit mobile and cloud computing in order to design and implement a cloud-based blood donation management information system. This system has as primary goal to actively help in the recruitment and retention of blood donors by taking advantage of mobile technologies and gamification. Emphasis was given in the architectural design for the system by combining current technological architectures based on a multi-tier architecture and cloud-based software development. In this context we have extensively studied cloud computing and the advantages that can offer to the health sector. Particularly the system that has been implemented as part of this thesis consists of the following interconnected subsystems: i) Smartphone application ii) Cloud application iii) Clinical decision support subsystem that helps to determine the eligibility of the blood donor based on his medical history. In the context of the development of the aforementioned subsystem, we have also studied issues related to interoperability with third party systems and we proposed interoperability mechanisms based on international standards and best practices.

## **Key words**

blood donation, voluntary blood donation, blood donation management information system, haemovigilance, cloud computing, clinical decision support system



# Περιεχόμενα

Ευχαριστίες . . . . .	7
Περίληψη . . . . .	9
Abstract . . . . .	11
Περιεχόμενα . . . . .	13
Κατάλογος πινάκων . . . . .	17
Κατάλογος σχημάτων . . . . .	19
<b>1. Εισαγωγή . . . . .</b>	<b>23</b>
1.1 Στόχοι διπλωματικής . . . . .	23
1.2 Οργάνωση κειμένου . . . . .	25
<b>2. Εθελοντική αιμοδοσία . . . . .</b>	<b>27</b>
2.1 Εθελοντική αιμοδοσία . . . . .	27
2.1.1 Το αίμα . . . . .	27
2.1.2 Αιμοδοσία . . . . .	32
2.1.3 Εθελοντές αιμοδότες . . . . .	36
2.1.4 Εθελοντική αιμοδοσία στην Ελλάδα . . . . .	38

2.2	Κινδυνoi που προκύπτουν μέσα από αιμοδοσία (ασθένειες, ποσοστά)	41
2.2.1	Haemovigilance	42
2.2.2	Υπόσυστηματα Haemovigilance Ελλάδα	44
<b>3.</b>	<b>Ανάλυση και σχεδιασμός του συστήματος</b>	<b>51</b>
3.1	Λειτουργικές απαιτήσεις	51
3.1.1	Λειτουργίες εφαρμογής (web, mobile)	52
3.2	Σενάρια χρήσης	54
3.3	Διαγράμματα UML - Διαγράμματα δραστηριότητας	77
3.4	Business Model	85
3.4.1	Gamification	85
3.4.2	Social Networking Integration	91
3.4.3	Ειδοποιήσεις και Υπενθυμίσεις	93
3.4.4	Haemovigilance	95
<b>4.</b>	<b>Τεχνολογίες</b>	<b>97</b>
4.1	Back-End	97
4.1.1	Cloud	97
4.1.2	NodeJs	110
4.1.3	Βάση Δεδομένων	116
4.2	Front-End	128
4.2.1	Αρχιτεκτονική	128
4.2.2	Frameworks	131
4.2.3	Jade Template Engine	138
4.2.4	Mockups	140

4.2.5	Διασύνδεση με Social Networks . . . . .	140
4.3	Mobile . . . . .	140
4.3.1	Mobile OS . . . . .	140
4.3.2	Android . . . . .	141
<b>5.</b>	<b>Υλοποίηση</b> . . . . .	<b>149</b>
5.1	Αρχιτεκτονική του συστήματος . . . . .	149
5.2	Βάση δεδομένων . . . . .	151
5.2.1	Σχήμα Βάσης . . . . .	151
5.3	Web App . . . . .	152
5.3.1	NodeJs - Server Establishment . . . . .	152
5.3.2	Routing . . . . .	153
5.3.3	Οθόνες . . . . .	156
5.4	Mobile App . . . . .	163
5.4.1	Εγγραφή στην εφαρμογή - Sign Up . . . . .	165
5.4.2	Δημιουργία νέου ραντεβού . . . . .	166
5.4.3	Εμφάνιση ιστορικού αιμοδοσιών . . . . .	167
5.4.4	Λήψη ειδοποίησης(push notification) . . . . .	168
5.5	Testing and Tools . . . . .	168
5.5.1	Android Application Testing . . . . .	168
5.5.2	NodeJs Testing . . . . .	172
<b>6.</b>	<b>Διασυνδεσιμότητα</b> . . . . .	<b>175</b>
6.1	Πρωτόκολλα επικοινωνίας . . . . .	175
6.1.1	Το πρότυπο Health Level Seven (HL7) . . . . .	175

6.1.2	CDA Documents . . . . .	184
6.2	Δυνατότητες διασύνδεσης με άλλα υποσυστήματα . . . . .	186
6.2.1	Ηλεκτρονική Συνταγογράφηση . . . . .	186
6.2.2	Ηλεκτρονικός Ιατρικός Φάκελος (EMR) . . . . .	191
6.2.3	epSOS . . . . .	195
6.3	Eligibility as a Service . . . . .	197
6.3.1	Motivation . . . . .	197
6.3.2	Decision Support Systems & Clinical Decision Support . . . . .	199
6.4	Υλοποίηση . . . . .	200
<b>7.</b>	<b>Επίλογος . . . . .</b>	<b>205</b>
7.1	Μελλοντική δουλειά . . . . .	206
7.1.1	Εφαρμογή έξυπνου ρολογιού - SmartWatch . . . . .	206
7.1.2	Υποσύστημα πρόβλεψης αιμοδοσιών . . . . .	206
	<b>Βιβλιογραφία . . . . .</b>	<b>209</b>
	<b>Παράρτημα</b>	<b>221</b>
	<b>A. Ερωτηματολόγιο Αιμοδότη . . . . .</b>	<b>221</b>
	<b>B. Διασύνδεση Κέντρων Αίματος και Νοσοκομειακών Υπηρεσιών Αιμοδοσίας . . . . .</b>	<b>223</b>



## Κατάλογος πινάκων

2.1	Συμβατότητα πλάσματος . . . . .	32
2.2	Διάρκεια ζωής συστατικών αίματος . . . . .	34
3.1	Σενάριο χρήσης εγγραφής αιμοδότη . . . . .	56
3.2	Σενάριο χρήσης εγγραφής αιμοδότη . . . . .	57
3.3	Σενάριο χρήσης σύνδεσης στο σύστημα . . . . .	58
3.4	Σενάριο χρήσης προβολής γεγονότων αιμοδοσίας . . . . .	59
3.5	Σενάριο χρήσης λήψης ειδοποιήσεων . . . . .	60
3.6	Σενάριο χρήσης διαχείρισης ειδοποιήσεων . . . . .	61
3.7	Σενάριο χρήσης διαχείρισης ραντεβού αιμοδοσίας . . . . .	62
3.8	Σενάριο χρήσης προσθήκης ραντεβού αιμοδοσίας στο ημερολόγιο . . . . .	63
3.9	Σενάριο χρήσης ενημέρωσης για το υπολειπόμενο διάστημα υποχρεωτικής αποχής του αιμοδότη . . . . .	64
3.10	Σενάριο χρήσης παιχνιδοποίησης . . . . .	65
3.11	Σενάριο χρήσης δημοσίευσης στοιχείων στα μέσα κοινωνικής δικτύωσης . . . . .	66
3.12	Σενάριο χρήσης δημιουργίας έκκλησης για αίμα . . . . .	67
3.13	Σενάριο χρήσης διαχείρισης προσωπικού προφίλ . . . . .	68
3.14	Σενάριο χρήσης δημιουργίας έκκλησης για αίμα . . . . .	69
3.15	Σενάριο χρήσης δημιουργίας έκκλησης για αίμα . . . . .	70

3.16	Σενάριο χρήσης διαχείρισης ραντεβού αιμοδοσίας . . . . .	71
3.17	Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας . . . . .	72
3.18	Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας . . . . .	73
3.19	Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας . . . . .	74
3.20	Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας . . . . .	75
3.21	Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας . . . . .	76
3.22	Σενάριο χρήσης δημιουργίας έκκλησης για αίμα . . . . .	77
3.23	Στόχοι και μηχανισμοί του Gamification . . . . .	88
3.24	Μηχανισμοί, δυναμικές και κίνητρα στο Gamification . . . . .	89
4.1	Ευκαιρίες και Προκλήσεις cloud στον χώρο της υγείας . . . . .	98
4.2	Σύγκριση βασικών στοιχείων Android & iOS. . . . .	140
6.1	Παραδείγματα τύπων μηνυμάτων HL7 V2 . . . . .	177
6.2	Χαρακτήρες οριοθέτησης του προτύπου HL7 V2 . . . . .	178
6.3	Βασικά πεδία κλάσεων του μοντέλου RIM . . . . .	181
6.4	Κόστος για την συλλογή 1 μονάδας αίματος στην Ελλάδα. . . . .	198

## Κατάλογος σχημάτων

2.1	Συστατικά του αίματος . . . . .	28
2.2	Χαρακτηριστικά των κυττάρων του αίματος . . . . .	29
2.3	Αντισώματα και αντιγόνα του αίματος . . . . .	31
2.4	Συμβατότητα ομάδων αίματος . . . . .	32
2.5	Κατανομή ομάδων αίματος . . . . .	33
2.6	Διάγραμμα λειτουργίας . . . . .	45
2.7	Οροεπικράτηση λοιμώξεων σε 9.459.863 ελεγχθείσες μονάδες αίματος και αφαίρεσης αιμοπεταλίων. . . . .	47
2.8	Μέση μεταβολή των ορολογικών δεικτών των λοιμώξεων που μεταδίδονται με το αίμα. . . . .	48
2.9	Κατανομή αντιδράσεων. . . . .	48
2.10	Ανεπιθύμητες αντιδράσεις σχετικά με την αιμοδοσία 2013. . . . .	49
2.11	Συμβάντα κατά και μετά την αιμοδοσία, 2013. . . . .	49
3.1	Διάγραμμα δραστηριότητας #1. Εγγραφή εθελοντή αιμοδότη στο σύστημα. . . . .	78
3.2	Διάγραμμα δραστηριότητας #2. Είσοδος εθελοντή αιμοδότη στο σύστημα. . . . .	79
3.3	Διάγραμμα δραστηριότητας #3. Προβολή γεγονότων αιμοδοσίας, από τον εθελοντή αιμοδότη. . . . .	80
3.4	Διάγραμμα δραστηριότητας #4. Διαχείριση των ειδοποιήσεων για τα γεγονότα αιμοδοσίας από τον εθελοντή αιμοδότη. . . . .	81

3.5	Διάγραμμα δραστηριότητας #5. Κλείσιμο ραντεβού για αιμοδοσία από τον εθελοντή αιμοδότη. . . . .	82
3.6	Διάγραμμα δραστηριότητας #6. Καταχώρηση αιμοδοσίας από τον εγκεκριμένο χρήστη/διαχειριστή του συστήματος αιμοδοσίας σύστημα. . . . .	83
3.7	Διάγραμμα δραστηριότητας #7. Έλεγχος καταλληλότητας του υποψήφιου αιμοδότη από τον εγκεκριμένο χρήστη/διαχειριστή του συστήματος αιμοδοσίας. . . . .	84
3.8	Δομικά στοιχεία παιχνιδιών . . . . .	85
3.9	Κύκλος ενασχόλησης . . . . .	86
3.10	Στοιχεία παιχνιδιού . . . . .	87
3.11	Πλαίσιο σχεδίασης . . . . .	87
4.1	Cloud Computing. . . . .	100
4.2	Υπηρεσίες IaaS cloud computing. . . . .	102
4.3	Υπηρεσίες PaaS cloud computing. . . . .	103
4.4	Υπηρεσίες SaaS cloud computing. . . . .	105
4.5	Υπηρεσίες BaaS cloud computing. . . . .	106
4.6	Πολυνηματικό Μοντέλο. . . . .	114
4.7	Το μονομηματικό μοντέλο του NodeJs. . . . .	115
4.8	Τάση αναζητήσεων Google για NoSQL σε σύγκριση με RDBMS. . . . .	117
4.9	Δημοφιλές βάσεις δεδομένων τύπου NoSQL. . . . .	118
4.10	Αποθήκευση κατά Σειρές και κατά Στήλες. . . . .	120
4.11	Βάσεις δεδομένων βασισμένες στα έγγραφα. . . . .	121
4.12	Κλιμακωσιμότητα RDBMS. . . . .	126
4.13	Κλιμακωσιμότητα NoSQL. . . . .	127
4.14	Αρχιτεκτονική MVC. . . . .	129

4.15	Android Model-View αρχιτεκτονικό πρότυπο. . . . .	130
4.16	Android MVP αρχιτεκτονικό πρότυπο. . . . .	130
4.17	Παράδειγμα MVP σε Android. . . . .	132
4.18	AngularJS κύκλος εκκίνησης. . . . .	134
4.19	Αρχιτεκτονική του λειτουργικού συστήματος Android . . . . .	141
4.20	Δομικά στοιχεία του Android. . . . .	144
4.21	Ο κύκλος ζωής των activities. . . . .	146
4.22	Ο κύκλος ζωής των services . . . . .	147
5.1	Σχηματικό διάγραμμα του συστήματος. . . . .	150
5.2	Σχεσιακό διάγραμμα της βάσης του συστήματος. . . . .	151
5.3	Παράδειγμα διαγράμματος κλάσεων - MVP Android. . . . .	164
5.4	Κατηγορίες δοκιμών στο Android . . . . .	169
6.1	Βασικές αρχές του προτύπου HL7 V2 . . . . .	179
6.2	Τύποι δεδομένων του προτύπου HL7 V2 . . . . .	180
6.3	Διάγραμμα κλάσεων του μοντέλου RMI . . . . .	182
6.4	Τύποι δεδομένων του προτύπου HL7 v3 . . . . .	183
6.5	Το σύστημα E-prescription στην Ελλάδα. . . . .	191
6.6	Αρχιτεκτονική του συστήματος EaaS (Eligibility as a Service). . . . .	202



# Κεφάλαιο 1

## Εισαγωγή

### 1.1 Στόχοι διπλωματικής

Η παρούσα διπλωματική έχει ως στόχο να αντιμετωπίσει τα σημαντικά προβλήματα που εμφανίζονται στον χώρο της εθελοντικής αιμοδοσίας, τόσο σε επίπεδο μηχανογράφησης όσο και σε επίπεδο ανεπάρκειας δωρεών αίματος. Συγκεκριμένα οι στόχοι που ευελπιστούμε να πετύχουμε με το προτεινόμενο σύστημα είναι οι παρακάτω:

- **Στρατολόγηση περισσότερων νέων εθελοντών:** Το σύστημα που προτείνουμε στην παρούσα διπλωματική εργασίας και κυρίως η εφαρμογή κινητού που το συνοδεύει πρόκειται να βοηθήσει σε πολύ μεγάλο βαθμό στην εξεύρεση περισσότερων εθελοντών αιμοδοτών και ειδικότερα άτομα νεαρότερης ηλικίας τα οποία χρησιμοποιούν κατά κόρων τα προϊόντα της νέας τεχνολογίας. Η προσθήκη ατόμων της νέας γενιάς στις τάξεις των εθελοντών αιμοδοτών μπορεί να κλείσει την όλο και αυξανόμενη ψαλίδα μεταξύ προσφοράς και ζήτησης προϊόντων αίματος, προσφέροντας ενεργούς και υγιείς αιμοδότες για τις επόμενες δεκαετίες ισοσταθμίζοντας το πρόβλημα της γήρανσης του πληθυσμού. Κάτι το οποίο είναι υψίστης σημασίας δεδομένου ότι τα στατιστικά δείχνουν ότι δεν έχουμε καταφέρει να ευαισθητοποιήσουμε τους νέους έτσι ώστε να γίνουν τακτικοί εθελοντές αιμοδότες [112], γεγονός που δεν περιορίζεται εντός των ελληνικών συνόρων [105]. Επιπροσθέτως εκμεταλλευόμενοι την κοινωνική επιρροή και κοινωνική ψυχολογία σε συνδυασμό με τα κοινωνικά δίκτυα της σύγχρονης εποχής, οι υπάρχοντες αιμοδότες μπορούν να αποτελέσουν πρότυπο και να χρησιμοποιηθούν ως εργαλείο στρατολόγησης νέων αιμοδοτών [118]. Για αυτό θεωρούμε ότι είναι υψίστης σημασίας η διασύνδεση του συστήματός μας με τα μέσα κοινωνικής δικτύωσης δίνοντας ένα γενικότερο κοινωνικό τόνο σε όλη την διαδικασία της εθελοντικής αιμοδοσίας.
- **Διατήρηση και επιπλέον ενεργοποίηση των υπάρχοντων εθελοντών αιμοδοτών:** Δυστυχώς

ένα συχνό φαινόμενο που εμφανίζεται στον χώρο της εθελοντικής αιμοδοσίας είναι να συναντάμε περιπτώσεις εθελοντών οι οποίοι πραγματοποιούν μόλις μία αιμοδοσία χωρίς να πραγματοποιούν επόμενες αιμοδοσίες και να μετατραπούν σε τακτικούς ενεργούς αιμοδότες που τόσο χρειαζόμαστε. Αν και ο πιο συχνός λόγος που αναφέρεται από τους εθελοντές για τη μη δωρεά αίματος αποτελούν τα προβλήματα υγείας άλλοι σημαντικοί λόγοι αποτελούν οι περιορισμοί χρόνου καθώς και οι έλλειψη τακτικών υπενθυμίσεων και επιπλέον κινήτρων [112]. Οπότε κατά τον σχεδιασμό και την υλοποίηση του συστήματος δώσαμε ιδιαίτερη έμφαση στην δημιουργία κατάλληλων μηχανισμών υπενθύμισης, έτσι ώστε να μπορούμε να κινητοποιήσουμε "επιλέξιμους" εθελοντές οι οποίοι επιθυμούν να πραγματοποιήσουν δωρεά αίματος και απλά θέλουν κάποια υπενθύμιση. Προς αυτή την κατεύθυνση μέσω του συστήματος γίνεται χρήση προσωποποιημένων ειδοποιήσεων προς έξυπνα κινητά (mobile push notifications) σε κατάλληλα χρονικά διαστήματα. Επιπλέον χρησιμοποιώντας κατάλληλο σύστημα gamification μπορούμε να προσφέρουμε επιπλέον κίνητρα ενεργοποιώντας σε μεγαλύτερο βαθμό τους εθελοντές [62]. Χρησιμοποιώντας τις παραπάνω λειτουργικότητες είμαστε αισιόδοξοι πως το προτεινόμενο σύστημα είναι σε θέση να αξιοποιήσει στο μέγιστο βαθμό τους υπάρχοντες εθελοντές και να μετατρέψει τους περιστασιακούς σε τακτικούς αιμοδότες. Τέλος θα πρέπει να αναφερθεί ότι σε περίπτωση ανάγκης αξιοποιώντας τα παραπάνω μπορεί να πραγματοποιηθεί άμεση και επιτυχημένη έκκληση των εθελοντών αιμοδοτών για να καλύψουν την ανάγκη.

- **Διευκόλυνση της διαχείρισης της εθελοντικής αιμοδοσίας:** Το σύστημα που αναλύουμε παρέχει την δυνατότητα στον εθελοντή αιμοδότη να κλείσει εύκολα και άμεσα ραντεβού για αιμοδοσία σε κάποιο κοντινό κέντρο αιμοδοσίας είτε μέσω της εφαρμογής έξυπνου κινητού είτε μέσω της διαδικτυακής εφαρμογής.
- **Αύξηση ασφάλειας, αξιοπιστίας και μείωση κόστους:** Λαμβάνοντας υπόψιν πληροφορίες από το υποσύστημα επικοινωνίας με τον ηλεκτρονικό φάκελο ασθενούς καθώς και από το σύστημα ηλεκτρονικής συνταγογράφησης, επιτελούμε έναν πρωταρχικό έλεγχο εκλεξιμότητας του εθελοντή αιμοδότη. Σε αυτό το σημείο κρίνεται σκόπιμο να αναφερθεί ότι το εν λόγω υποσύστημα λειτουργεί ως σύστημα υποστήριξης απόφασης (Decision support system) και τον τελικό λόγο για την απόρριψη ή μη του εθελοντή την έχει ο γιατρός. Για κάθε εθελοντή που απορρίπτεται χωρίς να χρειαστεί να περάσει από το στάδιο αιματολογικών εξετάσεων εξοικονομείται χρόνος και χρήμα.

Στην συνέχεια της παρούσας διπλωματικής εργασίας αναλύονται με λεπτομέρεια οι παραπάνω στόχοι καθώς και οι τρόποι που προσπαθούμε να τους πετύχουμε.



## 1.2 Οργάνωση κειμένου

Η διπλωματική χωρίζεται σε 7 επιμέρους κεφάλαια. Στο κεφάλαιο 1 της διπλωματικής παρουσιάζεται το αντικείμενο και οι στόχοι που θέλει να πετύχει η παρούσα διπλωματική στο πεδίο των Ιατρικών Πληροφοριακών Συστημάτων. Στο κεφάλαιο 2 αναφέρονται γενικές πληροφορίες και στατιστικά για την εθελοντική αιμοδοσία στην Ελλάδα αλλά και στον υπόλοιπο κόσμο, όπου και γίνεται εμφανής η ανάγκη για μια βιώσιμη λύση. Στο τέλος του κεφαλαίου 2 γίνεται μια εισαγωγή στην αιμοεπαγρύπνηση και στην ελληνική μέριμνα σε αυτόν τον τομέα. Στο Κεφάλαιο 3 παρουσιάζεται λεπτομερώς η ανάλυση και η σχεδίαση του προτεινόμενου συστήματος αιμοδοσίας, που αποτελεί και το βασικό αντικείμενο της παρούσας διπλωματικής εργασίας. Στο κεφάλαιο 4 αναλύονται διεξοδικά οι διάφορες τεχνολογίες που χρησιμοποιήθηκαν τόσο στο Back-End όσο και στο Front-End των διάφορων υποσυστημάτων της εφαρμογής. Στο Κεφάλαιο 5 παρουσιάζεται λεπτομερώς η υλοποίηση του συστήματος αιμοδοσίας καθώς και μέθοδοι τεκμηρίωσης και αξιολόγησής τους. Στο κεφάλαιο 6 γίνεται αναφορά σε θέματα διασυνδεσιμότητας με τρίτα συστήματα και ανταλλαγής πληροφοριών με το σύστημα ηλεκτρονικής συνταγογράφησης και τον ηλεκτρονικό φάκελο ασθενούς και στην δημιουργία ενός clinical decision support system με μορφή υπηρεσίας στο υπολογιστικό νέφος. Επίσης παρουσιάζονται αναλυτικά πρωτόκολλα επικοινωνίας και μετάδοσης ευαίσθητων ιατρικών δεδομένων (HL7, CDA). Στο κεφάλαιο 7 γίνεται η σύνοψη της παρούσας διπλωματικής και παρουσιάζονται ιδέες και προτάσεις για την μελλοντική επέκταση της με επιπλέον υποσυστήματα και λειτουργικότητες.



## Κεφάλαιο 2

# Εθελοντική αιμοδοσία

## 2.1 Εθελοντική αιμοδοσία

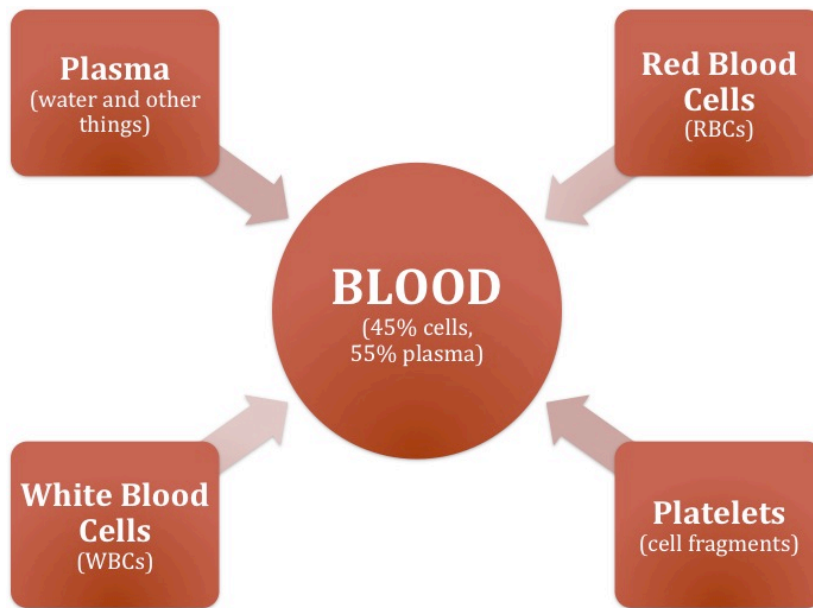
Στο παρόν κεφάλαιο αναλύουμε τις βασικές έννοιες σχετικά με την αιμοδοσία, οι οποίες εμπλέκονται στο θέμα της παρούσας διπλωματικής εργασίας.

### 2.1.1 Το αίμα

Το αίμα είναι το σπουδαιότερο βιολογικό υγρό του ανθρώπινου οργανισμού και αποτελεί ένα ανεκτίμητο προϊόν ζωής, το οποίο δεν μπορεί να κατασκευαστεί με συνθετικό τρόπο. Χωρίς την επαρκή ποσότητα αίματος, τα κύτταρα του ανθρώπινου σώματος δεν μπορούν να λάβουν το οξυγόνο και τα θρεπτικά συστατικά που τους είναι απαραίτητα για να επιβιώσουν [123]. Επιπλέον το σώμα δεν θα μπορούσε να αντιμετωπίσει βλαπτικούς παράγοντες, να αποβάλει τοξικά προϊόντα και να ρυθμίσει τις παραμέτρους του εσωτερικού του περιβάλλοντος [151]. Αν και η τεράστια αξία και χρησιμότητα του αίματος είναι αδιαμφισβήτητη, η μη ύπαρξη αρκετών εθελοντικών αιμοδοσιών για να καλυφθούν οι ανάγκες για αίμα αποτελεί μείζων πρόβλημα.

Το αίμα αποτελείται από ερυθρά αιμοσφαίρια, λευκά αιμοσφαίρια, αιμοπετάλια -τα οποία εναιωρούνται μέσα στο πλάσμα- και πρωτεΐνες. Στο σχήμα 2.1 παρακάτω παρουσιάζονται τα διάφορα συστατικά από τα οποία αποτελείται το αίμα:

Το πλάσμα αποτελεί το μεγαλύτερο και κυριότερο συστατικό του αίματος, καταλαμβάνοντας το 55% του συνολικού όγκου του. Είναι ένα υποκίτρινο υγρό μέσω του οποίου μεταφέρονται αιμοσφαίρια, πρωτεΐνες και άλλες ουσίες. Αποτελείται κατά 91,5% από νερό, κατά 7% από πρωτεΐνες, όπως η λευκωματίνη (αλβουμίνη), οι σφαιρίνες και το ινωδογόνο, και κατά 1,5% από άλλες ουσίες, όπως θρεπτικά συστατικά, ορμόνες, αναπνευστικά αέρια, ηλεκτρολύτες, βιταμίνες και άχρηστες αζωτού-



**Σχήμα 2.1:** Συστατικά του αίματος

χες ουσίες [29]. Η κύρια λειτουργία που επιτελεί είναι η μεταφορά υγρών και υδατοδιαλυτών ουσιών, όπως είναι οι ορμόνες.




Το υπόλοιπο 45% του αίματος αποτελείται από ερυθρά αιμοσφαίρια, λευκά αιμοσφαίρια και αιμοπετάλια. Τα ερυθρά αιμοσφαίρια είναι τα πιο πολυάριθμα κύτταρα σε κυκλοφορία και δίνουν στο αίμα το χαρακτηριστικό κόκκινο χρώμα του. Τα ερυθρά αιμοσφαίρια χρησιμοποιούνται ευρέως, για να αναπληρώνουν την απώλεια αίματος που προκαλείται από αιμορραγία κατά τη γέννα, κατά τη διάρκεια χειρουργικής επέμβασης και κατά τη διάρκεια ατυχημάτων. Η μετάγγιση ερυθρών αιμοσφαιρίων μπορεί επίσης να είναι σωτήρια για τη ζωή του ασθενούς σε συγκεκριμένους τύπους αναιμίας. Η λειτουργία τους αφορά τη διατήρηση των ιστών στη ζωή καθώς μεταφέρουν σε αυτούς οξυγόνο και απομακρύνουν το διοξείδιο του άνθρακα. Η εκατοστιαία αναλογία ερυθρών αιμοσφαιρίων ανά μονάδα όγκου αίματος ονομάζεται αιματοκρίτης [122].

Τα λευκά αιμοσφαίρια ή λευκοκύτταρα (WBC) αποτελούν λιγότερο από το 1% του ολικού αίματος. Η πρωταρχική λειτουργία των λευκοκυττάρων είναι η καταπολέμηση των λοιμώξεων μέσω της επίθεσης και της καταστροφής επιβλαβών ξένων ουσιών. Σχηματίζονται στον μυελό των οστών, στον σπλήνα και στους λεμφαδένες [28].

Τα αιμοπετάλια ή θρομβοκύτταρα παράγονται από τον μυελό των οστών και αποτελούν λιγότερο από το 1% του πλήρους αίματος. Παίζουν καθοριστικό ρόλο στην πήξη του αίματος και την αιμόσταση, δηλαδή στην αναστολή της αιμορραγίας ή της κυκλοφορίας. Σχηματίζουν θρόμβους ώστε να αποτρέπεται η διαρροή αίματος από τις πληγές και αν ο αριθμός τους είναι χαμηλός, αυτό μπο-

ρεί να οδηγήσει σε εύκολη δημιουργία μωλώπων και σε μεγάλη αιμορραγία. Οι ασθενείς που έχουν λευχαιμία ή ανεπάρκεια του μυελού των οστών, συνήθως έχουν χαμηλό ποσοστό αιμοπεταλίων με αποτέλεσμα να χρειάζονται αιμοπετάλια, για να διαφυλάξουν τη λειτουργία της πήξης του αίματός τους [122].

Στο σχήμα 2.2 βλέπουμε μια σύνοψη των βασικών χαρακτηριστικών των συστατικών του αίματος.

Element	Diameter (in $\mu\text{m}$ )	Number (per $\text{mm}^3$ )	Scientific notation <sub>3</sub> (per $\text{mm}^3$ )	Main function
 red blood cells	7 - 8	4,500,000 - 52500,000	$4.5 \times 10^6$ $5.5 \times 10^6$	oxygen transport
 white blood cells	9 - 12	7,000 - 10,000	$7 \times 10^3$ $1 \times 10^4$	defense against microorganisms
 platelets	2 - 4	300,000	$3 \times 10^5$	blood-clotting

**Σχήμα 2.2:** Χαρακτηριστικά των κυττάρων του αίματος

Τα κύτταρα του αίματος ανανεώνονται συνεχώς, όπως φαίνεται από τη διάρκεια ζωής στο αίμα των ερυθρών αιμοσφαιρίων (120 ημέρες), των αιμοπεταλίων (10 ημέρες) και των κοκκιοκυττάρων (9 ώρες). Ο χρόνος ζωής των λεμφοκυττάρων (T και B κυττάρων) ποικίλλει εξαιρετικά από ώρες έως χρόνια. Η παραγωγή ενεργών κυττάρων του αίματος λαμβάνει χώρα κατά κύριο λόγο στον μυελό των οστών. Ωστόσο ο σπλήνας, οι λεμφαδένες και οι βοηθητικοί λεμφοειδείς ιστοί είναι επίσης θέσεις συνεχιζόμενης παραγωγής κυττάρων, κυρίως λεμφικής σειράς [66].

Ένας υγιής ενήλικας έχει περίπου 5-6 λίτρα αίματος [104], και μπορεί να υποστεί απώλεια 500 ml χωρίς να υποστεί προβλήματα υγείας και να χρειαστεί κάποια μετάγγιση αίματος. Βέβαια άμα υποστεί απώλεια της τάξης των 1000-1500 ml σε μικρό χρονικό διάστημα ή κάποια συστατικά του αίματος (αιμοπετάλια, ερυθρά αιμοσφαίρια) είναι κάτω από τα απαιτούμενα επίπεδα λόγω ασθένειας (καρκίνος, αναιμία κτλπ) ή λόγω κάποιας εγχείρησης, τότε χρειάζεται μετάγγιση αίματος [144].

### Ομάδες αίματος και συμβατότητα

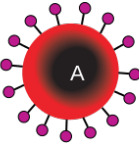
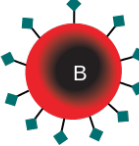
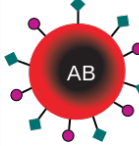



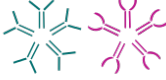



Στην επιφάνεια του ερυθρού αιμοσφαιρίου υπάρχουν διάφορα αντιγόνα ή ουσίες των ομάδων αίματος (blood group substances). Σήμερα είναι γνωστά 23 συστήματα ομάδων αίματος, γενετικά ανεξάρτητα το ένα από το άλλο. Κληρονομούνται σύμφωνα με τους νόμους του Mendel και η γνώση

τους είναι εξαιρετικά χρήσιμη για τη σωστή και ασφαλή μετάγγιση αίματος στη κλινική πράξη [44]. Κάθε σύστημα ομάδων αίματος περιλαμβάνει μία σειρά αντιγόνων που σχετίζονται ως προς τη δομή. Συνολικά, τα συστήματα αυτά περιλαμβάνουν περισσότερα από 400 αντιγόνα. Τα συστήματα ομάδας αίματος συμπεριλαμβάνουν τα ABO, τα Rh, MNS, Kell, Duffy, Kidd και άλλα.

Το πιο γνωστό και πιο ευρείας χρήσης είναι το σύστημα ABO. Είναι το πρώτο σύστημα που ανακαλύφθηκε το 1900 από τον Landsteiner (Βραβείο Νομπέλ Ιατρικής 1930) [101]. Η ανακάλυψη αυτή άνοιξε το δρόμο για την ασφαλή μετάγγιση αίματος. Το σύστημα ABO εξακολουθεί και σήμερα να είναι το πιο σημαντικό στη μετάγγιση. Το σύστημα συνδέεται με τρία αντιγόνα A, B και H. Το σύστημα χαρακτηρίζεται από την παρουσία ή την απουσία των αντιγόνων (συγκολλητινογόνων) στα ερυθρά αιμοσφαίρια. Με συνδυασμό αυτών διακρίνονται τέσσερις ομάδες αίματος, η AB, η A, η B, και η O. Η ομάδα AB χαρακτηρίζεται από την παρουσία στα ερυθρά αιμοσφαίρια και των δύο αντιγόνων A και B. Η ομάδα A χαρακτηρίζεται από την παρουσία του αντιγόνου A. Η ομάδα B χαρακτηρίζεται από την παρουσία του αντιγόνου B. Τέλος η ομάδα O δεν περιέχει κανένα από τα αντιγόνα A ή B, αλλά περιέχει το αντιγόνο H. Το τελευταίο υπάρχει σε όλες τις ομάδες αλλά ιδιαιτέρως παρατηρείται στην ομάδα O.

Στον ορό του αίματος των διαφόρων ατόμων παρατηρούνται φυσιολογικές συγκολλητίνες ομόλογοι των συγκολλητινογόνων A και B. Οι φυσιολογικές συγκολλητίνες είναι οι α (anti-A) και οι β (anti-B). Στον ορό δεν υπάρχει ποτέ η συγκολλητίνη η ομόλογος προς το συγκολλητινογόνο που υπάρχει στα ερυθρά αιμοσφαίρια του ίδιου ατόμου. Έτσι, στον ορό του αίματος AB δεν υπάρχει καμία συγκολλητίνη. Στον ορό της ομάδας A υπάρχει η συγκολλητίνη anti-B ή β. Στον ορό της ομάδας B υπάρχει η συγκολλητίνη anti-A ή α. Τέλος στον ορό της ομάδας O υπάρχουν και οι δύο συγκολλητίνες α και β. Η συγκολλητίνη α αντιδρά με το συγκολλητινογόνο A και η συγκολλητίνη β αντιδρά με το συγκολλητινογόνο B. Εάν επομένως σε μία μετάγγιση αίματος ο ορός του ασθενούς (δέκτη) έχει συγκολλητίνες (α ή β, α και β), τότε αυτές θα συγκολληθούν τα ερυθρά αιμοσφαίρια του δότη, όταν στα αιμοσφαίρια αυτά υπάρχουν συγκολλητινογόνα A ή B ή A και B. Στην περίπτωση αυτή τα συγκολλημένα ερυθρά αιμοσφαίρια μπορεί να προκαλέσουν ακόμη και τον θάνατο του δέκτη. Στο σχήμα 2.3 φαίνονται συγκεντρωτικά τα στοιχεία που αναλύσαμε παραπάνω.

Στις μεταγγίσεις πρέπει να λαμβάνεται υπόψιν και ένας άλλος παράγοντας, που λέγεται παράγοντας Rhesus, επειδή ανακαλύφθηκε πρώτα στα ερυθρά αιμοσφαίρια του πιθήκου Rhesus Maccacus. Το σύστημα Rhesus είναι το δεύτερο κατά σπουδαιότητα σύστημα ομάδων αίματος μετά το ABO. Η γνώση του είναι απαραίτητη για την ασφαλή μετάγγιση αίματος, ενώ είναι το σύστημα, που κυρίως ευθύνεται για την αιμολυτική νόσο του νεογνού. Σήμερα είναι γνωστά περί τα 50 αντιγόνα που ανήκουν στο σύστημα Rhesus. Από αυτά τα πέντε είναι τα πιο κύρια και βασικά. Το κυριότερο αντιγόνο

	Ομάδα A	Ομάδα B	Ομάδα AB	Ομάδα O
Τύπος Ερυθρού Κυττάρου				
Αντισώματα στο Πλάσμα	 Αντι-B	 Αντι-A	Κανένα	 Αντι-A και Αντι-B
Αντιγόνα στο Ερυθρό Κύτταρο	A αντιγόνο 	B αντιγόνο 	A και B αντιγόνα 	Κανένα

**Σχήμα 2.3:** Αντισώματα και αντιγόνα του αίματος

είναι το D και άτομα που το έχουν στα ερυθρά αιμοσφαίρια είναι Rh-Θετικά ή Rh (+), ενώ αυτά που δεν το έχουν είναι Rh-Αρνητικά ή Rh (-). Το αντιγόνο D είναι εξαιρετικά ανοσογόνο (immunogenic) και άτομα Rh-Αρνητικά, όταν εκτεθούν σε αυτό, μπορούν να σχηματίσουν anti-D αντισώματα. Το 85% των λευκών ανθρώπων έχουν τον παράγοντα αυτόν, δηλαδή είναι Rh-Θετικοί και το 15% δεν τον έχουν, δηλαδή είναι Rh-Αρνητικοί [102].

Συμβάντα μπορεί να παρατηρηθούν, αν δεν προσδιοριστεί ο παράγοντας Rhesus, στις εξής περιπτώσεις.

1. Σε άτομα στα οποία έγινε μια πρώτη μετάγγιση και στα οποία μια δεύτερη μετάγγιση μπορεί να είναι θανατηφόρα
2. Στις γυναίκες στις οποίες γίνεται μετάγγιση κατά τη διάρκεια της εγκυμοσύνης τους
3. Στις γυναίκες που γέννησαν ήδη το πρώτο τους παιδί και στις οποίες μετά από λίγο γίνεται μετάγγιση
4. Στα έμβρυα λόγω του παράγοντα Ρέζους μπορεί να προκληθεί μια πολύ σοβαρή πάθηση που λέγεται ερυθροβλάστωση των εμβρύων (αν η μητέρα είναι Ρέζους αρνητική, ο πατέρας Ρέζους θετικός και το έμβρυο επίσης Ρέζους θετικό). Κατά την αρρώστια αυτή τα αιμοσφαίρια του εμβρύου συγκολλούνται και προκαλείται τελικά ο θάνατος του. Μπορεί να σωθεί μόνο, αν γεννηθεί ζωντανό και γίνει αλλαγή του αίματος του (αφαιμαξομετάγγιση) με άλλο αίμα Ρέζους αρνητικό

Βάση όσων έχουν αναφερθεί παραπάνω, στο σχήμα 2.4 βλέπουμε τη συμβατότητα των διαφόρων

συνδυασμών.

Δότης

Ομάδα	O <sup>-</sup>	O <sup>+</sup>	B <sup>-</sup>	B <sup>+</sup>	A <sup>-</sup>	A <sup>+</sup>	AB <sup>-</sup>	AB <sup>+</sup>
AB <sup>+</sup>	🩸	🩸	🩸	🩸	🩸	🩸	🩸	🩸
AB <sup>-</sup>	🩸		🩸		🩸		🩸	
A <sup>+</sup>	🩸	🩸			🩸	🩸		
A <sup>-</sup>	🩸				🩸			
B <sup>+</sup>	🩸	🩸	🩸	🩸				
B <sup>-</sup>	🩸		🩸					
O <sup>+</sup>	🩸	🩸						
O <sup>-</sup>	🩸							

Δ  
έ  
κ  
τ  
η  
ς

Σχήμα 2.4: Συμβατότητα ομάδων αίματος

Από την άλλη μεριά η μετάγγιση πλάσματος έχει τους αντίθετους κανόνες αφού τα αντιγόνα βρίσκονται στο πλάσμα. Για παράδειγμα ένας ασθενής με τύπο αίματος O μπορεί να λάβει πλάσμα από τις ομάδες αίματος A,B και AB αφού το πλάσμα του τύπου O περιέχει αντιγόνα και της A και της B. Στον πίνακα 2.1 βλέπουμε τη συμβατότητα πλάσματος των διάφορων συνδυασμών.

Ασθενής	Αιμοδότης			
	O	A	B	AB
O	Ναι	Ναι	Ναι	Ναι
A	Όχι	Ναι	Όχι	Ναι
B	Όχι	Όχι	Ναι	Ναι
AB	Όχι	Όχι	Όχι	Ναι

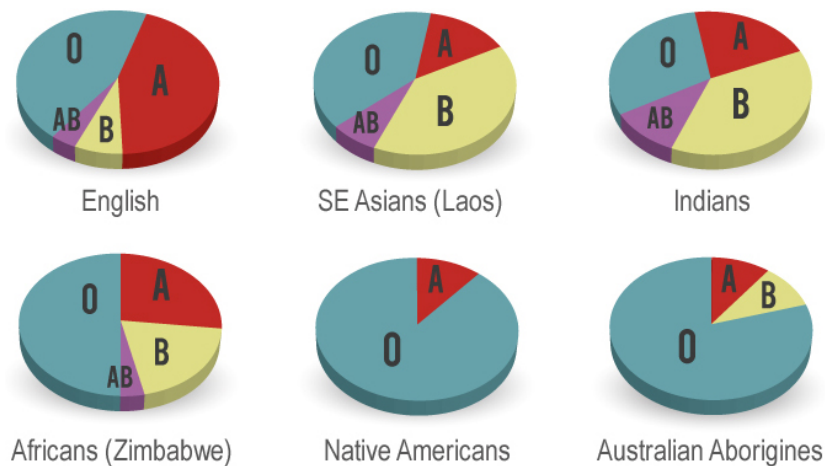
Πίνακας 2.1: Συμβατότητα πλάσματος

Σε αυτό το σημείο, θα πρέπει να αναφερθεί ότι η κατανομή του πληθυσμού στις διάφορες ομάδες αίματος δεν είναι ισόποση. Στο σχήμα 2.5 βλέπουμε την κατανομή σε επιλεγμένες χώρες.

### 2.1.2 Αιμοδοσία

Με τον όρο αιμοδοσία, εννοούμε τη χορήγηση αίματος από υγιείς δότες σε άτομα στα οποία η κατάσταση της υγείας τους απαιτεί μετάγγιση. Κατ' επέκταση με τον όρο αιμοδοσία εννοούμε την





**Σχήμα 2.5:** Κατανομή ομάδων αίματος

όλη οργάνωση που ασχολείται με τη λήψη, επεξεργασία, συντήρηση και διάθεση του αίματος και των παραγώγων του [34]. Η αιμοδοσία καλείται εθελοντική, επειδή αφορά σε πράξη που εκτελεί κάποιος με τη θέλησή του και με μοναδικά κίνητρα τα αισθήματα αλληλεγγύης και αλtruισμού [59].

Ως επιστημονικός τομέας, η αιμοδοσία αποτελεί ιδιαίτερο κλάδο της αιματολογίας με τεράστια ανάπτυξη τα τελευταία 20 χρόνια. Η καθιέρωση της αιμοδοσίας ως εξειδικευμένου τομέα, καθώς και η αλματώδης ανάπτυξή της, οδήγησαν στην ανάγκη να πλαισιώνεται από ιατρικό, νοσηλευτικό και τεχνικό προσωπικό υψηλής στάθμης με εξειδίκευση στον τομέα της αιμοδοσίας.

Η μετάγγιση αίματος γίνεται τακτικά σε εγχειρήσεις, τραυματίες, γαστρορραγίες και σε τοκετούς για την αναπλήρωση της απώλειας σημαντικής ποσότητας αίματος. Στις περισσότερες περιπτώσεις η μετάγγιση αίματος χρησιμοποιείται ως προσωρινή μορφή θεραπείας. Βέβαια, υπάρχουν και περιπτώσεις όπου ο ασθενής χρειάζεται μεταγγίσεις αίματος εφόρου ζωής. Μερικές ευρέως γνωστές ασθένειες που απαιτούν κάτι τέτοιο είναι η Β-Θαλασαιμία (η Ελλάδα έχει τα υψηλότερα ποσοστά), αιμοφιλία και λευχαιμία [146].

Αν και υπάρχουν κάποιες περιπτώσεις οι οποίες απαιτούν μετάγγιση ολικού αίματος, η πλειονότητα του αίματος διασπάται μετά τη δωρεά και τα απαιτούμενα προϊόντα αίματος μεταφέρονται στον ασθενή. Ακολουθώντας την παραπάνω προσέγγιση, από τη μία μεριά ο ασθενής δεν λαμβάνει περιττά συστατικά και από την άλλη μεριά από μία μονάδα ολικού αίματος μπορούν να επωφεληθούν περισσότεροι του ενός ατόμου. Η διάρκεια ζωής των διαφόρων συστατικών του αίματος φαίνεται στον πίνακα 2.2 παρακάτω [17]:

Συστατικό	Διάρκεια	Θερμοκρασία
Ολικό αίμα	24 ώρες	20-24
Ερυθρά αιμοσφαίρια	42 μέρες	4
Αιμοπετάλια	3-5 μέρες	20-24
Πλάσμα	1 χρόνος	-18

Πίνακας 2.2: Διάρκεια ζωής συστατικών αίματος

Από τα δεδομένα του παραπάνω πίνακα γίνεται εύκολα εμφανές ότι είναι προτιμότερο να αποθηκεύονται τα συστατικά του αίματος πάρα το ολικό αίμα, ενώ παράλληλα γίνεται εμφανής η ανάγκη για συνεχή παροχή αιμοδοσιών.

Το αίμα που χρησιμοποιείται στις μεταγγίσεις πρέπει να προέρχεται από υγιή άτομα. Το αίμα δεν είναι μόνο ζωντανός ιστός, αλλά έχει επιπλέον την ιδιότητα να ανανεώνεται και τα υγιή άτομα διαθέτουν μηχανισμούς αύξησης της παραγωγής αίματος. Έτσι με την αιμοδοσία προσφέρεται εύκολα το δώρο της ζωής χωρίς τον φόβο ότι η τακτική αιμοδοσία θα προκαλέσει εξασθένηση του οργανισμού και θα οδηγήσει σε αδυναμία η επιτάχυνση της γήρανσης.

Στόχος είναι οι εθελοντές, που πληρούν τα κριτήρια για αιμοδοσία να γίνονται τακτικοί αιμοδότες, δηλαδή να πραγματοποιούν δωρεά αίματος αρκετές φορές το χρόνο και να παραμένουν στον κατάλογο των ενεργών αιμοδοτών για πολλά χρόνια. Η διατήρηση ενός υψηλού επιπέδου ποιότητας παρεχομένων υπηρεσιών στην υπηρεσία αιμοδοσίας συνίσταται στην προτεραιότητα ικανοποίησης των αναγκών και των προσδοκιών των εθελοντών αιμοδοτών. Ωστόσο είναι σημαντικό να επισημάνουμε ότι μια επένδυση στην προσέλκυση και στη διατήρηση εθελοντών θα αποδώσει ασφαλή αποθέματα αίματος και προστασία της υγείας τόσο στον δότη όσο και στον λήπτη. Επιπλέον θα υπάρχει και σημαντική εξοικονόμηση κόστους για την υπηρεσία μέσω της μείωσης του αριθμού μονάδων αίματος που πρέπει να απορριφθούν λόγω της ανεύρεσης θετικών δεικτών λοιμωδών νοσημάτων.

Η προσέλκυση και η διατήρηση των εθελοντών αιμοδοτών είναι μια δυναμική λειτουργία που σχεδιάζεται κάθε φορά ανάλογα με τη μελέτη και την ανάλυση των παραμέτρων της συγκεκριμένης κοινωνικής ομάδας που απευθυνόμαστε σε σχέση με την αξιολόγηση και εκτίμηση των αναγκών σε αίμα και την υπάρχουσα κατάσταση στον χώρο.

## Διαδικασία της Αιμοδοσίας

Η επιλογή των αιμοδοτών γίνεται από άρτια εκπαιδευμένο και επαγγελματικά καταρτισμένο προσωπικό που έχει ως σκοπό την ασφάλεια τόσο του αιμοδότη όσο και του δέκτη. Στην είσοδο της αιμοδοσίας υπάρχουν ειδικά έντυπα αυτοαποκλεισμού, τα οποία καλούνται να διαβάσουν όλοι οι άνθρωποι που προσέρχονται προκειμένου να προσφέρουν αίμα [57]. Στην συνέχεια αν ο δυνητικός αιμοδότης κρίνει ότι πληροί τα κριτήρια αποδοχής και δεν ισχύει κάποιο από τα κριτήρια αποκλεισμού για την περίπτωση του, θα πρέπει να συμπληρώσει το ερωτηματολόγιο του αιμοδότη [56] (περισσότερα στο Παράρτημα Α). Το προαναφερθέν ερωτηματολόγιο είναι εμπιστευτικό διασφαλίζοντας το ιατρικό απόρρητο και θα πρέπει να συμπληρώνεται από τον αιμοδότη κάθε φορά που θέλει να δώσει αίμα. Ο αιμοδότης έχει τη δυνατότητα να συζητήσει με τον ιατρό προβλήματα υγείας ή άλλους λόγους που θέτουν ενδεχομένως σε κίνδυνο την ασφάλεια του ίδιου ή του δέκτη.

Στην συνέχεια ο αιμοδότης εξετάζεται για αρτηριακή πίεση, σφυγμό, βάρος, αιματοκρίτη ή αιμοσφαιρίνη, πιθανές δερματικές αλλοιώσεις στο σημείο της φλεβοκέντησης, καθώς και σχετικά με τη φυσική του κατάσταση. Ο γιατρός της αιμοδοσίας, εφόσον συλλέξει τις απαραίτητες πληροφορίες και λαμβάνοντας υπόψιν τις απαντήσεις στο ερωτηματολόγιο του αιμοδότη, κρίνει αν κάποιος, ο οποίος προσέρχεται στην υπηρεσία αιμοδοσίας, είναι σε θέση να προσφέρει αίμα. Την τελική ευθύνη για την επιλογή του αιμοδότη την επωμίζεται ο ιατρός της αιμοδοσίας.

Πρέπει να σημειωθεί, σε αυτό το σημείο, πως είναι απαραίτητη η έγγραφη συγκατάθεση του αιμοδότη ότι δέχεται να αιμοδοτήσει και να εξεταστεί το αίμα του για μεταδοτικά νοσήματα. Σε περίπτωση απόρριψης δίνονται οι απαραίτητες ιατρικές πληροφορίες και εξηγήσεις οι οποίες αφορούν στον λόγο και στη διάρκεια αποκλεισμού (για περισσότερα βλ: 2.1.3).

Αφού κριθεί ότι ο αιμοδότης πληροί τις προϋποθέσεις, οδηγείται στην ειδική καρέκλα της αιμοδοσίας, γίνεται περίδεση του βραχίονα και καλή αντισηψία στην περιοχή της φλεβοκέντησης. Κατά τη διαδικασία της αιμοληψίας συστήνεται στον αιμοδότη να ανοιγοκλείνει τη γροθιά του, ώστε να διευκολύνεται η ροή του αίματος. Λαμβάνονται 450ml αίματος. Ο όγκος του αίματος, που προσφέρεται, αναπληρώνεται μέσα σε 24 ώρες μετά την αιμοδοσία.

Μετά την αιμοληψία, το αίμα συλλέγεται σε ειδικούς σάκους, που περιέχουν αντιπηκτικές και άλλες ουσίες, οι οποίες βοηθούν στην άριστη διατήρηση του αίματος. Πριν αφαιρεθεί η βελόνα λαμβάνονται δείγματα για τις εξετάσεις, όπως προβλέπεται από το νόμο. Στο τέλος, ο αιμοδότης καλείται να παραμείνει καθιστός για λίγη ώρα, καθώς παράλληλα σε κάποια κέντρα του προσφέρεται χυμός και κάποιο μικρό πρόχειρο γεύμα.

Πρόκειται για μια ανώδυνη διαδικασία. Μοναδικό ενόχλημα είναι ένας μικρός πόνος από τη βελόνα. Η αιμοδοσία είναι ακίνδυνη για τον αιμοδότη. Η περίπτωση να μολυνθεί ο αιμοδότης από AIDS ή άλλο μεταδιδόμενο νόσημα είναι μηδενική, αφού οι βελόνες που χρησιμοποιούνται είναι μιας χρήσης και αποστειρωμένες.

Η αιμοδοσία διαρκεί περίπου δέκα λεπτά. Για την όλη διαδικασία, από τη συμπλήρωση του ερωτηματολογίου μέχρι να φύγει ο αιμοδότης, απαιτείται περίπου μισή ώρα [57].

### **2.1.3 Εθελοντές αιμοδότες**

#### **Κριτήρια επιλογής αιμοδοτών**

Αίμα μπορούν να δώσουν όλοι οι υγιείς άντρες και γυναίκες ηλικίας 18 - 65 ετών οι οποίοι έχουν σωματικό βάρος μεγαλύτερο των 50 κιλών, κάθε 3 - 4 μήνες. Βασικός στόχος και υποχρέωση των υπηρεσιών μετάγγισης αίματος είναι να συλλέξουν αίμα από υγιείς αιμοδότες, ώστε αφενός να προφυλαχθεί η δική τους υγεία και αφετέρου να προστατευθεί ο αιμολήπτης ασθενής από τη μετάδοση ασθενειών ή φαρμακευτικών ουσιών που μπορεί δυνητικά να είναι βλαβερά για την υγεία του. Ο υποψήφιος αιμοδότης κατά τη λήψη του ιστορικού πρέπει να αναφέρει τυχόν συμπτώματα, ώστε να βοηθήσει το ιατρικό προσωπικό να κρίνει με ασφάλεια την πιθανότητα κινδύνου. Κάθε πρόβλημα υγείας που ενδεχομένως έχει ο υποψήφιος αιμοδότης, πρέπει να συζητείται με τον υπεύθυνο γιατρό της αιμοδοσίας, ο οποίος και κρίνει τελικά για τη καταλληλότητα της αιμοληψίας. Η διαδικασία θα πρέπει να είναι τέτοια έτσι ώστε να εξασφαλίζεται ασφαλή αιμοδοσία και παροχή αίματος αλλά και να μην απορρίπτονται υγιείς αιμοδότες οι οποίοι θα μπορούσαν να συνεισφέρουν [124].

Ο αιμοδότης πρέπει να είναι σε καλή υγεία και απαλλαγμένος από μεταδοτικές ασθένειες. Όμως κάθε άνθρωπος είναι επιρρεπής σε μικρο-αδιαθεσίες. Αυτές είναι πόνοι κάθε είδους, ακμή, πονόλαιμοι και δυσπεψία. Όλα αυτά δεν αποτελούν στοιχεία απόρριψης του αιμοδότη. Εάν ο δότης υποβάλλεται σε φαρμακευτική αγωγή ή έχει υποβληθεί στο άμεσο παρελθόν, πρέπει να σημειωθούν τα παρακάτω σχετικά με τα φάρμακα που παίρνει. Η ποσότητα του φαρμάκου, δηλαδή η πυκνότητα του φαρμάκου στον οργανισμό του δότη και η ταχύτητα απορρόφησης ή η αποβολή του. Το φάρμακο μπορεί να έχει δυσμενή επίπτωση στον δέκτη εφόσον η περιεκτικότητα του φαρμάκου στον δότη είναι αυξημένη. Εάν ο δέκτης είναι αλλεργικός σ' αυτό το φάρμακο ή εάν ο δέκτης είναι έγκυος γυναίκα μπορεί να προκληθεί μέχρι και τερατογένεση. Το φάρμακο μπορεί να διαταράξει το αίμα του δότη, π.χ. τη λειτουργικότητα των αιμοπεταλίων. Παρόλο ότι τα φαινόμενα αυτά έχουν αναγνωριστεί, η συχνότητά τους δεν μελετήθηκε ακόμη καλά. Εάν λοιπόν ο αιμοδότης λαμβάνει φάρμακα, δεν σημαίνει

ότι αναγκαστικά δεν μπορεί να προσφέρει αίμα. Σε κάθε περίπτωση όμως είναι ορθό να ενημερώνεται ο γιατρός και το προσωπικό για τα φάρμακα που λαμβάνει και ανάλογα θα κριθεί εάν μπορεί να προσφέρει αίμα [124].

Υπάρχουν καταστάσεις και νοσήματα που αποκλείουν δια παντός την αιμοδοσία, όπως είναι το AIDS, οι ηπατίτιδες, η ελονοσία, η χρήση ενδοφλεβίων ναρκωτικών, οι κακοήθειες, η υπέρταση, ο σακχαρώδης διαβήτης ή σοβαρά χρόνια νοσήματα. Στις περισσότερες περιπτώσεις, όμως, ο αποκλεισμός είναι μόνο πρόσκαιρος. Ο αποκλεισμός αυτός γίνεται για να μην επιβαρυνθεί η υγεία του αιμοδότη και για να διασφαλιστεί η ποιότητα του αίματος που θα μεταγγιστεί στο λήπτη. Μελέτες έχουν δείξει ότι ο προσωρινός αποκλεισμός από την αιμοδοσία έχει έντονο αρνητικό αντίκτυπο στη μελλοντική επιστροφή του αιμοδότη [41][40]. Για αυτό είναι υψίστης σημασίας να δίνονται ξεκάθαρα διαστήματα αποκλεισμού και να ενθαρρύνονται οι εθελοντές να επιστρέψουν μετά το πέρας αυτού του διαστήματος. Στο σύστημα που προτείνουμε στην παρούσα διπλωματική έχει γίνει μέριμνα για τέτοιες περιπτώσεις και υλοποιήθηκε ένα σύστημα κατάλληλων ειδοποιήσεων.

Εκτός από τις γενικές αυτές αρχές και επειδή οι λεπτομέρειες στις ενδείξεις αναπροσαρμόζονται, ο υποψήφιος αιμοδότης πρέπει να συμβουλευέται το προσωπικό της αιμοδοσίας που είναι το πλέον αρμόδιο για την επιλογή των αιμοδοτών.

### **Κατηγορίες αιμοδοτών**

Οι αιμοδότες μπορούν να χωριστούν σε τρεις μεγάλες κατηγορίες:

1. Οι εθελοντές αιμοδότες (Volunteer Donors - VDs) οι οποίοι αιμοδοτούν με δική τους πρωτοβουλία καθαρά για ανθρωπιστικούς λόγους, χωρίς να λαμβάνουν κάποιο οικονομικό αντάλλαγμα ή οτιδήποτε που θα μπορούσε να θεωρηθεί ως αντικαταστατό του χρήματος.
2. Οι δότες αντικατάστασης (Replacement Donors - RDs), οι οποίοι αιμοδοτούν προκειμένου να καλύψουν τις ανάγκες που προκύπτουν από συγγενείς ή φίλους οι οποίοι νοσηλεύονται.
3. Δότες αίματος επί πληρωμή οι οποίοι λαμβάνουν πληρωμή από την οικογένεια η οποία δεν μπορεί να παρέχει η ίδια δότη αντικατάστασης για το συγγενικό τους πρόσωπο.

Σε αυτό το σημείο πρέπει να αναφέρουμε, ότι η τρίτη κατηγορία αιμοδοτών εμφανίζεται μόνο σε τριτοκοσμικές χώρες και κυρίως σε χώρες της Αφρικής. Στατιστικά στοιχεία έχουν δείξει ότι οι τακτικοί εθελοντές αιμοδότες (VDs) σχετίζονται με ασφαλέστερες παροχές αίματος σε σχέση με τους δότες αντικατάστασης όσον αφορά στις μεταδιδόμενες κατά την μετάγγιση ασθένειες [107].

Στοιχεία του Παγκόσμιου Οργανισμού Υγείας (WHO) και του Συμβουλίου της Ευρώπης υποδεικνύουν ότι το αίμα και τα παράγωγα του αίματος θα πρέπει να συλλέγονται αποκλειστικά από τακτικούς μη αμειβόμενους εθελοντές [49]. Τα συστήματα αιμοδοσίας τα οποία στηρίζονται στους εθελοντές αιμοδότες οι οποίοι δίνουν αίμα σε σταθερή βάση έχουν τη δυνατότητα να διαχειριστούν καλύτερα τις παροχές αίματος και να προγραμματίσουν τις μεταγγίσεις, επιταχύνοντας την όλη διαδικασία. Τέλος, από ηθική άποψη, δεν είναι σωστό να αναγκάζονται οι συγγενείς ενός ασθενούς σε ανάγκη, να αναζητούν κάτω από ψυχολογική πίεση άτομα για να προσφέρουν αίμα προκειμένου να καλυφθούν οι ανάγκες του δικού τους ανθρώπου.

Οι εθελοντές αιμοδότες (VDs) μπορούν να χωριστούν περαιτέρω στις ακόλουθες κατηγορίες [159]:

1. Στους συστηματικούς και αυτόνομους, οι οποίοι προσέρχονται να αιμοδοτήσουν με δική τους αποκλειστικά πρωτοβουλία.
2. Στους οργανωμένους σε συλλόγους ή τράπεζες αίματος, που καλούνται να δώσουν αίμα.
3. Στους περιστασιακούς, που απαντούν σε εκκλήσεις ραδιοφωνικών σταθμών και άλλων μέσων.
4. Στους εποχιακούς, που δίνουν αίμα κατά την ημέρα της αιμοδοσίας του Δήμου, του πολιτιστικού συλλόγου που ανήκουν και άλλων οργανώσεων.

Ο πραγματικός βέβαια εθελοντής αιμοδότης είναι εκείνος που προσέρχεται να δώσει αίμα με μοναδικό κίνητρο την κοινωνική αλληλεγγύη και τον αλtruισμό, δεν τον απασχολεί σε ποιον θα δοθεί το αίμα που πρόσφερε και δεν περιμένει κανένα απολύτως αντάλλαγμα.

#### **2.1.4 Εθελοντική αιμοδοσία στην Ελλάδα**

Στην Ελλάδα το "Εθνικό Κέντρο Αιμοδοσίας" (Ε.ΚΕ.Α) το οποίο στεγάζεται στο Υπουργείο Υγείας και Πρόνοιας αποτελεί το κεντρικό όργανο για την οργάνωση των Υπηρεσιών Αιμοδοσίας. Οι Υπηρεσίες Αιμοδοσίας μπορούν να διαχωριστούν στις παρακάτω τρεις κατηγορίες:

1. Κέντρα Αίματος
2. Σταθμοί Αιμοδοσίας Α' Τάξης
3. Σταθμοί Αιμοδοσίας Β' Τάξης

Τα κέντρα Αίματος καλύπτουν τις ανάγκες μιας ευρύτερης γεωγραφικής περιοχής ή μεγάλων πληθυσμιακών ομάδων και εδρεύουν σε νοσοκομεία. Κάθε κέντρο αίματος εκτελεί μοριακούς έλεγχους για τους σταθμούς αιμοδοσίας της περιοχής του (για περισσότερα βλ. παράρτημα Β) Η Ελλάδα διαθέτει 4 μεγάλα κέντρα αίματος:

1. Ε.ΚΕ.Α - Εθνικό κέντρο αιμοδοσίας.
2. Πανεπιστημιακό Νοσοκομείο ΑΧΕΠΑ Θεσσαλονίκης.
3. Πανεπιστημιακό Νοσοκομείο Ρίο Πατρών.
4. Πανεπιστημιακό Νοσοκομείο Βενιζέλειο Ηρακλείου Κρήτης.

Οι σταθμοί Αιμοδοσίας Α' Τάξης είναι μικρότερες υπηρεσίες και καλύπτουν τις ανάγκες του νοσοκομείου στο οποίο εδρεύουν και άλλες τοπικές ανάγκες. Οι Σταθμοί Αιμοδοσίας Β' Τάξης καλύπτουν αποκλειστικά τις ανάγκες του νοσοκομείου που στεγάζονται.

Το σύστημα αιμοδοσίας στην Ελλάδα είναι αποκεντρωμένο και αποτελείται από 101 υπηρεσίες αιμοδοσίας υπό την αιγίδα και εποπτεία του Υπουργείου Υγείας[121]. Κάθε υπηρεσία αιμοδοσίας αποτελεί ένα ενσωματωμένο μέρος ενός δημόσιου νοσοκομείου και οι αρμοδιότητές της περιλαμβάνουν α) τη στρατολόγηση νέων αιμοδοτών, β) τη συλλογή και τον έλεγχο του αίματος και γ) και τη διακίνηση του αίματος και των παραγώγων του στις νοσοκομειακές κλινικές [112].

Σύμφωνα με επίσημα δεδομένου του Υπουργείου Υγείας οι ανάγκες για αίμα το έτος 2014 στην Ελλάδα ήταν 750.000 μονάδες. Ένα σημαντικός παράγοντας ο οποίος αυξάνει σημαντικά τις ανάγκες σε αίμα στην χώρα μας και δικαιολογεί το παραπάνω νούμερο, είναι τα υψηλά ποσοστά μεσογειακής αναιμίας για τα οποία χρειάζονται 144.000 μονάδες αίματος ανά χρόνο.

Σύμφωνα με στοιχεία που έδωσε στη δημοσιότητα το Εθνικό Κέντρο Αιμοδοσίας (Ε.ΚΕ.Α) κατά την διάρκεια του έτους 2013 έγινε εθελοντική αιμοδοσία 584.088 μονάδων αίματος, εκ των οποίων 254.198 (43.52 %) προήλθε από τους επονομαζόμενους δότες αντικατάστασης (Replacement Donors - RDs), οι οποίοι αιμοδοτούν προκειμένου να καλύψουν τις ανάγκες που προκύπτουν από συγγενείς ή φίλους. Ένα ποσοστό 54.85 % δηλαδή 320.411 μονάδες προέρχονται από εθελοντές αιμοδότες (Volunteer Donors - VDs), οι οποίοι αιμοδοτούν με δική τους πρωτοβουλία καθαρά για ανθρωπιστικούς λόγους. Οι υπόλοιπες 9.479 μονάδες (0.016%) προέρχονται από τις ένοπλες δυνάμεις. Η τελευταία αυτή κατηγορία αιμοδοτών έχει δυνατά κίνητρα να αιμοδοτήσει εθελοντικά, καθώς αποζημιώνονται με άδειες και αποχή από τα καθήκοντά τους. Σε αυτό το σημείο κρίνεται σκόπιμο να αναφερθεί ότι τα τελευταία τρία χρόνια έχει παρατηρηθεί αύξηση στις εθελοντικές αιμοδοσίες. Συγκεκριμένα το 2013 παρασχεθήκανε από εθελοντές αιμοδότες (VDs) 21.234 περισσότερες μονάδες

αίματος σε σύγκριση με το 2012 [55]. Παρότι τα παραπάνω νούμερα είναι ενθαρρυντικά και παρότι διακρίνεται μια αυξητική τάση των εθελοντικών αιμοδοσιών, αξίζει να σημειωθεί ότι 24.000 μονάδες αίματος εισήχθησαν από την Ελβετία, προκειμένου να καλυφθούν οι εθνικές ανάγκες για αίμα [112].

Βάση επισήμων στοιχείων, η Ελλάδα διαθέτει έναν ευρύ κατάλογο αιμοδοτών βάσει του οποίου 6 αιμοδότες αντιστοιχούν σε 100 πολίτες, γεγονός που την κατατάσσει τρίτη ανάμεσα στις χώρες- μέλη της Ευρωπαϊκής Ένωσης όσον αφορά στον αριθμό των ατόμων οι οποίοι έχουν δωρίσει αίμα έστω και μία φορά στη ζωή τους (Eurobarometer, 2009). Επίσης, η Ελλάδα έρχεται πρώτη όσον αφορά στους μη αιμοδότες, οι οποίοι όμως έχουν σκεφτεί να δώσουν αίμα (Eurobarometer, 2005). Παρόλα τα στοιχεία αυτά όμως, η Ελλάδα που είναι μία χώρα 11.000.000 κατοίκων, πολύ συχνά βρίσκεται στη δυσάρεστη αλλά αναπόφευκτη θέση να εισάγει αίμα από το εξωτερικό, καθώς ο ετήσιος αριθμός μονάδων αίματος δεν επαρκεί για να καλυφθούν οι ανάγκες της χώρας. Αυτό συμβαίνει λόγω των υψηλών ποσοστών μεσογειακής αναιμίας, οι οποίες αναφέρθηκαν και παραπάνω, αλλά και λόγω του ότι οι ανάγκες αίματος κατά τη διάρκεια διάφορων χειρουργικών επεμβάσεων είναι μεγαλύτερη στην Ελλάδα από ότι σε άλλες χώρες της Κεντρικής και Βόρειας Ευρώπης, όπως προκύπτει από την έρευνα του ασφαλούς και καλού αίματος [71] και οι προσπάθειες να ελαχιστοποιηθεί αυτή η αλόγιστη χρήση έχουν αποβεί προς το παρόν άκαρπες.

Όπως γίνεται εμφανές από τα δεδομένα που παρουσιάσαμε παραπάνω, η Ελλάδα όπως και οι περισσότερες χώρες της Ευρώπης πρέπει να καθορίσουν μια στρατηγική με στόχο την αύξηση των τακτικών εθελοντών αιμοδοτών και κατ' επέκτασιν των μονάδων αίματος που συλλέγονται κάθε χρόνο. Η στρατηγική αυτή θα πρέπει να έχει ως κύριο άξονα την αύξηση των εθελοντών αιμοδοτών (VDs) και σταδιακή απογαλάκτιση από τους δότες αντικατάστασης (RDs) για τους λόγους που αναφέραμε στην παραπάνω ενότητα (κατηγορίες εθελοντών αιμοδοτών).

Η ευρύτατη εφαρμογή των μεταγγίσεων, σε συνάρτηση με τις δυσκολίες εξασφάλισης των απαιτούμενων ποσοτήτων αίματος για την κάλυψη των αναγκών, δημιουργούν ένα οξύ ιατρό-κοινωνικό πρόβλημα το οποίο απασχολεί τους υπεύθυνους φορείς υγείας ανά τον κόσμο. Η έλλειψη αίματος συνεπάγεται αναβολές χειρουργικών επεμβάσεων, παράταση της παραμονής των ασθενών στα νοσοκομεία, απώλεια εισοδημάτων από την επιβράδυνση της θεραπείας, καθώς και ευρύτερες ψυχολογικές και κοινωνικές επιπτώσεις, οι οποίες επιβαρύνουν τόσο τους ίδιους τους ασθενείς, όσο και το οικογενειακό τους περιβάλλον. Η αντιμετώπιση του προβλήματος απαιτεί την εφαρμογή Εθνικής Αιμοδοτικής Πολιτικής, η οποία στηριζόμενη σε αρχές μη κερδοσκοπικού μάρκετινγκ, θα αποσκοπεί κυρίως στην ενημέρωση και στην ευαισθητοποίηση του κοινού ως προς την εθελοντική αιμοδοσία [128][112].



Επομένως, η προσπάθεια του συστήματος αιμοδοσίας στην Ελλάδα πρέπει να έχει δύο βασικούς στόχους, οι οποίοι και ταυτίζονται με τους στόχους του συστήματος μας όπως περιγράφηκαν στο κεφάλαιο 1 της παρούσας διπλωματική εργασία.

1. Τη συνολική αύξηση των μονάδων αίματος που συλλέγονται για να διασφαλιστεί η αυτάρκεια στην παροχή αίματος
2. Τη μετατροπή των αιμοδοτών αντικατάστασης σε τακτικούς εθελοντές αιμοδότες, προκειμένου να αυξηθεί η ασφάλεια του αίματος και να διευκολυνθεί η διαχείριση των διαθέσιμων μονάδων αίματος και των παραγώγων του
3. Τη στρατολόγηση νέων εθελοντών αιμοδοτών με έμφαση σε άτομα τις νεαρής ηλικίας.

Τέλος, θα πρέπει να τονιστεί η επιτακτική ανάγκη για την πραγματοποίηση τόσο οργανωτικών, όσο και επιστημονικών αλλαγών στη χώρα μας στο εγγύς μέλλον, οι οποίες θα μπορέσουν να οδηγήσουν με γοργούς ρυθμούς, στην επίτευξη του πρωταρχικού στόχου, δηλαδή στην επίτευξη αυτάρκειας σε αίμα και παράγωγα αίματος. Μία επάρκεια, η οποία θα πρέπει να προέρχεται αποκλειστικά και μόνο από Εθελοντική Αιμοδοσία. Στην κατεύθυνση αυτή είναι και το ολοκληρωμένο πληροφοριακό σύστημα που μελετήθηκε και υλοποιήθηκε στην παρούσα διπλωματική εργασία.

## **2.2 Κινδυνoi που προκύπτουν μέσα από αιμοδοσία (ασθένειες, ποσοστά)**

Ο Παγκόσμιος Οργανισμός Υγείας (WHO) συνιστά την ακόλουθη ολοκληρωμένη στρατηγική με στόχο την προώθηση της ασφάλειας και της άμεσης πρόσβασης σε αίμα με ταυτόχρονη μείωση των κινδύνων που εγκυμονούν στις αιμοδοσίες:

1. Δημιουργία μίας εθνικής, άρτιας οργανωμένης υπηρεσίας αιμοδοσίας, η οποία θα μπορεί να εξασφαλίσει με αξιοπιστία και ασφάλεια τα απαραίτητα αποθέματα αίματος για τις ανάγκες του πληθυσμού.
2. Συλλογή αίματος μόνο από μη-αμειβόμενους εθελοντές αιμοδότες, οι οποίοι τηρούν όλες τις απαραίτητες προϋποθέσεις και φέρουν χαμηλό κίνδυνο απόκτησης επίφοβων λοιμώξεων.
3. Πλήρεις και ποιοτικές εξετάσεις όλων των αιμοδοσιών για μεταδοτικές μολύνσεις (όπως HIV, ηπατίτιδα Β, ηπατίτιδα C, σύφιλη και άλλες λοιμώξεις) καθώς και εξετάσεις για τις ομάδες αίματος και τη συμβατότητα τους.

4. Μείωση των άσκοπων μεταγγίσεων με κατάλληλη κλινική χρήση του αίματος και με διασφάλιση της ασφαλούς χορήγησης τόσο του αίματος και όσο και των παραγώγων του [24].

Οι υπηρεσίες Αιμοδοσίας έχουν καθήκον αφενός μεν την φροντίδα για την ασφάλεια του αιμοδότη, αφετέρου δε την εγγύηση του αίματος και των προϊόντων του αίματος που προορίζονται για μετάγγιση. Για τον σκοπό αυτό έχει δημιουργηθεί και αναπτυχθεί η διαδικασία της αιμοεπαγρύπνησης (haemovigilance).

### 2.2.1 Haemovigilance

Με τον όρο αιμοεπαγρύπνηση ορίζουμε ως ένα σύνολο οργανωμένων διαδικασιών επιτήρησης, που σχετίζονται με τα ανεπιθύμητα και μη αναμενόμενα συμβάντα και αντιδράσεις στους δότες και τους λήπτες των προϊόντων του αίματος και με την επιδημιολογική παρακολούθηση των αιμοδοτών [45]. Η αιμοεπαγρύπνηση εστιάζει στις επιπλοκές των δωρητών αίματος και στις ανεπιθύμητες αντιδράσεις των ληπτών και συμπεριλαμβάνει όλες τις επιπλοκές της “γραμμής παραγωγής αίματος, την αναλυτική και αναδρομική καταγραφή των συμβαμάτων και την προοπτική έγκαιρης προειδοποίησης με χρήση ενός συστήματος ταχείας έγερσης συναγεμίων. Η αιμοεπαγρύπνηση είναι ένα ισχυρό εργαλείο το οποίο στοχεύει στην βελτίωση της ποιότητας των διαδικασιών μετάγγισης αίματος, δίνοντας προτεραιότητα στην ασφάλεια. Απώτερος στόχος της αιμοεπαγρύπνησης είναι η πρόληψη ανεπιθύμητων συμβαμάτων και αντιδράσεων. Εισαγάγει μεθόδους εντοπισμού σφαλμάτων και αντιδράσεων και συμπεριλαμβάνει συστήματα συναγεμιού, συστήματα ιχνηλασιμότητας, συστήματα ειδοποιήσεων και τακτικούς ελέγχους των πρακτικών. Οι ουσιαστικοί στόχοι της διαδικασίας της αιμοεπαγρύπνησης είναι οι εξής:

- Να προληφθεί η επανεμφάνιση ανεπιθύμητων συμβαμάτων και αντιδράσεων
- Η βελτίωση της πρακτικής μετάγγισης στα νοσοκομεία με τη λήψη προληπτικών ή διορθωτικών μέτρων όπου επιβάλλεται
- Η ανάπτυξη εθνικών κατευθυντήριων οδηγιών και νοσοκομειακών πρωτοκόλλων
- Η διαμόρφωση αποφάσεων σε εθνικό επίπεδο που αφορούν στην ασφάλεια των μεταγγίσεων
- Η εκπαίδευση των γιατρών που χρησιμοποιούν το αίμα, καθώς και των ασθενών που μεταγγίζονται

Απαραίτητη προϋπόθεση για την αποτελεσματική λειτουργία της αιμοεπαγρύπνησης είναι η ύπαρξη συστήματος ιχνηλασιμότητας. Ιχνηλασιμότητα ή ανιχνευσιμότητα (traceability) ορίζεται η ικανότητα

πλήρους εντοπισμού καθεμίας μονάδας αίματος ή παραγώγων της από τον δότη μέχρι τον τελικό αποδέκτη. Επιτυγχάνεται με ακριβείς και πλήρεις διαδικασίες αναγνώρισης κάθε δότη, κάθε συλλεγόμενης μονάδας αίματος, καθώς και όλων των παραγώγων του, τήρησης αρχείων και μέσω πλήρους επαλήθευσης της παροχής και μετάγγισης αίματος. Με καταγραφή και μελέτη των απαραίτητων δεδομένων για την ιχνηλασιμότητα παράγουμε στατιστικά στοιχεία τα οποία αφορούν στο σύνολο των ασθενών που έχουν μεταγγισθεί, στις μονάδες ή στα παραγώγα αίματος που έχουν χρησιμοποιηθεί, στους εθελοντές αιμοδότες που έχουν δώσει τις μονάδες αίματος και στα προϊόντα αίματος που μεταγγίσθηκαν. Τα ανεπιθύμητα συμβάματα τα οποία μπορούν να προκύψουν χωρίζονται στις εξής βασικές κατηγορίες:

- Σοβαρή ανεπιθύμητη αντίδραση (Serious Adverse Reaction, SAR) ”μια άνευ προθέσεως αντίδραση του δότη ή του ασθενούς που σχετίζεται με τη συλλογή ή τη μετάγγιση αίματος ή παραγώγων του και η οποία είναι θανατηφόρα, απειλητική για τη ζωή, προκαλεί αναπηρία ή ανικανότητα ή έχει ως αποτέλεσμα ή παρατείνει τη νοσηλεία ή τη νοσηρότητα”, (Οδηγία 2002/98/EK)
- Σοβαρό ανεπιθύμητο συμβάν (Serious Adverse Event, SAE) ”κάθε ατυχές περιστατικό που σχετίζεται με τη συλλογή, τον έλεγχο, την επεξεργασία, την αποθήκευση και τη διανομή αίματος ή παραγώγου του, που θα μπορούσε να προκαλέσει το θάνατο, να απειλήσει τη ζωή, ή να προκαλέσει αναπηρία ή ανικανότητα ή να έχει ως αποτέλεσμα ή να παρατείνει τη νοσηλεία ή τη νοσηρότητα”, (Οδηγία 2002/98/EK)
- Παρ’ ολίγον συμβάματα (“near miss” events) ”σφάλματα που αν δεν ανιχνευθούν θα μπορούσαν να οδηγήσουν σε λανθασμένο προσδιορισμό ομάδας αίματος, αποτυχία ανίχνευσης ενός ερυθροκυτταρικού αντισώματος, ή σε διανομή, συλλογή ή χορήγηση εσφαλμένου ή ακατάλληλου προϊόντος αίματος”, (Οδηγός, Συμβούλιο της Ευρώπης, Έκδοση 14)
- Σφάλματα των μεταγγίσεων χωρίς συμβάματα (uneventful transfusion errors) ”η μετάγγιση οποιουδήποτε εσφαλμένου, ακατάλληλου ή μη ενδεικνυόμενου προϊόντος αίματος που δεν προκαλεί βλάβη στον λήπτη” (Οδηγός, Συμβούλιο της Ευρώπης, Έκδοση 14η)

Κάθε σοβαρό ανεπιθύμητο συμβάν θα πρέπει να περιγράφεται αναλόγως. Για τον σκοπό αυτόν, έχει καθιερωθεί μία διεθνώς αποδεκτή κλίμακα ταξινόμησης. Η εύρεση της αιτιακής συσχέτισης μεταξύ ανεπιθύμητης αντίδρασης και μετάγγισης, έχει πολύ μεγάλη σημασία, καθώς έτσι μπορούμε να αναγνωρίσουμε αν ευθύνεται κάποιο συστατικό του αίματος ή όχι. Υπάρχουν πολλοί διαφορετικοί τύποι αντιδράσεων μετάγγισης. Η πιο κοινή υποδιαίρεση που χρησιμοποιείται είναι η ταξινόμηση με βάση την εμφάνιση των αντιδράσεων σε οξείες (λιγότερο από 24 ώρες μετά) και καθυστερημένες (περισσότερο από 24 ώρες μετά την μετάγγιση) αντιδράσεις. Πέραν από τους παραλήπτες αίματος,

ανεπιθύμητες καταστάσεις μπορούν να προκύψουν και στους εθελοντές αιμοδότες. Τα αίτια και οι παράγοντες, που προκαλούν επιπλοκές στους αιμοδότες, διαφέρουν από τα αντίστοιχα των αιμοληπτών, και χωρίζονται σε τοπικές αντιδράσεις (κυρίως λόγω της εισαγωγής βελόνας), σε γενικότερες αντιδράσεις και σε διάφορους άλλους τύπους επιπλοκών. Οι επιπλοκές ταξινομούνται σε μία διαφορετική διεθνή κλίμακα, έτσι ώστε να μπορούν να γίνουν συγκρίσεις και τίθενται τα σημεία στα οποία πρέπει να γίνουν παρεμβάσεις.

Όλες οι υπηρεσίες αιμοδοσίας και τα νοσοκομεία, υποχρεώνονται να κοινοποιούν στις αρμόδιες αρχές με κατάλληλη διαδικασία αναφοράς τις σοβαρές ανεπιθύμητες αντιδράσεις και τα συμβλήματα τα οποία σχετίζονται με τη μετάγγιση αίματος. Οι αναφορές αυτές περιλαμβάνουν πληροφορίες για την κλινική έκβαση των αντιδράσεων και τα μέτρα που έλαβαν οι υπεύθυνοι, σε σχέση με τα άλλα παράγωγα του αίματος που μεταγγίστηκαν, καθώς και τον προσδιορισμό των σοβαρών συμβάντων εξαιτίας ελαττωματικού προϊόντος, εξοπλισμού, ανθρώπινου λάθους και άλλων προβλημάτων [160]. Οι μη σοβαρές αντιδράσεις καθώς και άλλα λάθη τα οποία συμβαίνουν κατά την μετάγγιση αίματος δεν αποτελούν αντικείμενο για αναφορά σύμφωνα με τις οδηγίες της Ευρωπαϊκής Ένωσης.

Για την πλήρη και ορθή γνωστοποίηση όλων των άνωθεν πληροφοριών είναι απαραίτητη η ανάπτυξη ολοκληρωμένων πληροφοριακών συστημάτων αιμοεπαγρύπνησης. Για την εφαρμογή ενός συστήματος αιμοεπαγρύπνησης πρέπει να δημιουργούνται λειτουργικοί σύνδεσμοι μεταξύ των Κλινικών τμημάτων, των Κέντρων Αιμοδοσίας, των νοσοκομειακών υπηρεσιών Αιμοδοσίας και των αρμόδιων εθνικών αρχών. Τα αποτελέσματα της ανάλυσης των δεδομένων που προκύπτουν από τα συστήματα αιμοεπαγρύπνησης πρέπει να ανατροφοδοτούνται περιοδικά σε εκείνους, που παρείχαν τα πρωτογενή δεδομένα και στις αρμόδιες αρχές.

### **2.2.2 Υπόσυστηματα Haemovigilance Ελλάδα**

Στην Ελλάδα το αρμόδιο όργανο για την αιμοεπαγρύπνηση είναι το Συντονιστικό Κέντρο Αιμοεπαγρύπνησης (ΣΚΑΕ), το οποίο έχει ιδρυθεί από το Ελληνικό Κέντρο Ελέγχου και Πρόληψης Νοσημάτων (ΚΕ.ΕΛ.Π.ΝΟ), το 1995. Το ΣΚΑΕ απαρτίζεται από εξειδικευμένο προσωπικό στα θέματα ιατρικής, δημόσιας υγείας, εργαστηριακών ελέγχων, στατιστικών αναλύσεων, εκπαίδευσης και οργάνωσης. Η ευρωπαϊκή προσπάθεια για την προώθηση αυστηρών προτύπων για την ποιότητα στην αιμοδοσία και στις μεταγγίσεις αίματος, και ιδίως για τον περιορισμό των κινδύνων που προκύπτουν σε αυτόν τον τομέα της δημόσιας υγείας, οδήγησαν στην ίδρυση του ΣΚΑΕ, το οποίο αποτελεί ιδρυτικό μέλος του Διεθνούς Δικτύου Αιμοεπαγρύπνησης. Η συλλογή των προς ανάλυση δεδομένων γίνεται με τη δημιουργία δικτύων σε τρία επίπεδα: τοπικό δίκτυο (επίπεδο νοσοκομείων, κέντρων αιμοδοσίας),

περιφερειακό δίκτυο (το οποίο έχει έξι βάσεις: στη Νότια Ελλάδα, στην Ανατολική Ελλάδα, Βόρεια Ελλάδα, Δυτική Ελλάδα, Θεσσαλία και την Κρήτη ) και στο εθνικό δίκτυο (ΣΚΑΕ). Το τελευταίο, κοινοποιεί τα αποτελέσματα στο ΚΕ.ΕΛ.Π.ΝΟ και στο Εθνικό Κέντρο Αιμοδοσίας( Ε.Κ.Ε.Α). Στο σχήμα 2.6 βλέπουμε το διάγραμμα οργάνωσης του συνολικού δικτύου.



**Σχήμα 2.6:** Διάγραμμα λειτουργίας

Οι βασικές λειτουργίες του ΣΚΑΕ είναι οι εξής:

- Η συνεχής επιδημιολογική επιτήρηση των λοιμώξεων που μεταδίδονται με το αίμα. Αυτό επιτυγχάνεται με χρήση συστημάτων ανιχνευσιμότητας σε όλα τα στάδια της αιματικής αλυσίδας. Σε περίπτωση εντοπισμού πιθανών μολυσματικών μονάδων, ακολουθεί η άμεση απόσυρση τους. Πραγματοποιούνται διαρκώς αναδρομικοί έλεγχοι, με σκοπό να μειωθούν στο ελάχιστο τα πιθανά λάθη.
- Η επιδημιολογική επιτήρηση ανεπιθύμητων αντιδράσεων και συμβάντων σχετικών με τη μετάγγιση αίματος. Οι παρατηρήσεις αυτές ακολουθούνται από αναφορές στις αρμόδιες αρχές και διαμόρφωση προτάσεων με αποτελεσματικά διορθωτικά μέτρα, τις οποίες ακολουθούν τα νοσοκομεία και τα κέντρα αιμοδοσίας.
- Η επαγρύπνηση για ανεπιθύμητες αντιδράσεις, ατυχήματα, βλάβες και επιπλοκές στην διάρκεια και μετά από την αιμοδοσία. Άμεση προϋπόθεση του συστήματος επαγρύπνησης είναι η ύπαρξη συστημάτων άμεσης ετοιμότητας εγέρσεων συναγερμών και η αποτελεσματική και αξιόπιστη δυνατότητα διαχείρισης κρίσεων. Το ΣΚΑΕ ενημερώνει τακτικά την ιατρική κοινότητα για τις εξελίξεις με δημοσιεύσεις και εκδόσεις.

Το ΣΚΑΕ έχει αναπτύξει πλήρεις και αποτελεσματικές μεθόδους εργασίας, με βάση τις κατευθυντήριες οδηγίες της Ευρωπαϊκής Ένωσης και του Διεθνούς Οργανισμού υγείας(WHO).

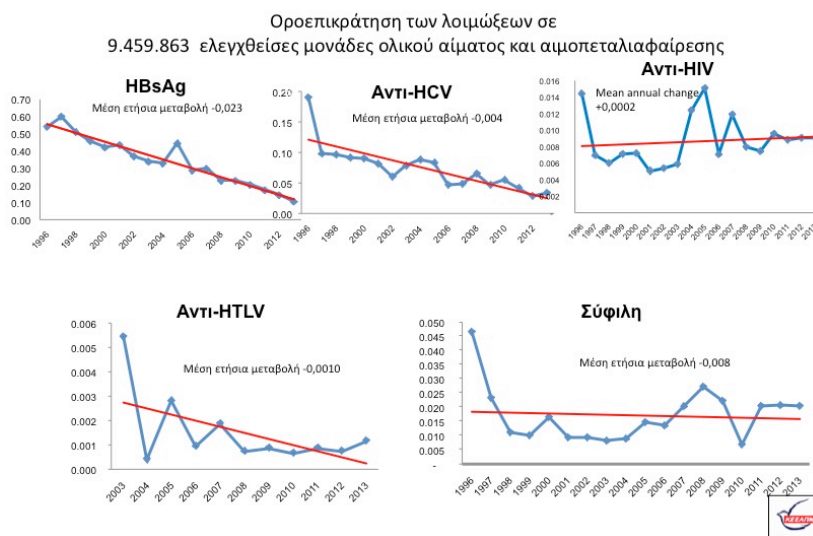
Όσον αφορά στις λοιμώξεις που μεταδίδονται με το αίμα εφαρμόζεται αναλυτική καταγραφή των οροθετικών αιμοδοτών για HIV, HBV, HCV, σύφιλη, HTLV καθώς και ανάλυση των επιδημιολογικών δεδομένων σε σχέση με τις μονάδες αίματος, την Υπηρεσία Αιμοδοσίας, τα δημογραφικά χαρακτηριστικά των αιμοδοτών (φύλο, ηλικία, τόπος διαμονής, οικονομικό και μορφωτικό επίπεδο), την κατηγορία αιμοδοτών (εθελοντές, συγγενείς, στρατιώτες) και την αιμοδοτική συχνότητα (αιμοδότες πρώτης φοράς, σποραδικοί και τακτικοί). Επιπλέον καταγράφονται αναλυτικά τα δεδομένα για ποιοτικούς ελέγχους, πιστοποιήσεις ποιότητας, δείκτες συλλογής και ελέγχου του αίματος ανά Υπηρεσία Αιμοδοσίας και διαχείρισης ανθρώπινου δυναμικού. Με βάση τους εγγεγραμμένους αιμοδότες πραγματοποιείται συνεχής παρακολούθηση και χαρτογράφηση των ρετροϊκών λοιμώξεων και των ηπατιτίδων στον αιμοδοτικό πληθυσμό. Οι μέθοδοι ορολογικού και μοριακού ελέγχου του αίματος για λοιμώξεις διευρύνονται διαρκώς. Πραγματοποιείται πλήρης καταγραφή δεδομένων για ποιοτικό έλεγχο, πιστοποίηση ποιότητας, δείκτες συλλογής και ελέγχου του αίματος ανά Υπηρεσία Αιμοδοσίας και διαχείρισης ανθρώπινου δυναμικού. Το ΣΚΑΕ έχει καθιερώσει πρωτόκολλα ανιχνευσιμότητας για λοιμώξεις που αναφέρονται ύστερα από μετάγγιση αίματος καθώς και πρωτόκολλα για αναδρομικό έλεγχο των ληπτών δυνητικά μολυσμένου αίματος, τα οποία είναι καθολικά και χρησιμοποιούνται σε όλες τις υπηρεσίες υγείας. Τέλος διεξάγονται διαρκώς μελέτες κόστους – ωφέλειας για τα μέτρα πρόληψης της διασποράς λοιμωδών νόσων με το αίμα και τα προϊόντα του αίματος καθώς και έρευνα πάνω στις νέες λοιμώξεις (Ηπατίτιδα Ε κλπ).

Στον τομέα των ανεπιθύμητων αντιδράσεων και των ανεπιθύμητων συμβάντων κατά και μετά την αιμοδοσία καταγράφονται αναλυτικά όλες οι ανεπιθύμητες αντιδράσεις και τα συμβάντα που προκύπτουν κατά την αιμοδοσία εθελοντών και με βάση αυτά τα καταγεγραμμένα γεγονότα αναλύονται οι πληροφορίες που προκύπτουν σχετικά με τον τύπο και τη σοβαρότητα των αντιδράσεων/ συμβαμάτων. Με βάση τις αναλύσεις αυτές προκύπτουν κάποια συμπεράσματα και προτάσεις για βελτίωση της ποιότητας των αιμοδοσιών και δίνονται κατευθυντήριες αρχές στα νοσοκομεία και στα αιμοδοτικά κέντρα.

Οι μέθοδοι εργασίας αναφορικά με τις ανεπιθύμητες αντιδράσεις και τα ανεπιθύμητα συμβάντα σχετικά με τη μετάγγιση αίματος και προϊόντων αίματος περιλαμβάνουν σε πρώτο στάδιο την καταγραφή ανεπιθυμητών αντιδράσεων, που σχετίζονται με λοιμογόνους παράγοντες (ιογενείς, βακτηριακοί, παρασιτικοί παράγοντες) καθώς και αυτών που σχετίζονται με μη λοιμογόνους παράγοντες, ανεξάρτητα από την σοβαρότητα των αντιδράσεων. Επιπλέον καταγράφονται τα ανεπιθύμητα σοβαρά και «παρ' ολίγον» συμβάματα και τα σφάλματα των μεταγγίσεων χωρίς σύμβαμα», που μπορεί να

επηρεάσουν την ασφάλεια και την ποιότητα του μεταγγιζόμενου προϊόντος αίματος, σε όλες τις διαδικασίες της συλλογής, του ελέγχου, της επεξεργασίας, της αποθήκευσης και της διανομής προϊόντων αίματος. Το ΣΚΑΕ διαμορφώνει τα δελτία αναφοράς των ανεπιθύμητων αντιδράσεων και ανεπιθύμητων συμβάντων και τις οδηγίες διερεύνησης και αντιμετώπισης τους, σύμφωνα με τις Ευρωπαϊκές Οδηγίες και τις συστάσεις του Διεθνούς Δικτύου Αιμοεπαγρύπνησης. Οι πληροφορίες σχετικά με τις ανεπιθύμητες αντιδράσεις και τα ανεπιθύμητα συμβάντα αναλύονται διεξοδικά ανάλογα με τον τύπο της αντίδρασης, τη συσχέτιση με τη μετάγγιση, τη σοβαρότητα και την έκβαση της αντίδρασης και το προϊόν αίματος (ερυθρά, πλάσμα, αιμοπετάλια). Τέλος το ΣΚΑΕ συλλέγει, αναλύει και καταγράφει την ποιότητα των ελληνικών αναφορών για τις μεταδιδόμενες μέσω μετάγγισης λοιμώξεις συγκριτικά με τις απαιτήσεις αιμοεπαγρύπνησης και τις συστάσεις των διεθνών οργανισμών.

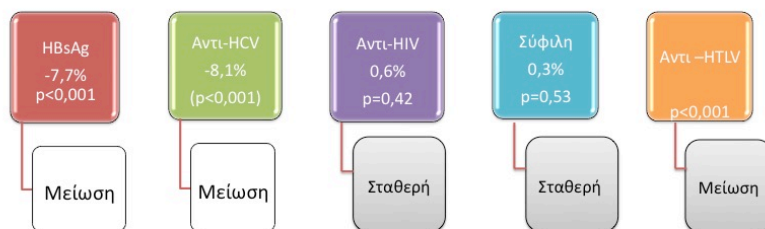
Στην συνέχεια παρουσιάζονται στοιχεία τα οποία έχουν προκύψει από μακροχρόνιες μελέτες του ΣΚΑΕ και αφορούν τις ελληνικές αιμοδοσίες [158]. Στο σχήμα 2.7 παρουσιάζονται τα αποτελέσματα της οροεπικράτησης των λοιμώξεων που αφορούν ελεγχθείσες μονάδες ολικού αίματος και αφαίρεσης αιμοπεταλίων το διάστημα των ετών 1996-2013. Στο σχήμα 2.8 παρουσιάζεται η μέση ετήσια μεταβολή ορολογικών δεικτών κατά το διάστημα των ετών 2003-2013. Η κατανομή και η ανάλυση



**Σχήμα 2.7:** Οροεπικράτηση λοιμώξεων σε 9.459.863 ελεγχθείσες μονάδες αίματος και αφαίρεσης αιμοπεταλίων.

των ανεπιθύμητων αντιδράσεων σχετικά με τη μετάγγιση ασθενών το έτος 2013 φαίνονται στο 2.9. Το έτος 2013 δηλώθηκαν, μελετήθηκαν και παρουσιάσθηκαν 1092 ανεπιθύμητες αντιδράσεις κατά τη μετάγγιση 729456 προϊόντων αίματος. Η συχνότητα των σοβαρών αντιδράσεων κατά τη μετάγγιση είναι 1/9994 προϊόντα. Κατά τα έτη 1997-2013, δηλώθηκαν επτά θάνατοι, με μεταγγίσεις 8972722 προϊόντων αίματος, συχνότητα 1/281817 προϊόντα με κύρια αίτια: ABO ασυμβατότητα, TRALI, βα-

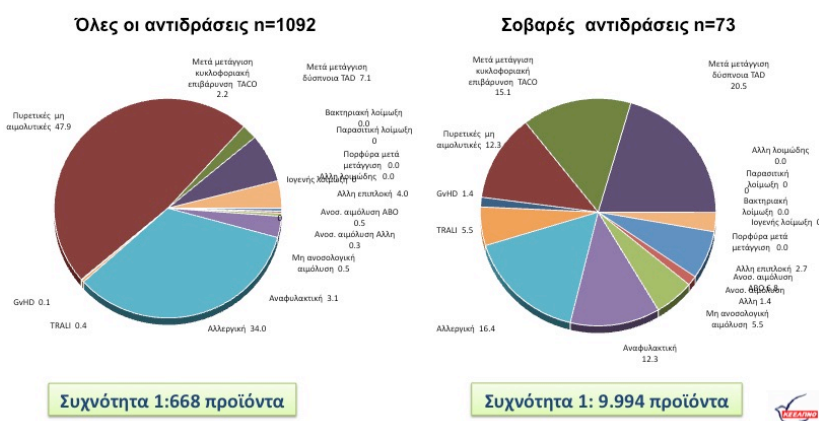
Λοιμώξεις που μεταδίδονται με το αίμα  
Μέση ετήσια μεταβολή ορολογικών δεικτών, 2003-2013



Σχήμα 2.8: Μέση μεταβολή των ορολογικών δεικτών των λοιμώξεων που μεταδίδονται με το αίμα.

κτηριακή επιμόλυνση. Στο σχήμα 2.10 παρουσιάζονται διαγραμματικά οι ανεπιθύμητες αντιδράσεις

Κατανομή αντιδράσεων 2013

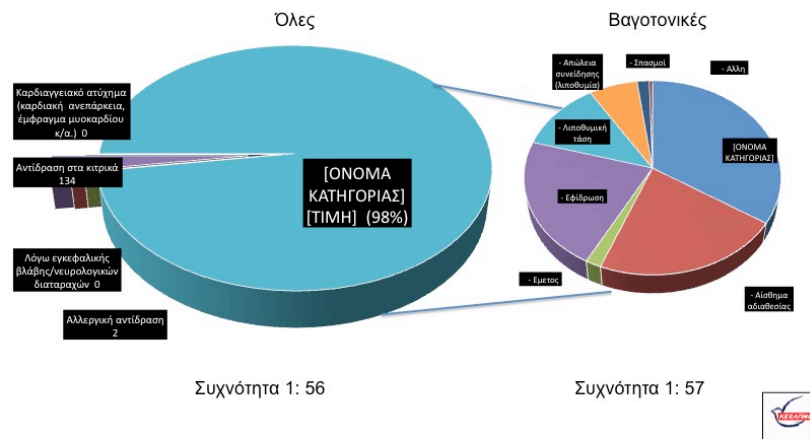


Σχήμα 2.9: Κατανομή αντιδράσεων.

σχετικές με την αιμοδοσία. Πιο συγκεκριμένα δηλώθηκαν 148 ανεπιθύμητες αντιδράσεις που αφορούσαν σε σύνολο μεταγίσεων 44611 μονάδων ερυθρών αιμοσφαιρίων. Καταγράφηκαν (σε ποσοστά επί τοις εκατό) 0.22 πυρετικές, 0.13 αλλεργικές, 0.13 αλλοανοσοποίηση και 0.6 άλλης αιτιολογίας αντίδρασης. Τέλος, στο σχήμα 2.11 παρουσιάζεται ποσοστιαία ο αριθμός των συμβάντων που έλαβαν χώρα πριν και κατά τη διάρκεια της αιμοδοσίας.

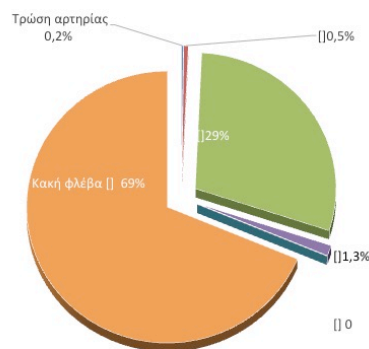


Ανεπιθύμητες αντιδράσεις σχετικά με την αιμοδοσία 2013  
n= 6020



Σχήμα 2.10: Ανεπιθύμητες αντιδράσεις σχετικά με την αιμοδοσία 2013.

Συμβάντα κατά και μετά την αιμοδοσία, 2013  
n= 960 Συχνότητα 1:352 δότες



Σχήμα 2.11: Συμβάντα κατά και μετά την αιμοδοσία, 2013.



## Κεφάλαιο 3

# Ανάλυση και σχεδιασμός του συστήματος

### 3.1 Λειτουργικές απαιτήσεις

Οι απαιτήσεις ενός συστήματος αποτελούν τη βάση των συστημάτων λογισμικού. Ουσιαστικά, είναι η περιγραφή των υπηρεσιών που πρέπει να παρέχει ένα σύστημα λογισμικού και οι περιορισμοί υπό τους οποίους πρέπει αυτές να λειτουργούν. Διακρίνονται σε δύο βασικές κατηγορίες τις λειτουργικές απαιτήσεις (functional requirements) και τις μη λειτουργικές απαιτήσεις (non functional requirements).

Οι λειτουργικές απαιτήσεις προσδιορίζουν τα βασικά χαρακτηριστικά του συστήματος και την λειτουργία του από την άποψη του ίδιου του προϊόντος και των χρηστών του. Στον όρο λειτουργία του συστήματος συμπεριλαμβάνεται οι συνολικές είσοδοι, η συμπεριφορά και οι έξοδοι του συστήματος. Στο σύστημα που υλοποιούμε, οι λειτουργικές απαιτήσεις χρησιμοποιούνται για να περιγράψουν ειδικές λειτουργίες που καθορίζουν αυτά τα οποία αναμένουμε να πετύχει το σύστημα. Αν και αναφέρονται ως "απαιτήσεις", στην πραγματικότητα αποτελούν μια μορφή σχεδιασμού, υψηλού επιπέδου. Οι λειτουργικές απαιτήσεις συχνά προσδιορίζονται ως λειτουργικές προδιαγραφές,» και ο όρος περιγραφή αποτελεί συνώνυμο του σχεδιασμού. Οι λειτουργικές απαιτήσεις είναι ουσιαστικά το "τι" λειτουργία θέλουμε να επιτελεί το σύστημα, χωρίς να παρέχουμε καμία πληροφορία για το "πώς".

Οι μη λειτουργικές απαιτήσεις είναι οι απαιτήσεις που καθορίζουν τα κριτήρια που χρησιμοποιούνται για να κρίνουν την λειτουργία του συστήματος και κατά επέκταση αν είναι επιτυχημένο ή όχι, για αυτό και συχνά αποκαλούνται και ιδιότητες του. Επεξεργάζονται τα χαρακτηριστικά απόδοσης του, και μπορούν επίσης να περιγράψουν τις πτυχές του συστήματος που δεν σχετίζονται με την εκτέλεση του, αλλά με την εξέλιξη του στο πέρασμα του χρόνου. Οι βασικές συνήθεις κατηγορίες των μη λειτουργικών απαιτήσεων σχετίζονται με την ασφάλεια( ασφαλή πρόσβαση σε δεδομένα αλλά και σε hardware), την απόδοση ( ορίζουν χαρακτηριστικά που έχουν να κάνουν με την ταχύτητα και την

ανταπόκριση του συστήματος) και την επεκτασιμότητα (σχετίζονται με το μέγεθος του συστήματος). Γενικότερα, οι μη λειτουργικές απαιτήσεις καθορίζουν το πώς πρέπει να είναι ένα σύστημα, σε αντίθεση με τις λειτουργικές απαιτήσεις που καθορίζουν τι πρέπει να κάνει ένα σύστημα. Οι λειτουργικές απαιτήσεις ανήκουν στο κομμάτι σχεδιασμού και ανάλυσης του συστήματος ενώ οι μη λειτουργικές απαιτήσεις ανήκουν στο κομμάτι της αρχιτεκτονικής του συστήματος.

### 3.1.1 Λειτουργίες εφαρμογής (web, mobile)

Οι χρήστες του συστήματος χωρίζονται σε τρεις μεγάλες ομάδες χρηστών.

1. Εθελοντές αιμοδότες
2. Εγκεκριμένοι χρήστες/διαχειριστές των συστημάτων αιμοδοσίας
3. Υπέρ-Διαχειριστές του συστήματος

Οι εθελοντές αιμοδότες αποτελούν την βασική κατηγορία χρηστών καθώς πραγματοποιούν συνεχή και εκτεταμένη χρήση των υπηρεσιών του συστήματος, τόσο στο επίπεδο της εφαρμογής έξυπνου κινητού όσο και μέσω του web portal. Οι εγκεκριμένοι χρήστες/διαχειριστές των συστημάτων αιμοδοσίας χρησιμοποιούν αποκλειστικά το web portal το οποίο βρίσκεται στο υπολογιστικό νέφος για ολοκληρωμένη διαχείριση ενός πλήρους κύκλου αιμοδοσίας. Οι λειτουργίες που μπορεί να επιτελέσει κάθε ομάδα χρηστών καθώς και οι λειτουργικές απαιτήσεις του συστήματος αναλύονται με λεπτομέρειες στην συνέχεια.

#### Mobile

Επιγραμματικά, οι λειτουργικές απαιτήσεις της εφαρμογής έξυπνου κινητού είναι οι εξής:

- Δυνατότητα εγγραφής στο σύστημα ως εθελοντής αιμοδότης με χρήση του αριθμού μητρώου κοινωνικής ασφάλισης (Α.Μ.Κ.Α.).
- Δυνατότητα σύνδεσης στο σύστημα αν έχει πραγματοποιήσει ήδη εγγραφή στο παρελθόν.
- Δυνατότητα προβολής γεγονότων αιμοδοσίας.
- Δυνατότητα λήψης ειδοποιήσεων (push notifications) όταν δημιουργείται κάποιο νέο γεγονός αιμοδοσίας για το οποίο ο εθελοντής αιμοδότης πληρεί τα απαραίτητα κριτήρια.

- Δυνατότητα αποδοχής, απόρριψης ή αναβολής της ειδοποίησης για το γεγονός αιμοδοσίας.
- Δυνατότητα διαχείρισης ραντεβού για αιμοδοσία (δημιουργία νέου ραντεβού, μετάθεση , προβολή, ακύρωση).
- Δυνατότητα προσθήκης και συγχρονισμού των ραντεβού αιμοδοσίας στο ημερολόγιό του χρήστη.
- Δυνατότητα ενημέρωσης σχετικά με το υπολειπόμενο διάστημα που πρέπει να παρέλθει ώστε ο εθελοντής να είναι σε θέση να συμμετέχει σε μία νέα αιμοδοσία.
- Δυνατότητα να κερδίζει πόντους και εμβλήματα βάση των αιμοδοσιών του (gamification system) και να τα μοιραστεί στα μέσα κοινωνικής δικτύωσης.
- Δυνατότητα να μοιραστεί και να δημοσιεύσει στοιχεία στα μέσα κοινωνικής δικτύωσης σε κάθε στάδιο της αιμοδοσίας.
- Δυνατότητα δημιουργίας έκκλησης για αίμα σε περίπτωση ανάγκης ατόμων του στενού κύκλου χρήστη και δημοσίευσής του στα μέσα κοινωνικής δικτύωσης.
- Δυνατότητα διαχείρισης προφίλ χρήστη.

## Web Portal

Το web portal που βρίσκεται στο υπολογιστικό νέφος χρησιμοποιείται από όλες τις ομάδες χρηστών. Όσον αφορά στους εθελοντές αιμοδότες ισχύουν οι ίδιες λειτουργικές απαιτήσεις όπως στην εφαρμογή του έξυπνου κινητού οι οποίες και αναλύθηκαν παραπάνω στο 3.1.1. Όσον αφορά στους χρήστες/διαχειριστές των κέντρων αιμοδοσίας έχουμε τις παρακάτω λειτουργικές απαιτήσεις:

- Δημιουργία νέων γεγονότων αιμοδοσίας.
- Προβολή στατιστικών ανά έτος σχετικά με τις ειδοποιήσεις που στάλθηκαν από το συγκεκριμένο κέντρο αιμοδοσίας και την ανταπόκριση των χρηστών (αποδοχή, απόρριψη, άνοιγμα ειδοποίησης, ολοκλήρωση αιμοδοσίας).
- Διαχείριση των ραντεβού για αιμοληψία.
- Διαχείριση του προφίλ του κέντρου αιμοδοσίας.
- Δυνατότητα εγγραφής στο σύστημα νέου εθελοντή αιμοδότη.

- Δυνατότητα καταχώρησης της ολοκλήρωσης της αιμοδοσίας.
- Δυνατότητα ελέγχου τήρησης προϋποθέσεων καταλληλότητας εθελοντή αιμοδότη.

Όσον αφορά στους υπέρ-διαχειριστές του συστήματος έχουμε τις παρακάτω λειτουργικές απαιτήσεις:

- Δυνατότητα διαχείρισης λογαριασμών χρηστών.
- Προβολή στατιστικών για τους χρήστες του συστήματος.

## 3.2 Σενάρια χρήσης

Η διαδικασία της μοντελοποίησης, δηλαδή ο αναλυτικός σχεδιασμός ενός συστήματος, πριν από την υλοποίηση του είναι ένα πολύ βασικό στάδιο κατά την ανάπτυξη εφαρμογών. Ένα σύστημα μοντελοποιείται επιτυχώς όταν οι λειτουργικότητες του συστήματος έχουν περιγραφεί πλήρως και ορθώς, έχουν καλυφθεί οι ανάγκες των τελικών χρηστών και υποστηρίζεται η επεκτασιμότητα του συστήματος. Η ενοποιημένη Γλώσσα Μοντελοποίησης (UML) είναι μια τυπική γλώσσα για τη σύνταξη λεπτομερών σχεδίων λογισμικού. Η UML μπορεί να χρησιμοποιηθεί για να απεικονίσει, να προσδιορίζει, να κατασκευάσει και να τεκμηριώσει τα προϊόντα του συστήματος εντάσεως λογισμικού.[25]. Δεν αποτελεί απλώς έναν κατάλογο από διαγράμματα, αλλά είναι μια γλώσσα αναπαράστασης γνώσης. Αυτό συνεπάγεται ότι κάθε στοιχείο διαγράμματος, π.χ. κουτί, βέλος, κύκλος, ορθογώνιο κ.λ.π., υποστηρίζεται από συγκεκριμένους κανόνες σύνταξης και σημασιολογία. Επιλέξαμε να μοντελοποιήσουμε το σύστημα μας με χρήση της γλώσσας UML για τους εξής λόγους:

- Είναι ένα βιομηχανικό πρότυπο από τον διεθνή, ανοιχτό και μη κερδοσκοπικό οργανισμό εταιριών Object Management Group (OMG).
- Η UML είναι χτισμένη πάνω σε θεμελιώσεις αντικειμενοστραφείς έννοιες, όπως η κλάση, με αποτέλεσμα να υποστηρίζει καλύτερα την ανάπτυξη και τον σχεδιασμό του αντικειμενοστραφούς λογισμικού.
- Δίνει έμφαση στην ανάλυση των σεναρίων χρήσης, το οποίο είναι ένα πολύτιμο εργαλείο για την παρακολούθηση των σταδίων μιας διαδικασίας από την οπτική γωνία του χρήστη.
- Δίνει στον σχεδιαστή την δυνατότητα να παρουσιάσει όσο λεπτομερώς επιθυμεί αυτός τις περιγραφές του συστήματος.

- Τμηματοποιεί τον σχεδιασμό του συστήματος με αποτέλεσμα να αυξάνεται η επεκτασιμότητα του.

Η UML 2.0 ορίζει δεκατρείς τύπους διαγραμμάτων, τα οποία μπορούν να χωριστούν σε δύο βασικές κατηγορίες:

- Τα διαγράμματα στατικής δομής, τα οποία δείχνουν τα πράγματα τα οποία απαρτίζουν το σύστημα που μοντελοποιείται και δεν αλλάζουν στο πέρασμα του χρόνου, για παράδειγμα τις κλάσεις, τα αντικείμενα κλπ. Στα διαγράμματα δομής ανήκουν: τα διαγράμματα κλάσεων, τα διαγράμματα αντικειμένων, τα ψηφιδικά διαγράμματα, τα διαγράμματα σύνθετης δομής, τα διαγράμματα πακέτου και τα διαγράμματα διάταξης.
- Τα διαγράμματα συμπεριφοράς, τα οποία δίνουν έμφαση στο τι πρέπει να γίνει στο σύστημα που μοντελοποιούμε, όπως τις αλληλεπιδράσεις με τους χρήστες, τις διάφορες καταστάσεις του συστήματος κλπ. Διαγράμματα συμπεριφοράς αποτελούν τα ακόλουθα: τα διαγράμματα χρήσης, τα διαγράμματα δραστηριότητας, τα διαγράμματα καταστάσεων μηχανής, τα ακολουθιακά διαγράμματα, τα συνεργατικά διαγράμματα και τα διαγράμματα χρονισμού.

Κάθε ένα από τα παραπάνω διαγράμματα βλέπει το σύστημα από μια διαφορετική οπτική γωνία, π.χ. τα διαγράμματα χρήσης δείχνουν πώς οι χρήστες αλληλεπιδρούν με το σύστημα, τα διαγράμματα κλάσης δείχνουν τις κλάσεις του συστήματος και τις σχέσεις μεταξύ τους κλπ.

Η πρακτική της ανάλυσης με σενάρια χρήσης τυποποιήθηκε για πρώτη φορά από τον Jacobson το 1994, ο οποίος περιέγραψε το σενάριο χρήσης σαν "μία σειρά από συναλλαγές που σχετίζονται ως προς την συμπεριφορά και εκτελούνται από έναν δράστη που βρίσκεται σε επικοινωνία με το σύστημα με σκοπό την παροχή κάποιας μετρήσιμης αξίας στον δράστη" [87]. Ένα σενάριο χρήσης είναι μία συλλογή από πιθανά σενάρια μεταξύ του συστήματος και των εξωτερικών του δραστών. Κάθε σενάριο χρήσης χαρακτηρίζεται από τον στόχο που έχει ο κύριος δράστης του. Ο στόχος αυτός μπορεί να επιτευχθεί ή όχι. Τα σενάρια χρήσης αποτελούν ένα πολύ διαδεδομένο και πολύτιμο εργαλείο για την πλήρη αποτύπωση των λειτουργικών απαιτήσεων του συστήματος. Οι λειτουργικές απαιτήσεις μπορούν να εκφραστούν είτε με μορφή κειμένου, είτε με μορφή πίνακα. Επιλέξαμε να χρησιμοποιήσουμε την μορφή πίνακα, καθώς έχει πιο αυστηρή δομή με αποτέλεσμα να μειώνονται οι ασυνέπειες και οι περιττές πληροφορίες. [35] και πιο συγκεκριμένα την μονή στήλη πρότυπο [35] :

<b>Όνομα σεναρίου χρήσης</b>
Ο τίτλος του σεναρίου χρήσης, που συνήθως είναι ένα μία πρόταση που περιγράφει τον στόχο του κύριου δράστη.
<b>Στόχος</b>
Μία μεγαλύτερη δήλωση η οποία περιγράφει τον στόχο του σεναρίου χρήσης με περισσότερες λεπτομέρειες.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Είναι οι δράστες που έχουν ως στόχο αυτόν που περιγράφεται από το σενάριο χρήσης.
<b>Προϋποθέσεις:</b> Αυτά που ισχύουν πριν ξεκινήσει να εκτελείται το σενάριο χρήσης. Εφόσον είναι γνωστό ότι μια συνθήκη είναι αληθής, δεν θα ελεγχθεί ξανά κατά την διάρκεια εκτέλεσης του σεναρίου χρήσης.
<b>Ενέργεια ενεργοποίησης:</b> Καθορίζει ποιο γεγονός εκκινεί το σενάριο χρήσης. Η ενέργεια εκκίνησης αποτελεί το πρώτο βήμα του σεναρίου χρήσης.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> Η κύρια ροή γεγονότων περιγράφει την περίπτωση που όλα βαίνουν αισίως. Είναι μια σειρά από αριθμημένα βήματα, όπου το κάθε βήμα είναι μία ρητή εντολή- μετάβαση.
<b>Επεκτάσεις:</b>
Οι επεκτάσεις περιγράφουν τι μπορεί να συμβεί και με διαφορετικό τρόπο κατά την διάρκεια εκτέλεσης του επιτυχημένου σεναρίου. Μπορούν να οδηγήσουν σε επιτυχία ή σε αποτυχίες. Κάθε επέκταση αποτελείται από δύο διαφορετικά μέρη: <ul style="list-style-type: none"> <li>• Τις προϋποθέσεις κάτω από τις οποίες το σύστημα υιοθετεί μία διαφορετική συμπεριφορά. Μετά από αυτές τοποθετείται μία άνω και κάτω τελεία (:).</li> <li>• Ο χειρισμός της επέκτασης που είναι μία ακολουθία από βήματα που απαιτούνται λόγω των προϋποθέσεων που υπάρχουν.</li> </ul>
<b>Παραλλαγές:</b> Οι παραλλαγές κάποιου βήματος από το επιτυχημένο σενάριο υπάρχουν επειδή ενώ αυτό που συμβαίνει σε κάποιο βήμα είναι πάντα το ίδιο υπάρχουν πολλοί τρόποι να το κάνουμε.

Πίνακας 3.1: Σενάριο χρήσης εγγραφής αιμοδότη



## Εφαρμογή

Θα αναπτύξουμε αρχικά τα σενάρια χρήσης της εφαρμογής.

- Σενάριο εγγραφής στο σύστημα ως εθελοντής αιμοδότης με χρήση του αριθμού μητρώου κοινωνικής ασφάλισης (Α.Μ.Κ.Α.).

<b>Όνομα σεναρίου χρήσης</b>
Εγγραφή στο σύστημα.
<b>Στόχος</b>
Ο χρήστης εισάγει τα προσωπικά του δεδομένα με σκοπό να δημιουργήσει το προσωπικό του μοναδικό αιμοδοτικό προφίλ.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Να διαθέτει αριθμό ΑΜΚΑ. <b>Ενέργεια ενεργοποίησης:</b> Μετάβαση στην sign up φόρμα
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης πηγαίνει στην επιλογή sign up και εισάγει το ΑΜΚΑ του.</li><li>2. Στην συνέχεια συμπληρώνει την sign up φόρμα και επιλέγει ολοκλήρωση εγγραφής.</li><li>3. Το σύστημα δημιουργεί ένα μοναδικό προφίλ για τον χρήστη, το οποίο εισάγει στην βάση δεδομένων του.</li><li>4. Ο χρήστης ανακατευθύνεται στην αρχική σελίδα</li></ol>
<b>Επεκτάσεις:</b>
<ol style="list-style-type: none"><li>1. Ο χρήστης εισάγει ΑΜΚΑ για το οποίο έχει ήδη δημιουργηθεί κάποιο προφίλ.</li><li>2. Το σύστημα τον ενημερώνει και τον καθοδηγεί ώστε να ανακτήσει τα στοιχεία του.</li></ol>

Πίνακας 3.2: Σενάριο χρήσης εγγραφής αιμοδότη

- Σύνδεση στο σύστημα αν έχει πραγματοποιήσει ήδη έγγραφη στο παρελθόν.

<b>Όνομα σεναρίου χρήσης</b>
Σύνδεση στο σύστημα.
<b>Στόχος</b>
Ο χρήστης εισάγει τα στοιχεία του ώστε να μεταφερθεί στον προσωπικό του λογαριασμό.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να είναι ήδη εγγεγραμμένος στο σύστημα. <b>Ενέργεια ενεργοποίησης:</b> Μετάβαση στην sign in φόρμα.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης εισάγει τα στοιχεία του και επιλέγει να συνδεθεί με το σύστημα.</li><li>2. Το σύστημα αποκρίνεται θετικά αν υπάρχει εγγεγραμμένος χρήστης με αυτά τα στοιχεία στην βάση δεδομένων του συστήματος, αλλιώς απαγορεύει την είσοδο στον χρήστη και τυπώνει το αντίστοιχο μήνυμα λάθους.</li><li>3. Ο χρήστης κατευθύνεται είτε στον προσωπικό του λογαριασμό (σωστή εισαγωγή στοιχείων) είτε του ζητείται να ξαναπροσπαθήσει να εισάγει τα σωστά στοιχεία στην sign in φόρμα.</li></ol>
<b>Επεκτάσεις:</b>
<ul style="list-style-type: none"><li>• Ο χρήστης εισάγει μη έγκυρης μορφής όνομα χρήστη ή κωδικό και επιλέγει να συνδεθεί με το σύστημα.</li><li>• Το σύστημα εμφανίζει μήνυμα λάθους, χωρίς να επικοινωνήσει με την βάση δεδομένων για ταυτοποίηση.</li><li>• Το σύστημα αναπροβάλλει την φόρμα με μήνυμα οδηγιών στον χρήστη.</li></ul>

Πίνακας 3.3: Σενάριο χρήσης σύνδεσης στο σύστημα

- Προβολή γεγονότων αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Προβολή γεγονότων αιμοδοσίας
<b>Στόχος</b>
Ο χρήστης θέλει να μπαίνει στην εφαρμογή και να βλέπει όλα τα ενεργά γεγονότα αιμοδοσίας, ταξινομημένα ανά την προτίμηση του(απόσταση, εναπομείνουσες ημέρες κλπ) καθώς και τις απαραίτητες πληροφορίες για αυτά (τοποθεσία διεξαγωγής, ημερομηνίες, ομάδα αίματος κλπ)
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να είναι ήδη εγγεγραμμένος στο σύστημα. <b>Ενέργεια ενεργοποίησης:</b> Επιλογή της καρτέλας "Γεγονότα αιμοδοσίας".
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει την καρτέλα "Γεγονότα αιμοδοσίας".</li><li>2. Το σύστημα φορτώνει από την βάση δεδομένων και την προσωρινή μνήμη τα γεγονότα αιμοδοσίας.</li><li>3. Ο χρήστης επιλέγει και αλλάζει την σειρά ως προς την οποία εμφανίζονται τα γεγονότα ανάλογα με κάποιο κριτήριο.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.4: Σενάριο χρήσης προβολής γεγονότων αιμοδοσίας

-Λήψης ειδοποιήσεων (push notifications) όταν δημιουργείται κάποιο νέο γεγονός αιμοδοσίας για το οποίο ο εθελοντής αιμοδότης πληρεί τα απαραίτητα κριτήρια.

<b>Όνομα σεναρίου χρήσης</b>
Λήψη ειδοποιήσεων (push notifications).
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να μπορεί να λαμβάνει ειδοποιήσεις από το σύστημα σχετικά με ενεργές αιμοδοσίες.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να είναι ήδη εγγεγραμμένος στο σύστημα. <b>Ενέργεια ενεργοποίησης:</b> Δημιουργία νέου γεγονότος αιμοδοσίας.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Δημιουργείται ένα νέο συμβάν αιμοδοσίας.</li><li>2. Το σύστημα με βάση το business model του αποφαινεται αν ο χρήστης είναι σε θέση ή όχι να συμμετάσχει στην αιμοδοσία.</li><li>3. Το σύστημα με βάση το αποτέλεσμα του προηγούμενου βήματος στέλνει ή όχι ειδοποίηση στον χρήστη.</li><li>4. Ο χρήστης βλέπει την ειδοποίηση στο κινητό του.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.5: Σενάριο χρήσης λήψης ειδοποιήσεων

-Αποδοχή, απόρριψη ή αναβολή της ειδοποίησης για το γεγονός αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση ειδοποιήσεων (push notifications).
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να μπορεί να διαχειρίζεται αποτελεσματικά τις ειδοποιήσεις από το σύστημα σχετικά με ενεργές αιμοδοσίες, είτε πατώντας αποδοχή, είτε απορρίπτοντας τες, είτε αναβάλλοντας τες.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να είναι ήδη εγγεγραμμένος στο σύστημα. <b>Ενέργεια ενεργοποίησης:</b> Λήψη νέας ειδοποίησης (push notification).
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Λήψη ειδοποίησης από το σύστημα για νέο γεγονός αιμοδοσίας.</li><li>2. Ο χρήστης επιλέγει μία από τις εξής ενέργειες: αποδοχή, απόρριψη, αναβολή.</li><li>3. Το σύστημα αν ο χρήστης επιλέξει αποδοχή μεταφέρει το γεγονός στην λίστα με τα γεγονότα αιμοδοσίας που έχει αποδεχτεί ο χρήστης, αν επιλέξει απόρριψη μεταφέρει το γεγονός στην λίστα με τα γεγονότα που έχει απορρίψει, αλλιώς προγραμματίζει εκ νέου την αποστολή υπενθύμισης.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.6: Σενάριο χρήσης διαχείρισης ειδοποιήσεων

-Διαχείριση ραντεβού για αιμοδοσία (δημιουργία νέου ραντεβού, μετάθεση , προβολή, ακύρωση).

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση ραντεβού για αιμοδοσία.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να αλληλεπιδράσει με το σύστημα των ραντεβού, είτε δημιουργώντας ένα νέο ραντεβού για αιμοδοσία, είτε ακυρώνοντας ένα ήδη υπάρχον, είτε μεταθέτοντας την ημερομηνία ενός ραντεβού.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να έχει αποδεχτεί κάποιο γεγονός αιμοδοσίας. <b>Ενέργεια ενεργοποίησης:</b> Αποδοχή κάποιου γεγονότος αιμοδοσίας ή μετάβαση στο ημερολόγιο.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης μεταφέρεται στο ημερολόγιο των ραντεβού, είτε αποδεχόμενος κάποιο γεγονός αιμοδοσίας, είτε με επιλογή της καρτέλας του ημερολογίου.</li><li>2. Ο χρήστης βλέπει τις διαθέσιμες ημερομηνίες και ώρες για ραντεβού για πραγματοποίηση αιμοδοσίας, με την επιλογή "Κλείσιμο ραντεβού", καθώς και τα ήδη υπάρχοντα ραντεβού, με τις επιλογές μετάθεση, διαγραφή.</li><li>3. Το σύστημα ανάλογα με τις προηγούμενες ενέργειες του χρήστη κατοχυρώνει τις τελικές αλλαγές.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.7: Σενάριο χρήσης διαχείρισης ραντεβού αιμοδοσίας

-Προσθήκη και συγχρονισμός των ραντεβού αιμοδοσίας στο ημερολόγιο του χρήστη.

<b>Όνομα σεναρίου χρήσης</b>
Προσθήκη των ραντεβού αιμοδοσίας στο ημερολόγιο του χρήστη.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να μπορεί να συγχρονίσει το σύστημα των ραντεβού του ολικού συστήματος αιμοδοσίας με το ημερολόγιο του, προσθέτοντας τα ραντεβού αιμοδοσίας στις εφαρμογές ημερολογίου που χρησιμοποιεί (google calendar, calendar κλπ.)
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να έχει κλείσει ραντεβού κάποιο γεγονός αιμοδοσίας. <b>Ενέργεια ενεργοποίησης:</b> Άνοιγμα ραντεβού αιμοδοσίας στο ημερολόγιο και επιλογή "προσθήκη στο ημερολόγιο μου".
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης μεταφέρεται στο ημερολόγιο των ραντεβού αιμοδοσίας της εφαρμογής και αφού ανοίξει κάποιο προγραμματισμένο ραντεβού επιλέγει το "προσθήκη στο ημερολόγιο μου".</li><li>2. Το σύστημα, εφόσον ο χρήστης το έχει εξουσιοδοτήσει με άδεια πρόσβασης στις εφαρμογές ημερολογίου του, συγχρονίζει τα ζητούμενα δεδομένα.</li></ol>
<b>Ελεκτάσεις:</b>
<ol style="list-style-type: none"><li>1. Το σύστημα δεν έχει τις απαραίτητες άδειες για πρόσβαση στις εφαρμογές ημερολογίων του χρήστη.</li><li>2. Το σύστημα αιτεί τις απαραίτητες άδειες από τον χρήστη.</li><li>3. Το σύστημα εκκινεί την διαδικασία συγχρονισμού αν ο χρήστης του παραχωρήσει δικαιώματα πρόσβασης, αλλιώς εμφανίζει μήνυμα αποτυχίας.</li></ol>

Πίνακας 3.8: Σενάριο χρήσης προσθήκης ραντεβού αιμοδοσίας στο ημερολόγιο

- Ενημέρωση σχετικά με το υπολειπόμενο διάστημα που πρέπει να παρέλθει ώστε ο εθελοντής να είναι σε θέση να συμμετέχει σε μία νέα αιμοδοσία.

<b>Όνομα σεναρίου χρήσης</b>
Ενημέρωση για το υπολειπόμενο διάστημα υποχρεωτικής αποχής του αιμοδότη.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να μπορεί να ενημερώνεται απλά και γρήγορα σχετικά με το διάστημα που πρέπει να παρέλθει ώστε να είναι σε θέση να συμμετάσχει ξανά σε αιμοδοσία.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να έχει πραγματοποιήσει κάποια αιμοδοσία. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης μπαίνει στον προσωπικό του λογαριασμό στην εφαρμογή.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b>  <ol style="list-style-type: none"><li>1. Ο χρήστης μπαίνει στον προσωπικό του λογαριασμό στην εφαρμογή.</li><li>2. Το σύστημα με βάση τους κανονισμούς των αιμοδοτικών κέντρων για το απαιτούμενο διάστημα αποχής και την ημερομηνία τελευταίας αιμοδοσίας του εθελοντή, υπολογίζει τις υπολειπόμενες μέρες και τις εμφανίζει στον χρήστη.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.9: Σενάριο χρήσης ενημέρωσης για το υπολειπόμενο διάστημα υποχρεωτικής αποχής του αιμοδότη



-Προβολή πόντων και εμβλημάτων βάση των αιμοδοσιών του (gamification system) και κοινοποίηση στα μέσα κοινωνικής δικτύωσης.

<b>Όνομα σεναρίου χρήσης</b>
Gamification System
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να μπορεί να κερδίζει πόντους και εμβλήματα βάση των αιμοδοσιών του και να κοινοποιεί στα μέσα κοινωνικής δικτύωσης.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης πρέπει να έχει πραγματοποιήσει αιμοδοσίες. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα "Πόντοι και εμβλήματα".
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"> <li>1. Ο χρήστης μπαίνει στην καρτέλα "Πόντοι και εμβλήματα" της εφαρμογής.</li> <li>2. Ο χρήστης βλέπει τα εμβλήματα του και επιλέγει το έμβλημα που θέλει να μοιραστεί στα κοινωνικά δίκτυα.</li> <li>3. Το σύστημα, εφόσον ο χρήστης το έχει εξουσιοδοτήσει με άδεια πρόσβασης στους λογαριασμούς του ζητούμενα κοινωνικά δίκτυα, μοιράζεται τα δεδομένα.</li> </ol>
<b>Επεκτάσεις:</b>
<ol style="list-style-type: none"> <li>1. Το σύστημα δεν έχει τις απαραίτητες άδειες για πρόσβαση στους λογαριασμούς των κοινωνικών δικτύων του χρήστη,</li> <li>2. Το σύστημα αιτεί τις απαραίτητες άδειες από τον χρήστη.</li> <li>3. Το σύστημα εκκινεί την διαδικασία κοινοποίησης αν ο χρήστης του παραχωρήσει δικαιώματα πρόσβασης, αλλιώς εμφανίζει μήνυμα αποτυχίας.</li> </ol>

Πίνακας 3.10: Σενάριο χρήσης παιχνιδιοποίησης

-Δημοσίευση στοιχείων στα μέσα κοινωνικής δικτύωσης σε κάθε στάδιο της αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Δημοσίευση στοιχείων από την αιμοδοσία στα μέσα κοινωνικής δικτύωσης.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να έχει την δυνατότητα σε διάφορα στάδια της αιμοδοσίας (αποδοχή αιτήματος, κλείσιμο ραντεβού) να κοινοποιεί δεδομένα στα μέσα κοινωνικής δικτύωσης.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης να έχει αποδεχτεί κάποια αιμοδοσία. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει να κάνει ανάρτηση μέσω της εφαρμογής στα μέσα κοινωνικής δικτύωσης.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει να κάνει ανάρτηση μέσω της εφαρμογής σε κάποιο στάδιο της διαδικασίας(αποδοχή, κλείσιμο ραντεβού, πραγματοποίηση αιμοδοσίας) .</li><li>2. Ο χρήστης συμπληρώνει την περιγραφή της δημοσίευσης και επιλέγει τα μέσα κοινωνικής δικτύωσης στα οποία θέλει να γίνει η ανάρτηση.</li><li>3. Το σύστημα, εφόσον ο χρήστης το έχει εξουσιοδοτήσει με άδεια πρόσβασης στους λογαριασμούς του στα ζητούμενα κοινωνικά δίκτυα, μοιράζεται τα δεδομένα.</li></ol>
<b>Επεκτάσεις:</b>
<ol style="list-style-type: none"><li>1. Το σύστημα δεν έχει τις απαραίτητες άδειες για πρόσβαση στους λογαριασμούς των κοινωνικών δικτύων του χρήστη,</li><li>2. Το σύστημα αιτεί τις απαραίτητες άδειες από τον χρήστη.</li><li>3. Το σύστημα εκκινεί την διαδικασία κοινοποίησης αν ο χρήστης του παραχωρήσει δικαιώματα πρόσβασης, αλλιώς εμφανίζει μήνυμα αποτυχίας.</li></ol>

Πίνακας 3.11: Σενάριο χρήσης δημοσίευσης στοιχείων στα μέσα κοινωνικής δικτύωσης

-Δημιουργία έκκλησης για αίμα.

<b>Όνομα σεναρίου χρήσης</b>
Δημιουργία έκκλησης για αίμα.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να έχει την δυνατότητα να δημιουργήσει ένα νέο αίτημα για αιμοδοσία και να μπορεί να το δημοσιεύσει μέσω της εφαρμογής στα μέσα κοινωνικής δικτύωσης.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγγεγραμμένος στην εφαρμογή. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα δημιουργία νέου γεγονότος αιμοδοσίας.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει να κάνει ανάρτηση μέσω της εφαρμογής στα κοινωνικά δίκτυα..</li><li>2. Ο χρήστης συμπληρώνει τα απαραίτητα πεδία στην φόρμα και επιλέγει δημιουργία νέου αιτήματος.</li><li>3. Ο χρήστης επιλέγει κοινοποίηση του αιτήματος και τα μέσα κοινωνικής δικτύωσης στα οποία θέλει να γίνει η ανάρτηση.</li><li>4. Το σύστημα, εφόσον ο χρήστης το έχει εξουσιοδοτήσει με άδεια πρόσβασης στους λογαριασμούς του στα ζητούμενα κοινωνικά δίκτυα, μοιράζεται τα δεδομένα.</li></ol>
<b>Επεκτάσεις:</b>
<ol style="list-style-type: none"><li>1. Το σύστημα δεν έχει τις απαραίτητες άδειες για πρόσβαση στους λογαριασμούς των κοινωνικών δικτύων του χρήστη,</li><li>2. Το σύστημα αιτεί τις απαραίτητες άδειες από τον χρήστη.</li><li>3. Το σύστημα εκκινεί την διαδικασία κοινοποίησης αν ο χρήστης του παραχωρήσει δικαιώματα πρόσβασης, αλλιώς εμφανίζει μήνυμα αποτυχίας.</li></ol>

Πίνακας 3.12: Σενάριο χρήσης δημιουργίας έκκλησης για αίμα

-Διαχείριση προφίλ χρήστη.

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση προσωπικού προφίλ.
<b>Στόχος</b>
Ο εθελοντής αιμοδότης να έχει την δυνατότητα να βλέπει και να επεξεργάζεται τα προσωπικά του στοιχεία.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εθελοντής αιμοδότης. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγγεγραμμένος στην εφαρμογή. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης μεταβαίνει στις ρυθμίσεις του προσωπικού του λογαριασμού.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης μεταβαίνει στις ρυθμίσεις του προσωπικού του λογαριασμού.</li><li>2. Τροποποιεί τα στοιχεία που επιθυμεί.</li><li>3. Το σύστημα αποθηκεύει τις αλλαγές του χρήστη και ανανεώνει το προφίλ του, ώστε να βλέπει τα νέα δεδομένα.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.13: Σενάριο χρήσης διαχείρισης προσωπικού προφίλ

## Web portal

Όσον αφορά στους χρήστες/διαχειριστές των κέντρων αιμοδοσίας έχουμε τα εξής σενάρια χρήσης:

-Δημιουργία νέων γεγονότων αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Δημιουργία γεγονότων αιμοδοσίας.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να έχει την δυνατότητα να δημιουργήσει ένα νέο γεγονός αιμοδοσίας και να το καταχωρήσει στο σύστημα ώστε να ειδοποιηθούν οι εθελοντές αιμοδότες.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα δημιουργία νέου γεγονότος αιμοδοσίας.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει την καρτέλα δημιουργίας νέου γεγονότος αιμοδοσίας.</li><li>2. Ο χρήστης συμπληρώνει τα απαραίτητα πεδία στην φόρμα και επιλέγει καταχώρηση του γεγονότος αιμοδοσίας.</li><li>3. Το σύστημα τρέχει τους απαραίτητους αλγορίθμους και επιλέγει ποιους χρήστες θα ενημερώσει με τα κατάλληλα push notifications.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.14: Σενάριο χρήσης δημιουργίας έκκλησης για αίμα

- Προβολή στατιστικών ανά έτος σχετικά με τις ειδοποιήσεις που στάλθηκαν από το συγκεκριμένο κέντρο αιμοδοσίας και την ανταπόκριση των χρηστών (αποδοχή, απόρριψη, άνοιγμα ειδοποίησης, ολοκλήρωση αιμοδοσίας).

<b>Όνομα σεναρίου χρήσης</b>
Προβολή στατιστικών ανά έτος.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να έχει την δυνατότητα να δει αναλυτικά στατιστικά ανά έτος σχετικά με τις ειδοποιήσεις που στάλθηκαν από το συγκεκριμένο κέντρο αιμοδοσίας, ποιοι και πόσοι χρήστες τις απέρριψαν, ποιοι και πόσοι χρήστες τις αποδέχτηκαν, ποιοι και πόσοι χρήστες πραγματοποίησαν αιμοδοσίες, ποιοι άνοιξαν τις ειδοποιήσεις, σε τι χρονικό διάστημα ανταποκρίθηκαν.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα προβολή στατιστικών στοιχείων.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει την καρτέλα προβολή στατιστικών στοιχείων.</li> <li>2. Το σύστημα φέρνει τα απαραίτητα δεδομένα από την βάση δεδομένων.</li> <li>3. Ο χρήστης πλοηγείται και επιλέγει το διάγραμμα που θέλει.</li> <li>4. Τα δεδομένα παρουσιάζονται στον χρήστη αναλυτικά σε νέα σελίδα.</li> </ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.15: Σενάριο χρήσης δημιουργίας έκκλησης για αίμα

-Διαχείριση των ραντεβού για αιμοληψία.

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση των ραντεβού για αιμοληψία.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να μπορεί να διαχειριστεί το σύστημα των ραντεβού, είτε ορίζοντας τα διαθέσιμα ραντεβού για αιμοδοσία, είτε βλέποντας τα υπάρχοντα, είτε αλλάζοντας την ημερομηνία ενός ραντεβού.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα ραντεβού αιμοδοσίας.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει την καρτέλα ραντεβού αιμοδοσίας.</li><li>2. Ο χρήστης ορίζει τις διαθέσιμες ημερομηνίες και ώρες για ραντεβού εάν πρόκειται για νέα αιμοδοσία, βλέπει τα προγραμματισμένα ραντεβού, είτε επιλέγει κάποιο ραντεβού και αλλάζει την ημερομηνία του.</li><li>3. Το σύστημα ανάλογα με τις προηγούμενες ενέργειες του χρήστη κατοχυρώνει τις τελικές αλλαγές.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.16: Σενάριο χρήσης διαχείρισης ραντεβού αιμοδοσίας

-Δυνατότητα διαχείρισης προφίλ κέντρου αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση προφίλ του αιμοδοτικού κέντρου.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να μπορεί να διαχειριστεί να δει και να επεξεργαστεί τα στοιχεία του αιμοδοτικού κέντρου.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης μεταβαίνει στις ρυθμίσεις του προσωπικού του λογαριασμού.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b>  <ol style="list-style-type: none"><li>1. Ο χρήστης μεταβαίνει στις ρυθμίσεις του λογαριασμού του αιμοδοτικού κέντρου.</li><li>2. Τροποποιεί τα στοιχεία που επιθυμεί.</li><li>3. Το σύστημα αποθηκεύει τις αλλαγές του χρήστη και ανανεώνει το προφίλ, ώστε να ορατά σε όλους τα νέα δεδομένα.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.17: Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας



-Δυνατότητα εγγραφής στο σύστημα νέου εθελοντή αιμοδότη.

<b>Όνομα σεναρίου χρήσης</b>
Δημιουργία λογαριασμού για νέο εθελοντή αιμοδότη.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να μπορεί να δημιουργήσει με χρήση μόνο ΑΜΚΑ νέο λογαριασμό για κάποιον εθελοντή αιμοδότη που δεν είναι εγγεγραμμένος στο σύστημα, στον οποίο λογαριασμό ο χρήστης να μπορεί να αποκτήσει πρόσβαση εισάγοντας το ΑΜΚΑ του, και να τον διαχειριστεί.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Έρχεται να δώσει αίμα εθελοντής που δεν είναι εγγεγραμμένος στο σύστημα.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Έρχεται να δώσει αίμα εθελοντής που δεν είναι εγγεγραμμένος στο σύστημα.</li><li>2. Ο χρήστης μεταβαίνει στην καρτέλα προσθήκη νέου χρήστη.</li><li>3. Εισάγει το ΑΜΚΑ και ζητά την δημιουργία λογαριασμού.</li><li>4. Το σύστημα δημιουργεί τον λογαριασμό του εθελοντή αιμοδότη.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.18: Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας

-Καταχώρηση της ολοκλήρωσης αιμοδοσίας.

<b>Όνομα σεναρίου χρήσης</b>
Καταχώρηση ολοκλήρωσης της αιμοδοσίας.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να μπορεί να ενημερώσει το προφίλ του χρήστη, όταν αυτός ολοκληρώσει επιτυχώς μια αιμοδοσία.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Ολοκληρώνει την αιμοδοσία ένας εθελοντής αιμοδότης.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο εθελοντής αιμοδότης ολοκληρώνει την αιμοδοσία.</li><li>2. Ο χρήστης μεταβαίνει στην καρτέλα καταχώρηση αιμοδοσίας</li><li>3. Επιλέγει το γεγονός αιμοδοσίας.</li><li>4. Εισάγει το ΑΜΚΑ για να βρει τον εθελοντή αιμοδότη και πατάει ολοκλήρωση.</li><li>5. Το σύστημα ενημερώνει τον λογαριασμό του εθελοντή αιμοδότη.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.19: Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας

-Έλεγχος καταλληλότητας εθελοντή αιμοδότη.

<b>Όνομα σεναρίου χρήσης</b>
Έλεγχος καταλληλότητας αιμοδότη.
<b>Στόχος</b>
Ο εγκεκριμένος χρήστης/διαχειριστής να μπορεί ελέγξει αυτόματα, αν ο εθελοντής τηρεί κάποια κριτήρια που τον καθιστούν κατάλληλο για αιμοδοσία.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Εγκεκριμένος χρήστης/διαχειριστής των συστημάτων αιμοδοσίας. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι εγκεκριμένος χρήστης/διαχειριστής εκ μέρους κάποιου νοσοκομείου. <b>Ενέργεια ενεργοποίησης:</b> Έρχεται για αιμοδοσία ένας εθελοντής αιμοδότης.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b> <ol style="list-style-type: none"><li>1. Ο εθελοντής αιμοδότης έρχεται για αιμοδοσία.</li><li>2. Ο χρήστης μεταβαίνει στην καρτέλα καταχώρηση αιμοδοσίας.</li><li>3. Εισάγει το ΑΜΚΑ και ζητάει πραγματοποίηση ελέγχου.</li><li>4. Το σύστημα επικοινωνεί με το epSOS (ηλεκτρονική συνταγογράφηση και ηλεκτρονικός ιατρικός φάκελος ασθενούς) και “τραβάει” τα απαραίτητα δεδομένα, τρέχει ειδικούς αλγορίθμους και αποφαινεται αν ο χρήστης είναι κατάλληλος ή όχι.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.20: Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας

Όσον αφορά στους υπέρ-διαχειριστές του συστήματος έχουμε τις παρακάτω λειτουργικές απαιτήσεις:

-Δυνατότητα διαχείρισης λογαριασμών χρηστών.

<b>Όνομα σεναρίου χρήσης</b>
Διαχείριση προφίλ των χρηστών.
<b>Στόχος</b>
Ο υπέρ-διαχειριστής να μπορεί να προσθέσει νέους χρήστες, να τροποποιήσει τα προφίλ ήδη υπαρχόντων χρηστών και να διαγράψει προφίλ χρηστών.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Υπέρ-διαχειριστής του συστήματος. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι υπέρ-διαχειριστής. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης μεταβαίνει στην καρτέλα διαχείρισης χρηστών.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων:</b>  <ol style="list-style-type: none"><li>1. Ο χρήστης μεταβαίνει στην καρτέλα διαχείρισης χρηστών.</li><li>2. Τροποποιεί τα στοιχεία που επιθυμεί.</li><li>3. Το σύστημα αποθηκεύει τις αλλαγές του χρήστη και αλλάζει τα προφίλ, στέλνοντας ειδοποιήσεις στους αντίστοιχους χρήστες.</li></ol>
<b>Επεκτάσεις:</b>

Πίνακας 3.21: Σενάριο χρήσης διαχείρισης προφίλ κέντρου αιμοδοσίας

- Προβολή στατιστικών για τους χρήστες του συστήματος.

<b>Όνομα σεναρίου χρήσης</b>
Προβολή στατιστικών.
<b>Στόχος</b>
Ο υπέρ-διαχειριστής να έχει την δυνατότητα να δει αναλυτικά στατιστικά σχετικά με τους χρήστες και το πόσο είναι ενεργοί / χρησιμοποιούν την εφαρμογή.
<b>Λεπτομέρειες</b>
<b>Δράστης:</b> Υπέρ-διαχειριστής του συστήματος. <b>Προϋποθέσεις:</b> Ο χρήστης να είναι υπέρ-διαχειριστής. <b>Ενέργεια ενεργοποίησης:</b> Ο χρήστης επιλέγει την καρτέλα προβολή στατιστικών στοιχείων.
<b>Ροή γεγονότων</b>
<b>Κύρια ροή γεγονότων</b> <ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει την καρτέλα προβολή στατιστικών στοιχείων.</li><li>2. Το σύστημα φέρνει τα απαραίτητα δεδομένα από την βάση δεδομένων.</li><li>3. Ο χρήστης πλοηγείται και επιλέγει το διάγραμμα που θέλει.</li><li>4. Τα δεδομένα παρουσιάζονται στον χρήστη αναλυτικά σε νέα σελίδα.</li></ol>
<b>Επεκτάσεις:</b>

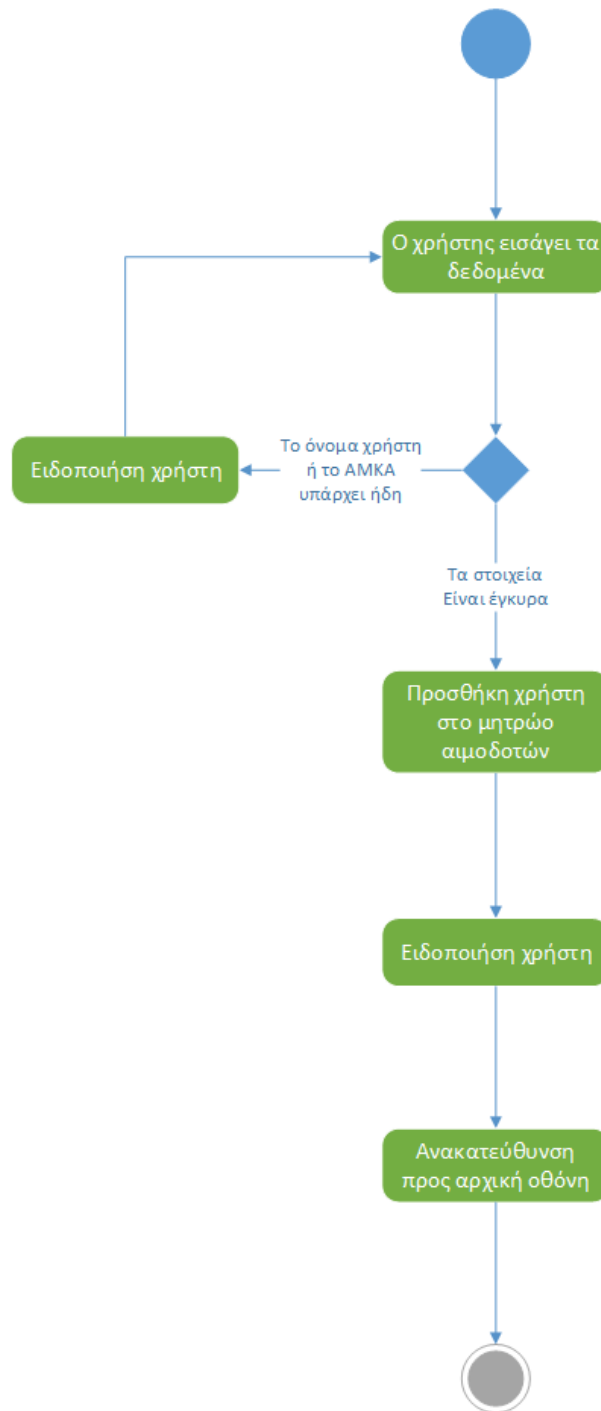
Πίνακας 3.22: Σενάριο χρήσης δημιουργίας έκκλησης για αίμα

### 3.3 Διαγράμματα UML - Διαγράμματα δραστηριότητας

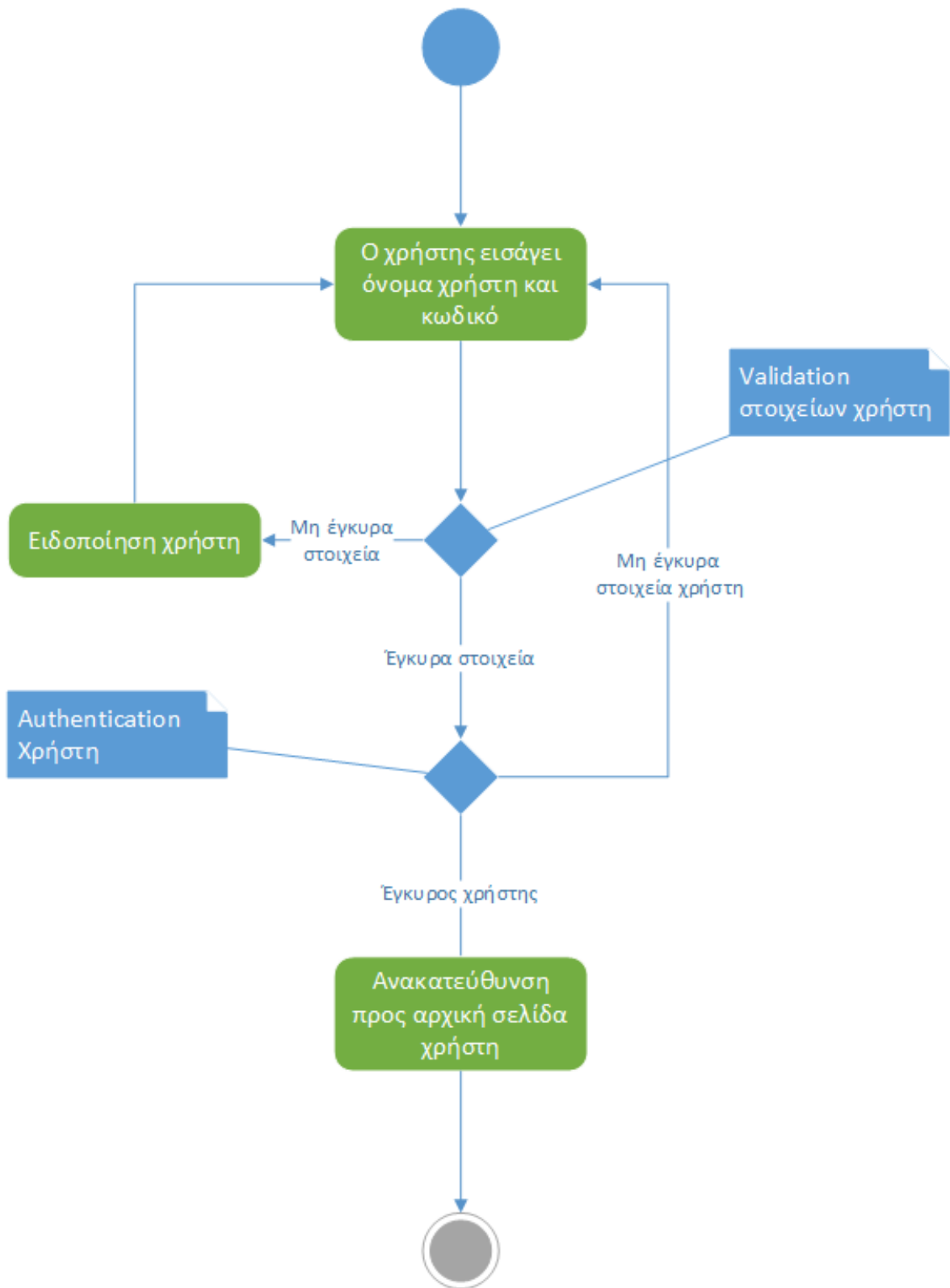
Τα διαγράμματα δραστηριότητας χρησιμοποιούνται για να μοντελοποιήσουν ένα σενάριο χρήσης με μεγάλη λεπτομέρεια. Κάθε διάγραμμα δραστηριότητας αρχίζει με ένα μαύρο κύκλο και σταματά σε τουλάχιστον έναν ομόκεντρο ασπρόμαυρο κύκλο. Οι δράσεις συμβολίζονται με ελλείψεις. Όταν μια δράση αναλύεται σε επιμέρους δράσεις, χρησιμοποιούνται κουτιά με στρογγυλεμένες άκρες. Από κάθε δράση βγαίνει ένα βέλος, που ονομάζεται μετάβαση και πραγματοποιεί την σύνδεση με την επόμενη. Το βέλος μπορεί επίσης να οδηγήσει σε ένα διαμάντι, το οποίο υποδεικνύει διακλάδωση σε δύο ή περισσότερες αλληλοαναιρούμενες μεταβάσεις. Οι εκφράσεις που βρίσκονται μέσα σε αγκύλες [], π.χ. [υπάρχει όνομα χρήστη] ή [δεν υπάρχει όνομα χρήστη], επισημαίνουν τις μεταβάσεις που

προέρχονται από ένα κλαδί. Επίσης, μία μετάβαση μπορεί να διακλαδωθεί σε δύο ή περισσότερες παράλληλες δραστηριότητες (fork activities). Όταν και οι δύο δραστηριότητες έχουν ολοκληρωθεί ενώνονται και η ροή συνεχίζεται. Κάθε διακλάδωση (fork) πρέπει να ακολουθείται από μία ένωση (join). Οι διακλαδώσεις(forks) και οι ενώσεις(joins) εμφανίζονται στο διάγραμμα δραστηριότητας σαν μπάρες.

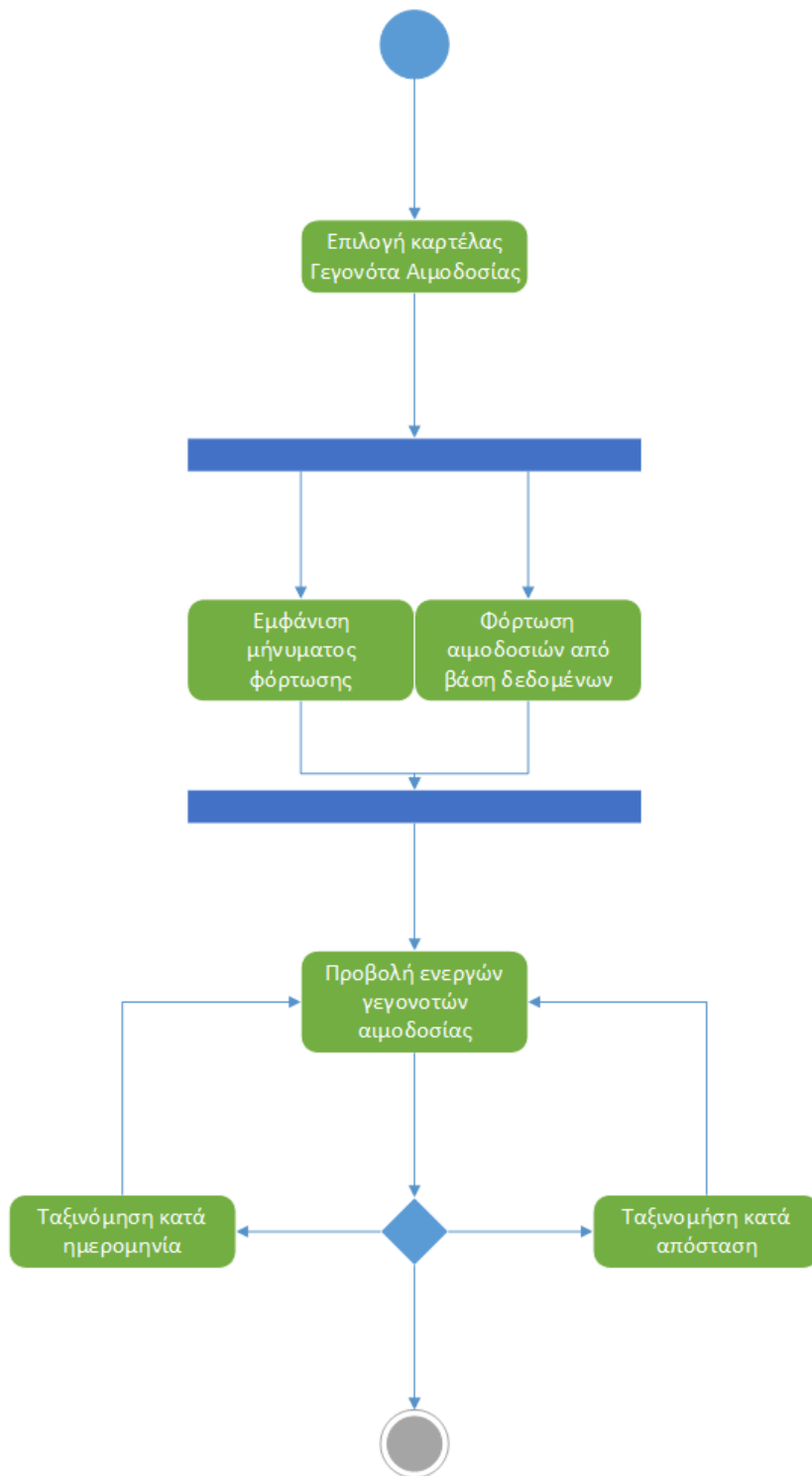
Στην συνέχεια ακολουθούν κάποια διαγράμματα δραστηριότητας:



**Σχήμα 3.1:** Διάγραμμα δραστηριότητας #1. Εγγραφή εθελοντή αιμοδότη στο σύστημα.

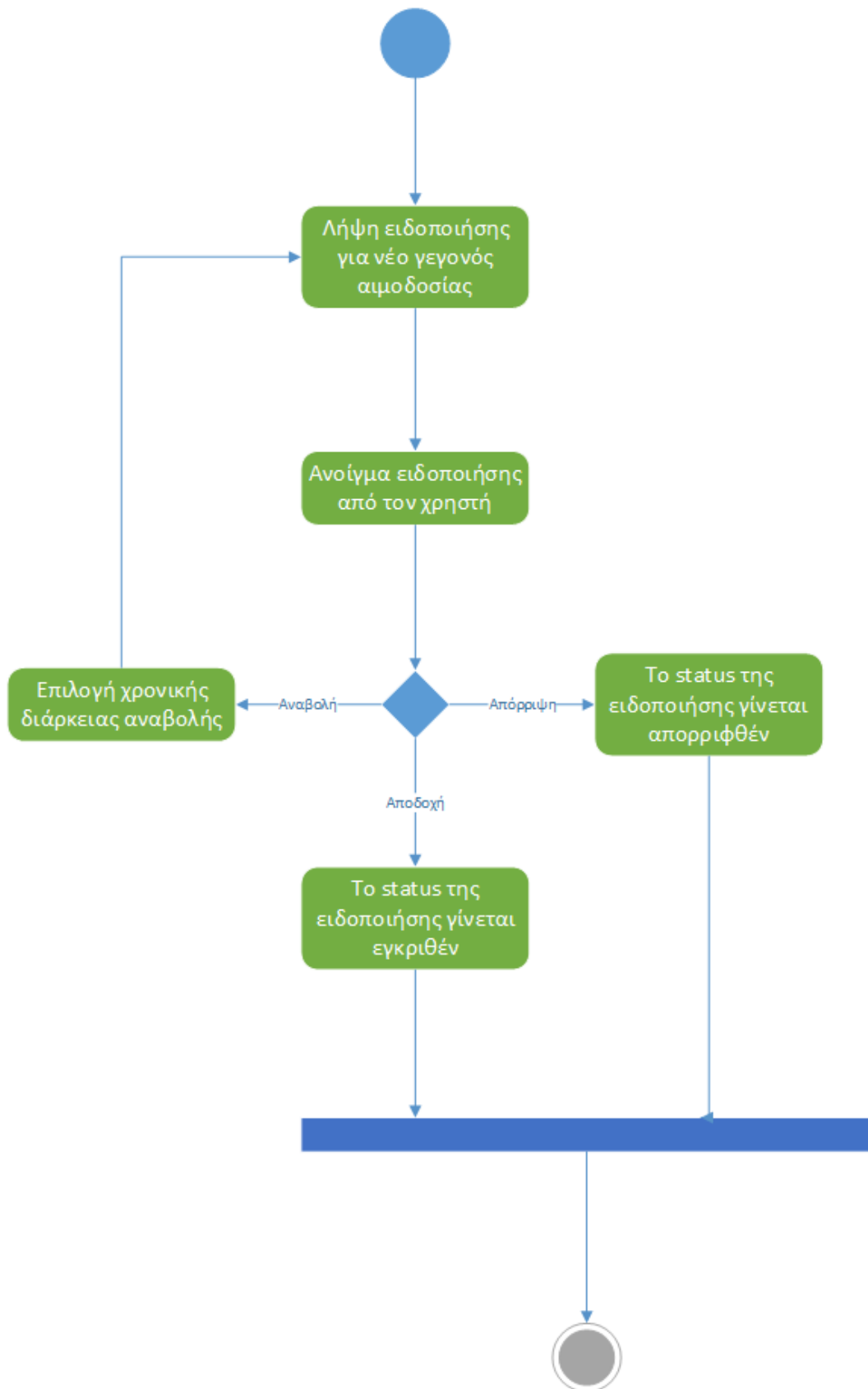


**Σχήμα 3.2:** Διάγραμμα δραστηριότητας #2. Είσοδος εθελοντή αιμοδότη στο σύστημα.

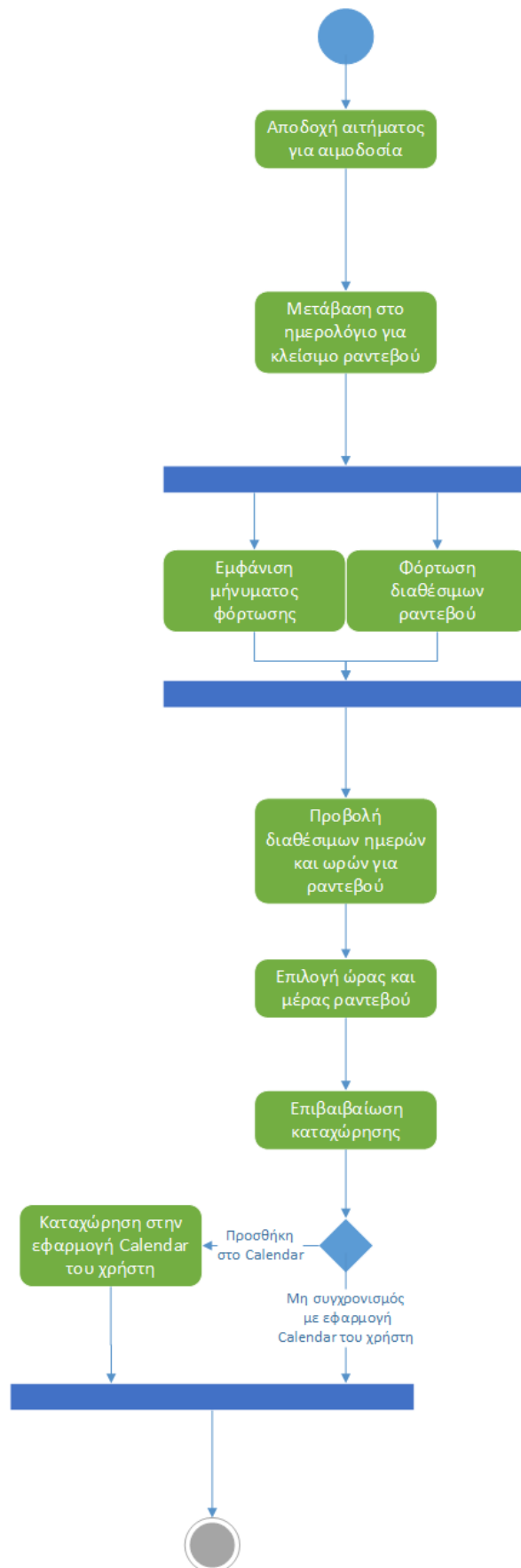


**Σχήμα 3.3:** Διάγραμμα δραστηριότητας #3. Προβολή γεγονότων αιμοδοσίας, από τον εθελοντή αιμοδότη.

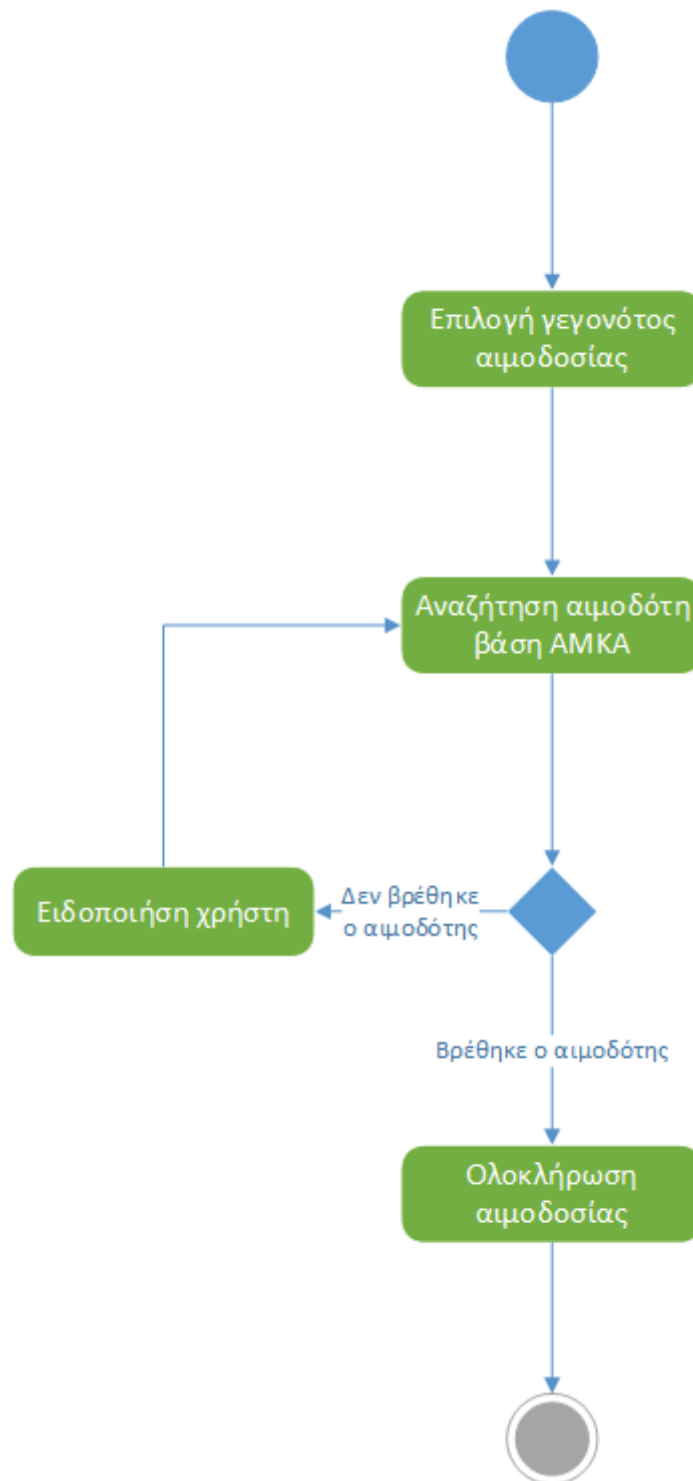




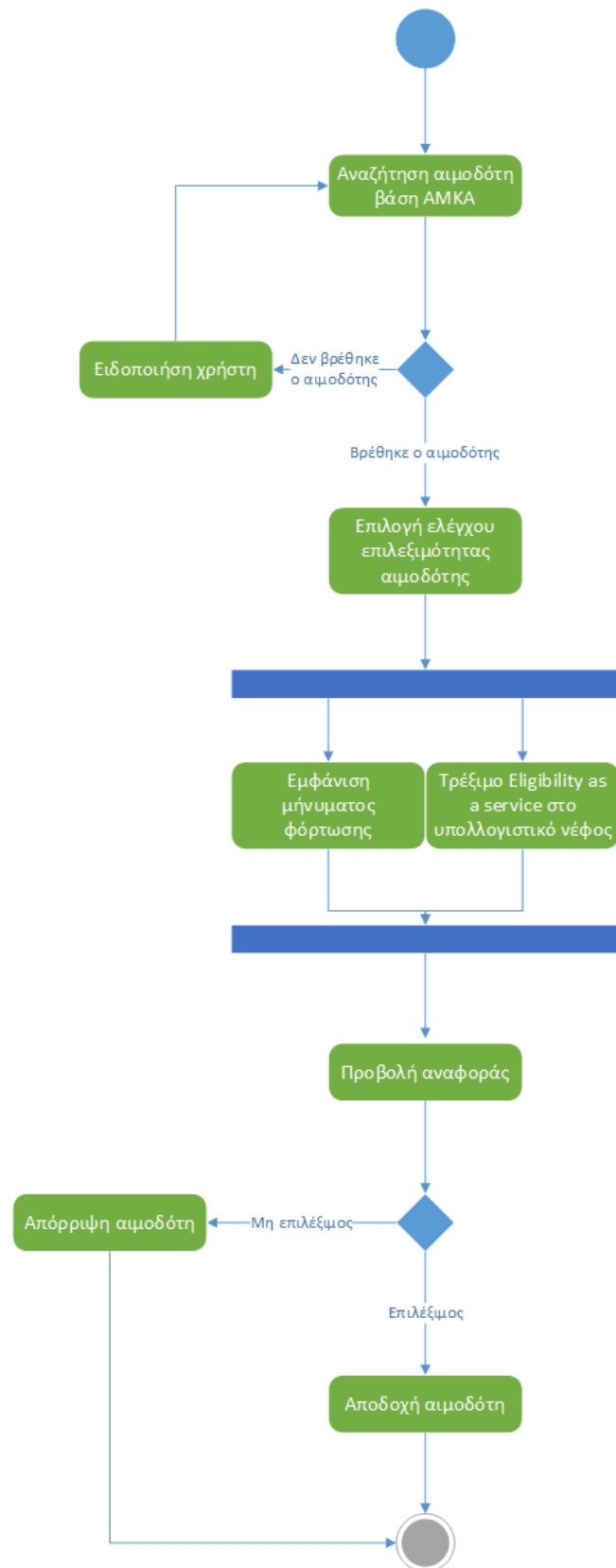
**Σχήμα 3.4:** Διάγραμμα δραστηριότητας #4. Διαχείριση των ειδοποιήσεων για τα γεγονότα αιμοδοσίας από τον εθελοντή αιμοδότη.



**Σχήμα 3.5:** Διάγραμμα δραστηριότητας #5. Κλείσιμο ραντεβού για αιμοδοσία από τον εθελοντή αιμοδότη.



**Σχήμα 3.6:** Διάγραμμα δραστηριότητας #6. Καταχώρηση αιμοδοσίας από τον εγκεκριμένο χρήστη/διαχειριστή του συστήματος αιμοδοσίας σύστημα.



**Σχήμα 3.7:** Διάγραμμα δραστηριότητας #7. Έλεγχος καταλληλότητας του υποψήφιου αιμοδότη από τον εγκεκριμένο χρήστη/διαχειριστή του συστήματος αιμοδοσίας.

## 3.4 Business Model

### 3.4.1 Gamification

Το gamification (παιχνιδοποίηση) ορίζεται ως “η ενσωμάτωση διάφορων πρακτικών και διαδικασιών παιχνιδιού σε καταστάσεις που δεν σχετίζονται με το παιχνίδι με στόχο τη λύση προβλημάτων μέσω της αύξησης της διαδραστικότητας και της συμμετοχής των χρηστών”[47][135]. Ως εκ τούτου, το gamification έχει βρει χρήση σε ένα ευρύ φάσμα εφαρμογών, από τον χώρο του marketing και της εκπαίδευσης μέχρι και τον χώρο της υγείας [80]. Το Gartner έχει εκτιμήσει ότι μέχρι το τέλος του 2015 περισσότερο από το 50% των επιχειρήσεων θα αξιοποιεί το gamification [65]. Στον ακαδημαϊκό χώρο συναντάμε όλο και περισσότερη έρευνα και δημοσιεύσεις με στόχο να μελετηθεί με μετρήσιμα, αντικειμενικά κριτήρια κατά πόσο είναι αποτελεσματική η χρήση των τεχνικών gamification.

#### Δομικά στοιχεία

Ένα παιχνίδι αποτελείται από έξι δομικά στοιχεία, όπως παρουσιάζονται στο σχήμα 3.8.



Σχήμα 3.8: Δομικά στοιχεία παιχνιδιών

Ένας από τους βασικούς στόχους ενός συστήματος gamification αποτελεί η αύξηση της ενασχόλησης του χρήστη. Τα στάδια με τα οποία ευελπιστεί να το πετύχει έχουν ως εξής:

1. Κίνητρα: Στην αρχή της διαδικασίας θα πρέπει να δοθεί στον χρήστη κάποιο κίνητρο το οποίο διαφέρει ανάλογα με τον τύπο χρήστη καθώς και τους στόχους του συστήματος. Παράδειγμα ενός τέτοιου κινήτρου αποτελεί η κοινωνική αναγνώριση [33].

2. Ενέργειες: είναι το δεύτερο στάδιο στο οποίο ο χρήστης οδηγείται μέσω των κινήτρων του πρώτου σταδίου. Σε αυτό το στάδιο ο χρήστης πραγματοποιεί την επιθυμητή από το σύστημα ενέργεια. Για παράδειγμα σε μια εφαρμογή εκμάθησης ξένων γλωσσών μια ενέργεια μπορεί να είναι η επανάληψη του λεξιλογίου.
3. Επιβραβεύσεις: Ύστερα από την επιτυχή ολοκλήρωση της ενέργειας ή των ενεργειών στο δεύτερο βήμα ο χρήστης λαμβάνει κάποια μορφή επιβράβευσης. Παραδείγματα επιβραβεύσεων που βρίσκουν ευρεία χρήση στο gamification αποτελούν τα εικονικά νομίσματα και τα εικονικά αγαθά.
4. Κατορθώματα: Στο τελευταίο αυτό στάδιο ο χρήστης φτάνει σε κάποιο κατόρθωμα το οποίο ενισχύει τα κίνητρα του και ο κύκλος επαναλαμβάνεται [156].

Η διαδικασία όπως την περιγράψαμε εμφανίζεται στο σχήμα 3.9.



**Σχήμα 3.9:** Κύκλος ενασχόλησης

## Σχεδιασμός συστήματος

Κατά τον σχεδιασμό του συστήματος gamification ακολουθήσαμε το πλαίσιο ΜΔΑ (MDA), το οποίο προέρχεται από τα αρχικά των λέξεων Μηχανισμοί (Mechanics), Δυναμικές (Dynamics) και Αισθητικά στοιχεία (Aesthetics) [133]. Σύμφωνα με το προαναφερθέν πλαίσιο τα βασικά στοιχεία που απαρτίζουν ένα παιχνίδι είναι οι κανόνες το σύστημα και η διασκέδαση όπως φαίνεται και στο σχήμα 3.10.

Ενώ για το κομμάτι του σχεδιασμού έχουμε το σχήμα 3.11:



Σχήμα 3.10: Στοιχεία παιχνιδιού



Σχήμα 3.11: Πλαίσιο σχεδίασης

Πριν προχωρήσουμε στην ανάλυση και παρουσίαση των σχεδιαστικών αποφάσεων που πήραμε, με σκοπό να γίνουν καλύτερα κατανοητές, κρίνουμε σκόπιμο να αναλύσουμε τις παραπάνω έννοιες δεδομένου ότι αποτελεί πολύ συχνό φαινόμενο η ύπαρξη σύγχυσης μεταξύ των εννοιών μηχανισμοί και δυναμικές.

- **Μηχανισμοί:** περιγράφουν συγκεκριμένα δομικά στοιχεία του παιχνιδιού σε επίπεδο αναπαράστασης δεδομένων και αλγορίθμων. Συνδυάζοντας διάφορους βασικούς μηχανισμούς μπορούμε να κατασκευάσουμε πιο πολύπλοκες δομές οι οποίες μπορούν να φέρουν το επιθυμητό αποτέλεσμα. Οι μηχανισμοί είναι αυτοί που υποστηρίζουν τις δυναμικές.
- **Δυναμικές:** περιγράφουν πως οι κανόνες του παιχνιδιού αλληλεπιδρούν με τον χρήστη και τους μηχανισμούς. Σε προγραμματιστική ορολογία μπορεί να περιγραφεί ως η συμπεριφορά του παιχνιδιού κατά την διάρκεια εκτέλεσης του. Ρόλος των δυναμικών είναι να κάνουν το παιχνίδι προοδευτικά δυσκολότερο αποφεύγοντας την ρουτίνα και διατηρώντας αμείωτο το ενδιαφέρον του χρήστη.
- **Αισθητικά στοιχεία:** περιγράφουν τα συναισθήματα που επιδιώκει το σύστημα να προξενήσει στον χρήστη του. Σύμφωνα με τους δημιουργούς του προτύπου MDA κάποια από αυτά είναι:
  - Αίσθηση ( Το παιχνίδι παρουσιάζεται σαν εσωτερική ευχαρίστηση).
  - Φαντασία (Το παιχνίδι σαν ιστορία με σκοπό να ταξιδέψει τον παίκτη μακριά από την πραγματικότητα).
  - Αφήγηση (Το παιχνίδι ως ιστορία με πλοκή).
  - Πρόκληση (Ο παίκτης πρέπει να ξεπεράσει κάποια εμπόδια).
  - Συνεργασία (Το παιχνίδι με την κοινωνική του υπόσταση).
  - Ανακάλυψη (Το παιχνίδι με κέντρο την ανακάλυψη του καινούριου).

- Έκφραση (Το παιχνίδι ως εξερεύνηση του εαυτού).
- Υποβολή (Το παιχνίδι ως απλό μέσο για αναψυχή κατά τον ελεύθερο χρόνο του ατόμου).

Κατά τον σχεδιασμό ενός συστήματος gamification, λαμβάνοντας υπόψιν τους στόχους του συστήματος γίνεται διαφορετικός συνδυασμός των παραπάνω στοιχείων. Οι πιο συχνά χρησιμοποιούμενοι μηχανισμοί είναι οι πόντοι, οι πίνακες κατάταξης καθώς και τα παράσημα. Στον πίνακα 3.23 βλέπουμε διάφορους μηχανισμούς καθώς και τους στόχους που θέλουν να πετύχουν [157][109].

Στόχος	Μηχανισμός
<b>Ενασχόληση</b>	Πίνακες Κατάταξης Επίπεδα Εικονικές επιβραβεύσεις προσπάθειας Ανταγωνισμός Αίσθηση συμμετοχής σε ομάδα Έλεγχος σε συμπαίκτες Διαφοροποίηση από συμπαίκτες Βοήθεια σε φίλο Χρονικοί περιορισμοί Λήξη χρόνου Περιορισμοί σε πόρους
<b>Αφοσίωση</b>	Περιορισμοί σε πρόσβαση Παράσημα Εικονικές επιβραβεύσεις
<b>Επίγνωση</b>	Προωθήσεις Λοταρίες

Πίνακας 3.23: Στόχοι και μηχανισμοί του Gamification

Για να γίνει ξεκάθαρος ο διαχωρισμός μεταξύ δυναμικών και μηχανισμών στον πίνακα 3.24 παρουσιάζουμε τους μηχανισμούς ενός παιχνιδιού με τις αντίστοιχες δυναμικές και κίνητρα [91].



Μηχανισμοί	Δυναμικές	Κίνητρα
Καταγραφή συμπεριφοράς	Εξερεύνηση	Περιέργεια
Συστήματα πόντων, εμβλήματα και επιβραβεύσεις	Συλλογή (εικονικών αγαθών)	Κατόρθωμα
Κατάταξη, επίπεδα	Ανταγωνισμός	Κοινωνική αναγνώριση
Ομαδικές δραστηριότητες	Συνεργασία	Επικοινωνία
Πίεση χρόνου, χρονική περιορισμοί	Πρόκληση	Διέγερση
Εικονικοί χαρακτήρες και κόσμοι	Ανάπτυξη, Οργάνωση	Αυτοδιάθεση

Πίνακας 3.24: Μηχανισμοί, δυναμικές και κίνητρα στο Gamification

### Gamification στο σύστημα μας

Έχοντας παρουσιάσει το θεωρητικό πλαίσιο του gamification, στην συνέχεια παρουσιάζουμε την χρήση του στο σύστημα που αναπτύξαμε στο πλαίσιο της παρούσας διπλωματικής. Μέσω του gamification επιδιώκουμε να πετύχουμε κυρίως τους δύο πρώτους στόχους της διπλωματικής όπως περιγράφηκαν στην ενότητα 1.1.

### Σύστημα επιβραβεύσεων

Οι επιβραβεύσεις δίνονται στον εθελοντή αιμοδότη (χρήστη του συστήματος) ανάλογα με την δραστηριότητα του και τους πόντους που έχει συγκεντρώσει. Οι επιβραβεύσεις μπορούν να χωριστούν σε δύο μεγάλες κατηγορίες:

- **Φυσικές Επιβραβεύσεις:** Ο χρήστης θα μπορεί να εξαργυρώνει τους πόντους που έχει μαζέψει για να κερδίσει υπηρεσίες ή προϊόντα από συνεργαζόμενες υπηρεσίες που τα έχουν διαθέσει τα πλαίσια κοινωνικής ευθύνης. Για παράδειγμα αν υπάρξει κάποια συνεργασία με την Cosmote μια μορφή φυσικής επιβράβευσης θα μπορούσε να είναι επιπλέον χρόνος ομιλίας.
- **Εικονικές Επιβραβεύσεις:** Η μορφή των εικονικών επιβραβεύσεων που χρησιμοποιήσαμε στο σύστημα μας είναι τα εμβλήματα, τα οποία ο χρήστης θα λαμβάνει για διάφορες δραστηριότητες που επιτελεί. Συγκεκριμένα ο χρήστης κερδίζει εμβλήματα όταν:
  - Εγγράφεται στο σύστημα ως εθελοντής αιμοδότης
  - Πραγματοποιεί την πρώτη του αιμοδοσία

- Συμπληρώνει τον μέγιστο αριθμό αιμοδοσιών που επιτρέπεται σε κάποιο ημερολογιακό έτος
- Καλεί κάποιον φίλο του να χρησιμοποιήσει την εφαρμογή
- Ανταποκρίνεται σε αιτήματα για αιμοδοσίες

Η παραπάνω λίστα είναι καθαρά ενδεικτική, δεδομένου ότι η δημιουργία εμβλημάτων αποτελεί μια δυναμική διαδικασία που έχει σκοπό να κρατήσει αμείωτο το ενδιαφέρον του εθελοντή. Για παράδειγμα ο διαχειριστής του συστήματος θα μπορεί να δημιουργεί ειδικά εμβλήματα για ένα συγκεκριμένο γεγονός αιμοδοσίας το οποίο έχει συγκεκριμένη ημερομηνία.

Σε αυτό το σημείο θα πρέπει να αναφέρουμε ότι, έχει αποτελέσει αντικείμενο τεράστιας διαμάχης το κατά πόσο πρέπει να δίνονται φυσικές επιβραβεύσεις στους εθελοντές και κατά πόσο αυτό αφαιρεί το νόημα της εθελοντικής αιμοδοσίας. Στην Ελλάδα μέχρι πρωτινός δεν επιτρεπόντουσαν κίνητρα που θα μπορούσαν να θεωρηθούν αντικαταστατό του χρήματος, αλλά ο πρόσφατος νόμος 4272/2014 άρθρο 27 δίνει την ελευθερία στον Ε.ΚΕ.Α. να ορίσει τέτοιου είδους κίνητρα σε συμφωνία με την οδηγία 2002/98 της Ευρωπαϊκής ένωσης.

### **Σύστημα πόντων**

Το σύστημα gamification που εφαρμόζουμε στην εφαρμογή που αναπτύχθηκε στα πλαίσια της παρούσας διπλωματικής περιλαμβάνει ολοκληρωμένο σύστημα πόντων. Ο χρήστης με το που εγγράφεται στο σύστημα λαμβάνει έναν μικρό αριθμό πόντων ως επιβράβευση της εγγραφής του ως εθελοντής αιμοδότης. Οι πόντοι μπορούν αν χωριστούν σε δύο επιμέρους κατηγορίες βάση της χρήσης τους:

- **Πόντοι εξαργύρωσης:** εξαργυρώνονται με σκοπό της απόκτησης κάποιας επιβράβευσης. Ανάλογα με τις δραστηριότητες που επιτελεί κερδίζει πόντους οι οποίοι αθροίζονται. Ενδεικτική λίστα για τις οποίες ο χρήστης θα μπορεί να κερδίζει πόντους:
  - Εγγραφή στο σύστημα.
  - Πραγματοποίηση αιμοδοσίας.
  - Πρόσκληση φίλων στην εφαρμογή.
  - Αλληλεπίδραση με τα κοινωνικά δίκτυα.

- **Πόντοι εμπειρίας:** ουσιαστικά είναι οι συνολικοί πόντοι που έχει αποκτήσει ο εθελοντής από την έναρξη χρήσης της εφαρμογής. Οι πόντοι εμπειρίας χρησιμοποιούνται για τον πίνακα κατάταξης μεταξύ των “φίλων” του, καθώς και για τον συνολικό πίνακα κατάταξης.

Κρίνεται σκόπιμο να αναφερθεί ότι οι πόντοι κατανέμονται με βάση την βαρύτητα της δραστηριότητας που εκτελεί ο εθελοντής αιμοδότης.

### **Πίνακες κατάταξης**

Οι πίνακες κατάταξης δείχνουν την επίδοση και την ‘πρόοδο’ ενός χρήστη σε σχέση με αυτή των ‘φίλων’ τους [108]. Υπάρχει ένας ενιαίος πίνακας στον οποίο και θα υπάρχουν όλοι οι εθελοντές που χρησιμοποιούν το σύστημα, εφόσον έχουν δηλώσει ότι επιθυμούν να συμμετέχουν στην κατάταξη. Ενώ υπάρχει και ένας δεύτερος πίνακας που δημιουργείται δυναμικά για κάθε χρήστη και περιλαμβάνει μόνο άτομα τα οποία ακολουθεί. Η κατάταξη διαμορφώνεται με βάση τους πόντους εμπειρίας που έχει μαζέψει κάθε χρήστης, οι οποίοι περιγράφηκαν παραπάνω. Για τους πρώτους στον κεντρικό πίνακα κατάταξης θα υπάρχει μέριμνα για κάποια ειδική επιβράβευση, ανάλογα τις συμφωνίες και χορηγίες που υπάρχουν με εταιρείες στα πλαίσια της κοινωνικής ευθύνης.

### **Συμπέρασμα**

Συμπερασματικά το σύστημα που περιγράψαμε παραπάνω μπορεί να βοηθήσει σημαντικά στον δεύτερο στόχο που θέσαμε στην εισαγωγή της διπλωματικής, στην ενότητα 1.1, δηλαδή στη διατήρηση και επιπλέον ενεργοποίηση των εθελοντών αιμοδοτών. Για την επίτευξη του πρώτου στόχου που θέσαμε δηλαδή τη στρατολόγηση περισσότερων νέων εθελοντών θεωρούμε πως μεγάλο ρόλο μπορεί να παίζει ο συνδυασμός του gamification με τα κοινωνικά δίκτυα όπως περιγράψαμε στην επόμενη ενότητα.

### **3.4.2 Social Networking Integration**

Η θεωρία της κοινωνικοποίησης των καταναλωτών προβλέπει ότι η επικοινωνία μεταξύ των καταναλωτών επηρεάζει τη γνωστική και συναισθηματική τους συμπεριφορά, καθώς και τις στάσεις τους απέναντι σε ένα προϊόν ή μια υπηρεσία[149]. Τα κοινωνικά δίκτυα μπορούν να επηρεάσουν δραστικά την συμπεριφορά ενός ατόμου[142] και οι υπάρχοντες εθελοντές μέσω των κοινωνικών δικτύων, είναι σε θέση να παίξουν τον ρόλο του πρέσβη προσελκύοντας νέους εθελοντές αιμοδότες. Η αξιοποίηση

των εθελοντών με στόχο την προαγωγή της εθελοντικής αιμοδοσίας, αποτελεί μια εξαιρετικά αποτελεσματική μέθοδο "στρατολόγησης" και αύξησης του αριθμού των αιμοδοσιών [106]. Πριν προχωρήσουμε στην ανάλυση των συγκεκριμένων ενεργειών που επιτελέσαμε με αυτό τον σκοπό, κρίνουμε σκόπιμο να αναφέρουμε μερικές εισαγωγικές έννοιες για τα κοινωνικά δίκτυα εν γένει.

Τα μέσα κοινωνικής δικτύωσης (Social Media), μέσω των οποίων επιτυγχάνεται η ηλεκτρονική κοινωνική δικτύωση, αποτελούν ουσιαστικά απόρροια του Web 2.0, το οποίο κατάφερε να αλλάξει την υφή του διαδικτύου προσδίδοντάς του μια πιο κοινωνική διάσταση. Ο όρος κοινωνικά δίκτυα (social media), αποτελεί μια γενική έννοια που χρησιμοποιείται για να περιγράψει διαδικτυακές εφαρμογές και υπηρεσίες στις οποίες οι χρήστες έχουν την δυνατότητα να δημιουργήσουν και να μοιραστούν μεταξύ τους περιεχόμενο [95]. Τα μέσα κοινωνικής δικτύωσης μπορούν να χωριστούν σε έξι κατηγορίες με βάση τον βαθμό της κοινωνικής παρουσίας που απαιτούν από τον χρήστη:

- Ιστολόγια (π.χ Blogger, Tumblr, Wordpress)
- Συνεργατικές ιστοσελίδες - έργα (π.χ Wikipedia)
- Ιστοσελίδες κοινωνικής δικτύωσης (π.χ Facebook)
- Κοινότητες διαμοιρασμού περιεχομένου (π.χ Reddit, Youtube)
- Εικονικοί κόσμοι (π.χ Second Life)
- Εικονικοί κόσμοι παιχνιδιών (π.χ World of Warcraft)

Τα μέσα κοινωνικής δικτύωσης βρήκαν τέτοια ευρεία αποδοχή και εδραιώθηκαν στην καθημερινότητα μας γιατί ικανοποιούν μια ισχυρή ανάγκη του ανθρώπου. Την ανάγκη του να συνεταιρίζεται, δημιουργώντας δίκτυα με άλλους ανθρώπους, ανταλλάζοντας ιδέες και απόψεις. Τα κύρια κίνητρα για τη χρήση των μέσων κοινωνικής δικτύωσης είναι τα εξής [18]:

- Κοινωνικοποίηση
- Ψυχική ευεξία
- Ατομική ικανοποίηση
- Κοινωνική αναγνώριση

Όπως αναφέραμε και παραπάνω ο κύκλος της ενασχόλησης (σχήμα 3.9) ξεκινάει από το κίνητρο. Ένα από τα βασικότερα κίνητρα του ανθρώπου είναι η επιθυμία για κοινωνική αναγνώριση και προ-

βολή [33] την οποία πολλές φορές επιδιώκει να ικανοποιήσει μέσω των κοινωνικών δικτύων (όπως φαίνεται από την παραπάνω λίστα).

Η εφαρμογή που αναπτύξαμε στα πλαίσια της διπλωματικής εργασίας δίνει την δυνατότητα στον εθελοντή (χρήστη) να μοιραστεί στα κοινωνικά δίκτυα τα επιτεύγματα του καθώς και πληροφορίες σε κάθε στάδιο της αιμοδοσίας, ικανοποιώντας με αυτό τον τρόπο το αίσθημα της κοινωνικής αναγνώρισης. Συγκεκριμένα υπάρχει άμεση διασύνδεση με το facebook και το twitter στα οποία ο χρήστης ενθαρρύνεται να δημοσιεύσει στις παρακάτω περιπτώσεις:

- Όταν λαμβάνει ειδοποίηση για γεγονός αιμοδοσίας, του δίνεται η δυνατότητα να το μοιραστεί στα μέσα κοινωνικής δικτύωσης. Με αυτό τον τρόπο αυξάνεται η προβολή του γεγονότος αιμοδοσίας και κατ' επέκταση οι πιθανότητες να καλυφθούν οι ανάγκες σε αίμα που έχουν τεθεί, ενώ προβάλλεται και η ίδια η εφαρμογή το οποίο οδηγεί στην στρατολόγηση περισσότερων εθελοντών αιμοδοτών.
- Όταν ολοκληρώνει κάποια αιμοδοσία ο χρήστης παροτρύνεται να βγάλει φωτογραφία και να τη μοιραστεί στα μέσα κοινωνικής δικτύωσης, προάγοντας την αιμοδοσία.
- Όταν ξεκλειδώνει κάποιο έμβλημα.

Επίσης ο χρήστης έχει τη δυνατότητα να προσκαλέσει τους φίλους του στην εφαρμογή. Σε αυτό το σημείο πρέπει να αναφερθεί ότι κάθε φορά που ο χρήστης αλληλεπιδρά μέσω της εφαρμογής με τα μέσα κοινωνικής δικτύωσης κερδίζει πόντους σύμφωνα με το σύστημα πόντων όπως περιγράφηκε στο 3.4.1 . Αξιοποιώντας τη δύναμη των κοινωνικών δικτύων είμαστε σε θέση να δώσουμε μεγαλύτερη προβολή και κοινωνικό χαρακτήρα στην εθελοντική αιμοδοσία. Μετατρέποντας ουσιαστικά τους εθελοντές αιμοδότες σε πρότυπα της εθελοντικής αιμοδοσίας με στόχο την προσέλκυση νέων εθελοντών αιμοδοτών.

### **3.4.3 Ειδοποιήσεις και Υπενθυμίσεις**

Όταν υπάρχει κάποια συγκεκριμένη ανάγκη για αίμα ή ένα καθορισμένο γεγονός αιμοδοσίας, το σύστημα στέλνει στους εθελοντές αιμοδότες νέες ειδοποιήσεις (push notifications). Οι ειδοποιήσεις αυτές δεν αποστέλλονται σε όλους τους χρήστες που είναι εγγεγραμμένοι στην εφαρμογή, αντιθέτως το σύστημα τρέχει κάποιους “έξυπνους αλγορίθμους” και επιλέγει την ομάδα χρηστών που τηρούν τις απαραίτητες προϋποθέσεις. Τα κριτήρια που ελέγχει το σύστημα αφορούν στο αν έχει παρέλθει το απαιτούμενο διάστημα μέχρι να μπορεί να δώσει ξανά αίμα ο εθελοντής, στην καταλληλότητα με

βάση τη γεωγραφική του τοποθεσία και στο αν έχει τη ζητούμενη ομάδα αίματος, στην περίπτωση που υπάρχουν .

Ο χρήστης έχει την επιλογή να αποδεχθεί, να απορρίψει και να αναβάλει την ειδοποίηση. Στην περίπτωση αποδοχής, ο χρήστης μπορεί να κλείσει ένα ραντεβού με το αντίστοιχο κέντρο αιμοδοσίας καθώς το σύστημα παρέχει ολοκληρωμένη διαχείριση ραντεβού. Η εφαρμογή έχει τη δυνατότητα να παρέχει υπενθυμίσεις εγκαίρως μέσα από πολλούς διαφορετικούς τρόπους, όπως μήνυμα ηλεκτρονικού ταχυδρομείου, επαναποστολή της ειδοποίησης και συγχρονισμός και καταχώρηση στο ημερολόγιο του χρήστη). Σε περίπτωση που ο χρήστης απορρίψει ένα αίτημα, αυτό μεταφέρεται και εμφανίζεται αυτόματα στο ιστορικό των χρηστών, στην ενότητα των απορριφθέντων αιτημάτων. Ως εκ τούτου ο χρήστης μπορεί να αναθεωρήσει τα αιτήματα του και να το εξετάσει σε μεταγενέστερο χρόνο.

Σύμφωνα με μελέτες που έχουν γίνει υπάρχει μεγάλος αριθμός αιμοδοτών που ενώ πληρούν τις απαραίτητες προϋποθέσεις και έχουν την θέληση να αιμοδοτήσουν, δεν προβαίνουν σε αιμοδοσία επειδή δεν υπάρχει κάποιος να τους το υπενθυμίσει [112]. Η αντιμετώπιση αυτού του προβλήματος επιτυγχάνεται με τις συστηματικές ειδοποιήσεις που λαμβάνει ο αιμοδότης, οι οποίες εκτός από μέσο για υπενθύμιση, δρουν ως κινητήριος δύναμη αφύπνισης και διατήρησης της εθελοντικής συνείδησης του αιμοδότη.

Ένα άλλο φαινόμενο το οποίο δυσχεραίνει ακόμα περισσότερο την ήδη δύσκολη κατάσταση στον τομέα των αιμοδοσιών είναι η προσωρινή απόρριψη των αιμοδοτών. Το φαινόμενο αυτό λαμβάνει χώρα σε περιπτώσεις που οι χρήστες αποκλείονται προσωρινά από την αιμοδοσία, λόγω του ότι δεν τηρούν τα κριτήρια ακαταλληλότητας (π.χ. πήραν ασπιρίνη την ημέρα της αιμοδοσίας). Από τον συνολικό αριθμό αιμοδοσιών το 14.13% αποκλείονται για λόγους προσωρινής ακαταλληλότητας [125]. Μελέτες έχουν δείξει ότι οι περισσότεροι εθελοντές δεν επιστρέφουν αν απορριφθούν μία φορά, συγκεκριμένα μόλις το 11% επιστρέφει μετά την απαιτούμενη περίοδο για να πραγματοποιήσει δωρεά αίματος [79]. Όταν τηρούνται οι κατάλληλες συνθήκες ώστε να είναι σε θέση ο αιμοδότης να προσφέρει πάλι αίμα, η εφαρμογή του στέλνει ειδοποιήσεις για τα γεγονότα αιμοδοσίας ώστε να τον παρακινήσει και να του υπενθυμίσει να δώσει αίμα.

Με βάση τα όσα προαναφέρθηκαν, η χρήση κατάλληλων ειδοποιήσεων και ως προς την χρονική συχνότητα και ως το προς τον παραλήπτη τους, βοηθάει στην τακτική συμμετοχή των εθελοντών σε αιμοδοσίες και συντελεί στην αντιμετώπιση κάποιων από τα προβλήματα που τους αποθαρρύνουν. Η σωστή διαχείριση και αποστολή των ειδοποιήσεων είναι ένα πολύ σημαντικό ζήτημα, καθώς δεν πρέπει να στέλνονται περιττές ή λανθασμένα επαναλαμβανόμενες ειδοποιήσεις, οι οποίες μπορεί να

έχουν ως αποτέλεσμα να κουράσουν τον χρήστη αλλά πρέπει η συχνότητα αποστολής τους να αποφέρει το βέλτιστο αποτέλεσμα. Στόχος του συστήματος ειδοποιήσεων είναι η ανταπόκριση να αυξηθεί με ικανοποιητικό ρυθμό και οι αιμοδοσίες να πολλαπλασιαστούν[112].

#### **3.4.4 Hemovigilance**

Ένα άλλο θέμα για το οποίο λαμβάνεται μέριμνα είναι το ζήτημα της αιμοεπαγρυπνήσης. Μία απαραίτητη προϋπόθεση για την αποτελεσματική εφαρμογή της αιμοεπαγρυπνήσης είναι η ύπαρξη συστήματος ιχνηλασιμότητας, δηλαδή πρέπει να εξασφαλίζεται η ικανότητα εντοπισμού καθεμίας από τις μονάδες αίματος ή καθενός από τα παράγωγα προϊόντα αίματος που προέρχονται από αυτήν, από τον εθελοντή αιμοδότη μέχρι τον τελικό προορισμό τους.

Το σύστημα μας, αρχικά καταγράφει τον αιμοδότη από τον οποίο προήλθε το αίμα. Στην συνέχεια, κάθε μονάδα αίματος παρακολουθείται με χρήση RFIDs και όταν τελικά φτάσει στον τελικό αποδέκτη το πληροφοριακό σύστημα του νοσοκομείου αποθηκεύει την πληροφορία αυτή στο πληροφοριακό του σύστημα. Με χρήση κατάλληλων πρωτοκόλλων και ασφαλούς διαύλου επικοινωνίας το σύστημα μας λαμβάνει από το πληροφοριακό σύστημα του νοσοκομείου το όνομα και το μοναδικό αναγνωριστικό του ασθενούς (ΑΜΚΑ) και τα αποθηκεύει, συσχετίζοντας τα με τα στοιχεία του αιμοδότη. Αυτή η διαδικασία έχει ως αποτέλεσμα να κρατείται πλήρες ηλεκτρονικό αρχείο το οποίο συσχετίζει σε περιπτώσεις ανεπιθύμητων συμβαμάτων τον αιμοδότη με τον αιμολήπτη. Η λειτουργία είναι ιδιαίτερα σημαντική για την ελαχιστοποίηση των ανεπιθύμητων κρουσμάτων και κατ'επέκταση για τη δημόσια υγεία.





## Κεφάλαιο 4

### Τεχνολογίες

#### 4.1 Back-End

##### 4.1.1 Cloud

###### Ευκαιρίες και προκλήσεις στον τομέα υγείας

Πρόσφατες έρευνες έδειξαν ότι το 75% των διευθυντών πληροφοριακών συστημάτων ανέφεραν ότι θα χρειαστεί να χρησιμοποιήσουν τις τεχνολογίες του υπολογιστικού νέφους στο άμεσο μέλλον [43][14]. Στον τομέα υγείας πολλοί οργανισμοί, διευθυντές και ειδικοί του χώρου πιστεύουν ότι η τεχνολογία του υπολογιστικού νέφους μπορεί να βελτιώσει σημαντικά τις παρεχόμενες υπηρεσίες υγείας, ενώ παράλληλα να μειώσει αισθητά τα λειτουργικά κόστη[53][138][22][93]. Επιπλέον μια αναφορά από τον Ευρωπαϊκό Οργανισμό για την Ασφάλεια Δικτύων και Πληροφοριών (ENISA) προβλέπει ότι ο νέος αυτός τεχνολογικός τομέας θα λάβει τεράστιες επενδύσεις στο χώρο της υγείας[16].

Παρακάτω στο 4.1 βλέπουμε σε μορφή πίνακα μια σύνοψη των ευκαιριών αλλά και των προκλήσεων που παρουσιάζει η τεχνολογία του υπολογιστικού νέφους για το χώρο της υγείας.

###### Θεωρητικό υπόβαθρο υπολογιστικού νέφους

Το cloud computing (υπολογιστικό νέφος) όπως ορίζεται από το NIST (Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας) είναι μια τεχνολογία, η οποία επιτρέπει την πρόσβαση από παντού, παρέχει μετά από αίτημα πρόσβαση στο δίκτυο για την κοινή χρήση προσαρμοστικών υπολογιστικών πόρων (π.χ., δίκτυα, servers, συστήματα αποθήκευσης, εφαρμογές και υπηρεσίες), μπορεί να ξεκινήσει και να αναπτυχθεί γρήγορα με ελάχιστη διαχείριση και χωρίς καμία αλληλεπίδραση με τον φορέα παροχής υπηρεσιών. [120] Το cloud computing παρέχει στους χρήστες και στις επιχειρήσεις διάφορες

Τομέας	Ευκαιρίες	Προκλήσεις
Διοίκησης	Χαμηλότερο κόστος υποδομών του IT Διάθεση υπολογιστικών πόρων on demand Σταδιακή πληρωμή ανάλογως τις ανάγκες	Έλειψη εμπιστοσύνης των προμηθευτών υγείας Οργανωτική αδράνεια Απώλεια άμέσου ελέγχου δεδομένων
Τεχνολογίας	Μείωση εργασιών συντήρησης του IT Ευελξία και επεκτασιμότητα υποδομών IT Οικολογική λύση	Προβλήματα εξάντλησης πόρων Απρόβλεπτη συμπεριφορά Απώλεια άμέσου ελέγχου δεδομένων
Ασφάλειας	Περισσότεροι πόροι για προστασία δεδομένων Αντιγραφή των δεδομένων σε πολλαπλές τοποθεσίες Δυναμικά επεκτάσιμοι πόροι ασφάλεια που αυξάνει την ελαστικότητα	Κατάχρηση προνομίων Ασφάλεια των φυσικών αποθηκευτικών μέσων Προβλήματα διαχείρισης κλειδιών κρυπτογράφησης

Πίνακας 4.1: Ευκαιρίες και Προκλήσεις cloud στον χώρο της υγείας

δυνατότητες για την αποθήκευση και την επεξεργασία των δεδομένων τους σε τρίτα κέντρα δεδομένων. [78]

Το cloud computing ουσιαστικά στηρίζεται στην κατανομή των πόρων με στόχο την επίτευξη συνοχής και οικονομίας μέσω ενός δικτύου και την μεγιστοποίηση της απόδοσης τους. Οι πόροι του cloud δεν είναι μόνο διαμοιραζόμενοι από πολλούς χρήστες αλλά μπορούν και να ανακαταμεμηθούν δυναμικά ανάλογα με την ζήτηση, γεγονός που μπορεί να χρησιμοποιηθεί όταν κατανέμονται στους διάφορους χρήστες. Για παράδειγμα, μια cloud εγκατάσταση ηλεκτρονικών υπολογιστών οι οποία εξυπηρετεί Ευρωπαίους χρήστες στην διάρκεια των Ευρωπαϊκών εργασιμών ωρών μπορεί να ανακαταναείμει τους ίδιους πόρους για να εξυπηρετήσει χρήστες στην Βόρεια Αμερική κατά τις εργασίμες ώρες της Βόρειας Αμερικής σε μία διαφορετική εφαρμογή. Αυτή η προσέγγιση βοηθά στην μεγιστοποίηση της χρήσης της υπολογιστικής ισχύος ενώ ταυτόχρονα μειώνεται το ολικό κόστος των φυσικών πόρων καθώς χρησιμοποιούμε λιγότερη ενέργεια, κλιματισμό κ.λ.π. για να διατηρείται το σύστημα. Με χρήση του cloud computing πολλοί χρήστες μπορούν να έχουν πρόσβαση σε έναν server και να ανακτούν και να ανανεώνουν τα δεδομένα τους χωρίς να αγοράζουν άδειες για όλες τις διαφορετικές εφαρμογές. [73]

Οι πάροχοι cloud συνήθως χρησιμοποιούν μοντέλο κοστολόγησης "πλήρωσε όσο χρησιμοποιήσεις". Παρουσιάζει σπουδαιότητα λοιπόν το γεγονός ότι οι "πελάτες" μπορούν να εξοικονομήσουν

χρηματικούς πόρους στο πέρασμα του χρόνου, καθώς χρησιμοποιώντας τους πόρους στο cloud, πληρώνουν μόνο για τους πόρους που χρησιμοποιούν και όχι για την αγορά και συντήρηση άφθονων μηχανημάτων (π.χ. εξυπηρετητών). Η λύση λοιπόν του cloud computing λοιπόν αποτελεί μία ιδιαίτερα ελκυστική προοπτική για τους χρήστες στους οποίους η χρήση των πόρων ποικίλλει σημαντικά καθώς και σε αυτούς στους οποίους η αγορά μηχανημάτων και το κόστος συντήρησης τους αποτελούν σημαντικό τμήμα του προϋπολογισμού.

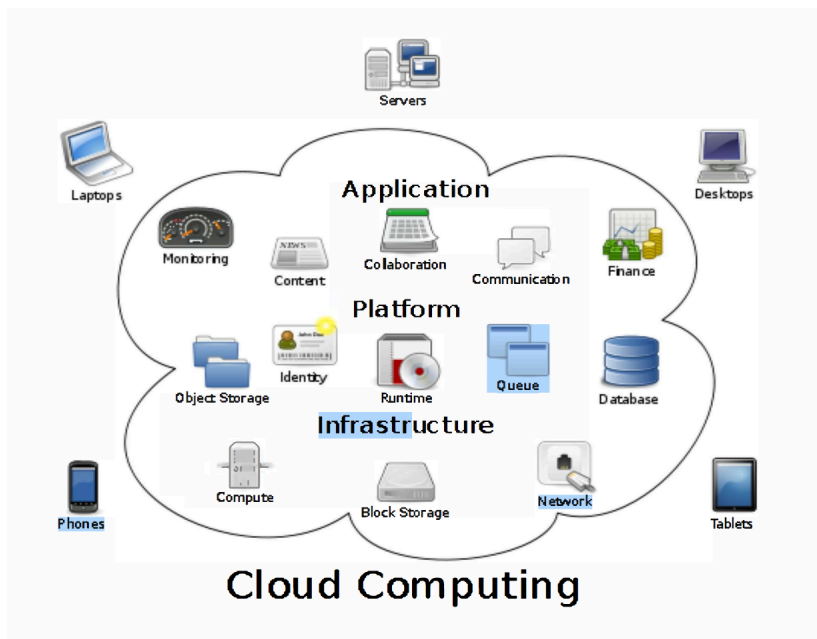
Ένα πλήρες σύστημα υπολογιστικού νέφους αποτελείται από τα εξής βασικά συστατικά :

- **Πελάτες - υπολογιστές.** Πρόκειται για τις συσκευές των τελικών χρηστών, μέσω των οποίων οι “πελάτες” αποκτούν πρόσβαση στο υπολογιστικό νέφος. Χωρίζονται σε τρεις κατηγορίες: τις κινητές συσκευές (mobile devices) που επιτρέπουν στον χρήστη την απομακρυσμένη πρόσβαση μετά από επεξεργασία του κατανεμημένου διακομιστή, τους παχείς πελάτες (thick clients) που είναι οι σταθεροί υπολογιστές ή τα λάπτοπ τα οποία συνδέονται στο cloud με χρήση ενός φυλλομετρητή και τους λεπτούς πελάτες (thin clients ) που είναι υπολογιστές με μικρή υπολογιστική ισχύ και χωρίς σκληρούς δίσκους, που εμφανίζουν τα δεδομένα αφού τα επεξεργαστεί πρώτα ο διακομιστής.
- **Κέντρο δεδομένων.** Είναι το σύνολο των διακομιστών, όπου τρέχουν οι εφαρμογές. Στο κέντρο δεδομένων emπίπτουν και οι εικονικοί διακομιστές (σε έναν υπολογιστή γίνεται εγκατάσταση ώστε να επιτρέπεται η ταυτόχρονη ύπαρξη πολλαπλών στιγμιotypών από εικονικούς διακομιστές).
- **Κατανεμημένους διακομιστές.** Οι διακομιστές αυτοί βρίσκονται σε διαφορετικές γεωγραφικές θέσεις, έτσι ώστε να εξασφαλίζεται μεγαλύτερη ευελιξία και υψηλότερη ασφάλεια στις παροχές που προσφέρει το cloud computing. Έτσι μπορούν να αντιμετωπιστούν αποτελεσματικά περιπτώσεις παραβίασης ασφάλειας ή ανάγκης επιπλέον αποθηκευτικού χώρου και υπολογιστικής δύναμης.

Στο σχήμα 4.1, φαίνεται μία σχηματική αναπαράσταση του συστήματος cloud computing.

Το μοντέλο cloud αποτελείται από πέντε βασικά χαρακτηριστικά:

- **Αυτοεξυπηρέτηση μετά από ζήτηση (On-demand self-service).** Ο χρήστης των υπηρεσιών του νέφους μπορεί να χρησιμοποιεί τους υπολογιστικούς πόρους που χρειάζεται, όπως χρόνο στον server και αποθηκευτικό χώρο, αυτόματα και χωρίς να απαιτείται ανθρώπινη αλληλεπίδραση με τον φορέα παροχής υπηρεσιών.



Σχήμα 4.1: Cloud Computing.

- Ευρεία πρόσβαση στο δίκτυο (Broad network access). Οι δυνατότητες που προσφέρονται είναι διαθέσιμες μέσω του δικτύου και προσβάσιμες μέσω τυποποιημένων μηχανισμών που προωθούν την χρήση ετερογενών πλατφορμών λεπτών ή παχέων πελατών (π.χ., κινητά τηλέφωνα, τάμπλετ, υπολογιστές).
- Διάθεση των πόρων. Οι υπολογιστικοί πόροι των cloud παρόχων είναι συγκεντρωμένοι με χρήση ενός μοντέλου πολλαπλών ενοικιαστών (κάθε οργανισμός εργάζεται με ένα προσαρμοσμένο εικονικό στιγμιότυπο της εφαρμογής), έτσι ώστε να μπορούν να ανατεθούν οι φυσικοί και οικονομικοί πόροι ανάλογα με την ζήτηση των καταναλωτών. Οι παρεχόμενοι πόροι μπορεί να περιλαμβάνουν πόρους όπως μνήμη, επεξεργαστική ισχύ, εύρος δικτύου, εικονικές μηχανές.
- Μεγάλη ελαστικότητα (Rapid Elasticity). Οι δυνατότητες του cloud computing παρέχονται με μεγάλη ταχύτητα και προσαρμοστικότητα, ώστε να είναι σε θέση να κλιμακώνουν πολύ γρήγορα ανάλογα με την ζήτηση που υπάρχει και να φαίνονται απεριόριστες σε οποιαδήποτε ποσότητα και χρονική στιγμή ζητηθούν.
- Μετρούμενη υπηρεσία (Measured Service). Τα cloud computing συστήματα πραγματοποιούν μετρήσεις, σε συγκεκριμένα επίπεδα αφαίρεσης ανάλογα με το είδος της υπηρεσίας (επεξεργασία, μνήμη, ενεργοί λογαριασμοί χρηστών). Με βάση αυτές τις μετρήσεις γίνεται έλεγχος και ενέργειες για τη βελτιστοποίηση της χρήσης των πόρων. Επιπλέον η χρήση των πόρων παρακολουθείται, ελέγχεται και καταγράφεται με αποτέλεσμα την παροχή διαφάνειας μεταξύ του χρήστη και του παρόχου του cloud.

[67][69].

Το cloud computing μπορεί να διαχωριστεί με βάση τις υπηρεσίες που προσφέρει σε διάφορες κατηγορίες. Οι πιο βασικές, είναι α) η υποδομή ως μία υπηρεσία (IaaS), β) η πλατφόρμα ως μία υπηρεσία (PaaS) γ) το λογισμικό ως μία υπηρεσία (SaaS) και το δ) backend των κινητών σαν υπηρεσία (MBaaS), που είναι γνωστό σαν backend σαν υπηρεσία (BaaS).

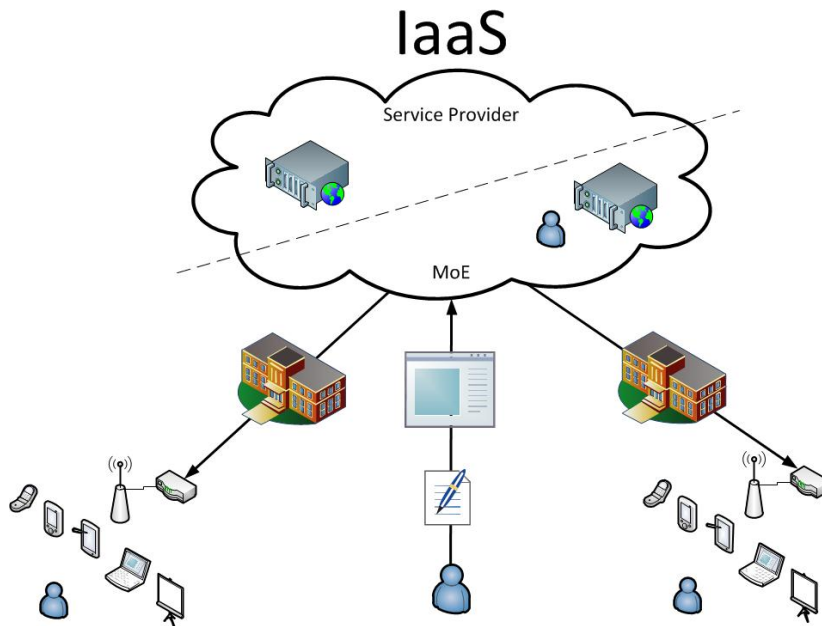
## IaaS

Η υποδομή σαν υπηρεσία προσφέρει πρόσβαση σε υπολογιστικούς πόρους στο εικονικό περιβάλλον του cloud, σε εικονικό υλικό (hardware). Το εικονικό υλικό περιλαμβάνει εικονικό χώρο server, συνδέσεις δικτύου, εύρος ζώνης, διευθύνσεις IP και εξισορροπιστές φορτίου. Το σύνολο των πόρων του υλικού προκύπτει από ένα πλήθος servers και δικτύων που είναι καταναμημένοι συνήθως σε διάφορα κέντρα δεδομένων και ο πάροχος cloud είναι υπεύθυνος για τη διατήρησή τους. Στον χρήστη δίνεται πρόσβαση στους πόρους του cloud ώστε να μπορεί να χτίσει τις δικές του πλατφόρμες πληροφορικής.

Οι IaaS υπηρεσίες μπορούν σε συνδυασμό με άλλες υπηρεσίες του cloud να χρησιμοποιηθούν για τη δημιουργία οικονομικά συμφέρουσων και εύκολα επεκτάσιμων λύσεων πληροφορικής, όπου οι δυσκολίες και τα έξοδα διαχείρισης έχουν ανατεθεί στον πάροχο cloud. Όταν οι δραστηριότητες του χρήστη κλιμακώνονται, και επιθυμεί μία επέκταση ή σμίκρυνση των υπολογιστικών πόρων, τότε μπορεί να αντλήσει ή να εγκαταλείψει πόρους cloud χωρίς να χρειάζεται να αγοραστεί, να εγκατασταθεί και να ενσωματωθεί νέο υλικό από τον ίδιο. [136]

Η υπηρεσία IaaS προσφέρει επεκτασιμότητα. Οι πόροι είναι διαθέσιμοι την χρονική στιγμή που τους χρειάζεται ο πελάτης και έτσι αποφεύγονται καθυστερήσεις στην επέκταση της παραγωγικής ικανότητας. Επιπλέον, δεν υπάρχει η ανάγκη για επενδύσεις στο υλικό (hardware). Το "φυσικό" υλικό, το οποίο βρίσκεται πίσω από την IaaS υπηρεσία, έχει στηθεί και συντηρείται από τον πάροχο cloud, με αποτέλεσμα να εξοικονομείται χρόνος και χρήμα στην πλευρά του χρήστη. Ένα βασικό πλεονέκτημα είναι ότι ο πελάτης κοστολογείται με βάση τη χρήση και πληρώνει μόνο τους πόρους που χρησιμοποιεί. Οι υπηρεσίες μπορούν να προσεγγισθούν από οποιαδήποτε τοποθεσία, αρκεί ο χρήστης να έχει πρόσβαση στο ίντερνετ και τα κατάλληλα διαπιστευτήρια, γεγονός που εξασφαλίζει ανεξαρτησία σε σχέση με την τοποθεσία. Τέλος, στις υπηρεσίες IaaS δεν υπάρχει περίπτωση κατάρρευσης ή βλάβης στο σύστημα, καθώς ακόμα και αν ένας server ή ένας διακόπτης δικτύου πέσει, η ευρύτερη υπηρεσία θα παραμείνει ανεπηρέαστη λόγω των υπόλοιπων πόρων υλικού καθώς και των επιπλέον ρυθμίσεων.

Στις περισσότερες περιπτώσεις, ακόμα και αν έφευγε ολόκληρο το κέντρο δεδομένων, και συνέχιζε να λειτουργεί μόνο ένας server, αυτό θα ήταν αρκετό ώστε οι IaaS υπηρεσίες να εξακολουθούν να τρέχουν ανεμπόδιστα. Ένα μειονέκτημα που υπάρχει είναι ότι σε περιπτώσεις που υπάρχει ανάγκη αλληλεπίδρασης υψηλής ταχύτητας μεταξύ του εσωτερικού λογισμικού ή του λογισμικού που υπάρχει σε κάποιο άλλο cloud και του παρόχου υπηρεσιών IaaS Cloud, μέσω μίας Internet σύνδεσης, μπορεί να μην γίνεται να επιτευχθεί η ταχύτητα που χρειάζεται. [42]



**Σχήμα 4.2:** Υπηρεσίες IaaS cloud computing.

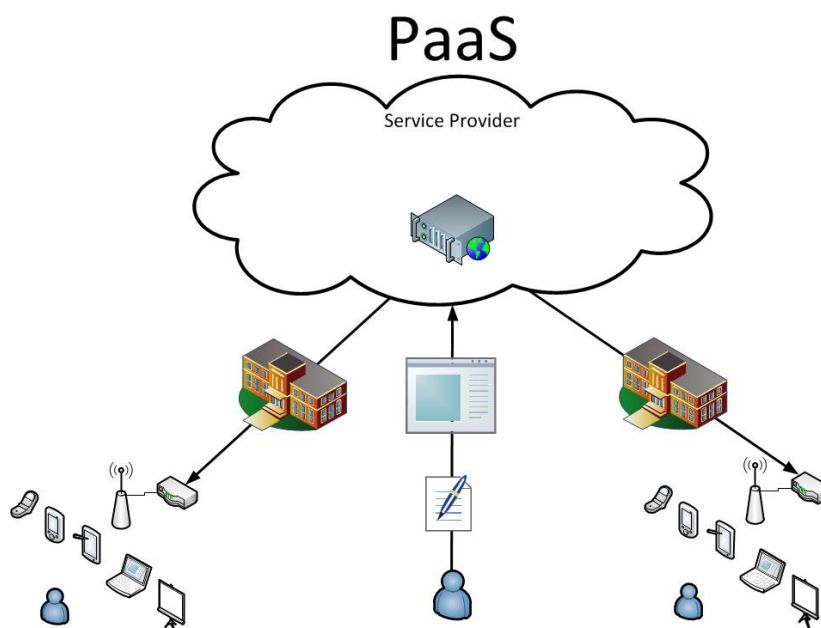
## PaaS

Η πλατφόρμα σαν υπηρεσία παρέχει στους πελάτες του υπολογιστικού νέφους όλους τους πόρους και τα εργαλεία που απαιτούνται για να έχουν την δυνατότητα να αναπτύξουν, να διαχειριστούν και να τρέξουν εφαρμογές στο διαδίκτυο χωρίς την πολυπλοκότητα και την ανάγκη εγκατάστασης της υποδομής (λογισμικού) που συνήθως συνδέονται με την ανάπτυξη και την έναρξη μιας εφαρμογής. Οι πλατφόρμες σαν υπηρεσίες προσφέρονται σε δύο διαφορετικές μορφές: είτε σαν δημόσιες cloud υπηρεσίες από τον πάροχο, στις οποίες ο χρήστης ελέγχει την ανάπτυξη του λογισμικού και τις ρυθμίσεις διαμόρφωσης, ενώ ο πάροχος εξασφαλίζει το απαραίτητο υπόβαθρο που χρειάζεται η εφαρμογή, είτε σαν λογισμικό, που εγκαθίσταται σε ιδιωτικά κέντρα δεδομένων ή δημόσια υποδομή ως υπηρεσία και διοικείται από εσωτερικά συστήματα πληροφορικής. [127]

Στο μοντέλο PaaS ο χρήστης δεν ελέγχει την υποκείμενη cloud υποδομή, η οποία συμπεριλαμβάνει

νει το δίκτυο, τους servers, το λειτουργικό σύστημα και τους αποθηκευτικούς χώρους αλλά έχει τον πλήρη έλεγχο στις εγκατεστημένες ή αναπτυγμένες εφαρμογές καθώς και στις ρυθμίσεις των παραμέτρων που γίνονται στο περιβάλλον. Το μοντέλο PaaS βασίζεται σε χρήση της HTML ή Javascript. Υπάρχουν διάφοροι τύποι του μοντέλου PaaS, δημόσιοι, ιδιωτικοί και υβριδικοί. Σε αυτού του τύπου το μοντέλο παροχής υπηρεσιών cloud απαιτούνται δύο επίπεδα προστασίας για τη διασφάλιση της ασφάλειας της ιδιωτικής ζωής. Στο χαμηλότερο επίπεδο του συστήματος, ο πάροχος cloud μπορεί να παρέχει βασικούς μηχανισμούς ασφαλείας, όπως end-to-end κρυπτογράφηση, έλεγχο ταυτότητας και εξουσιοδότηση. Στο υψηλότερο επίπεδο, ο χρήστης των υπηρεσιών cloud χρειάζεται να καθορίσει τις πολιτικές ελέγχου της πρόσβασης στην εφαρμογή.[94]

Στα κύρια πλεονεκτήματα του μοντέλου PaaS συγκαταλέγεται αρχικά το γεγονός ότι επιτρέπει τον προγραμματισμό υψηλού επιπέδου. Επιπροσθέτως, καθιστά την ολική ανάπτυξη μίας εφαρμογής πιο αποτελεσματική, εφόσον έχει ενσωματωμένη δομή, και τη συντήρηση-βελτίωση της πιο εύκολη. Ένα άλλο σημαντικό πλεονέκτημα αποτελεί η υψηλή διαθεσιμότητα, η ελαστικότητα και η ευελιξία που προσφέρει σε συνδυασμό με τις ικανότητες πλήρους αυτοδιαχείρισης και αυτό-κλιμάκωσης της υποδομής και της πλατφόρμας εφαρμογών. Βασικό μειονέκτημα αποτελεί η έλλειψη δια-λειτουργικότητας και η ανικανότητα μεταφοράς της εφαρμογής από έναν πάροχο σε κάποιον άλλον.



Σχήμα 4.3: Υπηρεσίες PaaS cloud computing.

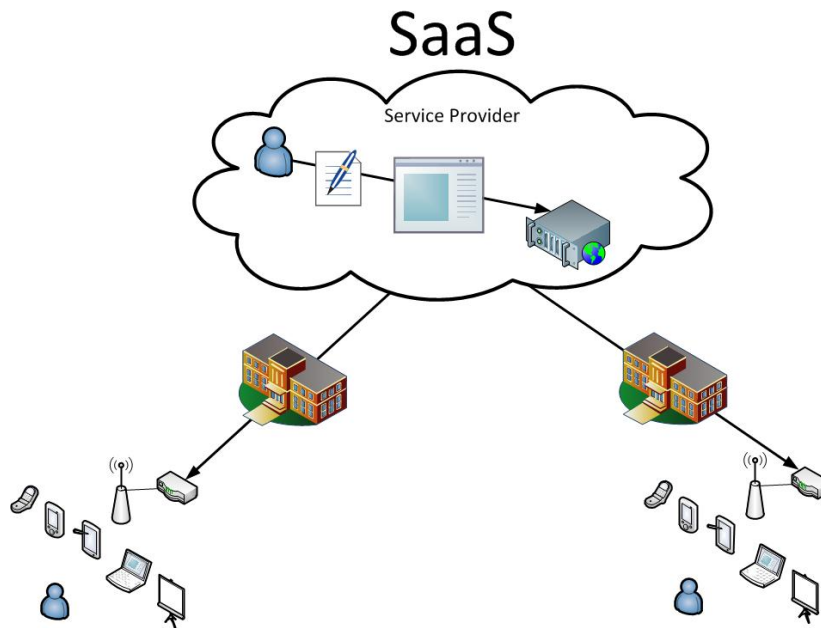
## SaaS

Το λογισμικό σαν υπηρεσία είναι ένα μοντέλο ανάπτυξης λογισμικού σύμφωνα με το οποίο παρέχονται κατόπιν ζήτησης στον χρήστη εφαρμογές και υπολογιστικοί πόροι. Ουσιαστικά, είναι όλες οι υπηρεσίες του cloud, μέσω των οποίων, οι χρήστες είναι σε θέση να αποκτήσουν πρόσβαση σε εφαρμογές μέσω σύνδεσης στο Internet. Οι εφαρμογές αυτές βρίσκονται στο cloud και μπορούν να χρησιμοποιηθούν για ένα ευρύ φάσμα δραστηριοτήτων, τόσο από εταιρείες όσο και από άτομα. Η χρήση των εφαρμογών αυτών αποτελεί περισσότερο ενοικίαση λογισμικού παρά αγορά του. Με την παραδοσιακές διαδικασία για τις εφαρμογές λογισμικού αρχικά αγοράζεται το λογισμικό ως πακέτο και εγκαθίσταται σε έναν υπολογιστή, ενώ συνήθως οι σχετικές άδειες περιορίζουν τον αριθμό των χρηστών και των συσκευών όπου μπορεί να αναπτυχθεί το λογισμικό. Εν αντιθέσει, οι χρήστες των υπηρεσιών SaaS εγγράφονται στο λογισμικό, χωρίς να το αγοράσουν και επομένως γλιτώνοντας όλη αυτή τη διαδικασία, και ανανεώνουν τη συνδρομή τους συνήθως σε μηνιαία βάση.

Οι περισσότεροι χρήστες χρησιμοποιούν έναν λεπτό πελάτη (thin client) για να αποκτήσουν πρόσβαση στις υπηρεσίες SaaS. Οι εταιρικοί χρήστες χρησιμοποιούν τις εφαρμογές που παρέχει ο πάροχος cloud για ανάγκες όπως λογισμικό γραφείου, λογισμικό ανταλλαγής μηνυμάτων, λογισμικό λογιστικής και τιμολόγησης, λογισμικό παρακολούθησης επιδόσεων, λογισμικό διαχείρισης επιχειρησιακών σχέσεων και άλλα.

Βασικά πλεονεκτήματα των υπηρεσιών SaaS είναι η ευελιξία και η επεκτασιμότητα που προσφέρουν, καθώς οι χρήστες δεν χρειάζεται να αγοράσουν άλλο λογισμικό ή άλλον server, απλά ενεργοποιούν νέες προσφορές του μοντέλου SaaS. Πολύ σημαντικό είναι επίσης, η υψηλή ποιότητα των υπηρεσιών η υψηλή σταθερότητα και το γεγονός ότι απαιτείται ελάχιστη συντήρηση. Υπάρχει μεγάλο χρονικό κέρδος, καθώς στις υπηρεσίες SaaS το λογισμικό είναι ήδη εγκατεστημένο και ρυθμισμένο και η εφαρμογή μπορεί να διατεθεί στον χρήστη έτοιμη για χρήση. Η πολιτική κοστολόγησης της χρήσης των υπηρεσιών είναι ιδιαίτερα συμφέρουσα συγκριτικά με την αγορά του και πολλές φορές επιτρέπει την χρήση εφαρμογών που σε άλλη περίπτωση θα ήταν απαγορευμένες, λόγω ιδιαίτερα υψηλού κόστους. Θεωρείται αξιόπιστη λύση ασφαλείας, καθώς υιοθετεί SSL (Secure Socket Layer) στις υπηρεσίες του με αποτέλεσμα τα δεδομένα να μεταδίδονται από και προς τον χρήστη με ασφάλεια. Βασικό μειονέκτημα αποτελεί το γεγονός ότι παρέχει πρόσβαση βασισμένη σε ένα δίκτυο που αποτελείται από τις περισσότερες εμπορικές εφαρμογές με αποτέλεσμα να μην μπορεί πάντα κάποιος να βρει το λογισμικό που ψάχνει διαθέσιμο στο SaaS.





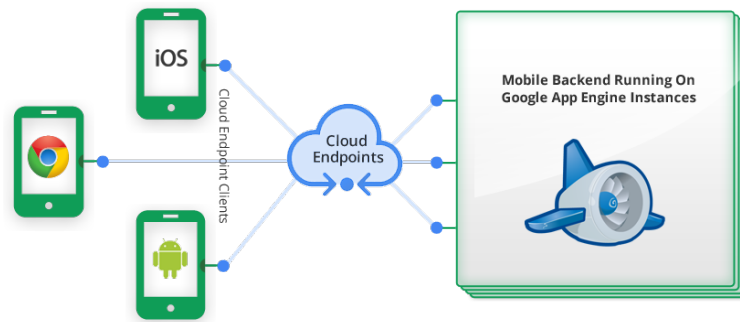
**Σχήμα 4.4:** Υπηρεσίες SaaS cloud computing.

## BaaS

Η υπηρεσία BaaS είναι ένα μοντέλο που παρέχει στους προγραμματιστές διαδικτυακών εφαρμογών και εφαρμογών για κινητά τηλέφωνα έναν τρόπο εύκολης και γρήγορης διασύνδεσης των εφαρμογών, μέσω παρεχόμενων APIs. Παρέχει υπηρεσίες όπως διαχείριση χρηστών, ειδοποιήσεις (push notifications), κώδικα για server και σύνδεση με υπηρεσίες κοινωνικών δικτύων. Όλες αυτές οι υπηρεσίες αξιοποιούνται μέσω χρήσης εργαλείων ανάπτυξης λογισμικού (SDKs) και έχουν την δική τους διεπαφή προγραμματισμού εφαρμογών (APIs), που τους επιτρέπει να ενσωματωθούν στις εφαρμογές χωρίς ιδιαίτερο κόπο. Η παροχή ενός σταθερού τρόπου για την διαχείριση των backend δεδομένων συνεπάγεται το γεγονός ότι οι προγραμματιστές δεν χρειάζεται να αναπτύξουν κώδικα backend για κάθε υπηρεσία που χρησιμοποιεί ή έχει πρόσβαση η εφαρμογή τους.

Ένα από τα κύρια πλεονεκτήματα της χρήσης υπηρεσιών BaaS είναι η τεράστια μείωση του χρόνου ολοκλήρωσης και συντήρησής της εφαρμογής, σε ποσοστό μέχρι και 60%. Επιπλέον ευνοείται η κλιμάκωση, καθώς το BaaS παρέχει ένα συγκεκριμένο backend το οποίο διαμορφώνει κάποιος εύκολα με βάση τις ανάγκες της εφαρμογής. Η ανάλυση της εφαρμογής γίνεται πιο αποδοτική, καθώς το BaaS διαχειρίζεται όλη τη "κίνηση" που γίνεται μέσα στην εφαρμογή και αναλύει τι συμβαίνει σε αυτήν (ποιοι χρήστες είναι πιο ενεργοί, ποιοι είναι πιο εμπλεκόμενοι κ.λ.π.). Παρά τα πολλά οφέλη του BaaS, είναι επίσης σημαντικό να ληφθεί υπόψη η κατασκευή του user-interface (UI), διότι αυτό βρίσκεται σε άμεση επικοινωνία με τους τελικούς χρήστες. Ο ρόλος του UI είναι να συνδέεται η εφαρμογή με άλλα ανεξάρτητα κομμάτια ή με αποκλειστικά APIs που συνδέονται στο backend, κάτι το

οποίο με χρήση των υπηρεσιών BaaS δεν είναι δυνατό πολλές φορές καθώς μια εφαρμογή "κλειδώνεται" στον πάροχο.



Σχήμα 4.5: Υπηρεσίες BaaS cloud computing.

Στην συνέχεια θα αναπτυχθούν τα μοντέλα ανάπτυξης του cloud computing.

## Public cloud

Το δημόσιο cloud computing είναι ένα μοντέλο μέσω του οποίου οι χρήστες έχουν πρόσβαση στο cloud μέσω διεπαφών που χρησιμοποιούν τα προγράμματα περιήγησης. Ένα cloud καλείται δημόσιο, όταν οι υπηρεσίες του παρέχονται μέσω ενός δημόσιου δικτύου, όπως το Internet και είναι ανοιχτές για δημόσια χρήση. Τα δημόσια cloud παρέχουν υπηρεσίες σε πολλούς πελάτες χρησιμοποιώντας την ίδια διαμοιραζόμενη υποδομή. Πολλά δημόσια cloud δεν χρεώνουν τον χρήστη.

Οι υπηρεσίες SaaS όπως η αποθήκευση στο cloud και οι online εφαρμογές γραφείου είναι οι υπηρεσίες που βρίσκονται κατά κόρον σε δημόσια cloud. Οι υπηρεσίες υποδομής (IaaS) καθώς και οι πλατφόρμες σαν υπηρεσία (PaaS), κυρίως οι περιπτώσεις του cloud web hosting και των περιβαλλόντων ανάπτυξης, αναπτύσσονται εξίσου συχνά και σε δημόσια και σε ιδιωτικά cloud. Τα δημόσια cloud χρησιμοποιούνται ευρέως σε προσφορές για ιδιώτες οι οποίοι είναι λιγότερο πιθανόν να χρειαστούν το επίπεδο των υποδομών και την ασφάλεια που προσφέρουν τα ιδιωτικά cloud. Ωστόσο, σε πολλές περιπτώσεις οι επιχειρήσεις μπορούν να χρησιμοποιούν δημόσια cloud σε περιπτώσεις που δεν τίθεται θέμα ευαίσθητων δεδομένων (π.χ. webmail, online συνεργασία) για να κάνουν τις λειτουργίες τους πιο αποτελεσματικά [90].

Τα βασικά πλεονεκτήματα του δημόσιου cloud είναι:

- Απόλυτη επεκτασιμότητα. Το γεγονός ότι οι υπολογιστικοί πόροι στο cloud είναι "ανεξάντητοι" και γίνονται διαθέσιμοι ανάλογα με τη ζήτηση έχει ως αποτέλεσμα οι εφαρμογές που τρέ-

χουν σε αυτούς να μπορούν να ανταποκριθούν στις διακυμάνσεις της δραστηριότητας.

- Υψηλή απόδοση. Τα δημόσια cloud συνεπάγονται υψηλότερα επίπεδα υπολογιστικών πόρων με ταυτόχρονα λιγότερο κόστος. Μερικές προτάσεις μαζικής αγοράς μπορούν ακόμη και να είναι δωρεάν για τον πελάτη και να στηρίζονται στις διαφημίσεις για τα έσοδα τους (Google).
- Το μοντέλο πληρωμής ανάλογα με τη χρήση. Ο χρήστης είναι σε θέση να αποκτήσει πρόσβαση στους πόρους που χρειάζεται, όποτε θέλει και να πληρώσει μόνο για αυτούς που χρησιμοποιεί αποφεύγοντας τους περιττούς πόρους.
- Ευελιξία. Υπάρχει ένας τεράστιος αριθμός των IaaS, PaaS και SaaS υπηρεσιών που διατίθενται στην αγορά, οι οποίες ακολουθούν το μοντέλο του δημόσιου cloud και τις οποίες ο χρήστης μπορεί να προσεγγίσει από οποιαδήποτε συσκευή με χρήση Internet. Οι υπηρεσίες αυτές μπορούν να εκπληρώσουν τις περισσότερες υπολογιστικές απαιτήσεις και μπορούν να χρησιμοποιηθούν εξίσου αποτελεσματικά από ιδιώτες και επιχειρήσεις.
- Ανεξαρτησία της τοποθεσίας. Η διαθεσιμότητα των υπηρεσιών του δημόσιου cloud μέσω του Internet, εξασφαλίζει ότι οι υπηρεσίες είναι διαθέσιμες όπου και αν βρίσκεται ο χρήστης. Αυτό το γεγονός πολλές φορές είναι ιδιαίτερα χρήσιμο όπως για παράδειγμα σε περιπτώσεις εταιριών, όπου δίνεται η δυνατότητα για απομακρυσμένη πρόσβαση σε υποδομές πληροφορικής (σε περίπτωση έκτακτης ανάγκης, κ.λ.π.).

## Private cloud

Το ιδιωτικό cloud, είναι ένα μοντέλο το οποίο λειτουργεί αποκλειστικά για έναν οργανισμό, διευθύνεται είτε από τον ίδιο τον οργανισμό είτε από τρίτους και εγκαθίσταται είτε εσωτερικά είτε εξωτερικά. Περιλαμβάνει ένα ξεχωριστό και ασφαλές περιβάλλον με το οποίο μπορούν να αλληλεπιδράσουν οι χρήστες. Τα ιδιωτικά cloud παρέχουν μεγάλη υπολογιστική ισχύ σαν υπηρεσία χρησιμοποιώντας ένα σύνολο από φυσικούς υπολογιστικούς πόρους, οι οποίοι είναι προσβάσιμοι μόνο από έναν οργανισμό, εξασφαλίζοντας περισσότερο έλεγχο και αυξημένη ασφάλεια. Τα δημόσια cloud περιλαμβάνουν πολλούς πελάτες οι οποίοι έχουν πρόσβαση σε εικονικές υπηρεσίες που αντλούν πόρους από την ίδια ομάδα από servers μέσω δημοσίων δικτύων. Τα ιδιωτικά cloud από την άλλη πλευρά, περιλαμβάνουν την οριοθέτηση ενός cloud για την αποκλειστική χρήση από έναν οργανισμού. Παρόλο που χρησιμοποιούνται ξεχωριστοί πόροι, μπορούν να γίνουν προσβάσιμοι μέσω ιδιωτικών γραμμών ή ασφαλών κρυπτογραφημένων συνδέσεων δημόσιου δικτύου [50] [110].

Τα ιδιωτικά cloud είναι κατάλληλα για περιπτώσεις στις οποίες υπάρχει η ανάγκη αποθήκευσης

και επεξεργασίας ιδιωτικών ή ευαίσθητων δεδομένων ή η ανάγκη διεξαγωγής ευαίσθητων εργασιών. Για παράδειγμα, μία ιδιωτική cloud υπηρεσία μπορεί να χρησιμοποιηθεί από ένα πληροφοριακό σύστημα που διαχειρίζεται και αποθηκεύει ευαίσθητα ιατρικά δεδομένα και θέλει να επωφεληθεί από τα πλεονεκτήματα του cloud computing στην επιχειρησιακή υποδομή του. Υιοθετεί τις βασικές αρχές ασφαλείας, νομοθεσίας και κανονιστικών απαιτήσεων αφού προσφέρει μεγάλο έλεγχο εγκατάστασης και χρήσης στην επιχείρηση που το χρησιμοποιεί. Το ιδιωτικό μοντέλο cloud μοιάζει αρκετά με το μοντέλο των δικτύων τοπικής πρόσβασης (LAN) που χρησιμοποιούταν στο παρελθόν αλλά παρέχει τα επιπλέον πλεονεκτήματα του cloud. Κάποια από τα πλεονεκτήματα του ιδιωτικού cloud είναι:

- Μεγαλύτερη προστασία και ασφάλεια. Τα δημόσια cloud εφαρμόζουν μηχανισμούς παροχής ασφαλείας, αλλά τα ιδιωτικά cloud με τους αποκλειστικούς φυσικούς πόρους, οι οποίοι επιτρέπουν πρόσβαση μόνο σε συνδέσεις που γίνονται μέσω συγκεκριμένου οργανισμού και προσωπικές ασφαλείς γραμμές εξασφαλίζουν ότι υπάρχει πλήρης ασφάλεια και αξιοπιστία.
- Περισσότερο έλεγχο με αποτέλεσμα περισσότερη ευελιξία. Εφόσον η πρόσβαση στους πόρους γίνεται μόνο από έναν οργανισμό, αυτός ο οργανισμός έχει την ικανότητα να ρυθμίσει και να διαχειριστεί το cloud όπως επιθυμεί, ώστε να επιτευχθεί το κατάλληλο δίκτυο.
- Βελτίωση της σχέσης κόστους και ενεργειακής απόδοσης. Η εφαρμογή ενός ιδιωτικού μοντέλου cloud μπορεί να συντελέσει στην καλύτερη κατανομή των πόρων σε έναν οργανισμό, εξασφαλίζοντας ότι η διαθεσιμότητα των πόρων ανταποκρίνεται άμεσα και ευέλικτα ανάλογα με τη ζήτηση τους. Παρόλο που είναι πιο ακριβά σε σχέση με τα δημόσια cloud, κάνουν πιο αποτελεσματική χρήση των υπολογιστικών πόρων σε σχέση με τα παραδοσιακά τοπικά δίκτυα (LANs), καθώς ελαχιστοποιείται η επένδυση σε αχρησιμοποίητη παραγωγική ικανότητα.
- Βελτιωμένη αξιοπιστία. Ακόμα και αν οι πόροι (servers, δίκτυα, κ.λ.π.) φιλοξενούνται εσωτερικά στον πελάτη, η δημιουργία εικονικών περιβάλλοντων λειτουργικού συνεπάγεται ότι το δίκτυο είναι πιο ανθεκτικό σε επιμέρους αστοχίες από ότι η φυσική υποδομή.

## **Community cloud**

Το κοινοτικό cloud είναι ένα μοντέλο νέφους το οποίο μοιράζεται σε διάφορους οργανισμούς και υποστηρίζει μια συγκεκριμένη κοινότητα που έχει κοινά ενδιαφέροντα (ασφάλειας, συμμόρφωσης, της διεθνούς δικαιοδοσίας, κλπ). Η διαχείριση του νέφους μπορεί να γίνει είτε εσωτερικά από τον ίδιο τον οργανισμό είτε από τρίτους και υλοποιείται είτε εντός είτε εκτός του οργανισμού. Οι δαπάνες κατανέμονται σε λιγότερους χρήστες από ότι σε ένα δημόσιο cloud (περισσότερους βέβαια από ότι σε

ένα ιδιωτικό cloud), και έτσι πραγματοποιούνται μόνο μερικές από τις δυνατότητες εξοικονόμησης κόστους του cloud computing [31] [39].

Το κοινοτικό cloud φιλοδοξεί να συνδυάσει την υπολογιστική παροχή των πόρων που διανέμονται από το δίκτυο, τον κατανεμημένο έλεγχο από τα ψηφιακά οικοσυστήματα και τη βιωσιμότητα της "πράσινης" πληροφορικής, με τις περιπτώσεις χρήσης του cloud computing, κάνοντας μεγαλύτερη χρήση των πλεονεκτημάτων της αυτοδιαχείρισης του υπολογιστικού συστήματος. Τα πλεονεκτήματα του κοινοτικού cloud περιλαμβάνουν:

- Χαμηλότερο κόστος. Το κόστος της δημιουργίας ενός κοινοτικού cloud έναντι των ιδιωτικών cloud μπορεί να είναι χαμηλότερο λόγω της κατανομή των κοστών σε όλους τους συμμετέχοντες.
- Ανάθεση της διαχείρισης σε έναν πάροχο cloud. Το πλεονέκτημα έγκειται στο γεγονός ότι ο πάροχος θα είναι ένας αμερόληπτος τρίτος που δεσμεύεται από σύμβαση και δεν έχει καμία προτίμηση σε κάποιον από τους πελάτες που εμπλέκονται.

Βασικό μειονέκτημα αποτελεί το γεγονός ότι το κόστος είναι πολύ υψηλότερο από ότι στα δημόσια cloud [68].

## **Hybrid cloud**

Τα υβριδικά cloud είναι μία ολοκληρωμένη υπηρεσία που αποτελείται από δύο ή περισσότερα cloud (ιδιωτικά, δημόσια ή κοινοτικά) τα οποία παραμένουν ξεχωριστές οντότητες αλλά συνδέονται μεταξύ τους, προσφέροντας τα πλεονεκτήματα των πολλαπλών μοντέλων ανάπτυξης. Μια υπηρεσία υβριδικού cloud ξεπερνά τα όρια του παρόχου σε σχέση με τις άλλες υπηρεσίες cloud, με αποτέλεσμα να μην μπορεί να κατηγοριοποιηθεί απλά σε μία κατηγορία ιδιωτικών, δημόσιων ή κοινοτικών υπηρεσιών cloud. Με τα υβριδικά cloud μπορεί να επεκταθεί η ικανότητα ή η χωρητικότητα μιας υπηρεσίας μέσω της ένταξης ή της προσαρμογής της με κάποια άλλη υπηρεσία cloud [130].

Η επιλογή υλοποίησης ενός υβριδικού cloud εξαρτάται από έναν αριθμό παραγόντων, όπως οι απαιτήσεις ασφάλειας των δεδομένων και των λειτουργιών πάνω σε αυτά, το επίπεδο ελέγχου που απαιτείται, καθώς και τις εφαρμογές που χρησιμοποιεί ο χρήστης. Όλες οι υπηρεσίες του cloud computing προσφέρουν βελτίωση της αποτελεσματικότητας, αλλά οι υπηρεσίες των δημόσιων cloud υπερτερούν σε σχέση με αυτές των ιδιωτικών λόγω χαμηλότερου κόστους και καλύτερης κλιμάκωσης. Ως εκ τούτου, ένας οργανισμός μπορεί να μεγιστοποιήσει την αποτελεσματικότητα του χρησιμοποιώντας υπηρεσίες από δημόσια cloud για όλες τις μη-ευαίσθητες λειτουργίες, στηριζόμενος σε χρήση

ιδιωτικού cloud μόνο όταν κρίνεται τελείως απαραίτητο, και διασφαλίζοντας ότι όλες οι πλατφόρμες συνεργάζονται άψογα μεταξύ τους [155].

Το εξειδικευμένο μοντέλο του υβριδικού cloud, το οποίο είναι χτισμένο πάνω σε ετερογενές hardware, ονομάζεται "Cross-platform Hybrid Cloud". Συνήθως χρησιμοποιούνται διαφορετικές αρχιτεκτονικές επεξεργαστών, για παράδειγμα, x86-64 και ARM. Οι χρήστες μπορούν να αναπτύξουν εφαρμογές με πλήρη διαφάνεια, χωρίς να γνωρίζουν την πολυμορφία του υλικού του cloud. Αυτό το είδος cloud έχει προκύπτει από την αύξηση των βασισμένων σε ARM system-on-chip αρχιτεκτονικών για servers.

Ένα ειδικά διαμορφωμένο υβριδικό cloud, προσφέρει τα εξής πλεονεκτήματα:

- **Επεκτασιμότητα.** Ενώ τα ιδιωτικά cloud προσφέρουν δυνατότητα κλιμάκωσης η οποία εξαρτάται από τις ρυθμίσεις τους (όπως το αν φιλοξενούνται εσωτερικά ή εξωτερικά), οι υπηρεσίες των δημοσίων cloud computing προσφέρουν πολύ αυξημένες δυνατότητες επεκτασιμότητας, διότι οι πόροι τους προέρχονται από μεγαλύτερες υποδομές. Υλοποιώντας τις μη-ευαίσθητες λειτουργίες σε δημόσιο cloud μπορεί ένας οργανισμός να επωφεληθεί από την επεκτασιμότητα του δημόσιου cloud και ταυτόχρονα να μειώσει τις απαιτήσεις στο ιδιωτικό cloud.
- **Εξοικονόμηση κόστους.** Τα δημόσια cloud προσφέρουν με χρήση υπηρεσιών (όπως η κεντρική διαχείριση) μεγαλύτερη αποδοτικότητα κόστους, από τα ιδιωτικά cloud. Ως εκ τούτου, τα υβριδικά σύννεφα επιτρέπουν στους οργανισμούς να έχουν πρόσβαση σε αυτές τις συμφέρουσες υπηρεσίες για όσες από τις λειτουργίες τους είναι εφικτό, διατηρώντας ωστόσο τις ευαίσθητες λειτουργίες ασφαλείς στο ιδιωτικό cloud.
- **Ασφάλεια.** Το ιδιωτικό cloud του μοντέλου υβριδικού cloud παρέχει την απαραίτητη ασφάλεια για τις ευαίσθητες λειτουργίες και επιπλέον σε πολλές περιπτώσεις τηρεί τις προϋποθέσεις για τον χειρισμό και την αποθήκευση δεδομένων.
- **Ευελιξία.** Η διαθεσιμότητα τόσο των ασφαλών πόρων του ιδιωτικού cloud και όσο και των οικονομικότερων των δημοσίων cloud, παρέχει στους οργανισμούς περισσότερες ευκαιρίες για να εξερευνήσουν διαφορετικές επιχειρησιακές κατευθύνσεις.

#### 4.1.2 NodeJs

Το Node.js, ή απλούστερα Node, είναι μια πλατφόρμα που μας επιτρέπει να γράψουμε JavaScript στην πλευρά του εξυπηρετητή (server-side JavaScript). Από το 2009, που δημιουργήθηκε, μέχρι σή-

μερα η πλατφόρμα έχει εξελιχθεί ραγδαία και χρησιμοποιείται σε πολλές μεγάλες εφαρμογές. Αποτελεί αυτήν την στιγμή το δεύτερο σε επισκεψιμότητα πρότζεκτ στο GitHub και έχει παραπάνω από 90000 πακέτα (modules) που έχουν δημοσιευτεί στον διαχειριστή πακέτων του Node (Node Package Manager, NPM). Οι NodeJs εφαρμογές είναι γραμμένες σε JavaScript και μπορούν να εκτελεστούν σε OS X, Microsoft Windows, Linux, FreeBSD, NonStop, IBM AIX, IBM System z και IBM i. Το πρότζεκτ του Node φιλοξενείται και υποστηρίζεται από το Node.js Foundation, ένα συλλογικό έργο του Linux Foundation [147].

Το Node είναι μια οδηγούμενη από γεγονότα (event-driven) πλατφόρμα, στην οποία μια διεργασία σε αντίθεση με τα περισσότερα σύγχρονα περιβάλλοντα ανάπτυξης εφαρμογών δεν στηρίζεται στην πολυνηματικότητα αλλά χρησιμοποιεί λειτουργίες εισόδου/εξόδου χωρίς αναμονή (non blocking I/O), ή ισοδύναμα χρησιμοποιεί ασύγχρονη E/E (asynchronous I/O) . Αυτό έχει ως σκοπό να αυξήσει την απόδοση και τη δυνατότητα κλιμάκωσης (scalability) των διαδικτυακών εφαρμογών πραγματικού χρόνου. Το Node χρησιμοποιεί την V8, μια εικονική μηχανή που αναπτύχθηκε από τη Google για τον Google Chrome και τρέχει κώδικα JavaScript. Η V8 δίνει στο Node μεγάλη αύξηση στην απόδοση, καθώς κάνει απευθείας μεταγλώττιση σε γλώσσα μηχανής αντί να εκτελεί bytecode ή να χρησιμοποιεί διερμηνευτή (interpreter).

Το Node επιτρέπει τη δημιουργία web servers και εργαλείων δικτύωσης με χρήση της γλώσσας JavaScript και μια συλλογή από πακέτα (modules), τα οποία χειρίζονται διάφορες λειτουργίες του πυρήνα [143]. Τα πακέτα αυτά χειρίζονται το σύστημα αρχείων E/E, το σύστημα δικτύωσης (HTTP, TCP, UDP, DNS, ή TLS / SSL), τα δυαδικά δεδομένα (buffers)] και άλλες βασικές λειτουργίες. Τα πακέτα του NodeJs χρησιμοποιούν ένα API το οποίο έχει σχεδιαστεί για να μειώσει την πολυπλοκότητα της ανάπτυξης εφαρμογών για διακομιστή.

Χρησιμοποιείται κυρίως για την κατασκευή προγραμμάτων δικτύου, όπως διακομιστές δικτύου (web servers), γεγονός που το καθιστά παρόμοιο με την PHP και την Python. Η μεγαλύτερη διαφορά μεταξύ PHP και Node είναι ότι η PHP είναι μια γλώσσα με αναμονή, όπου η εκτέλεση των εντολών προχωρά μόνο όταν έχουν ολοκληρωθεί οι προηγούμενες, ενώ στο Node οι εντολές εκτελούνται παράλληλα και χρησιμοποιούνται κλήσεις με επιστροφή (callbacks) για να σηματοδοτηθεί η ολοκλήρωση. Το Node υλοποιεί οδηγούμενο από γεγονότα προγραμματισμό για διαδικτυακές εφαρμογές σε JavaScript. Οι προγραμματιστές μπορούν να δημιουργήσουν εξαιρετικά επεκτάσιμους servers χωρίς τη χρήση νημάτων, χρησιμοποιώντας ένα απλοποιημένο οδηγούμενο από γεγονότα μοντέλο προγραμματισμού που χρησιμοποιεί κλήσεις επιστροφή για να σηματοδοτήσει την ολοκλήρωση ενός στόχου[119].

## JavaScript

Η JavaScript αναπτύχθηκε αρχικά από τον Brendan Eich του Netscape και ονομαζόταν Μόκα, η οποία αργότερα μετονομάστηκε σε LiveScript, και τελικά σε JavaScript. LiveScript ήταν το επίσημο όνομα για τη γλώσσα, όταν εισήχθη για πρώτη φορά σε beta εκδόσεις του Netscape Navigator 2,0 αλλά μετονομάστηκε σε JavaScript σε μια κοινή ανακοίνωση με την Sun Microsystems όταν προοριζόταν να αναπτυχθεί στη νέα έκδοση του Netscape Navigator 2.0B3.

Η JavaScript, επίσης γνωστή ως ECMAScript, είναι μία πρότυπη αντικειμενοστραφής scripting γλώσσα η οποία είναι δυναμική και έχει πρώτης τάξεως λειτουργίες. Η JavaScript είναι μια εφαρμογή του προτύπου γλώσσας ECMAScript και χρησιμοποιείται κυρίως με τη μορφή client-side, όπου υλοποιείται ως μέρος ενός web browser, ώστε να παρέχεται ενισχυμένη διεπαφή χρηστών και δυναμικές ιστοσελίδες. Αυτό επιτρέπει την πρόσβαση μέσω προγραμματισμού στα υπολογιστικά αντικείμενα μέσα σε ένα περιβάλλον υποδοχής. Η JavaScript χρησιμοποιεί σύνταξη επηρεασμένη από αυτή της C. Πολλά ονόματα και συμβάσεις ονοματολογίας έχουν αντιγραφεί από τη Java, αλλά οι δύο γλώσσες είναι διαφορετικές και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού στο πλαίσιο της JavaScript, λαμβάνονται από Self και Scheme γλώσσες προγραμματισμού. Είναι μια γλώσσα συγγραφής σεναρίων (scripting language) για την προσθήκη διαδραστικότητας (interactivity) σε ιστοσελίδες.

Με την JavaScript μπορούμε να φτιάξουμε σενάρια που να εκτελούν αυτόματες εργασίες, για παράδειγμα όταν μια σελίδα του Web ανοίγει ή κλείνει. Επίσης μπορούμε να χρησιμοποιήσουμε την JavaScript για να εκτελέσει ενέργειες που να ανταποκρίνονται σε ένα συγκεκριμένο γεγονός. Για παράδειγμα, όταν ο χρήστης επιλέγει ένα κουμπί, έναν σύνδεσμο ή όταν επιλέγει να εστιάσει από ένα στοιχείο μιας φόρμας σε ένα άλλο στοιχείο της, εκτελείται κώδικας JavaScript.

Κάθε φυλλομετρητής υλοποιεί τον δικό του διερμηνευτή για την JavaScript (συνήθως καλείται μηχανή JavaScript), ο οποίος μάλιστα είναι και καθοριστικής σημασίας για την απόδοση του φυλλομετρητή όσον αφορά στη συμβατότητα, σταθερότητα, ταχύτητα και ασφάλεια. Συνήθως ο κώδικας JavaScript διερμηνεύεται, δε μεταγλωττίζεται. Εντούτοις για να βελτιωθεί η απόδοση, υπάρχουν φυλλομετρητές που ακολουθούν μεθόδους just-in-time μεταγλώττισης, όπως ο Google Chrome με μια καινοτόμο μηχανή JavaScript, την V8.

Η σύνταξη JavaScript είναι κοινή μεταξύ των διάφορων φυλλομετρητών, παρόλο που υπάρχουν διαφορές στην ονοματολογία των βασικών μεθόδων. Η σύνταξη θυμίζει php, με τη διαφορά ότι δεν χρησιμοποιείται το χαρακτηριστικό «\$» στα ονόματα των μεταβλητών. Πρόκειται για μια γλώσσα χαλαρών τύπων όπου οι μεταβλητές μπορούν να αλλάξουν τύπο δυναμικά. Αλφαριθμητικά μεταβλητού



μεγέθους υποστηρίζονται εγγενώς. Για τη δήλωση μιας μεταβλητής χρησιμοποιείται η λέξη-κλειδί "var" ενώ για τον ορισμό συνάρτησης χρησιμοποιείται η λέξη "function" και δεν απαιτείται να δηλωθεί ο τύπος του αποτελέσματος. Δεν υπάρχουν δείκτες, ενώ υποστηρίζονται τάξεις.

### **Javascript στον Εξυπηρετητή (server-side Javascript)**

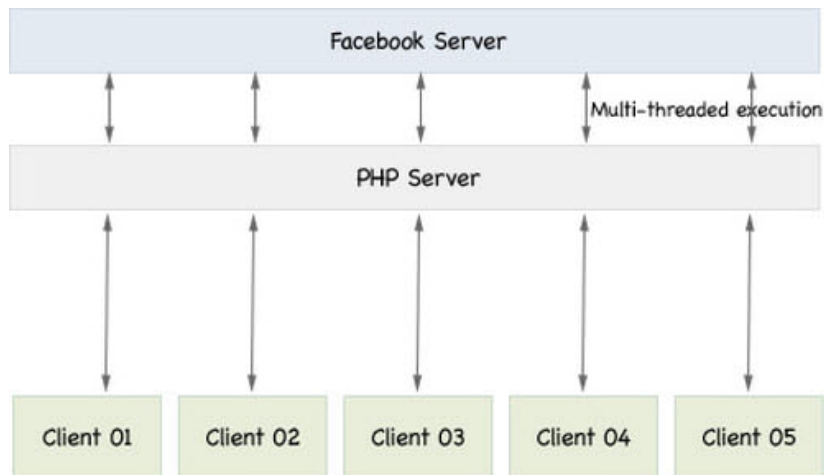
Μέχρι πρόσφατα η Javascript χρησιμοποιούνταν μόνο στο front-end των προγραμμάτων περιήγησης και οι προγραμματιστές ήταν υποχρεωμένοι να χρησιμοποιούν άλλες γλώσσες για τον προγραμματισμό στον εξυπηρετητή (server) και στον εξυπηρετούμενο (client). Το γεγονός ότι το Node μας δίνει τη δυνατότητα να χρησιμοποιήσουμε την ίδια γλώσσα και στις δύο άκρες, απλουστεύει σημαντικά την ανάπτυξη των διαδικτυακών εφαρμογών. Με αυτόν τον τρόπο, το μόνο προαπαιτούμενο για να αναπτύξει κανείς μια διαδικτυακή εφαρμογή είναι η γνώση Javascript. Εκτός από τη μεγάλη διευκόλυνση που προσφέρει, το γεγονός ότι χρησιμοποιούμε Javascript και στις δύο πλευρές συντελεί σημαντικά και στην αύξηση της απόδοσης των διαδικτυακών εφαρμογών. Με χρήση του προγράμματος περιήγησης Google Chrome, η ταχύτητα με την οποία εκτελείται η JavaScript αυξάνεται με τρομερά ταχείς ρυθμούς.

Η JavaScript δεν παρέχει κάποια βιβλιοθήκη για λειτουργίες E/E με σύγχρονο τρόπο και εξαιτίας αυτού οι λειτουργίες που παρέχονται στην πλατφόρμα Node είναι χωρίς αναμονή (non blocking). Οι συναρτήσεις στην JavaScript αποτελούν αντικείμενα πρώτης τάξης (first-class functions). Αυτό σημαίνει ότι η γλώσσα υποστηρίζει το πέρασμα μιας συνάρτησης σαν παράμετρο σε μια άλλη συνάρτηση και την επιστροφή μιας συνάρτησης σαν αποτέλεσμα από μια άλλη συνάρτηση. Επιπλέον, η JavaScript παρέχει το μοτίβο κλεισίματος (closure pattern). Με αυτόν τον τρόπο οι συναρτήσεις επιστρεφόμενης κλήσης (callback functions) έχουν την ιδιότητα να κρατάνε στοιχεία και να ξέρουν το περιβάλλον λειτουργίας στο οποίο δηλώθηκαν και τις μεταβλητές που υπήρχαν σε αυτό το περιβάλλον. Το Node στηρίζεται στα παραπάνω δύο χαρακτηριστικά της Javascript.

### **Νήματα, μονομηματικό μοντέλο και πολυμηματισμός**

Οι παραδοσιακές προγραμματιστικές τεχνικές παγώνουν την συνέχεια της εκτέλεσης μιας διεργασίας όταν μεσολαβήσει μια διακοπή ή μια λειτουργία E/E μέχρι αυτή να ολοκληρωθεί. Στις διαδικτυακές εφαρμογές όμως ο χρήστης πρέπει να μπορεί να αλληλεπιδράσει με το σύστημα και να μην αναμένει την ολοκλήρωση της κάθε λειτουργίας E/E. Μία μέθοδος που δημιουργήθηκε για να αντιμετωπίσει αυτό το πρόβλημα είναι η χρήση νημάτων. Νήμα (thread) είναι ένα είδος ελαφριάς διεργασίας που μπορεί να χρησιμοποιήσει την κεντρική μονάδα επεξεργασίας και μοιράζεται τη μνήμη με κάθε

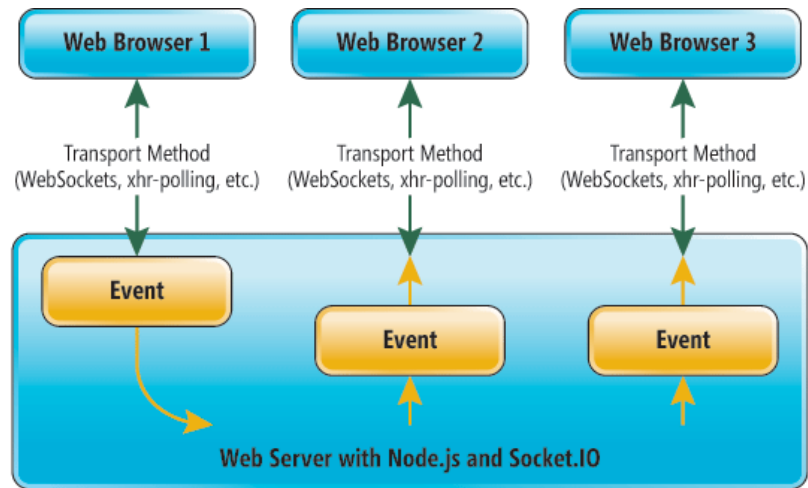
άλλο νήμα μέσα σε μια διεργασία. Όσο ένα νήμα βρίσκεται σε αναμονή ώσπου μια λειτουργία E/E να ολοκληρωθεί, ένα άλλο νήμα της ίδιας διεργασίας μπορεί να χρησιμοποιήσει την κεντρική μονάδα επεξεργασίας. Μόλις η λειτουργία E/E ολοκληρωθεί, η εκτέλεση περνάει πάλι στο πρώτο νήμα και το δεύτερο κοιμάται. Η διαδικασία αυτή ονομάζεται πολυνηματισμός.



**Σχήμα 4.6:** Πολυνηματικό Μοντέλο.

Ο πολυνηματισμός παρόλο που λύνει το πρόβλημα της απόδοσης και των μεγάλων νεκρών διαστημάτων της πρώτης υλοποίησης παρουσιάζει κάποια προβλήματα. Αρχικά απαιτεί πολύ χρόνο εργασίας από τον προγραμματιστή καθώς και τη χρήση εξειδικευμένων λύσεων συγχρονισμού των νημάτων, όπως κλειδώματα (locks) και σημαφόρους (semaphores), με σκοπό να μην πραγματοποιηθεί ταυτόχρονη πρόσβαση δύο νημάτων στο ίδιο σημείο της μνήμης, η οποία θα έχει ως αποτέλεσμα ασυνέπειες [3]. Τέτοιου είδους λάθη (bugs) δεν είναι πάντα εύκολο να προβλεφθούν και πολλές φορές συμβαίνουν τυχαία και είναι πολύ δύσκολος ο εντοπισμός τους. Ένα άλλο μειονέκτημα του μοντέλου του πολυνηματισμού είναι ότι το μοντέλο δεν κλιμακώνει ιδιαίτερα καλά καθώς ο αριθμός των χρηστών αυξάνεται. Η διαχείριση όλων αυτών των νημάτων είναι ένα μεγάλο βάρος για το λειτουργικό σύστημα, τόσο λόγω των απαιτήσεων μνήμης, όσο και λόγω της εναλλαγής περιβάλλοντος λειτουργίας. Η εναλλαγή της ΚΜΕ ανάμεσα στα νήματα απαιτεί την αποθήκευση της κατάστασης του τρέχοντος νήματος και την επαναφορά της κατάστασης του προς εκτέλεσης νήματος.

Το Node για να αντιμετωπίσει όλα αυτά τα προβλήματα, υιοθετεί μία διαφορετική προσέγγιση δημιουργώντας μόνο ένα νήμα το οποίο χειρίζεται όλες τις λειτουργίες E/E. Με αυτόν τον τρόπο επιτυγχάνει υψηλά επίπεδα κλιμάκωσης, που μπορεί να φτάσουν τις ένα εκατομμύριο ταυτόχρονες συνδέσεις. Στο σχήμα 4.7 φαίνεται ένα παράδειγμα από το μονονηματικό μοντέλο του NodeJs.



Σχήμα 4.7: Το μονομηματικό μοντέλο του NodeJs.

### Event-driven model

Ο οδηγούμενος από τα γεγονότα (event-driven) προγραμματισμός είναι ένα προγραμματιστικό μοντέλο όπου η ροή της εκτέλεσης “οδηγείται” από γεγονότα. Τα γεγονότα αυτά τα χειρίζονται είτε οι επιστρεφόμενες κλήσεις των γεγονότων είτε οι χειριστές συμβάντος. Μια επιστρεφόμενη κλήση είναι μια συνάρτηση η οποία καλείται όταν συμβαίνει ένα συγκεκριμένο γεγονός. Στα πλαίσια του εξυπηρετητή το γεγονός στο οποίο αναφερόμαστε συνήθως αποτελεί η ολοκλήρωση μιας λειτουργίας E/E. Για παράδειγμα, σε μια γραφική διεπαφή χρήστη έχουμε ως γεγονότα τα πατήματα πλήκτρων ή επιλογών με το ποντίκι.

Στις περιπτώσεις που οι λειτουργίες E/E επιστρέφουν μία συνάρτηση και όχι απλά μία τιμή έχουμε το μοντέλο του ασύγχρονου ή οδηγούμενου από γεγονότα προγραμματισμού. Έτσι επιτυγχάνεται η παραλληλοποίηση των E/E λειτουργιών, ενώ κάθε φορά η συνάρτηση επιστρεφόμενης κλήσης ενημερώνει για την ολοκλήρωση της αντίστοιχης λειτουργίας. Η JavaScript ανήκει σε αυτό το είδος προγραμματισμού. Η JavaScript υποστηρίζει επιπλέον το μοτίβο κλεισίματος (closure pattern) καθώς και το πέρασμα συναρτήσεων σαν παραμέτρους σε άλλη συνάρτηση (first-class functions).

Ο οδηγούμενος από τα γεγονότα προγραμματισμός συνοδεύεται από έναν βρόχο γεγονότων (event loop). Ο βρόχος γεγονότων είναι ένα loop το οποίο σε κάθε επανάληψη ανιχνεύει γεγονότα και ενεργοποιεί τον αντίστοιχο χειριστή συμβάντος. Ουσιαστικά κοιτάει σε κάθε επανάληψη ποια λειτουργία E/E έχει ολοκληρωθεί και καλεί την αντίστοιχη συνάρτηση επιστρεφόμενης κλήσης. Ο βρόχος γεγονότων υλοποιείται από ένα νήμα μέσα σε μια διεργασία και με βάση όσα αναφέρθηκαν για τα νήματα εύκολα προκύπτει ότι κάθε χρονική στιγμή τρέχει το πολύ ένα νήμα, που στην περιπτωσή μας είναι ένας διαχειριστής γεγονότος, και ότι κάθε διαχειριστής γεγονότος τρέχει μέχρι να ολοκληρωθεί χωρίς

διακοπή.

## **Ο Διαχειριστής Πακέτων του Node (Node Package Manager ,NPM)**

Ο διαχειριστής πακέτων του Node (Node Package Manager ή NPM) παρέχει την δυνατότητα να κατεβάσουμε, να εγκαταστήσουμε και να τρέξουμε στην εφαρμογή μας πακέτα (modules) τρίτων. Η εγκατάσταση του διαχειριστή πακέτων πραγματοποιείται αυτόματα όταν εγκαθιστούμε την πλατφόρμα του Node. Μετά την εγκατάσταση του Node, έχουμε πρόσβαση σε κάποια ενσωματωμένα πακέτα, τα οποία μας παρέχουν κάποιες διεπαφές εφαρμογών χαμηλού επιπέδου (low-level APIs) ώστε να μπορούμε να υλοποιήσουμε κάποιες βασικές λειτουργίες που χρειάζονται οι εφαρμογές. Αυτά τα πακέτα ονομάζονται ως ο πυρήνας του NodeJs . Στα πλαίσια σύνθετων εφαρμογών είναι σχεδόν απαραίτητο να συμπεριλάβουμε πακέτα τρίτων.

Το σύνολο όλων των πακέτων που έχουν φτιάξει και δημοσιοποιήσει οι προγραμματιστές διατηρείται σε μία κεντρική αποθήκη από τον διαχειριστή πακέτων του Node. Ο διαχειριστής πακέτων παρέχει ένα εργαλείο γραμμής-εντολών (command-line tool), όπου ο χρήστης μπορεί να εγκαθιστά και να διαχειρίζεται τα πακέτα αυτά. Η αναζήτηση τέτοιων πακέτων μπορεί να πραγματοποιηθεί είτε μέσω της γραμμής εντολών είτε μέσω της σελίδας [www.npmjs.org](http://www.npmjs.org). Σε αυτήν την σελίδα παρουσιάζονται και διάφορα στατιστικά στοιχεία, όπως πόσα πακέτα υπάρχουν συνολικά, από ποια πακέτα εξαρτώνται περισσότερο άλλα πακέτα κ.λ.π. [116]

## **Ο Εξυπηρετητής (Server)**

Το NodeJs κάνει την δημιουργία διαφόρων τύπων εξυπηρετητών (servers) μια πολύ εύκολη διαδικασία καθώς στο NodeJs ο εξυπηρετητής και η εφαρμογή είναι το ίδιο.

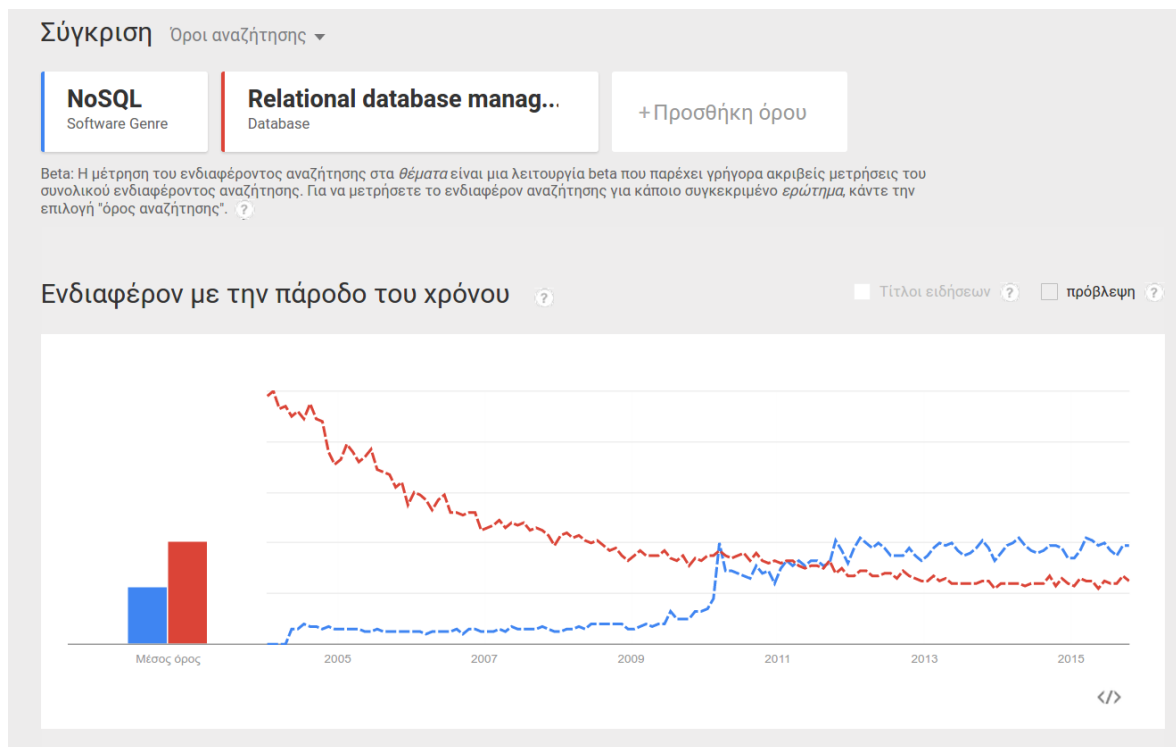
### **4.1.3 Βάση Δεδομένων**

Στο παρόν κεφάλαιο κάνουμε μια σύντομη αναφορά στα διάφορα συστήματα βάσεων δεδομένων τύπου NoSQL, τα πλεονεκτήματά τους καθώς και τις εφαρμογές για τις οποίες ενδείκνυται η χρήση τους.

Η ιδέα ενός συστήματος βάσης δεδομένων τύπου NoSQL παρουσιάστηκε για πρώτη φορά το 1998 χωρίς να προχωρήσει όμως ιδιαίτερα και επανήλθε πάλι στο προσκήνιο το 2009, όταν και αναπτύχθηκαν αρκετές NoSQL λύσεις [148] [36]. Η διαδικασία απόδοσης κάποιου συγκεκριμένου ορισμού

για τον όρο NoSQL δεν αποτελεί απλή διαδικασία, δεδομένου ότι δεν αναφέρεται σε ένα συγκεκριμένο σύστημα διαχείρισης βάσεων δεδομένων, αλλά αποτελεί έναν γενικό όρο που περικλείει πολλά συστήματα βάσεων δεδομένων.

Στον τομέα της επιστήμης των υπολογιστών όταν αναφερόμαστε στον όρο NoSQL (Not Only SQL), μιλάμε για ένα ευρύ φάσμα συστημάτων διαχείρισης βάσεων δεδομένων τα οποία παρέχουν ένα μηχανισμό προς αποθήκευση και ανάκτηση δεδομένων διαφορετικό από αυτό του σχεσιακού μοντέλου των κλασικών RDBMS (Relational database management system)[32][81]. Τα συστήματα NoSQL ξεφεύγουν από την παραδοσιακή χρήση πινάκων για την οργάνωση των δεδομένων και χρησιμοποιούν διαφορετικές δομές οργάνωσης, όπως αυτή του κλειδιού-τιμής. Επιπροσθέτως η επεξεργασία αυτών των δεδομένων δεν γίνεται με χρήση ερωτημάτων SQL όπως έχουμε συνηθίσει. Λόγω των εγγενών διαφορών μεταξύ των δύο αυτών συστημάτων, κάποιες λειτουργίες είναι ταχύτερες στα συστήματα NoSQL ενώ κάποιες άλλες στα συστήματα σχεσιακών βάσεων δεδομένων.



**Σχήμα 4.8:** Τάση αναζητήσεων Google για NoSQL σε σύγκριση με RDBMS.

Η χρήση των βάσεων δεδομένων τύπου NoSQL έχει γνωρίσει ραγδαία αύξηση τα τελευταία χρόνια σε εφαρμογές big data καθώς και σε εφαρμογές πραγματικού χρόνου. Αποκαλούνται με τον όρο Not Only SQL για να δοθεί έμφαση στο γεγονός ότι οι δύο τεχνολογίες δεν είναι αμοιβαία αποκλειόμενες. Έχουν τη δυνατότητα να συνυπάρχουν, κρατώντας τα δυνατά σημεία της κάθε μιας, προσφέροντας σε διαφορετικές καταστάσεις διαφορετικές υπηρεσίες. Οι βάσεις δεδομένων τύπου NoSQL χρησιμοποιούνται από εταιρείες κολοσσούς του τεχνολογικού τομέα όπως το Facebook, Amazon,

Google, LinkedIn[148]. Ο όγκος των δεδομένων που διαχειρίζονται τέτοιες εταιρείες είναι της τάξης πολλών petabyte και επομένως θα ήταν αδύνατη η κλιμάκωσή τους με κάποιο σχεσιακό σύστημα ΒΔ. Κι αυτό γιατί οι σχεσιακές ΒΔ συναντούν περιορισμούς όσον αφορά την αντιμετώπιση προβλημάτων όπως είναι η εξόρυξη δεδομένων, το Web 2.0, το cloud computing, τα μη γραμμικά σε εκτέλεση ερωτήματα κ.τ.λ.



**Σχήμα 4.9:** Δημοφιλές βάσεις δεδομένων τύπου NoSQL.

Τα NoSQL συστήματα δεν εμφανίστηκαν για να αντικαταστήσουν τα σχεσιακά, αλλά για να τα συμπληρώσουν. Όπως και σε πολλές άλλες περιπτώσεις έτσι και εδώ, το ζητούμενο είναι να γίνει η επιλογή του κατάλληλου εργαλείου για τη λύση του κάθε προβλήματος. Έτσι στις περισσότερες περιπτώσεις είναι απαραίτητο να γίνει μια εκτίμηση των δυνατοτήτων των διαθέσιμων συστημάτων και στη συνέχεια να υλοποιηθεί ένας συνδυασμός τους ώστε κάθε επιμέρους ανάγκη να καλύπτεται από το σύστημα που της ταιριάζει καλύτερα. Έτσι, όσο τα δεδομένα του διαδικτύου αυξάνονται με εκθετικούς ρυθμούς, τόσο πιο πολλές θα είναι και οι περιπτώσεις όπου θα καθίσταται καλή επιλογή η χρήση τους. Η ανάπτυξη των NoSQL συστημάτων είναι πολύ γρήγορη τα τελευταία χρόνια. Τη στιγμή της συγγραφής της παρούσας διπλωματικής εργασίας, η λίστα με τις NoSQL βάσεις δεδομένων περιλαμβάνει περίπου 225 καταχωρήσεις όταν πριν από τέσσερα χρόνια οι καταχωρήσεις έφταναν τις 35[54].

## Τύποι NoSQL

Όπως αναφέραμε και προηγουμένως με τον όρο NoSQL αναφερόμαστε σε ένα ευρύ φάσμα τεχνολογιών. Στην παρούσα ενότητα θα κάνουμε μια σύντομη παρουσίαση των πιο διαδεδομένων τε-

χνολογιών.

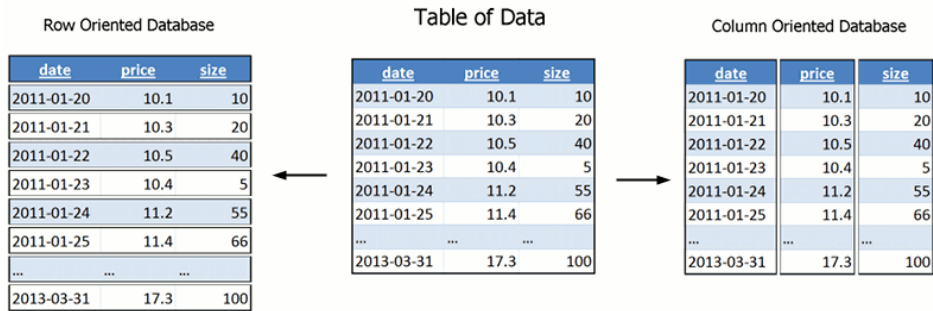
### **Key-value Stores (Αποθήκες κλειδιών-τιμών)**

Το μοντέλο κλειδιού - τιμής είναι το πιο απλό από όλες τις κατηγορίες που εξετάζουμε. Τα δεδομένα αποθηκεύονται σε ζεύγη κλειδιού - τιμής, με τρόπο παρόμοιο με μία απεικόνιση (map) ή με έναν πίνακα κατακερματισμού (hash table). Κάποια συστήματα μπορεί να επιτρέπουν η τιμή να είναι κάτι πιο πολύπλοκο, όπως για παράδειγμα μία λίστα. Αποτελούν ουσιαστικά αποθήκες δεδομένων, οι οποίες λειτουργούν με τη χρήση ενός hashtable, στο οποίο αποθηκεύονται μοναδικά κλειδιά και δείκτες για τα δεδομένα του κάθε αντικειμένου. Τέτοιου τύπου αποθήκες δεδομένων προσφέρουν υπηρεσίες για τη διαχείριση δεδομένων χωρίς τον ορισμό συγκεκριμένου σχήματος. Το γεγονός ότι το μοντέλο είναι τόσο απλό, κάνει τις βάσεις αυτού του είδους να πετυχαίνουν πολύ καλές επιδόσεις όταν η συγκεκριμένη δομή ταιριάζει στο πρόβλημα. Τα δεδομένα για να είναι κατάλληλα για μια τέτοια βάση, δεν θα πρέπει να έχουν υψηλά επίπεδα συσχέτισεων μεταξύ τους. Αν το ζητούμενο είναι να υπάρχει δυνατότητα για πολύπλοκα ερωτήματα στη βάση, τότε το μοντέλο αυτό δεν διευκολύνει. Από τα πιο γνωστά συστήματα που ανήκουν σε αυτή την κατηγορία είναι τα memcached, membase, cachedb, Voldemort, Scalaris, Dynamo, Redis και Riak[46][139].

### **Column-oriented Databases (Βάσεις δεδομένων προσανατολισμένες στη στήλη)**

Δημιουργήθηκαν με σκοπό την αποθήκευση και την επεξεργασία μεγάλης ποσότητας δεδομένων, τα οποία είναι κατανεμημένα σε πολλούς διαφορετικούς υπολογιστικούς κόμβους. Υπάρχουν και στην περίπτωση αυτή κλειδιά το οποία δείχνουν σε διαφορετικά σύνολα στηλών. Οι γραμμές διαχωρίζονται από κλειδιά και οι στήλες χωρίζονται σε οικογένειες. Οι column-oriented βάσεις δεδομένων από άποψη δομής βρίσκονται κάπου ανάμεσα στους σχεσιακούς πίνακες και στο μοντέλο κλειδιού - τιμής. Θα μπορούσαμε να πούμε ότι αποτελούνται από πίνακες οι οποίοι όμως διαφέρουν από τους παραδοσιακούς των σχεσιακών βάσεων ως προς τον τρόπο με τον οποίο αποθηκεύουν τα δεδομένα. Η διαφορά βρίσκεται στο ότι στους πίνακες κατά στήλες, τα δεδομένα από μία στήλη αποθηκεύονται μαζί στον δίσκο (σε αντίθεση με τους πίνακες των σχεσιακών βάσεων που αποθηκεύονταν κατά γραμμές). Αυτή η αλλαγή, παρ' ότι φαίνεται μικρή δημιουργεί πολύ διαφορετικές συνθήκες και ιδιότητες στο σύστημα. Η προσθήκη επιπλέον στηλών είναι μια διαδικασία που δεν κοστίζει, κάθε γραμμή μπορεί να έχει διαφορετικές στήλες ή και καθόλου, και επίσης οι πίνακες μπορούν να παραμένουν αραιοί χωρίς αυτό όμως να έχει επίπτωση στον αποθηκευτικό χώρο που απαιτούν.

Ένα σημαντικό χαρακτηριστικό που έχουν τέτοια συστήματα είναι οι ενσωματωμένες δυνατότη-



**Σχήμα 4.10:** Αποθήκευση κατά Σειρές και κατά Στήλες.

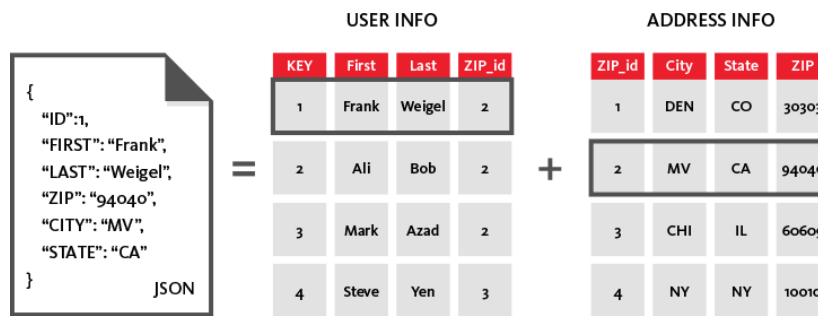
τες για συμπίεση των δεδομένων, αφού η οργάνωση κατά στήλες κάνει τα δεδομένα που βρίσκονται σε μία στήλη να είναι όμοια και έτσι οι αλγόριθμοι συμπίεσης πετυχαίνουν καλύτερα αποτελέσματα. Το χαρακτηριστικό αυτό γίνεται πολύ σημαντικό όταν ο όγκος των δεδομένων είναι μεγάλος. Είναι γενικά καλό με τις συγκεκριμένες βάσεις να υπάρχει ένα πλάνο για το πώς θα χρησιμοποιηθούν τα δεδομένα και τι ερωτήματα θα γίνονται σε αυτά, έτσι ώστε να γίνει κατάλληλη επιλογή της δομής των δεδομένων και να είναι αποδοτική η βάση στη συνέχεια. Αν δεν είναι δυνατόν να γίνει μία τέτοια εκτίμηση εκ των προτέρων, τότε τα συγκεκριμένα συστήματα μπορεί να μην αναδεικνύουν τα δυνατά τους χαρακτηριστικά. Παραδείγματα τέτοιων συστημάτων βάσεων δεδομένων είναι τα BigTable, Cassandra, Hyperbase και HBase[1][2].

### Document-oriented Databases (Βάσεις δεδομένων βασισμένες στα έγγραφα)

Οι βάσεις οργανωμένες κατά αρχεία, όπως λέει και το όνομά τους, χρησιμοποιούν αρχεία για να αποθηκεύσουν τα δεδομένα. Τα αρχεία αυτά έχουν ένα μοναδικό αναγνωριστικό ID το καθένα και συνήθως η βάση διατηρεί ευρετήριο (index) ώστε να γίνεται γρήγορα η αναζήτηση και η ανάκτηση τους. Σαν τιμή μπορεί να περιέχουν κάθε είδους δεδομένα με μοναδικό περιορισμό να μπορούν να εκφραστούν σαν αρχείο, γεγονός που παρέχει μεγάλη ευελιξία. Τα αρχεία είναι συνήθως τύπου JSON (JavaScript Object Notation) είτε παραλλαγές όπως BSON (Binary JSON) τα οποία είναι κατάλληλα τόσο για την μεταφορά και την αποθήκευση, όσο και για την ανάγνωση και τον χειρισμό των δεδομένων, αφού είναι ένα πρότυπο με το οποίο τα δεδομένα περιγράφουν από μόνα τους τη δομή τους, χωρίς αυτό να προσθέτει κάποιο περιορισμό στη βάση. Λόγω της φύσης των αρχείων, συνήθως τέτοιου είδους συστήματα ταιριάζουν με αντικειμενοστραφείς γλώσσες προγραμματισμού. Μία ιδιαιτερότητα των βάσεων που οργανώνονται κατά αρχεία είναι ότι τα δεδομένα κατά βάση δεν είναι κανονικοποιημένα όπως στις σχεσιακές βάσεις, αλλά κάθε αρχείο προβλέπεται να διατηρεί τις περισσότερες αν όχι όλες τις πληροφορίες που μπορεί να χρειαστούν όταν πραγματοποιείται η ενέργεια που το αφορά. Παραδείγματα τέτοιων συστημάτων βάσεων δεδομένων είναι τα MongoDB, CouchDB, Couchbase,



RethinkDB, Terrastore, Elasticsearch[15][131].



Σχήμα 4.11: Βάσεις δεδομένων βασισμένες στα έγγραφα.

### Graph databases (Βάσεις δεδομένων γράφου)

Βασίζονται στη θεωρία των γράφων για την κατασκευή συστημάτων με κόμβους, οι οποίοι τοποθετούνται ανάλογα με τις σχέσεις που προκύπτουν μεταξύ των δεδομένων. Οι βάσεις δεδομένων αυτής της κατηγορίας δεν χρησιμοποιούνται συχνά, αφού είναι κατάλληλη επιλογή μόνο σε περιπτώσεις όπου τα δεδομένα παρουσιάζουν μεγάλη συσχέτιση μεταξύ τους και η συσχέτιση αυτή πρέπει να αποτυπωθεί. Μοντελοποιούνται με κόμβους και με σχέσεις που τους συνδέουν. Χαρακτηριστική λειτουργία σε τέτοιου είδους βάσεις είναι να διασχίζονται οι κόμβοι χρησιμοποιώντας τις σχέσεις που τους συνδέουν. Τέλος, οι κόμβοι και οι σχέσεις μπορούν να έχουν ιδιότητες στις οποίες αποθηκεύονται δεδομένα. Η κλασική χρήση τέτοιων συστημάτων είναι σε κοινωνικά δίκτυα, όπου οι σχέσεις μεταξύ των χρηστών είναι πολλές και πολύπλοκες. Σε τέτοιες περιπτώσεις τα υπόλοιπα συστήματα βάσεων δεδομένων ταιριάζουν αρκετά λιγότερο και προσθέτουν δυσκολίες. Το γεγονός όμως ότι οι εξαρτήσεις μεταξύ των δεδομένων είναι τόσο πολλές, καθιστά δύσκολο το να γίνουν αυτά τα συστήματα καταναμημένα, αφού σε περίπτωση κατάτμησης λόγω δικτύου οι σχέσεις που δεν θα είναι πλέον προσβάσιμες μπορεί να είναι πάρα πολλές. Παράδειγμα τέτοιων συστημάτων είναι τα Neo4j, AllegroGraph, GraphDB, FlockDB, OrientDB[82][134].

### Object-oriented databases (Αντικειμενοστραφείς βάσεις)

Οι αντικειμενοστρεφείς βάσεις δεδομένων αποθηκεύουν τα δεδομένα τους, όπως υποδεικνύει το όνομά τους με τη μορφή αντικειμένων, όπως αυτά χρησιμοποιούνται στις αντικειμενοστρεφείς γλώσσες προγραμματισμού. Αυτό κάνει πιο άμεση και απλή τη χρησιμοποίησή τους από εφαρμογές γραμμένες σε τέτοιου είδους γλώσσες, αφού δε χρειάζεται η χρήση επιπλέον κώδικα για την αντιστοίχιση των δεδομένων με την εφαρμογή. Το γεγονός ότι δε χρειάζεται κάποια μετάφραση των αντικειμένων που χρησιμοποιούνται, σε κάποια άλλη μορφή (πχ κατάλληλη για σχεσιακές βάσεις), κάνει τις

αντικειμενοστρεφείς βάσεις να παρουσιάζουν συχνά αυξημένη επίδοση σε σχέση με άλλα συστήματα. Επίσης δεν είναι απαραίτητη η δημιουργία κλειδιών για να επιτευχθεί η συσχέτιση και η αντιστοίχιση των δεδομένων, τα οποία μπορούν να αναζητηθούν με τη χρήση δεικτών (pointers).

Οι αντικειμενοστρεφείς βάσεις δεδομένων έχουν εμφανιστεί από τα μέσα της δεκαετίας του 70 και βρήκαν ανταπόκριση σε κάποιες εφαρμογές όπως CAD (Computer Aided Design), CAM (Computer Aided Manufacturing) και σε ενσωματωμένα συστήματα, όμως ποτέ δεν έγινε εκτεταμένη χρήση τους. Αυτό οφείλεται σε κάποια μειονεκτήματα που παρουσιάζουν σε σχέση με άλλα συστήματα. Το κυριότερο από αυτά είναι ότι η βάση που δημιουργείται συσχετίζεται στενά με την εφαρμογή που τη χρησιμοποιεί και πιθανώς και με τη γλώσσα στην οποία είναι γραμμένη η εφαρμογή. Αυτό κάνει τη βάση λιγότερο ευέλικτη σε σχέση με άλλες εναλλακτικές. Επίσης για απλές περιπτώσεις βάσεων που περιέχουν απλές σχέσεις, οι αντικειμενοστρεφείς βάσεις μπορεί να είναι λιγότερο αποδοτικές και πιο πολύπλοκες σε σχέση με ένα σύστημα RDBMS. Παράδειγμα τέτοιων συστημάτων είναι τα VelocityDB, Versand, Objectivity, Starcounter, Perst, HSS Database, Sterling[20][99][100][103].

## **Χαρακτηριστικά NoSQL βάσεων δεδομένων**

### **Ανθεκτικότητα**

Όπως και στις σχεσιακές βάσεις, έτσι και στις NoSQL βάσεις όταν κάτι γραφτεί στο δίσκο είμαστε σίγουροι πως δε θα χαθεί. Η διαφορά από τις σχεσιακές βάσεις βέβαια είναι πως στα κατανεμημένα συστήματα κάτι τέτοιο σημαίνει πως η πληροφορία έχει γραφτεί εκ προεπιλογής σε παραπάνω από έναν δίσκο και επομένως δε θα χαθεί ακόμα και σε περίπτωση αποτυχίας ενός δίσκου ή υπολογιστή[137][145].

### **Προσαρμοστικότητα**

Οι NoSQL βάσεις είναι ευέλικτες σχετικά με τα δεδομένα που αποθηκεύουν. Δεν υπάρχει περιορισμός στον τύπο των δεδομένων που μπορούν να αποθηκευτούν, ενώ στις SQL βάσεις ο τύπος έχει καθοριστεί κατά τη δημιουργία τους. Ακόμη, οι NoSQL βάσεις μπορούν να αντιμετωπίσουν με ευκολία πιο πολύπλοκες και προηγμένες δομές δεδομένων. Οι περισσότερες από τις υπάρχουσες NoSQL βάσεις, παρέχουν πλούσιες δομές δεδομένων και εργασιών σε λίστες, σύνολα, ταξινομημένα σύνολα και κλειδιά κατακερματισμού που κάνουν πράγματα όπως εξισορρόπηση φορτίου και message-queuing απλούστατα στην υλοποίηση[145][70].

## **Διαχειρισιμότητα**

Η ύπαρξη ενός άριστα εκπαιδευμένου, με πολλή εμπειρία και γνώση, διαχειριστή βάσεων δεδομένων είναι απαραίτητη για τη συντήρηση ενός υψηλού επιπέδου RDBMS. Είναι άρρηκτα συνδεδεμένος με όλες τις λειτουργίες που αφορούν τη βάση, δηλαδή με το σχεδιασμό, την εγκατάσταση και τη συνεχή παρακολούθηση και ρύθμιση της. Οι μη σχεσιακές βάσεις από τα θεμέλιά τους έχουν σχεδιαστεί για να χρειάζονται λιγότερη διαχείριση. Η δυνατότητα αυτόματης επισκευής, η διανομή δεδομένων, το απλούστερο μοντέλο δεδομένων είναι όλα χαρακτηριστικά που συντελούν στη μείωση των απαιτήσεων σε θέματα ρύθμισης και διαχείρισης[145].

## **Παράλληλη επεξεργασία**

Χάρη στις δυνατότητες που προσφέρουν τα συστήματα NoSQL, όπως εύκολη αναπαραγωγή και διαμοιρασμός δεδομένων, καθίσταται εξίσου εύκολη και η εκμετάλλευση των φυσικών πόρων του συμπλέγματος όπου λειτουργεί η βάση. Από τη στιγμή που τα δεδομένα βρίσκονται σε διαφορετικούς δίσκους, υπολογιστές ή ακόμα και δίκτυα, τα ερωτήματα μπορούν χωρίς καθόλου επιπλέον κώδικα να εκτελούνται παράλληλα και να δίνουν άμεσα απαντήσεις[145].

## **Διαθεσιμότητα**

Όπως θα ήταν αναμενόμενο, δε συμφέρει καθόλου κάποια web ή mobile based επιχείρηση (π.χ. facebook) να είναι πεσμένη. Θέλει δηλαδή να υπάρχει όσο το δυνατόν λιγότερο downtime. Χάρη πάλι στο γεγονός ότι οι NoSQL βάσεις είναι κατανεμημένες, οι ενημερώσεις λογισμικού, αναβαθμίσεις υλικού αλλά και τυχόν αποτυχιές υλικού δε σημαίνουν πως θα πέσει η εφαρμογή, αφού υπάρχει σε πολλαπλούς servers. Αντιθέτως, αν η εφαρμογή βασίζεται σε μία σχεσιακή βάση, τα παραπάνω ισοδυναμούν με δυσκολίες και downtime[145].

## **Υψηλή απόδοση**

Εκτός από το γεγονός ότι με την προσθήκη υπολογιστών στο cluster έχουμε ήδη περισσότερη υπολογιστική ισχύ και άρα καλύτερη απόδοση, η ίδια η αρχιτεκτονική των NoSQL εργαλείων κάνει τις βάσεις αυτές πιο αποδοτικές. Αν μία σχεσιακή βάση είχε μερικές εκατοντάδες χιλιάδες πίνακες, η επεξεργασία των δεδομένων θα δημιουργούσε πάρα πολλά locks και θα υποβάθμιζε την απόδοση. Λόγω όμως της ασθενέστερης συνέπειας που χαρακτηρίζει τα μοντέλα δεδομένων των NoSQL βά-

σεων, δίνεται βάρος στην αποτελεσματικότητα αντί της συνοχής και επομένως δεν υπάρχει πρόβλημα όσο μεγάλη και να είναι η βάση[145].

### **Αναπαραγωγή δεδομένων**

Τα δεδομένα διανέμονται μεταξύ πολλών κόμβων με πολύ εύκολο τρόπο. Σε όλα τα NoSQL συστήματα κάτι τέτοιο ρυθμίζεται πολύ απλά βάζοντας σε ένα .conf αρχείο την ip ή το domain name των servers στους οποίους θέλουμε να αναπαράγονται τα δεδομένα. Η αυτοματοποίηση της αναπαραγωγής έχει ως αποτέλεσμα την υψηλή διαθεσιμότητα των δεδομένων και επομένως την εύκολη αποκατάσταση από καταστροφή τους, χωρίς να εμπλέκονται ξεχωριστές εφαρμογές για το σκοπό αυτό. Ακόμη, χάρη σε αυτό το χαρακτηριστικό, η προσθήκη ή αφαίρεση υλικού (υπολογιστές, servers κλπ) γίνεται χωρίς downtime[145].

### **Schema-less persistence – Δυναμικά Schemas**

Στις σχεσιακές βάσεις είναι απαραίτητο το σχήμα να είναι ορισμένο πριν αρχίσουμε να προσθέτουμε δεδομένα. Για παράδειγμα, αν θέλουμε να κρατάμε σε κάποια SQL βάση 90 δεδομένα όπως όνομα, επώνυμο και τηλέφωνο πελάτη, είναι απαραίτητο η βάση αυτή να το γνωρίζει από πριν.

Τώρα όμως που ο ρυθμός αύξησης των δυνατοτήτων ιστοσελίδων/εφαρμογών είναι ταχύτατος, κάθε φορά που ολοκληρώνεται κάποιο νέο χαρακτηριστικό, είναι πολύ πιθανό να χρειάζεται αλλαγή και το σχήμα της βάσης. Αν θέλω δηλαδή στη βάση του παραδείγματος να ξεκινήσω να κρατάω και κάτι παραπάνω για τους πελάτες, πρέπει να αλλάξω το σχήμα και να προσθέσω στήλη και ίσως να φτιάξω επιπλέον indexes/foreign keys κλπ. Αν η βάση είναι μεγάλη, η διαδικασία αυτή θα πάρει πολλή ώρα οπότε η βάση δε θα λειτουργεί για το διάστημα αυτό. Επίσης, με ένα σχεσιακό σύστημα, δεν υπάρχει τρόπος να αντιμετωπίσουμε δεδομένα που είναι αδόμητα ή άγνωστα από πριν. Επομένως, το φλέγον θέμα της διαχείρισης των αλλαγών (change management) είναι μεγάλος πονοκέφαλος σε ένα μεγάλο RDBMS.

Αντιθέτως, οι NoSQL βάσεις είναι πολύ ευέλικτες και έχουν χαλαρούς (ή και ανύπαρκτους) περιορισμούς όσον αφορά στο μοντέλο δεδομένων. Είναι σχεδιασμένες να επιτρέπουν την εισαγωγή δεδομένων χωρίς την ύπαρξη προκαθορισμένου σχήματος. Έτσι, οι σημαντικές αλλαγές που χρειάζονται γίνονται σε πραγματικό χρόνο, χωρίς την ανησυχία για διακοπή της λειτουργίας της βάσης. Επομένως, η ανάπτυξη καθίσταται γρηγορότερη, η ενσωμάτωση κώδικα πιο αξιόπιστη και χρειάζεται λιγότερη ώρα ενασχόλησης του διαχειριστή της βάσης[145].

## Κλιμάκωση

Είναι ίσως το πιο σημαντικό χαρακτηριστικό των NoSQL βάσεων. Όταν οι επισκέψεις σε ένα site γίνονται μερικές εκατοντάδες ή και χιλιάδες το δευτερόλεπτο, μία σχεσιακή βάση θα έπρεπε να είναι διαμοιρασμένη και αναπαραγμένη σε μεγάλο βαθμό για να καταφέρει να ανταποκριθεί. Μία NoSQL βάση όμως, λόγω της κατανεμημένης της φύσης, το μόνο που χρειάζεται για σωστή κλιμάκωση, είναι προσθήκη υπολογιστών στο σύμπλεγμα (cluster). Δε χρειάζεται ούτε σπατάλη του χρόνου του διαχειριστή της βάσης, ούτε πολύπλοκος κώδικας για partitioning και replication. Ιδιαίτερα με τις νέες τεχνολογίες του cloud και των virtual machines, η κλιμάκωση είναι ιδιαίτερα εύκολη και φθηνή λύση[145]. Σε αυτό το σημείο θα πρέπει να αναφερθεί ότι για την κλιμάκωση μιας βάσης δεδομένων υπάρχουν δύο επιλογές α) κατακόρυφη, β) οριζόντια κλιμάκωση. Κατακόρυφη σημαίνει να αυξάνονται συνεχώς οι δυνατότητες των υπάρχοντων εξυπηρετητών ενώ οριζόντια σημαίνει απλά προσθήκη υπολογιστών στο σύμπλεγμα όπως αναφέρθηκε και παραπάνω.

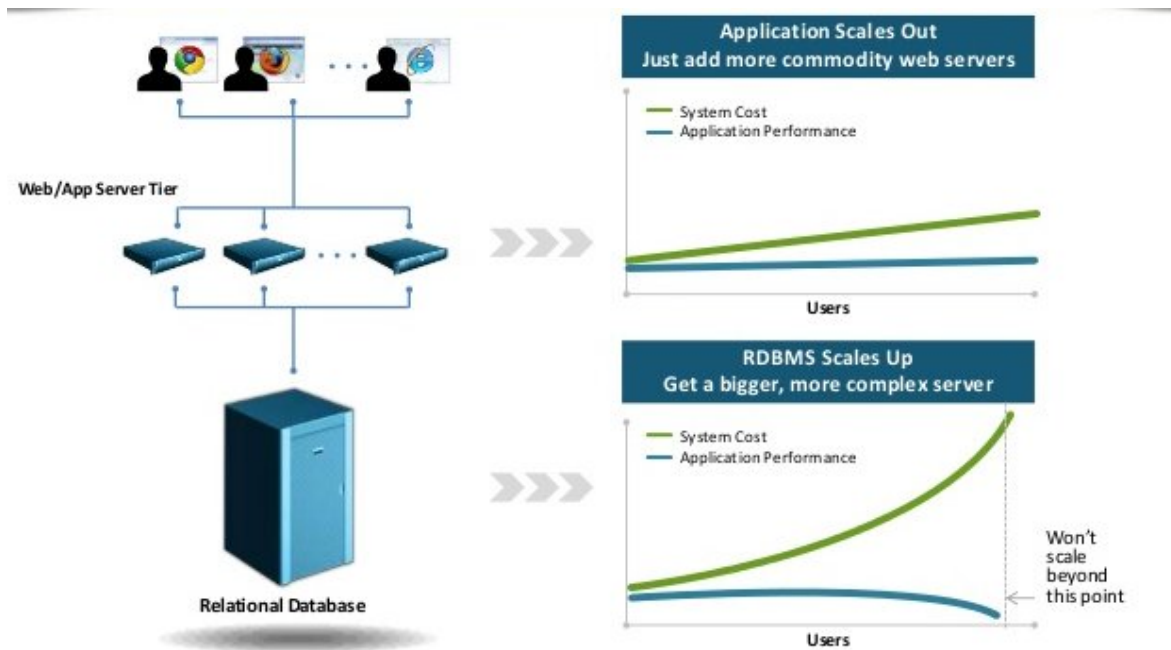
- **Κατακόρυφη κλιμάκωση**

Στο επίπεδο της εφαρμογής της αρχιτεκτονικής διαδικτύου τριών επιπέδων, η οριζόντια κλιμάκωση ήταν η προεπιλεγμένη λύση για πολλά χρόνια και λειτουργούσε πολύ καλά. Καθώς όλο και περισσότεροι άνθρωποι χρησιμοποιούν μια εφαρμογή, περισσότεροι εξυπηρετητές του εμπορίου προστίθενται στο επίπεδο της εφαρμογής, η απόδοση διατηρείται με την κατανομή του φόρτου εργασίας σε ένα αυξημένο αριθμό εξυπηρετητών και το κόστος κλιμακώνεται γραμμικά σε σχέση με τον αριθμό των χρηστών

Πριν από τις NoSQL βάσεις δεδομένων, η προεπιλεγμένη προσέγγιση ως προς την κλιμάκωση στο επίπεδο της βάσης δεδομένων ήταν η κατακόρυφη κλιμάκωση. Αυτό υπαγορευόταν από τη θεμελιωδώς συγκεντρωτική αρχιτεκτονική της σχεσιακής τεχνολογίας βάσεων δεδομένων. Για την υποστήριξη περισσότερων ταυτόχρονων χρηστών και/ή την αποθήκευση περισσότερων δεδομένων, χρειάζονται όλο και υψηλότερων δυνατοτήτων εξυπηρετητές με περισσότερους επεξεργαστές, περισσότερη μνήμη και περισσότερο αποθηκευτικό χώρο για όλους τους πίνακες. Οι μεγάλοι εξυπηρετητές τείνουν να είναι πολύ πολύπλοκοι, ιδιοταγείς και δυσανάλογα ακριβοί εν αντιθέσει με αυτούς του εμπορίου.

- **Οριζόντια κλιμάκωση**

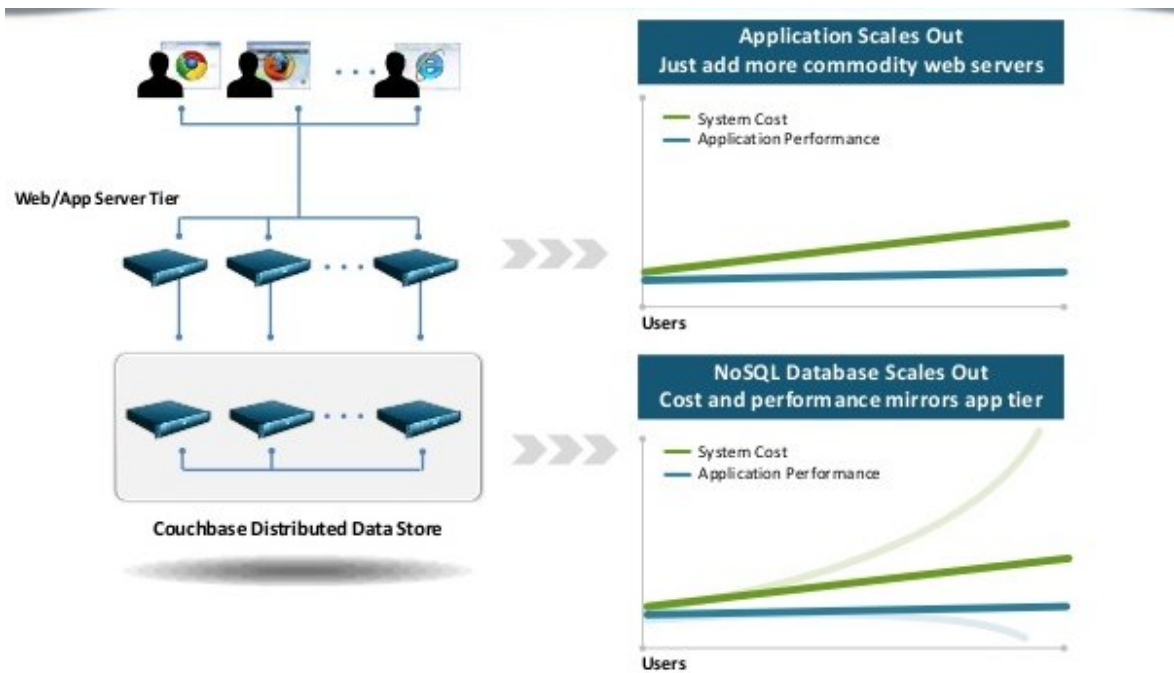
Οι NoSQL βάσεις δεδομένων αναπτύχθηκαν εξ αρχής ώστε να είναι κατανεμημένες και να ευνοούν την οριζόντια κλιμάκωση. Χρησιμοποιούν ένα σύμπλεγμα τυπικών, φυσικών ή εικονικών εξυπηρετητών για να αποθηκεύουν δεδομένα και να υποστηρίζουν τις λειτουργίες της βάσης δεδομένων. Για κλιμάκωση, επιπλέον εξυπηρετητές προστίθενται στο σύμπλεγμα και τα δεδο-



**Σχήμα 4.12:** Κλιμακωσιμότητα RDBMS.

μένα και οι λειτουργίες της βάσης δεδομένων μοιράζονται στο μεγαλύτερο σύμπλεγμα. Επίσης, επειδή οι εξυπηρετητές του εμπορίου αναμένεται να παρουσιάσουν σφάλματα ανά κάποια χρονικά διαστήματα, οι βάσεις δεδομένων NoSQL έχουν δημιουργηθεί έτσι ώστε να είναι ανεκτικές και να ανακάμπτουν από τέτοια προβλήματα και αυτό τις καθιστά πολύ ανθεκτικές. Οι NoSQL βάσεις δεδομένων παρέχουν μια πολύ ευκολότερη και γραμμική προσέγγιση στην κλιμάκωση των βάσεων δεδομένων. Για κάθε 10,000 νέους χρήστες που αρχίζουν να χρησιμοποιούν μια εφαρμογή, τότε πρέπει απλά να προστεθεί μια ένας νέος εξυπηρετητής βάσης δεδομένων στο σύμπλεγμα. Με αυτόν τον τρόπο η εφαρμογή δεν χρειάζεται τροποποίηση καθώς αυτή βλέπει συνέχεια μόνο μια (κατανεμημένη) βάση δεδομένων.

Μια κατανεμημένη προσέγγιση οριζόντιας κλιμάκωσης επίσης συνήθως καταλήγει να είναι φθηνότερη από την εναλλακτική κατακόρυφης κλιμάκωσης. Αυτό είναι συνέπεια του ότι οι μεγάλοι, πολύπλοκοι και ανεκτικοί στα σφάλματα εξυπηρετητές έχουν υψηλά κόστη σχεδιασμού, κατασκευής και συντήρησης. Τα κόστη των αδειών των εμπορικών σχεσιακών βάσεων δεδομένων μπορεί επίσης να είναι απαγορευτικά, λόγω του ότι κοστολογούνται θεωρώντας ότι θα χρησιμοποιηθεί ένας μόνο εξυπηρετητής. Από την άλλη, οι βάσεις δεδομένων NoSQL είναι γενικά ανοιχτού κώδικα και κοστολογούνται με βάση το ότι θα χρησιμοποιηθούν σε ένα σύμπλεγμα εξυπηρετητών και είναι περισσότερο οικονομικές.



Σχήμα 4.13: Κλιμακωσιμότητα NoSQL.

### Ενσωματωμένο caching

Οι περισσότερες NoSQL βάσεις παρέχουν τεχνολογίες ενσωματωμένου caching. Τα συχνά χρησιμοποιούμενα δεδομένα αποθηκεύονται στη μνήμη όσο το δυνατόν περισσότερο και επομένως η επικοινωνία με το δίσκο ελαχιστοποιείται. Έτσι δεν υπάρχει η ανάγκη για ξεχωριστό επίπεδο caching, ενώ στις περισσότερες SQL βάσεις χρειάζεται ξεχωριστή υποδομή για να επιτευχθεί κάτι τέτοιο [145].

### Μικρότερο κόστος

Στις NoSQL βάσεις χρησιμοποιούνται φθινοί, απλοί servers που μπορεί να βρίσκονται και στο cloud ή να είναι εικονικοί. Από την άλλη, τα RDBMS βασίζονται σε ακριβούς, ιδιωτικούς servers και συστήματα αποθήκευσης. Το κόστος gb ή δοσοληψία ανά δευτερόλεπτο είναι επομένως πολύ χαμηλότερο σε ένα σύστημα NoSQL από ένα σύστημα SQL [145].

## 4.2 Front-End

### 4.2.1 Αρχιτεκτονική

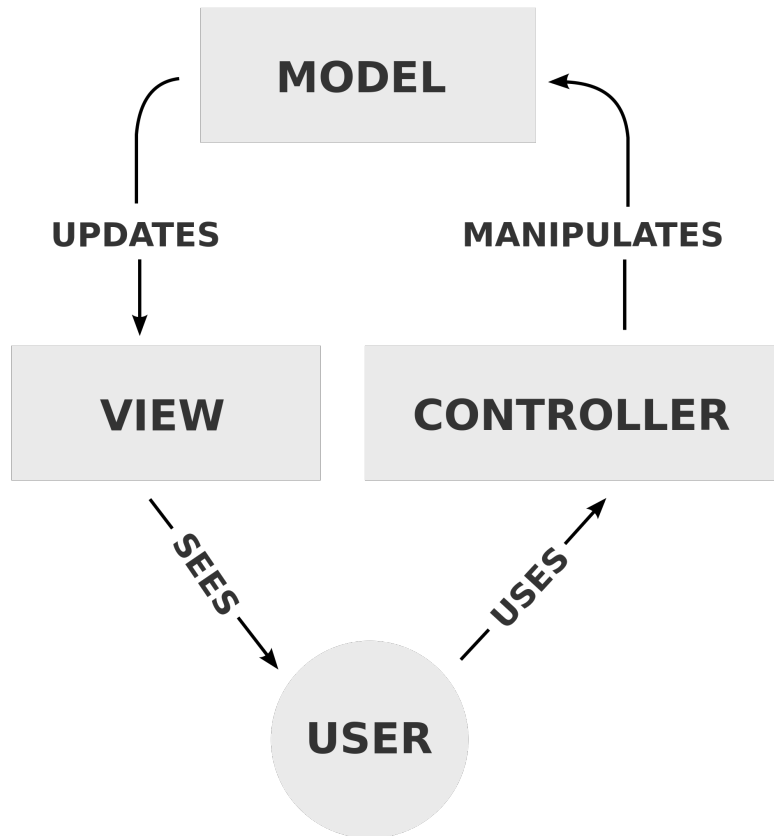
#### MVC αρχιτεκτονικό πρότυπο

Το σχεδιαστικό μοτίβο Model View Controller (MVC) είναι ένα αρχιτεκτονικό πρότυπο που χρησιμοποιείται στην τεχνολογία λογισμικού και συνήθως στην υλοποίηση web εφαρμογών. Χωρίζει την εφαρμογή σε τρία διασυνδεδεμένα μέρη, με σκοπό τον πλήρη διαχωρισμό της εσωτερικής αναπαράστασης της πληροφορίας στο σύστημα από τους τρόπους παρουσίασης της πληροφορίας στον χρήστη. Το MVC αποτελείται από τρία στοιχεία: α) Model, β) View, γ) Controller.

- **Model:** Το model είναι το κομμάτι της εφαρμογής που αναπαριστά την πληροφορία, δηλαδή τα δεδομένα που χρησιμοποιεί η εφαρμογή και επιβάλλει τις ενέργειες και τους περιορισμούς πάνω σε αυτά. Στο model τοποθετούμε τις λειτουργίες της εφαρμογής που σχετίζονται με την πρόσβαση στη βάση δεδομένων. Οι λειτουργίες αυτές είναι συναρτήσεις με τις οποίες διαχειριζόμαστε αποτελεσματικά τα δεδομένα που λαμβάνουμε από τη βάση. Το μοντέλο ανταποκρίνεται σε ερωτήματα από το view καθώς και σε οδηγίες από τον controller για να πραγματοποιήσει ενημερώσεις.
- **View:** Αντιστοιχεί στη γραφική παρουσίαση των δεδομένων του model στο interface του χρήστη. Ο controller, είναι αυτός ο οποίος αποφασίζει πότε θα παρουσιαστούν τα δεδομένα ενώ το view καθορίζει τον τρόπο με τον οποίο θα παρουσιαστούν τα δεδομένα στον χρήστη. Το επίπεδο view θα πρέπει να προσαρμόζεται στις αλλαγές του επιπέδου του model, για αυτό και τα views θα πρέπει να επικεντρώνονται μόνο στην εμφάνιση δεδομένων και να μην εμπλέκονται με την επιχειρηματική λογική του model.
- **Controller:** Ο controller είναι αυτός που παίρνει αποφάσεις, δηλαδή διαχειρίζεται τη σύνδεση των δεδομένων του model με τη λογική του προγράμματος και καθορίζει ποια από τα δεδομένα του model θα παρουσιαστούν από το view στο interface του χρήστη. Ο controller ενημερώνει το view κάθε φορά που αλλάζει το model. Επιπλέον προσθέτει event listeners στο view και ενημερώνει το model όταν ο χρήστης αλληλεπιδρά με το view.

Η χρήση της αρχιτεκτονικής MVC κατά την δημιουργία μίας εφαρμογής μας προσφέρει τα εξής βασικά πλεονεκτήματα:





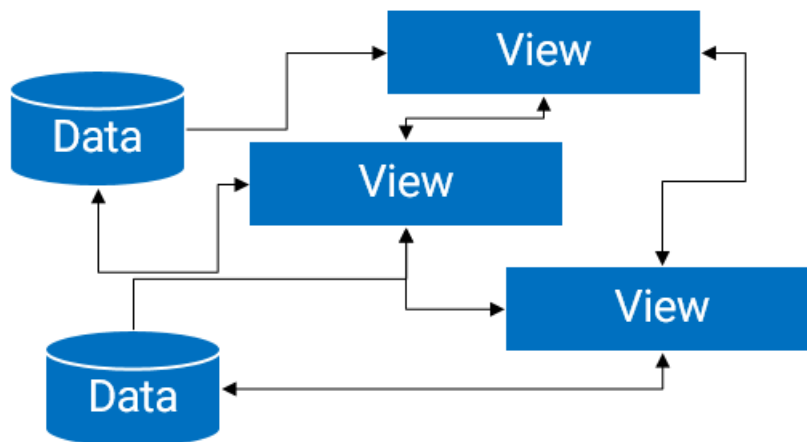
**Σχήμα 4.14:** Αρχιτεκτονική MVC.

- **Διαχωρισμός των προβλημάτων (Separation of Concerns).** Ουσιαστικά δημιουργείται μία εφαρμογή η οποία έχει τρία επίπεδα(models, views, controllers) και το κάθε επίπεδο επιτελεί ξεχωριστό έργο και ταυτόχρονα συνεργάζεται με τα άλλα επίπεδα. Στην σωστή υλοποίηση πρέπει τα τρία επίπεδα να είναι πλήρως καθορισμένα και να μην συμπλέκονται. Ο διαχωρισμός αυτός, επιτρέπει την επαναχρησιμοποίηση της επιχειρηματικής λογικής σε διάφορες εφαρμογές και την ανάπτυξη πολλαπλών user interfaces χωρίς να προβληματιζόμαστε από τον κώδικα που διαχειρίζεται την βάση.
- **Επεκτασιμότητα (Scaleability).** Είναι η δυνατότητα που διαθέτει μία εφαρμογή , να μπορούμε μελλοντικά να προσθέσουμε λειτουργίες σε αυτή ή να αλλάξουμε κάποιες από τις ήδη υπάρχουσες λειτουργίες με σκοπό να επιτύχουμε διαφορετικά αποτελέσματα.
- **Ελεγχιμότητα (Testability).** Οι MVC εφαρμογές έχουν τη δυνατότητα να είναι πιο εύκολα ελέγξιμες και με τον τρόπο αυτόν συντηρούνται πιο εύκολα. Εφόσον τα συστατικά της αρχιτεκτονικής είναι διακριτά, είναι πιο εύκολο να γράφουμε κώδικα που να τεστάρει ξεχωριστά το κάθε κομμάτι, γρήγορα και αποτελεσματικά.

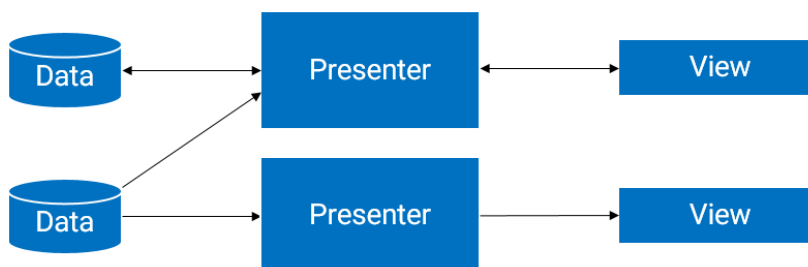
- **Παράλληλη ανάπτυξη από διαφορετικές ομάδες.** Οι προγραμματιστές επιχειρηματικής λογικής μπορούν να δημιουργούν τις κλάσεις, ενώ οι προγραμματιστές του user interface μπορούν να σχεδιάζουν ταυτόχρονα τις οθόνες. Επίσης μπορούν να γίνονται ενημερώσεις και αλλαγές στο user interface χωρίς να επιβραδύνεται η διαδικασία του business logic.

### MVP αρχιτεκτονικό πρότυπο

Το αρχιτεκτονικό πρότυπο Model View Presenter (MVP) αποτελεί ένα παράγωγο αρχιτεκτονικό πρότυπο του MVC [129], το οποίο είναι πολύ δημοφιλές στην ανάπτυξη εφαρμογών για το λειτουργικό σύστημα Android. Στην θεωρία τουλάχιστον, οι εφαρμογές Android είναι στημένες με τη λογική του MVC [30], όπως περιγράφηκε στην ενότητα 4.2.1, με το Activity ή το Fragment να αποτελεί τον controller και τα xml layouts τα views. Στην πράξη όμως καταλήγει η πλειονότητα του view logic να βρίσκεται στο Activity το οποίο το μετατρέπει σε δομή model-view όπως φαίνεται στο σχήμα 4.15, κάτι το οποίο σε καμία περίπτωση δεν είναι το επιθυμητό. Στόχος λοιπόν του MVP είναι να λύσει ακριβώς αυτό το πρόβλημα και να πετύχει πλήρη διαχωρισμό μεταξύ του view και του view logic όπως βλέπουμε στο σχήμα 4.16.



Σχήμα 4.15: Android Model-View αρχιτεκτονικό πρότυπο.



Σχήμα 4.16: Android MVP αρχιτεκτονικό πρότυπο.

Πλεονεκτήματα και λόγοι χρήσης του αρχιτεκτονικού προτύπου Model View Presenter[140]:

- **Επεκτασιμότητα** η οποία επιτυγχάνεται μέσω του πλήρη διαχωρισμού μεταξύ της εμφάνισης (view) και της λογικής της εμφάνισης (view logic).
- **Ευκολία Testing** του κάθε στρώματος (layer) της εφαρμογής αφού έχουμε πετύχει σαφή διαχωρισμό μεταξύ τους. Εν γένει η χρήση του μοτίβου MVP προάγει το test driven development της εφαρμογής με όλα τα πλεονεκτήματα που συνεπάγεται η εν λόγω τεχνική ανάπτυξης.

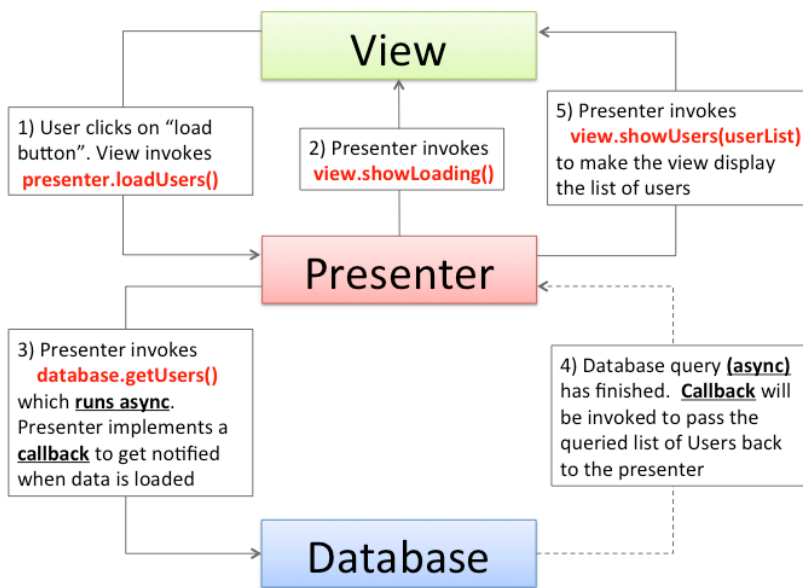
Το αρχιτεκτονικό πρότυπο MVP όπως δηλώνει και το όνομα του αποτελείται από τα παρακάτω τρία στοιχεία α) Model β) View γ) Presenter :

- **Model:** Το model στο αρχιτεκτονικό πρότυπο MVP επιτελεί της ίδιες λειτουργίες με αυτές στο MVC όπως περιγράφηκαν παραπάνω στην ενότητα 4.2.1.
- **View:** Το view υλοποιείται συνήθως μέσω κάποιου Activity ή Fragment (εξαρτάται από τη συγκεκριμένη αρχιτεκτονική της εφαρμογής), και περιέχει αναφορά στον presenter ο οποίος το χειρίζεται. Η εν λόγω αναφορά πραγματοποιείται με χρήση του σχεδιαστικού μοτίβου dependency injection και ιδανικά με χρήση του Dagger. Κάθε φορά που ο χρήστης αλληλεπιδρά με κάποια λειτουργία (π.χ κουμπί) του view, καλείται η αντίστοιχη συνάρτηση του presenter ο οποίος επικοινωνεί με το model και λέει εν τέλει στο view τι θα εμφανίσει.
- **Presenter:** Ο Presenter είναι λειτουργεί ως το ενδιάμεσο στάδιο - εννοείται μεταξύ του view και του model. Είναι υπεύθυνος για την απόκτηση των δεδομένων από το model και την παρουσίαση τους στο view, αλλά σε αντίθεση με την κλασσική αρχιτεκτονική MVC είναι υπεύθυνος για τις αλλαγές που γίνονται στο view όταν ο χρήστης αλληλεπιδρά μαζί του.

Για να αποσαφηνιστεί πλήρως η λειτουργία του και να γίνει πιο εύκολα κατανοητό στο σχήμα 4.17 βλέπουμε ένα χαρακτηριστικό παράδειγμα.

## 4.2.2 Frameworks

Το framework, στον προγραμματισμό ηλεκτρονικών υπολογιστών, είναι μια αφηρημένη έννοια με την οποία κοινός κώδικας ο οποίος παρέχει γενικές λειτουργίες, μπορεί να επανεγγραφεί επιλεκτικά ή να εξειδικευτεί μέσω του κώδικα κάποιου χρήστη ώστε να χρησιμοποιηθεί για συγκεκριμένες λειτουργίες. Τα frameworks είναι μια ειδική περίπτωση βιβλιοθηκών λογισμικού που εμπεριέχουν



Σχήμα 4.17: Παράδειγμα MVP σε Android.

επαναχρησιμοποιήσιμα κομμάτια κώδικα σε μια καλά καθορισμένη διεπαφή προγραμματισμού εφαρμογών (API), που όμως περιέχουν κάποια βασικά χαρακτηριστικά γνωρίσματα που τα διαφοροποιούν από τις κανονικές βιβλιοθήκες. Κάποια από τα γνωρίσματα αυτά είναι: αντιστροφή του ελέγχου (η ροή του προγράμματος υπαγορεύεται από το framework και όχι από αυτόν που το χρησιμοποιεί), προκαθορισμένη συμπεριφορά, επεκτασιμότητα (μπορεί να επεκταθεί από το χρήστη συνήθως με επανεγγραφή ή εξειδίκευση του κώδικα ώστε να εκτελούνται συγκεκριμένες λειτουργίες. [154]

## Express Framework

Το Express είναι ένα δρομολόγησης και middleware framework, το οποίο έχει ελάχιστη λειτουργικότητα από μόνο του. Μια Express εφαρμογή ουσιαστικά αποτελεί μία σειρά από middleware κλήσεις. Ο δημιουργός του Express framework, TJ Holowaychuk, το έχει περιγράψει ως "Sinatra Inspired Server", που σημαίνει ότι είναι σχετικά ελάχιστο, αλλά παρέχει πολλά επιπλέον χαρακτηριστικά, τα οποία είναι διαθέσιμα ως πρόσθετα.

Το middleware είναι μια συνάρτηση με πρόσβαση στο αντικείμενο της αίτησης (req), το αντικείμενο απάντηση (res), και το επόμενο middleware στον κύκλο αίτησης-απάντησης της εφαρμογής, που συνήθως συμβολίζεται με μια μεταβλητή με το όνομα next.

Το middleware μπορεί να:

- Εκτελέσει οποιοδήποτε κώδικα.

- Να κάνει αλλαγές στην αντικείμενα αίτησης και απάντησης(request and response objects).
- Να τερματίζει τον κύκλο αίτησης-απάντησης.
- Να καλέσει το επόμενο middleware της στοίβας.
- Εάν το τρέχον middleware δεν τελειώσει τον κύκλο αίτησης-απάντησης, θα πρέπει να κληθεί το επόμενο, next(), για να περάσει ο έλεγχος στο επόμενο middleware, διαφορετικά η αίτηση θα μείνει ξεκρέμαστη.

Μία εφαρμογή Express μπορεί να χρησιμοποιήσει τα ακόλουθα είδη middleware: middleware επιπέδου εφαρμογής (application-level middleware), middleware επιπέδου δρομολόγησης, middleware χειρισμού σφαλμάτων(error-handling middleware), ενσωματωμένο middleware (built-in middleware), middleware τρίτων κατασκευαστών middleware (third-party middleware).

## Angular

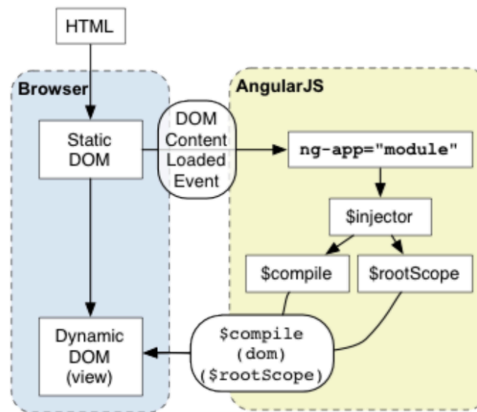
Το AngularJS είναι ένα από τα πιο δημοφιλή ανοιχτού κώδικα JavaScript framework που χρησιμοποιείται για την κατασκευή και τη δομή σύγχρονων διαδικτυακών εφαρμογών, κυρίως εφαρμογών μίας σελίδας. Αναπτύχθηκε από τη Google και από μία κοινότητα προγραμματιστών για την αντιμετώπιση των πολλών προκλήσεων που ανέκυψαν κατά την ανάπτυξη των εφαρμογών μίας σελίδας. Στόχος του είναι να απλοποιήσει τόσο την ανάπτυξη όσο και το testing αυτών των εφαρμογών, παρέχοντας ένα framework για client-side model-View-Controller (MVC) και το model-view-viewmodel (MVVM) αρχιτεκτονικές, μαζί με τα συστατικά που χρησιμοποιούνται συνήθως στις εφαρμογές Διαδικτύου.

### Αρχιτεκτονική

Κατά την εκκίνηση της εφαρμογής, το πρόγραμμα περιήγησης φορτώνει το HTML αρχείο και το κάνει “parse” στο DOM αρχείο, και μαζί με αυτό στο AngularJs script. Μόλις το DOM αρχείο έχει φορτωθεί, το AngularJS αναζητά την εντολή ng-app η οποία ορίζει τα όρια της εφαρμογής. Εάν ορίζεται κάποιο module μέσα στην οδηγία, τότε χρησιμοποιείται για να ρυθμιστεί το \$injector, που δημιουργεί το \$compile service καθώς και το \$rootScope. Τότε μεταγλωττίζεται το DOM και συνδέεται στο \$rootScope μετά από το οποίο εκτελούνται οι υπόλοιπες εντολές (αν υπάρχουν). Στο σχήμα 4.18 φαίνεται ο κύκλος εκκίνησης του AngularJS.

Κάποια από τα βασικά χαρακτηριστικά του AngularJS είναι:

- Two Way Data-binding. Αυτό πρακτικά σημαίνει ότι όταν αλλάζει το model αλλάζει και το view,



Σχήμα 4.18: AngularJS κύκλος εκκίνησης.

και όταν αλλάζει το view αλλάζει και το model. Το Two Way Data-binding εξηγείται καλύτερα με ένα παράδειγμα:

```

1 <input type="text" ng-model="myName">
2 <h1>Hello {{myName}}!</h1>

```

Αν ο χρήστης αλλάζει κάποια τιμή του model μέσω της εισόδου στο view, τότε το AngularJs αποθηκεύει την τιμή αυτή σε μια μεταβλητή. Το στοιχείο h1 στην συνέχεια ενημερώνεται αυτόματα ώστε να αντικατοπτρίζονται οι αλλαγές στην τιμή της μεταβλητής του model. Το view ενημερώνεται ακόμα και αν η μεταβλητή αλλάξει χειροκίνητα με χρήση JavaScript.[77]

- HTML Templates. Ενώ άλλα JavaScript frameworks χρησιμοποιούν ένα template system που βασίζεται στην HTML με ειδική σήμανση το οποίο μπορεί να είναι δυσκολεύει τους προγραμματιστές, το AngularJS δεν στηρίζεται σε εξωτερικές template engines αλλά χρησιμοποιεί ένα templating σύστημα, που είναι χτισμένη πάνω στην HTML με έξυπνη χρήση των ng- χαρακτηριστικών. Το πρόγραμμα περιήγησης διαβάζει την HTML και “ψάχνει” ng - directives τα οποία, όταν εκτελεστούν, δεσμεύουν το view στο model.
- Dependency Injection. Το dependency injection ενθαρρύνεται πολύ από το AngularJS και είναι ένα από τα συστατικά του, που βελτιώνει σε μεγάλο βαθμό το testability του. Το dependency injection είναι ένα design pattern λογισμικού στο οποίο σε ένα αντικείμενο ορίζονται οι εξαρτήσεις του και δεν τις δημιουργεί μόνο του. Πρόκειται ουσιαστικά για την αφαίρεση των hard-coded εξαρτήσεων, κάνοντας δυνατό να αλλάξουν οι εξαρτήσεις όποτε χρειαστεί. Ένα παράδειγμα είναι όταν υπάρχει μία συνάρτηση η οποία παίρνει μία παράμετρο, όπως:

```

1 addThing = function ( thing )
2   {
3     $scope . add ( thing ) ;
4   }

```

Το παραπάνω παράδειγμα είναι εύκολο να το τεστάρουμε, επειδή το στοιχείο που εισάγεται στην συνάρτηση μπορεί να είναι είτε ένα πραγματικό αντικείμενο ή ένα αντικείμενο για τους σκοπούς των τεστ. Ο διαχωρισμός σε μικρότερα πιο εύκολα ελέγξιμα συστατικά μπορεί να βοηθήσει πολύ στην ανίχνευση των σφαλμάτων.

- **Deep Linking.** Σε μια εφαρμογή μίας σελίδας είναι σημαντικό να διατηρείται η κατάσταση της εφαρμογής στο url, έτσι ώστε οι χρήστες είναι σε θέση να βάλουν κάποιο σελιδοδείκτη ή να μοιραστούν κάποιον σύνδεσμο σε διάφορες καταστάσεις της εφαρμογής. Το AngularJS χρησιμοποιεί το HTML5 history API μαζί με ένα (#!) εφεδρικό για παλαιότερα προγράμματα περιήγησης. Η λειτουργία αυτή είναι εξαιρετικά ισχυρή στη σύγχρονη εποχή των κοινωνικών δικτύων και του sharing.
- **Directives.** Τα directives είναι κάτι μοναδικό στο AngularJS και επιτρέπουν την επέκταση της λειτουργικότητας της HTML. Ενώ το AngularJS έχει μια συλλογή από προκαθορισμένα directives, μπορεί να επεκταθεί με συγκεκριμένες λειτουργικότητες σε σημείο που να επιτρέπει στον χρήστη να δημιουργήσει το δικό του DSL (Domain specific language). Επίσης επιτρέπει να δημιουργηθούν προσαρμοσμένα στοιχεία DOM, καθώς και χαρακτηριστικά και κλάσεις στα οποία μπορούν να προστεθούν λειτουργικότητες. Το επόμενο παράδειγμα από το επίσημο documentation του AngularJS δείχνει ότι για παράδειγμα γίνεται να επιτραπούν data-bindings για ένα html στοιχείο, αν υπάρχει ένα συγκεκριμένο χαρακτηριστικό. [11]

#### HTML:

```

1 <div contentEditable="true" ng-model="content">Edit Me</div>
2   <pre>{{ content }}</pre>

```

#### JavaScript:

```

1 angular . module ( ' directive ' , [ ] ) . directive ( ' contenteditable ' , ↵
2   function ( )
3   return {

```

```

4   require: 'ngModel',
5   link: function(scope, elm, attrs, ctrl) {
6     // view -> model
7     elm.bind('blur', function() {
8       scope.$apply(function() {
9         ctrl.$setViewValue(elm.html());
10      });
11    });
12
13    // model -> view
14    ctrl.$render = function(value) {
15      elm.html(value);
16    };
17
18    // load init value from DOM
19    ctrl.$setViewValue(elm.html());
20  }
21  };
22  });

```

- \$http service. Το AngularJS είναι εξαιρετικά ευέλικτο στον τρόπο που επικοινωνεί με τα διαφορετικά backends. Αντί να βασίζεται αποκλειστικά στο REST, επικοινωνεί ελεύθερα μέσω των XMLHttpRequests του προγράμματος περιήγησης ή μέσω του JSONP. Παρόλο που το \$ http service προσθέτει http headers ως προεπιλογή στα αιτήματα του, είναι εύκολα διαμορφώσιμα μέσω του \$httpProvider.defaults.headers.

## Bootstrap

Το bootstrap είναι ένα πάρα πολύ δυνατό front-end framework για την ταχύτερη και ευκολότερη ανάπτυξη ιστοσελίδων. Είναι ουσιαστικά μία συλλογή εργαλείων ανοιχτού κώδικα και περιλαμβάνει HTML και CSS πρότυπα σχεδιασμού για κοινά στοιχεία διεπαφής χρήστη, όπως τυπογραφίας, φόρμες, κουμπιά, πίνακες, dropdowns, alerts, μπάρες, και πολλά άλλα, καθώς και προαιρετικές επεκτάσεις JavaScript. Το bootstrap είναι συμβατό με όλες τις τελευταίες εκδόσεις των προγραμμάτων πλοήγησης και υποστηρίζει "responsive web design". Αυτό σημαίνει πως η διάταξη των ιστοσελίδων προσαρμόζεται δυναμικά, λαμβάνοντας υπόψιν τα χαρακτηριστικά της συσκευής που το χρησιμοποιεί (σταθερός υπολογιστής, κινητό, τάμπλετ κ.τ.λ.).



Το bootstrap είναι modular και αποτελείται από μικρότερα LESS stylesheets τα οποία υλοποιούν τα διάφορα στοιχεία του πακέτου εργαλείων. Ένα stylesheet, με όνομα bootstrap.less περιλαμβάνει τα επιμέρους stylesheets. Οι προγραμματιστές μπορούν να προσαρμόσουν και οι ίδιοι το bootstrap, καθώς τους δίνεται η δυνατότητα να επιλέξουν τα στοιχεία που θέλουν να χρησιμοποιήσουν. Αναπροσαρμογές γίνονται σε περιορισμένο όμως βαθμό, μέσω ενός κεντρικού stylesheet. Πιο ουσιαστικές αλλαγές πραγματοποιούνται με LESS δηλώσεις. Η χρήση των LESS stylesheets επιτρέπει τη χρήση μεταβλητών, συναρτήσεων καθώς και mixins. Επιπλέον, ο σχεδιαστής μπορεί να επιλέγει σε μια φόρμα τα επιθυμητά στοιχεία και να τα προσαρμόζει, σε τιμές διαφόρων εναλλακτικών λύσεων για τις ανάγκες του. Στη συνέχεια δημιουργείται ένα πακέτο που περιλαμβάνει ήδη το προ-χτισμένο CSS stylesheet.

Το bootstrap παρέχει ένα σύνολο από stylesheets που παρέχουν βασικούς ορισμούς στυλ για όλα τα βασικά HTML στοιχεία. Αυτά παρέχουν ενιαία, σύγχρονη εμφάνιση για πίνακες, μορφοποίηση κειμένου, καθώς και για τα στοιχεία μιας φόρμας. Επιπλέον των HTML στοιχείων, το bootstrap περιέχει και άλλα στοιχεία περιβάλλοντος τα οποία χρησιμοποιούνται συχνά. Αυτά περιλαμβάνουν κουμπιά με προηγμένα χαρακτηριστικά ( π.χ. ομαδοποίηση κουμπιών ή drop-down επιλογή, οριζόντιες και κάθετες καρτέλες, σελιδοποίηση, κ.λ.π. ), ετικέτες, προηγμένες τυπογραφικές δυνατότητες, εικονίδια, προειδοποιητικά μηνύματα και μια γραμμή προόδου. Τέλος, το bootstrap περιέχει πολλά συστατικά JavaScript σε μια μορφή jQuery plugin, τα οποία παρέχουν πρόσθετα στοιχεία διεπαφής χρήστη όπως παράθυρα διαλόγου, επεξηγήσεις, και καρουσέλ. Μπορούν επίσης να επεκτείνουν τη λειτουργικότητα ορισμένων υφιστάμενων στοιχείων της διασύνδεσης, όπως για παράδειγμα μια αυτόματη πλήρη λειτουργία για πεδία εισαγωγής[153]. Τα πιο σημαντικά πλεονεκτήματα που προκύπτουν από την χρήση του bootstrap είναι:

- **Ταχύτητα ανάπτυξης:** Αδιαμφισβήτητα, ένα από τα σημαντικότερα πλεονεκτήματα της χρήσης του bootstrap είναι η ταχύτητα της ανάπτυξης. Το bootstrap παρέχει τη δυνατότητα χρήσης έτοιμων μπλοκ κώδικα, έτσι ώστε να γίνεται η αρχή του front-end από εκεί και να μην χρειάζεται να γραφτεί κώδικας από το μηδέν ("from scratch"). Το άνωθεν σε συνδυασμό με τη συμβατότητα σε όλα τα προγράμματα περιήγησης που προσφέρει καθώς και τις λειτουργίες του CSS-LESS, εξοικονομεί πολύτιμο χρόνο στους προγραμματιστές.
- **Ανταποκρισιμότητα:** Λόγω του διαρκώς αυξανόμενου αριθμού των κινητών συσκευών υπάρχει η ανάγκη να έχουμε διαδραστικές ιστοσελίδες. Το bootstrap διευκολύνει πάρα πολύ την δημιουργία οθονών για κινητά χάρη στη ρευστή διάταξη πλέγματος που έχει, η οποία προσαρμόζεται δυναμικά στην κατάλληλη ανάλυση οθόνης και δεν χρειάζεται να προβούμε σε επιπλέον ενέργειες για να πετύχουμε την σωστή αποκρισιμότητα.

- **Συνοχή:** Το bootstrap βασίζεται σε αυτή την αρχή καθώς αρχικά αναπτύχθηκε από μερικούς υπαλλήλους του Twitter ως ένα framework για να υπάρχει συνέπεια μεταξύ των εσωτερικών εργαλείων που χρησιμοποιούνταν. Το bootstrap ουσιαστικά χτίστηκε ώστε να "ταιριάζει" τους σχεδιαστές με τους προγραμματιστές και να διασφαλίζει τη συνέπεια, ανεξάρτητα του ποιος εργάζεται στο έργο ή σε ποια πλατφόρμα θα τρέξει.
- **Προσαρμοστικότητα:** Το bootstrap, μπορεί να προσαρμοστεί σύμφωνα με τις προδιαγραφές του κάθε έργου. Οι προγραμματιστές έχουν τη δυνατότητα να επιλέξουν τα χαρακτηριστικά που χρειάζονται και να πετάξουν τα υπόλοιπα.

### 4.2.3 Jade Template Engine

Μια μηχανή πρότυπο είναι μια βιβλιοθήκη ή ένα framework που χρησιμοποιεί ορισμένους κανόνες ή μία συγκεκριμένη γλώσσα για να ερμηνεύσει τα δεδομένα και να παρουσιάσει τα views. Στην περίπτωση των διαδικτυακών εφαρμογών, τα views είναι σελίδες HTML (ή τμήματα αυτών), αλλά γενικότερα μπορεί να είναι JSON ή XML αρχεία, ή graphic user interfaces. Με βάση το μοντέλο Model-View-Controller (MVC), τα πρότυπα ανήκουν στα views.

Υπάρχουν πολλοί λόγοι που οδηγούν στη χρήση μιας μηχανή προτύπων αντί για την συγγραφή του front-end κώδικα σε μια συγκεκριμένη γλώσσα. Το μεγάλο πλεονέκτημα αυτής της προσέγγισης, είναι ότι παρουσιάζει HTML στην πλευρά του εξυπηρετητή (server), την οποία στέλνει στον πελάτη (client) και στη συνέχεια όταν ο πελάτης αλληλεπιδράσει με τη σελίδα και η HTML αλλάξει χρησιμοποιώντας τα ίδια bits του πρότυπου και ίσως ακόμη και κάποια κοινή λογική. Άλλοι πολύ σημαντικοί λόγοι αποτελούν η συνέπεια του κώδικα και η οργάνωση. Εάν υπάρχει καλός διαχωρισμός της σήμανσης (markup) από τη λογική γίνεται πιο εύκολος εντοπισμός σφαλμάτων του κώδικα, κοινή χρήση του κώδικα με συνεργάτες, καθώς και επιτυγχάνεται ταχύτερη ανάπτυξη μέσω της απόκτησης κοινών μεθόδων για την έξοδο της σήμανσης (markup). Οι μηχανές προτύπων που λειτουργούν, εξίσου καλά και από την πλευρά του εξυπηρετητή (server), όπως και από την πλευρά του πελάτη (client) αξιοποιούν το πλεονέκτημα ότι της ισομορφικής JavaScript, δηλαδή τη δυνατότητα που υπάρχει να τρέξει ο ίδιος κώδικας παντού. [113]

Η μηχανή προτύπων Jade (Jade Template Engine) είναι μία μηχανή υψηλής απόδοσης η οποία έχει επηρεαστεί σε μεγάλο βαθμό από το Haml και υλοποιήθηκε με χρήση Javascript για το Node. Η μηχανή Jade παράγει HTML, που χρησιμοποιείται στο πρόγραμμα περιήγησης στην πλευρά του εξυπηρετητή (server-side). Το πρόγραμμα περιήγησης (browser) εκτελεί μια αίτηση στον web-server, ο web-server εκτελεί την Jade, η οποία θα δημιουργήσει το HTML που θα πρέπει να σταλεί στον

browser. Η μηχανή προτύπων Jade χρησιμοποιεί τα κενά και τις εσοχές ως μέρος της γλώσσας. Παράγει αυτόματα συμβατή σήμανση (markup) που βασίζεται σε μια ένθετη ιεραρχία με αναγνωριστικά ονόματα (tag names). Η μηχανή προτύπων Jade χρησιμοποιείται σε συνδυασμό με το framework ExpressJs σε πλατφόρμες ανάπτυξης NodeJs.

Τα πλεονεκτήματα της μηχανής προτύπων Jade είναι η ταχεία ανάπτυξη που εξασφαλίζεται, η συνοχή του παραγόμενου κώδικα και τα κοινά πρότυπα συμμόρφωσης. Επιπλέον, μας προσφέρει τη δυνατότητα να δημιουργήσουμε αμέτρητο αριθμό σελίδων δυναμικά με ένα ενιαίο πρότυπο. Τέλος, όταν προκύψει η ανάγκη να πραγματοποιηθεί μία αλλαγή, μπορεί να αλλάξει ο κώδικας σε ένα μόνο σημείο και να εμφανισθεί σε όλες τις απαιτούμενες σελίδες. Παρακάτω φαίνεται ένα παράδειγμα γραφής σε μηχανή προτύπων Jade.

```
1 doctype html
2 html(lang="en")
3   head
4     title= pageTitle
5     script(type='text/javascript').
6       if (foo) {
7         bar(1 + 5)
8       }
9   body
10    h1 Jade – node template engine
11    #container.col
12      if youAreUsingJade
13        p You are amazing
14      else
15        p Get on it!
16    p.
17      Jade is a terse and simple
18      templating language with a
19      strong focus on performance
20      and powerful features.
```

#### 4.2.4 Mockups

#### 4.2.5 Διασύνδεση με Social Networks

### 4.3 Mobile

#### 4.3.1 Mobile OS

Στην παρούσα ενότητα θα πραγματοποιήσουμε μια σύντομη σύγκριση μεταξύ των λειτουργικών συστημάτων iOS και Android. Περιορίζουμε την ανάλυση μόνο στα δύο προαναφερθείσα λειτουργικά γιατί αφενός έχουν τον μεγαλύτερο αριθμό ενεργών προγραμματιστών και αφετέρου τον μεγαλύτερο αριθμό χρηστών, κάτι που τα καθιστά τα πλέον διαδεδομένα λειτουργικά στην αγορά των έξυπνων κινητών. Στον πίνακα 4.2 βλέπουμε μια στοιχειώδη σύγκριση μεταξύ των δύο λειτουργικών [88][9][13][8]. Θα πρέπει να σημειωθεί ότι οι τιμές και τα στοιχεία που αναφέρονται στον πίνακα 4.2 είναι αυτά που ισχύουν κατά τη συγγραφή της παρούσας διπλωματικής (Οκτώβριος 2015).

Λειτουργικό	Android	iOS
Γλώσσα Προγραμματισμού	Java	Swift
Περιβάλλον Ανάπτυξης	Android Studio	X-Code
Προσαρμοστικότητα	Μεγάλη	Ελάχιστη
Ανοιχτού Κώδικα	Ναι	Όχι
Μερίδιο Αγοράς	82.8%	13.9%
Κόστος δημοσίευσης εφαρμογής	25\$ εφάπαξ	99\$/έτος
Έλεγχος Εφαρμογών	Όχι	Ναι

Πίνακας 4.2: Σύγκριση βασικών στοιχείων Android & iOS.

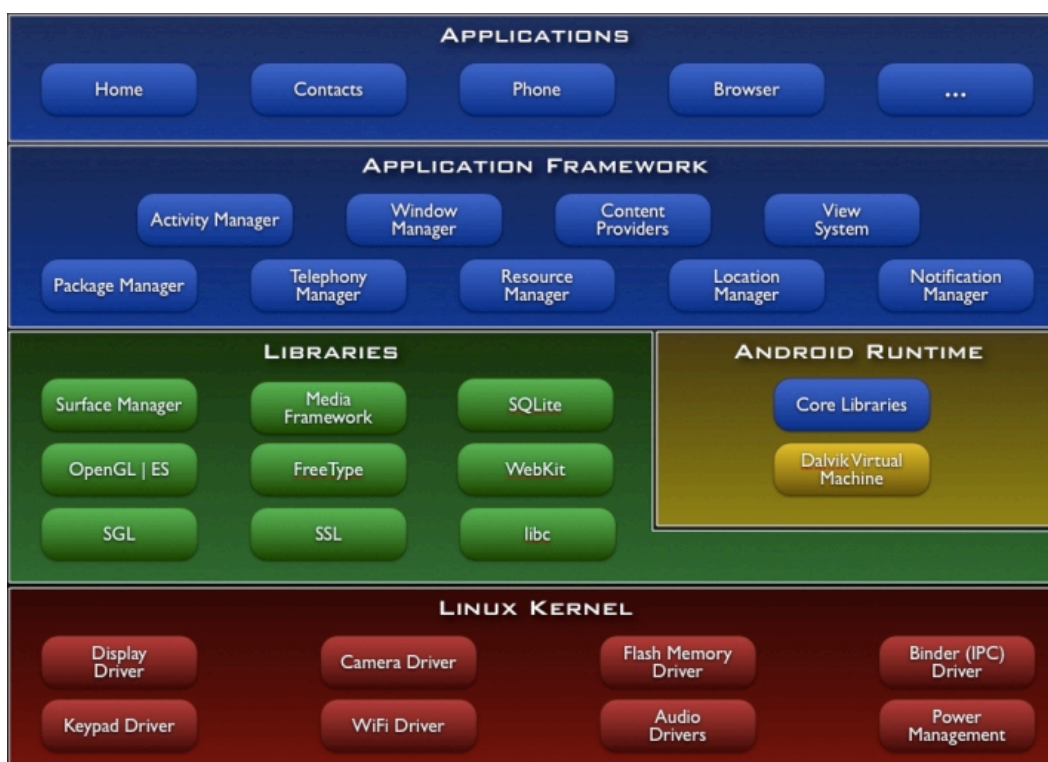
Σε αυτό το σημείο πρέπει να αναφέρουμε ότι το iOS Developer Toolset λειτουργεί μόνο στα πλαίσια του λειτουργικού Mac OS X. Επομένως, η ανάπτυξη μιας εγγενούς iOS εφαρμογής προϋποθέτει την ύπαρξη ενός Apple υπολογιστή. Δεδομένου του μεγαλύτερου μεριδίου αγοράς που διαθέτει το Android καθώς και τις ευκολίες που παρέχει προς τους προγραμματιστές, το επιλέξαμε για την πιλοτική έκδοση της εφαρμογής μας.

### 4.3.2 Android

Στην παρούσα ενότητα θα αναφερθούμε στα βασικά δομικά στοιχεία και στην αρχιτεκτονική του λειτουργικού συστήματος Android.

#### Αρχιτεκτονική του Android

Το λειτουργικό σύστημα Android αποτελεί ουσιαστικά μια στοίβα πρωτοκόλλων για κινητά τερματικά που χωρίζεται σε πέντε κατηγορίες και τέσσερα επίπεδα όπως φαίνεται στο σχήμα 4.19.



Σχήμα 4.19: Αρχιτεκτονική του λειτουργικού συστήματος Android

Ξεκινώντας από το χαμηλότερο επίπεδο έχουμε [37][12][30]:

- **Linux Kernel:** ο οποίος είναι ειδικά σχεδιασμένος για το λειτουργικό σύστημα Android και περιέχει τους οδηγούς για το υλικό (κάμερα, πληκτρολόγιο κ.τ.λ.). Επίσης ο πυρήνας είναι υπεύθυνος για τη διαχείριση της μνήμης και των διεργασιών, τις συνδέσεις δικτύου και κατ' επέκταση την διαχείριση των διεπαφών δικτύου που διαθέτει κάθε συσκευή. Επιπρόσθετα, χειρίζεται τους πόρους και την ενεργεία, κάτι το οποίο είναι πολύ σημαντικό δεδομένου της περιορισμένης παροχής ενέργειας των κινητών συσκευών.

- **Libraries** Στο αμέσως επόμενο επίπεδο έχουμε τις βιβλιοθήκες του συστήματος, οι οποίες πολλές φορές αποκαλούνται και εγγενείς βιβλιοθήκες (Native Libraries). Οι εν λόγω βιβλιοθήκες έχουν αναπτυχθεί με χρήση των γλωσσών προγραμματισμού C και C++. Μερικές από τις πιο σημαντικές βιβλιοθήκες αυτού του επιπέδου είναι:

- WebKit, για την υποστήριξη των φυλλομετρητών
- Media Framework, για την αναπαραγωγή αρχείων ήχου και βίντεο
- Surface Manager, για τη δημιουργία παραθύρων και ανανέωση της οθόνης
- SQLite, για τη διαχείριση των βάσεων δεδομένων
- OpenGL, για την απόδοση γραφικών
- Text, για τον χειρισμό και την εμφάνιση κειμένου

Σε αυτό το επίπεδο βρίσκεται επίσης και το κομμάτι της εκτέλεσης του Android (Android Runtime), το οποίο υποστηρίζει τη διαδικασία εγγραφής και εκτέλεσης των εφαρμογών. Αποτελείται από δυο βασικές συνιστώσες:

- Βασικές βιβλιοθήκες για τη διεπαφή των εφαρμογών Java με το περιβάλλον της συσκευής στην οποία εκτελούνται. Πιο αναλυτικά, οι εν λόγω βιβλιοθήκες είναι δυναμικές, οι οποίες εισάγονται στην αρχή κάθε κλάσης.
- Την εικονική μηχανή Dalvik (Dalvik Virtual Machine) η οποία είναι υπεύθυνη για τη δημιουργία των εκτελέσιμων αρχείων των εφαρμογών. Συγκεκριμένα ο πηγαίος κώδικας Java μετατρέπεται σε μορφή ενδιάμεσου κώδικα (bytecode) και στην συνέχεια μεταφράζεται σε Dalvik bytecode που αποθηκεύεται σε αρχεία της μορφής .dex. Η διαδικασία της μετατροπής των αρχείων κλάσεων java σε μορφή .dex γίνεται από το εργαλείο dx, ενώ παράλληλα γίνεται και βελτιστοποίηση της πλεονάζουσας πληροφορίας με αποτέλεσμα τα αρχεία .dex να είναι μικρότερα σε μέγεθος από τα αντίστοιχα αρχεία κλάσεων. Τέλος, το αρχείο .dex μαζί με τους πόρους της εφαρμογής μετατρέπονται σε αρχείο της μορφής .apk (Android Package). Το .apk αρχείο είναι αυτό που χρησιμοποιείται από τον χρήστη για την εγκατάσταση της εφαρμογής στην συσκευή του. Όταν ο χρήστης εκτελέσει την εφαρμογή, τότε η εφαρμογή αυτή θα εκτελεστεί στην εικονική μηχανή Dalvik αντί της JVM, καθώς το περιβάλλον εκτέλεσης των κινητών συσκευών διαθέτει περιορισμένους πόρους.

- **Application Framework** Στο τρίτο επίπεδο βρίσκεται το πλαίσιο λογισμικού των εφαρμογών, το οποίο ουσιαστικά είναι ένα σύνολο υπηρεσιών, οι οποίες δημιουργούν το περιβάλλον μέσα στο οποίο εκτελούνται οι εφαρμογές Android. Μερικές από τις πιο βασικές υπηρεσίες που περιλαμβάνονται στο επίπεδο του applicaiton framework είναι οι παρακάτω:

- Package Manager: Ο διαχειριστής πακέτων είναι το σύστημα μέσω του οποίου οι εφαρμογές μπορούν να βρουν πληροφορίες για τις υπόλοιπες εφαρμογές που είναι εγκατεστημένες στη συσκευή. Αποτελεί ουσιαστικά μια βάση δεδομένων όλων των εφαρμογών που είναι εγκατεστημένες στην συσκευή και επιτρέπει σε μια εφαρμογή να χρησιμοποιήσει μια άλλη και να μοιραστεί δεδομένα με αυτήν.
  - Activity Manager: Ο διαχειριστής δραστηριοτήτων διαχειρίζεται τον κύκλο ζωής των εφαρμογών και τη στοίβα δραστηριοτήτων (activity stack).
  - Content Providers: Οι πάροχοι περιεχομένου επιτρέπουν στις εφαρμογές να δημοσιεύουν και να μοιράζονται δεδομένα με άλλες εφαρμογές. Αποτελούν ουσιαστικά βάσεις δεδομένων οι οποίες δίνουν την δυνατότητα στις εφαρμογές να αποθηκεύσουν και να μοιραστούν δομημένες πληροφορίες. Παράδειγμα τέτοιων πληροφοριών αποτελούν οι επαφές της συσκευής.
  - View System: Περιλαμβάνει γραφικά στοιχεία τα οποία χρησιμοποιούνται για τη δημιουργία της διεπαφής χρήστη. Παράδειγμα γραφικών στοιχείων που περιλαμβάνει είναι κουμπιά, κουτιά κειμένων, πλαίσια κ.τ.λ.
  - Resource Manager: Ο διαχειριστής πόρων είναι υπεύθυνος για τη διαχείριση των πόρων που δεν αποτελούν κώδικα και κατ' επέκταση δεν μπορούν να μεταγλωττιστούν. Παράδειγμα τέτοιων στοιχείων αποτελούν τα γραφικά στοιχεία και οι συμβολοσειρές.
  - Notification Manager: Ο διαχειριστής ειδοποιήσεων επιτρέπει στις εφαρμογές να εμφανίζουν ειδοποιήσεις προς το χρήστη. Οι ειδοποιήσεις αυτές εμφανίζονται στη μπάρα κατάστασης της συσκευής, η οποία βρίσκεται στο πάνω μέρος της οθόνης και είναι σχεδόν πάντα ορατή στον χρήστη.
  - Telephony Manager: Ο διαχειριστής τηλεφωνίας παρέχει πληροφορίες στην εφαρμογή σχετικά με τις υπηρεσίες τηλεφώνου που είναι διαθέσιμες στην συσκευή.
  - Location Manager: Ο διαχειριστής τοποθεσίας παρέχει στις εφαρμογές πρόσβαση σε πληροφορίες σχετικά με την τοποθεσία και την κίνηση της συσκευής.
- **Applications** Στο τέταρτο και τελευταίο επίπεδο της αρχιτεκτονικής του λειτουργικού συστήματος Android βρίσκονται οι εφαρμογές. Σε αυτό το επίπεδο περιλαμβάνονται τόσο οι προεγκατεστημένες εφαρμογές όσο και οι εφαρμογές τρίτων.

## Δομικά στοιχεία του Android

Υπάρχουν τέσσερα βασικά δομικά στοιχεία στο λειτουργικό σύστημα Android (βλ. σχήμα 4.20). Κάθε δομικό στοιχείο επιτελεί ένα ξεχωριστό σκοπό και έχει ανεξάρτητο κύκλο ζωής. Κάθε δομικό στοιχείο αποτελεί ουσιαστικά ένα διαφορετικό σημείο, μέσω του οποίου το λειτουργικό σύστημα μπορεί να αποκτήσει πρόσβαση στην εφαρμογή. Στην συνέχεια θα κάνουμε μια συνοπτική παρουσίαση των βασικών αυτών δομικών στοιχείων, θεμελιώνοντας έτσι το θεωρητικό πλαίσιο για ανάπτυξη εφαρμογών Android [74][4].



Σχήμα 4.20: Δομικά στοιχεία του Android.

### Activities

Η κλάση Activity αποτελεί τη βασικότερη κλάση μιας εφαρμογής Android. Κάθε κλάση Activity αποτελεί μια ξεχωριστή γραφική διεπιφάνεια (παράθυρο/οθόνη) μέσω της οποίας ο χρήστης αλληλεπιδρά με την εφαρμογή. Συνήθως το παράθυρο αυτό καλύπτει όλη την οθόνη της συσκευής αλλά υπάρχουν και κάποιες εξεζητημένες περιπτώσεις στις οποίες καλύπτει μόνο ένα μέρος της οθόνης.

Κάθε εφαρμογή αποτελείται κατά κανόνα από πολλές Activities (οθόνες), οι οποίες είναι συνδεδεμένες μεταξύ τους, ενώ έχουν και την δυνατότητα ανταλλαγής πληροφοριών. Η οθόνη που παρουσιάζεται πρώτη στον χρήστη με το που πραγματοποιήσει εκκίνηση της εφαρμογής ονομάζεται 'κύρια' (main), μέσω της οποίας μπορεί να πλοηγηθεί και στις υπόλοιπες οθόνες. Εν γένει κάθε activity έχει τη δυνατότητα να πραγματοποιήσει εκκίνηση μια άλλης activity. Το σύστημα κάθε φορά που γίνεται



μετάβαση σε μια νέα activity εισχωρεί (push) την προηγούμενη σε μια στοίβα μορφής LIFO, που ονομάζεται back-stack, έτσι ώστε αν ο χρήστης πατήσει το κουμπί επιστροφής (back) να μπορεί με ένα pop στην στοίβα να επαναφέρει την αμέσως προηγούμενη activity. Δηλαδή αν ήμασταν αρχικά στην activity A μέσω της οποίας μεταβήκαμε στην B, η activity A βρίσκεται στην κορυφή της στοίβας οπότε με το πλήκτρο back οδηγούμαστε πάλι στην οθόνη που βρισκόμασταν προηγουμένως την A. Σε αυτό το σημείο θα πρέπει να αναφερθεί ότι αν πατηθεί το κουμπί back όταν ο χρήστης βρίσκεται στην αρχική οθόνη και η στοίβα είναι άδεια, πραγματοποιείται έξοδος από την εφαρμογή.

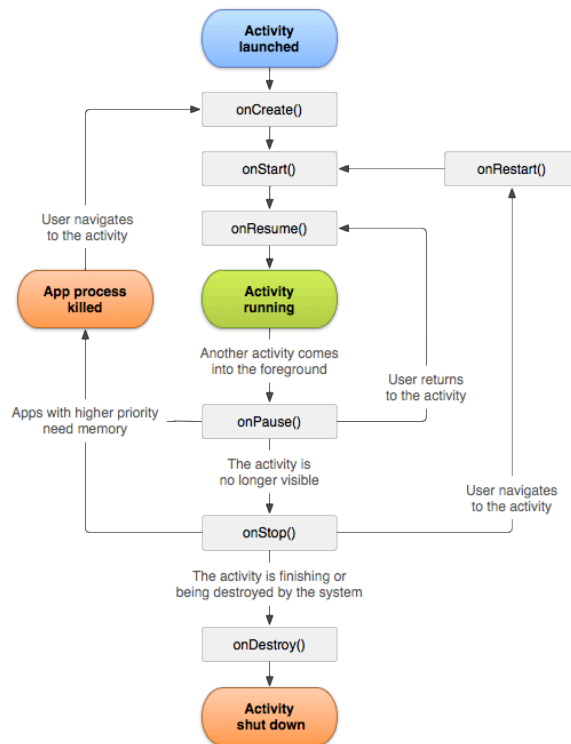
Για τη δημιουργία μιας οθόνης θα πρέπει να κάνουμε extend τη βασική κλάση Activity και να υλοποιήσουμε κάποιες απαραίτητες μεθόδους της. Συγκεκριμένα κάθε activity πρέπει να υλοποιεί τη μέθοδο onCreate(), η οποία καλείται αυτόματα κατά την πρώτη εκκίνηση της activity όπως υπονοεί και το όνομα της. Θα πρέπει να τονιστεί ότι εκτελείται μόνο κατά την πρώτη δημιουργία του activity και όχι όταν επιστρέφουμε σε αυτό με χρήση του κουμπιού της επιστροφής. Μέσα στην εν λόγω μέθοδο γίνεται αρχικοποίηση των βασικών συστατικών που αποτελούν τη διεπαφή του χρήστη.

Μια επίσης εκ των βασικών μεθόδων που πρέπει να υλοποιήσουμε αποτελεί το onPause(). Όπως υποδηλώνει και το όνομά της καλείται από το σύστημα όταν ο χρήστης πραγματοποιεί έξοδο από το συγκεκριμένο activity χωρίς όμως να το καταστρέφει, για παράδειγμα όταν κάνει μετάβαση σε άλλο activity της ίδιας εφαρμογής.

Για να πραγματοποιήσουμε τον τερματισμό ενός activity αρκεί να εκτελεστεί η εντολή finish(), αν και συνήθως δεν χρειάζεται να τερματίζουμε εμείς τις activities αφού είναι κάτι που αποτελεί ουσιαστικά δικαιοδοσία του Dalvik. Για να γίνουν καλύτερα κατανοητά τα παραπάνω στο σχήμα 4.21 βλέπουμε τον κύκλο ζωής των activities και τις αντίστοιχες μεθόδους.

## Services

Οι υπηρεσίες αποτελούν τα στοιχεία μιας εφαρμογής, οι οποίες εκτελούν χρονοβόρες λειτουργίες στο παρασκήνιο (background) της εφαρμογής. Θεμελιώδη διαφορά μεταξύ μιας υπηρεσίας και ενός activity αποτελεί το γεγονός ότι η υπηρεσία δεν παρέχει κάποια διεπαφή προς τον χρήστη της εφαρμογής, δεδομένου ότι όπως προαναφέρθηκε δρουν παρασκηνιακά. Μια υπηρεσία έχει την δυνατότητα να συνεχίζει την εκτέλεση της στο παρασκήνιο ακόμα και στην περίπτωση όπου ο χρήστης έχει πραγματοποιήσει μετάβαση σε κάποια τελείως διαφορετική εφαρμογή. Ένα κλασικό παράδειγμα υπηρεσίας αποτελεί η αναπαραγωγή μουσικής, η οποία συνεχίζει να εκτελείται στο παρασκήνιο ακόμα και όταν έχουμε κλείσει την εφαρμογή αναπαραγωγής μουσικής και εκτελούμε τελείως διαφορετικές εφαρμογές. Υπάρχουν κάποιες έτοιμες υπηρεσίες που μας παρέχει το λειτουργικό σύστημα όπως είναι ο

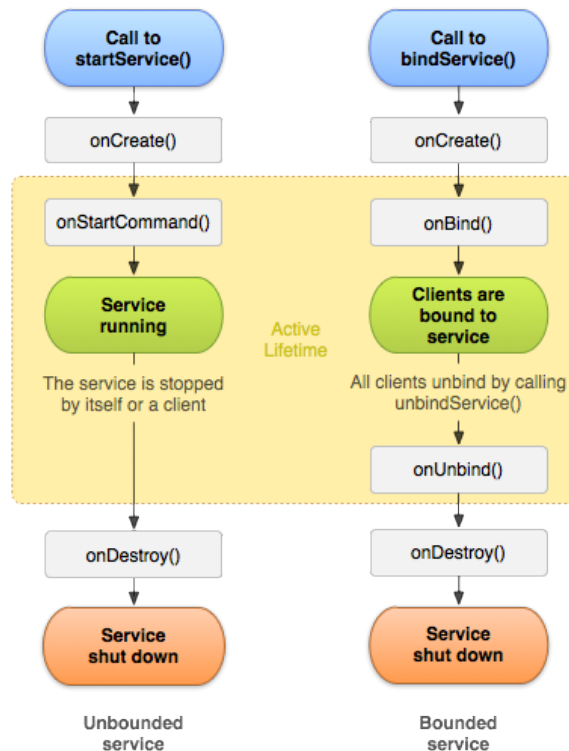


**Σχήμα 4.21:** Ο κύκλος ζωής των activities.

Location Manager, αλλά μπορούμε να ορίσουμε και τις δικές μας υπηρεσίες. Μια υπηρεσία μπορεί ουσιαστικά να πάρει δυο διαφορετικές μορφές[7]:

- **Started:** Μια υπηρεσία έχει εκκινηθεί όταν κάποιο συστατικό της εφαρμογής (π.χ ένα activity) έχει καλέσει τη μέθοδο `startService()`. Μόλις η υπηρεσία εκκινηθεί συνεχίζει να εκτελείται επ' αόριστον στο παρασκήνιο ακόμα και αν το activity που την κάλεσε έχει τερματιστεί. Συνήθως επιτελεί μία και μόνο εργασία χωρίς να επιστρέφει κάποιο αποτέλεσμα στο activity που την κάλεσε. Για παράδειγμα, μπορεί να κατεβάζει κάποιο αρχείο και όταν ολοκληρωθεί η μεταφόρτωση του θα τερματίσει η ίδια τον εαυτό της.
- **Bound:** Λέμε ότι μια υπηρεσία είναι “δεμένη” όταν κάποιο συστατικό της εφαρμογής (π.χ ένα activity) έχει καλέσει την μέθοδο `bindService()`. Μια ‘δεμένη’ υπηρεσία παρέχει διεπαφή πελάτη - εξυπηρετητή (client - server) και επιτρέπει στα συστατικά της εφαρμογής που είναι συνδεδεμένα μαζί της να αλληλεπιδρούν με αυτήν, στέλνοντας αιτήματα και λαμβάνοντας τα αποτελέσματα ακόμα και αν βρίσκονται σε διαφορετικές διεργασίες χρησιμοποιώντας IPC (interprocess communication). Υπηρεσίες τέτοιου τύπου συνεχίζουν τη λειτουργία τους όσο τα συστατικά με τα οποία είναι συνδεδεμένα εξακολουθούν να τρέχουν. Η υπηρεσία τερματίζεται μόνο όταν τερματιστεί (ή αποσυνδεθεί) και το τελευταίο συστατικό που είναι συνδεδεμένο με αυτήν.

Όπως αναφέρθηκε προηγουμένως οι υπηρεσίες δεν παρέχουν κάποια διεπαφή προς τον χρήστη, οπότε δημιουργείται και το εύλογο ερώτημα για το πως θα ενημερώνεται ο χρήστης. Οι υπηρεσίες ειδοποιούν τους χρήστες για τα αποτελέσματα τους είτε με χρήση ειδοποίησης μορφής Toast είτε μέσω ειδοποίησης στο status bar. Για να γίνουν καλύτερα κατανοητά οι έννοιες που αναλύθηκαν παραπάνω στο σχήμα 4.22 βλέπουμε τον κύκλο ζωής των υπηρεσιών.



Σχήμα 4.22: Ο κύκλος ζωής των services

## Content Providers

Οι πάροχοι περιεχομένου διαχειρίζονται την πρόσβαση στους αποθηκευτικούς χώρους των δεδομένων. Ουσιαστικά, δημιουργείται η δυνατότητα αποθήκευσης και διαμοιρασμού δεδομένων στις εφαρμογές. Ένας πάροχος δεδομένων παρουσιάζει τα δεδομένα στις εφαρμογές σε μορφή πινάκων σαν τους πίνακες σε μια σχεσιακή βάση δεδομένων. Παράδειγμα ενός ενσωματωμένου παροχέα δεδομένων είναι το λεξικό χρήστη στο οποίο αποθηκεύονται οι λέξεις του χρήστη. Για αυτή τη χρήση υπάρχει και η ενσωματωμένη βάση δεδομένων SQLite, η οποία δίνει την δυνατότητα αποθήκευσης και ανάγνωσης δεδομένων στους παρόχους περιεχομένου.[6]

## Broadcast Receivers

Οι δέκτες εκπεμπόμενων προθέσεων είναι υπεύθυνοι για τη λήψη και τη διαχείριση των γεγονότων που πραγματοποιούνται στην συσκευή. Συγκεκριμένα, ενημερώνονται από το λειτουργικό σύστημα όταν πραγματοποιείται κάποιο γεγονός και τότε ενεργοποιούνται. Χαρακτηριστικά παραδείγματα τέτοιων γεγονότων είναι όταν το κινητό μπαίνει σε κατάσταση αναμονής, όταν η στάθμη της μπαταρίας είναι χαμηλή κ.τ.λ. Οι δέκτες εκπεμπόμενων προθέσεων δεν παρέχουν κάποια διεπαφή προς τον χρήστη όπως και οι υπηρεσίες και μπορούν να επικοινωνήσουν με τον χρήστη μέσω ειδοποιήσεων στο status bar.[5]

## Κεφάλαιο 5

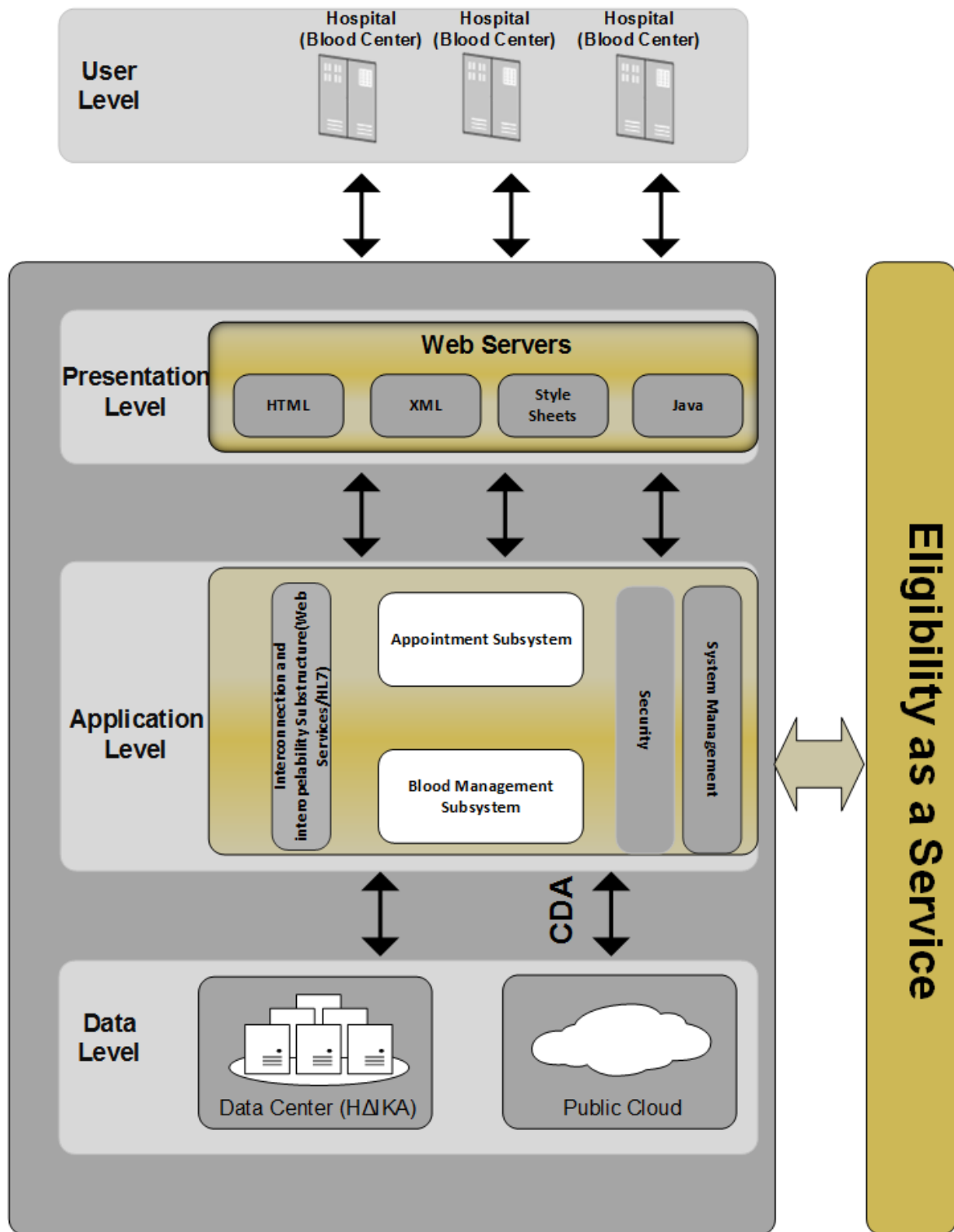
### Υλοποίηση

#### 5.1 Αρχιτεκτονική του συστήματος

Για την υλοποίηση του προτεινόμενου συστήματος αιμοδοσίας, χρησιμοποιείται η διαδομένη αρχιτεκτονική πολλαπλών επιπέδων (multitier architecture) όπως απεικονίζεται στο σχήμα 5.1. Συγκεκριμένα χρησιμοποιήθηκε αρχιτεκτονική τριών επιπέδων (3-tier).

Τα επίπεδα που χρησιμοποιούνται στο σχήμα είναι τα εξής:

- Επίπεδο Βάσης Δεδομένων (Database layer). Αποτελώντας το βασικότερο επίπεδο του συστήματος, στο επίπεδο αυτό τηρείται το σύνολο των επιχειρησιακών δεδομένων αλλά και των βοηθητικών αρχείων για την εύρυθμη λειτουργία των εφαρμογών του συστήματος.
- Επίπεδο Επιχειρησιακής Λογικής (Business logic layer). Αποτελεί το κύριο τμήμα του λογισμικού, στο οποίο εκτελούνται οι περισσότερες λειτουργίες, εκτός εκείνων που σχετίζονται με τη διαμόρφωση των οθονών εργασίας. Στο επίπεδο αυτό διαμορφώνεται η επιχειρησιακή λογική ώστε να παραμετροποιείται το σύστημα με βάση τις ανάγκες λειτουργίας του φορέα.
- Επίπεδο Παρουσίασης (Presentation layer). Το τρίτο επίπεδο του λογισμικού αποτελεί τη επαφή του χρήστη με το σύστημα (User Interface). Στο επίπεδο αυτό, πραγματοποιείται η διαχείριση των Οθονών Εργασίας (User Screens) καθώς επίσης και η μορφοποίηση των δεδομένων που εμφανίζονται. Το προτεινόμενο σύστημα βασίζεται στις τεχνολογίες WEB και στις τεχνολογίες παρουσίασης του Android.

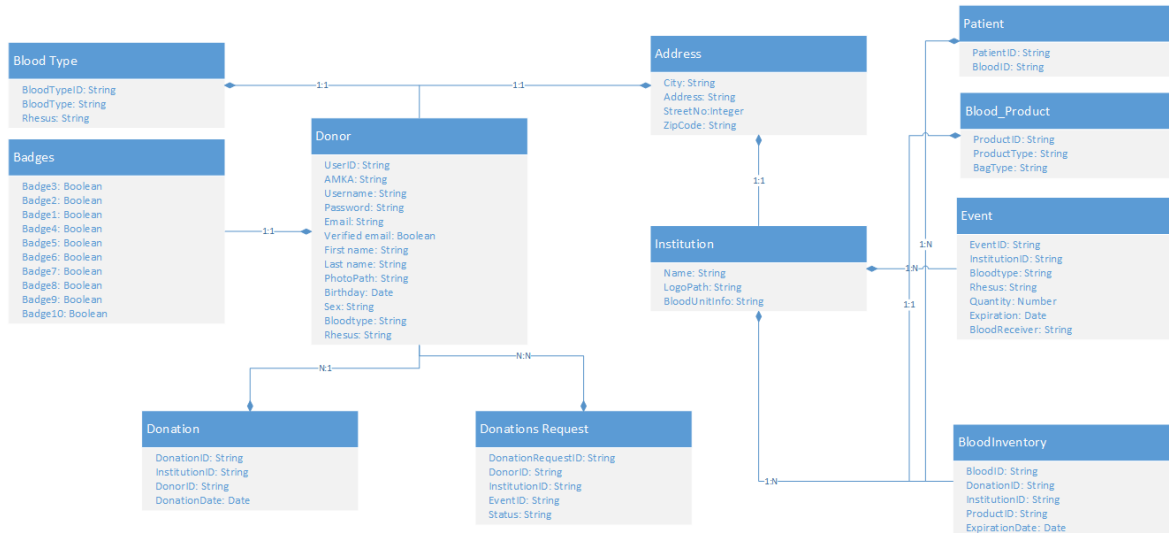


Σχήμα 5.1: Σχηματικό διάγραμμα του συστήματος.

## 5.2 Βάση δεδομένων

### 5.2.1 Σχήμα Βάσης

Στην ενότητα αυτή παρουσιάζεται το σχεσιακό διάγραμμα της βάσης NoSQL για το προτεινόμενο σύστημα ενώ παράλληλα αποτυπώνονται συνοπτικά οι προτεινόμενες δομές δεδομένων που υλοποιούν το προτεινόμενο σχήμα δεδομένων. Στο σχήμα 5.2 απεικονίζεται η βάση δεδομένων.



Σχήμα 5.2: Σχεσιακό διάγραμμα της βάσης του συστήματος.

Κεντρικοί πίνακες είναι ο πίνακας Donor και ο πίνακας Institution. Ο πίνακας Donor περιέχει τα στοιχεία του εθελοντή αιμοδότη τα οποία είναι τα εξής: AMKA, όνομα, επώνυμο, photopath (για αποθήκευση φωτογραφίας), ημερομηνία γέννησης, φύλο, userid, username, password, e-mail, ημερομηνία δημιουργίας λογαριασμού. Ο πίνακας Institution περιέχει τα στοιχεία του κέντρου αιμοδοσίας, τα οποία είναι: userID, username, password, email, ημερομηνία δημιουργίας λογαριασμού, logopath, λοιπά στοιχεία του ινστιτούτου. Υπάρχει επιπλέον ο πίνακας Address ο οποίος περιέχει τα στοιχεία: πόλη, διεύθυνση, νούμερο και ταχυδρομικός κώδικας και ενώνεται με σχέση 1-1 με το κέντρο αιμοδοσίας και με σχέση 1-1 με τον εθελοντή αιμοδότη και ο πίνακας BloodType ο οποίος αποθηκεύει ένα id για τον τύπο του αίματος, την ομάδα αίματος και συνδέεται με σχέση 1:1 με τον εθελοντή αιμοδότη.

Ο πίνακας Donation περιεχί στοιχεία για τις πραγματοποιηθείσες αιμοδοσίες, όπως: donationid, το id του κέντρου αιμοδοσίας στο οποίο έλαβαν χώρα, το id του εθελοντή και την ημερομηνία κατά την οποία πραγματοποιήθηκε η αιμοδοσία. Ο πίνακας Donation συνδέεται με σχέση N:1 με τον πίνακα Donor, καθώς ένας Donor μπορεί να έχει πραγματοποιήσει N αιμοδοσίες. Ο πίνακας DonationRequest

περιέχει τα αιτήματα αιμοδοσίας που στέλνονται στους χρήστες, με τα εξής στοιχεία: το id του εθελοντή, το id του κέντρου αιμοδοσίας, το id του γεγονότος αιμοδοσίας στο οποίο αναφέρεται και τέλος ένα πεδίο που λέγεται κατάσταση και περιγράφει την κατάσταση του αιτήματος (σε αναμονή, εγκεκριμένο, απορριφθέν). Ο πίνακας DonationRequest ενώνεται με σχέση N:N με τους εθελοντές, καθώς πολλά αιτήματα αιμοδοσίας αντιστοιχούν σε πολλούς αιμοδότες.

Στον πίνακα BloodProduct αποθηκεύονται στοιχεία για τα παράγωγα αίματος που προκύπτουν από κάποια αιμοδοσία, ενώ στον πίνακα Patient αποθηκεύονται στοιχεία για τον ασθενή στον οποίο καταλήγει το αίμα, και συσχετίζεται με σχέση 1-1 με τον πίνακα BloodInventory. Ο πίνακας BloodInventory αντιστοιχεί σε κάθε σακούλα αίματος που έχει περάσει εργαστηριακούς ελέγχους και έχει εγκριθεί και περιέχει στοιχεία: id του αίματος, id της αιμοδοσίας από την οποία προήλθε, το id του κέντρου αιμοδοτών, το id του αντικειμένου, ημερομηνία λήξης του αίματος. Τέλος, ο πίνακας badges περιέχει τις δέκα διαφορετικές κατηγορίες εμβλημάτων που μπορεί να κερδίσει ένας εθελοντής αιμοδότης και συνδέεται με σχέση 1:1 με τον εθελοντή αιμοδότη.

## 5.3 Web App

### 5.3.1 NodeJs - Server Establishment

Όπως αναφέρθηκε και στην ενότητα ?? χρησιμοποιήσαμε την πλατφόρμα NodeJs για να αναπτύξουμε την εφαρμογή μας. Για να δουλέψουμε πάνω στο NodeJS χρησιμοποιούμε το το Express framework, εγκαθιστώντας το πακέτο Express από τον Node Package Manager (NPM) και συνέχεια ενσωματώνουμε το πακέτο Express.

```
1 var express = require('express');
```

Αφού έχουμε εξάγει και ενσωματώσει το πακέτο Express δημιουργούμε ένα νέο Express αντικείμενο.

```
1 var app = express();
```

Το αντικείμενο "app" που έχουμε φτιάξει ουσιαστικά ξεκινάει τον server μας και εμπεριέχει μεθόδους για

- Να δρομολογεί HTTP αιτήσεις.



- Να ρυθμίζει το middleware.
- Να προβάλλει ένα view.
- Να αρχικοποιεί και να χρησιμοποιεί μία μηχανή δημιουργίας προτύπων.

### 5.3.2 Routing

Η δρομολόγηση αναφέρεται στον ορισμό των τελικών σημείων (URIs) σε μια εφαρμογή και πώς αυτή ανταποκρίνεται στα αιτήματα του πελάτη. Μια διαδρομή είναι ένας συνδυασμός ενός URI, μια μέθοδος αίτησης HTTP (GET, POST, κ.π.λ.), και έναν ή περισσότερους χειριστές για αυτό το τελικό σημείο. Στην υλοποίησή μας χρησιμοποιήσαμε ένα ξεχωριστό αρχείο για δρομολόγηση (routing) στο οποίο "ενώνεται" μία middleware συνάρτηση με κάθε μονοπάτι που χρησιμοποιεί η εφαρμογή μας. Δημιουργήσαμε ένα αντικείμενο Router που παρέχει το Express framework, το οποίο εκτελεί μόνο middleware και routing συναρτήσεις. Στο αρχείο αυτό γράψαμε τον απαραίτητο κώδικα για τις δρομολογήσεις. Ακολουθεί παράδειγμα από τον κώδικα δρομολόγησης.

```
1
2   var express = require('express');
3   var router = express.Router();
4
5   var login = require('../controllers/LoginController');
6
7
8   router.get('/', login.loginForm);
9   router.post('/login', login.loginAttempt);
```

### Μηχανή δημιουργίας προτύπων - Μοντέλο εφαρμογής

Έχουμε υλοποιήσει μοντέλο Model View Controller (MVC). Αυτό πρακτικά σημαίνει ότι έχουμε χωρίσει την εφαρμογή μας σε τρεις διαφορετικούς φακέλους ./Views, ./Models, ./Controllers. Κάθε οθόνη έχει το αντίστοιχο view και κάθε view ελέγχεται από τον αντίστοιχο controller. Η εφαρμογή μας χρησιμοποιεί την μηχανή δημιουργίας προτύπων Jade Template Engine. Ακολουθεί ο κώδικας αρχικοποίησης της Jade.

```
1
2 app.set('views', path.join(__dirname, 'views'));
3 app.set('view engine', 'jade');
```

Η μηχανή προτύπων Jade μας παρέχει την εξής λειτουργικότητα την οποία και αξιοποιήσαμε: Το Jade δίνει την δυνατότητα να δημιουργηθεί ένα βασικό View και στην συνέχεια κάνοντας το extend και συμπληρώνοντας κώδικα μόνο για τα επιπλέον τμήματα που θέλουμε να εμφανίσουμε - υλοποιήσουμε να αναπτυχθούν τα υπόλοιπα Views. Αυτό είχε ως αποτέλεσμα να αποφύγουμε την επαναληψιμότητα του κώδικα, να έχουμε καλύτερη δομή και οργάνωση και λιγότερο χρόνο ανάπτυξης της εφαρμογής. Οι οθόνες μας χρησιμοποιούν NavigationBar. Δημιουργήσαμε ένα βασικό NavigationBarView, το οποίο παρατίθεται στην συνέχεια και κάνοντας το extend αναπτύξαμε τις υπόλοιπες οθόνες.

```
1
2 doctype html
3 html(lang='en')
4   head
5     meta(charset='utf-8')
6     meta(http-equiv='X-UA-Compatible', content='IE=edge')
7     meta(name='viewport', content='width=device-width, initial-↵
8       scale=1')
9     meta(name='description', content='')
10    meta(name='author', content='')
11    link(href = "/stylesheets/bootstrap.min.css", rel = "stylesheet↵
12      ")
13    script(src='http://cdn.datatables.net/1.10.2/js/jquery.↵
14      dataTable.min.js')
15    link(href="http://cdn.datatables.net/1.10.2/css/jquery.↵
16      dataTable.min.css",rel="stylesheet")
17    link(href="http://cdnjs.cloudflare.com/ajax/libs/bootstrap↵
18      datepicker/1.3.0/css/datepicker.css",rel="stylesheet",type="↵
19      text/css")
20    // IE10 viewport hack for Surface/desktop Windows 8 bug
21    script(src="//ajax.googleapis.com/ajax/libs/jquery/2.1.4/jquery↵
22      .min.js")
23    script(src="/javascripts/bootstrap.js")
24    script(src="//www.parsecdn.com/js/parse-1.5.0.min.js")
```

```

18     script(src='http://cdnjs.cloudflare.com/ajax/libs/bootstrap↵
19         datepicker/1.3.0/js/bootstrap-datepicker.js')
20
21     body
22         .navbar.navbar-default.navbar-fixed-top(role='navigation')
23             .container
24                 .navbar-header
25                     button.navbar-toggle(type='button', data-toggle='↵
26                         collapse', data-target='.navbar-collapse')
27                     span.sr-only Toggle navigation
28                     span.icon-bar
29                     span.icon-bar
30                     span.icon-bar
31                     a.navbar-brand(href='#') LifeDonor
32                 .navbar-collapse.collapse
33                     #re.ul.nav.navbar-nav
34                     li
35                         a(href='homepage') Donation Events
36                     li
37                         a(href='donationrequests') Donation ↵
38                             Requests
39                     li
40                         a(href='create') Create a New Donation ↵
41                             Event
42                     li
43                         a(href='create') Complete a Donation
44                     li
45                         a(href='create') View Blood Supplies
46                     li
47                         a(href='create') Appointment Management
48
49                 li.dropdown
50                     a.dropdown-toggle(href='#', data-toggle='↵
51                         dropdown')
52                         | Dropdown
53                         span.caret
54                     ul.dropdown-menu(role='menu')
55                         li

```

```

52         a(href='#') Profile Management
53     li
54         a(href='#') Volunteer Informations
55
56     li.divider
57     li.dropdown-header Nav header
58     li
59         a(href='#') View Statistics
60
61     ul.nav.navbar-nav.navbar-right
62     li
63         a(href='../ navbar/') Default
64     li
65         a(href='../ navbar-static-top/') Static top
66     li.active
67         a(href='../') Fixed top
68     // /.nav-collapse
69 .container
70     .jumbotron
71     block content

```

### 5.3.3 Οθόνες

Για τις ανάγκες λειτουργίας του συστήματος, μελετήθηκαν προσεκτικά και αξιολογήθηκαν οι αναγκαίες βασικές καρτέλες και τα μενού τα οποία κατ'ελάχιστον θα πρέπει να διαθέτει το σύστημα ώστε να καλύπτει τις ανάγκες λειτουργίας του σε φορείς παροχής υπηρεσιών υγείας. Το σύστημα λοιπόν προτείνεται να απαρτίζεται από τις παρακάτω βασικές καρτέλες και μενού:

Στην συνέχεια παρουσιάζονται ενδεικτικά οι βασικές οθόνες λειτουργίας που διαθέτει το σύστημα μας για να καλύπτει τις λειτουργικές απαιτήσεις που αναφέρθηκαν στην ενότητα ???. Παρακάτω θα μελετήσουμε κάποιες ενδεικτικές οθόνες καθώς και τις λειτουργικότητες που παρέχουν, της διαδικτυακής εφαρμογής του υπολογιστικού νέφους από την πλευρά των εγκεκριμένων χρηστών/διαχειριστών των νοσοκομείων - κέντρων αιμοδοσίας.

## Register

Η οθόνη Register υλοποιείται με ένα View (RegisterView) και έναν Controller(RegisterController). Για την εμφάνιση του RegisterView χρησιμοποιείται ένα κατάλληλα προσαρμοσμένο css βασισμένο στο bootstrap framework. Αρχικά εμφανίζεται μία λίστα από όλα τα νοσοκομεία και τα κέντρα αιμοδοσίας, τα οποία κατά την φόρτωση της φόρμας ανακτώνται από την βάση δεδομένων. Ο RegisterViewController είναι ο χειριστής της όλης διαδικασίας. Μόλις λάβουμε ένα αίτημα GET :

```
1
2   var register = require( '../ controllers / RegisterController ' );
3
4   router . get( '/ register ' , register . registerForm );
```

μεταβαίνουμε στον RegisterController ο οποίος χειρίζεται το αίτημα. Με την κλήση κατάλληλης συνάρτησης της αντίστοιχης κλάσης του μοντέλου, λαμβάνει τα δεδομένα και στην συνέχεια καλεί το RegisterView , περνώντας τα ζητούμενα δεδομένα σαν όρισμα.

```
1
2       res . render( ' RegisterView ' , {
3           title : ' Register ' ,
4           data : results ,
5           results : resultsVol2
6       })
```

Για την παρουσίαση των δεδομένων στην λίστα καθώς και την επιστροφή της επιλογής του χρήστη χρησιμοποιήθηκε το framework AngularJS. Έπειτα θα πρέπει να συμπληρωθεί μία φόρμα με τα στοιχεία τού διαχειριστή του νοσοκομείου που εκπροσωπείται. Αυτά αφορούν στο όνομα, στο επώνυμο, στον αριθμό αστυνομικής ταυτότητας ή διαβατηρίου, στο σταθερό τηλέφωνο, στο κινητό τηλέφωνο και στο e-mail. Ο χρήστης δημιουργεί τα αναγνωριστικά εισόδου (όνομα χρήστη και κωδικό πρόσβασης) και επιλέγει εγγραφή. Τα στοιχεία που εισήγαγε ο χρήστης επιστρέφονται στον RegisterController, όπου καλείται η κατάλληλη συνάρτηση της αντίστοιχης κλάσης του μοντέλου για να εισαχθούν στην βάση δεδομένων.

## Login

Η οθόνη σύνδεσης του χρήστη στο σύστημα υλοποιείται με ένα View (LoginView) και έναν Controller(LoginController). Ο LoginViewController είναι ο χειριστής της όλης διαδικασίας. Μόλις λάβουμε ένα αίτημα GET :

```
1
2   var login = require( '../ controllers / LoginController ');
3
4   router . get( '/ login ', login . loginForm );
```

μεταβαίνουμε στον LoginController ο οποίος χειρίζεται το αίτημα. Ο LoginController καλεί το LoginView:

```
1
2       res . render( ' LoginView ', {
3           title : ' Welcome to DonorLife Login Page '
4       })
```

Ο χρήστης καλείται να συμπληρώσει το όνομα χρήστη και τον κωδικό πρόσβασης με τα οποία είναι εγγεγραμμένος στο σύστημα. Ο LoginController λαμβάνει τα στοιχεία και καλεί το authentication service. Αν επιτευχθεί η ταυτοποίηση των στοιχείων, μεταβαίνουμε στην αρχική σελίδα αλλιώς εμφανίζεται μήνυμα λάθους και ο χρήστης καλείται να ξαναπροσπαθήσει. Επίσης αξίζει να σημειωθεί ότι στην παρούσα οθόνη δίνεται η δυνατότητα επαναφοράς κωδικού.

## Home Page - Donation Events

Η οθόνη αυτή υλοποιείται με ένα View (HomePageView) και έναν Controller(HomeController). Ο HomeController είναι ο χειριστής της όλης διαδικασίας. Κατά την φόρτωση της σελίδας εμφανίζονται όλα τα γεγονότα αιμοδοσίας του νοσοκομείου. Στο πάνω μέρος εμφανίζεται η λίστα με όλες τις ενεργές αιμοδοσίες και από κάτω εμφανίζεται μία λίστα με το ιστορικό όλων των αιμοδοσιών που έχουν γίνει στο συγκεκριμένο ιατρικό κέντρο. Μόλις λάβουμε ένα αίτημα GET :

```
1
2   var homepage = require( '../ controllers / HomeController ');
```

```
3 |
4 | router.get( '/homepage', homepage.homePageForm );
```

μεταβαίνουμε στον RegisterController ο οποίος χειρίζεται το αίτημα. Με την κλήση κατάλληλης συνάρτησης της αντίστοιχης κλάσης του μοντέλου, λαμβάνει τα δεδομένα, και στην συνέχεια καλεί το HomePageView , περνώντας τα ζητούμενα δεδομένα σαν όρισμα.

```
1 |
2 |         res.render( 'HomePageView', {
3 |             title: 'Donation Events',
4 |             dataNow: results,
5 |             dataExpired: resultsVol2
6 |         })
```

Σε κάθε event υπάρχουν αναλυτικά οι πληροφορίες του, καθώς και:

- ένα button με την επιλογή "Επεξεργασία" το οποίο μας οδηγεί σε νέο παράθυρο στο οποίο γίνονται edit τα στοιχεία του γεγονότος αιμοδοσίας
- ένα button με την επιλογή "Αναλυτική Προβολή" μέσω του οποίου μεταβαίνουμε σε νέα σελίδα στην οποία βλέπουμε αναλυτικά ποιοι αιμοδότες έχουν δώσει αίμα και ποιοι έχουν κλείσει ραντεβού για να δώσουν.
- μία γραφική απεικόνιση του ποσοστού του αίματος που έχει δοθεί σε σχέση με το ζητούμενο (αν υπάρχει κάποια απαίτηση).

## Requests

Η οθόνη αυτή υλοποιείται με ένα View (RequestsView) και έναν Controller(RequestsController). Ο RequestsController είναι ο χειριστής της όλης διαδικασίας. Μόλις λάβουμε ένα αίτημα GET :

```
1 |
2 | var requests = require( '../controllers/RequestsController' );
3 |
4 | router.get( '/requests', requests.RequestsForm );
```

μεταβαίνουμε στον `RequestsController` ο οποίος χειρίζεται το αίτημα. Με την κλήση κατάλληλης συνάρτησης της αντίστοιχης κλάσης του μοντέλου, λαμβάνει τα δεδομένα, και στην συνέχεια καλεί το `RequestsView`, περνώντας τα ζητούμενα δεδομένα σαν όρισμα.

```
1
2     res.render('RequestsView', {
3         title: 'Donation Requests',
4         dataGiven: results,
5         dataAppoint: resultsVol2
6     })
```

Σε κάθε event υπάρχουν αναλυτικά οι πληροφορίες του, καθώς και:

- ένα button με την επιλογή "Επεξεργασία" το οποίο μας οδηγεί σε νέο παράθυρο στο οποίο γίνονται edit τα στοιχεία του γεγονότος αιμοδοσίας
- ένα button με την επιλογή "Αναλυτική Προβολή" μέσω του οποίου μεταβαίνουμε σε νέα σελίδα στην οποία βλέπουμε αναλυτικά ποιοι αιμοδότες έχουν δώσει αίμα και ποιοι έχουν κλείσει ραντεβού για να δώσουν.
- μία γραφική απεικόνιση του ποσοστού του αίματος που έχει δοθεί σε σχέση με το ζητούμενο (αν υπάρχει κάποια συγκεκριμένη απαίτηση αίματος).

## Donation Completion

Η οθόνη αυτή υλοποιείται με ένα View (`CompletionView`) και έναν Controller (`CompletionController`). Ο `CompletionController` είναι ο χειριστής της όλης διαδικασίας. Κατά την φόρτωση της σελίδας εμφανίζονται όλοι οι χρήστες οι οποίοι έχουν δώσει αίμα ή έχουν κλείσει ραντεβού για να δώσουν αίμα στο συγκεκριμένο γεγονός αιμοδοσίας. Στο πάνω μέρος εμφανίζεται η λίστα με τους εθελοντές που έχουν κλείσει ραντεβού να δώσουν αίμα και από κάτω εμφανίζεται μία λίστα με τα στοιχεία των εθελοντών που έχουν πραγματοποιήσει ήδη αιμοδοσία. Μόλις λάβουμε ένα αίτημα GET :

```
1
2     var completion = require('../controllers/CompletionController');
3     router.get('/completion', completion.CompletionForm);
```

μεταβαίνουμε στον `CompletionController` ο οποίος χειρίζεται το αίτημα. Με κλήση του `CompletionView`:



```
1
2     res.render( 'CompletionView', {
3         title: 'Complete a donation',
4     })
```

Ο χρήστης καλείται να συμπληρώσει το ΑΜΚΑ του εθελοντή και να κάνει έλεγχο καταλληλότητας. Το σύστημα ξεκινάει να τρέχει το Eligibility as a Service, και επιστρέφει απάντηση ναι ή όχι. Ο RequestsController ανάλογα με την απάντηση ενεργοποιεί ή όχι την δυνατότητα να ολοκληρωθεί αιμοδοσία από τον συγκεκριμένο εθελοντή.

### Create a New Donation Event

Η οθόνη Create a New Donation Event υλοποιείται με ένα View (CreateDonationView) και έναν Controller(CreateDonationController).Μόλις λάβουμε ένα αίτημα GET :

```
1
2     var createDonation = require( '../ controllers / ←
3         CreateDonationController ' );
4
5     router . get( '/ createDonation ' , createDonation . CreateDonationForm );
```

μεταβαίνουμε στον CreateDonationController ο οποίος χειρίζεται το αίτημα. Ο CreateDonationController καλεί το CreateDonationView:

```
1
2     res.render( 'CreateDonationView', {
3         title: 'Create a New Donation',
4     })
```

Στην συνέχεια θα πρέπει να συμπληρωθεί μία φόρμα με τα στοιχεία του γεγονότος αιμοδοσίας. Αυτά αφορούν στην ονομασία της, στην ποσότητα αίματος, στις συγκεκριμένες απαιτήσεις αν υπάρχουν (π.χ. συγκεκριμένη ομάδα αίματος), στο χρονικό διάστημα διεξαγωγής της αιμοδοσίας και στα διαθέσιμα ραντεβού. Τα στοιχεία που εισήγαγε ο χρήστης επιστρέφονται στον CreateDonationController, όπου καλείται η κατάλληλη συνάρτηση της αντίστοιχης κλάσης του μοντέλου για να εισαχθούν στην βάση δεδομένων. Στην συνέχεια τρέχει το Notification as a Service, το οποίο στέλνει τα κατάλληλα

push notifications στους εθελοντες αιμοδότες.

## Appointment Management

Η οθόνη αυτή υλοποιείται με ένα View (AppointmentView) και έναν Controller (AppointmentController). Κατά την φόρτωση της σελίδας εμφανίζονται όλα τα ραντεβού αιμοδοσίας του νοσοκομείου. Ο AppointmentController είναι ο χειριστής της όλης διαδικασίας. Μόλις λάβουμε ένα αίτημα GET :

```
1
2   var appointment = require( '../ controllers / AppointmentController ' );
3
4   router . get( '/ appointment ' , appointment . AppointmentForm );
```

μεταβαίνουμε στον AppointmentController ο οποίος χειρίζεται το αίτημα. Με την κλήση κατάλληλης συνάρτησης της αντίστοιχης κλάσης του μοντέλου, λαμβάνει τα δεδομένα, και στην συνέχεια καλεί το AppointmentView , περνώντας τα ζητούμενα δεδομένα σαν όρισμα.

```
1
2       res . render( ' AppointmentView ' , {
3           title : ' Appointment Management ' ,
4           data : results ,
5       })
```

Ο χρήστης μπορεί να επιλέξει ένα ραντεβού και να δει αναλυτικά περαιτέρω πληροφορίες, όπως το γεγονός της αιμοδοσίας και τα στοιχεία του εθελοντή αιμοδότη καθώς και:

- Έχει την επιλογή 'Ακύρωση', με την οποία ακυρώνει το ραντεβού, εισάγοντας την αιτιολογία για αυτή την ενέργεια ή οποία κοινοποιείται και στον εθελοντή αιμοδότη.
- Έχει την επιλογή 'Μετάθεση' με την οποία μεταθέτει χρονικά ένα ραντεβού, η οποία ολοκληρώνεται ύστερα από συναίνεση του εθελοντή αιμοδότη.

## Supplies Management

Η οθόνη αυτή υλοποιείται με ένα View (SuppliesView) και έναν Controller (SuppliesController). Κατά την φόρτωση της σελίδας εμφανίζεται ένας πίνακας ο οποίος περιέχει τις βασικές κατηγορίες

των παραγώγων του αίματος (ΣΕ, Αιμοπετάλια(κοινά), Αιμοπετάλια, FFP, Κοινό πλάσμα) και την ποσότητα που υπάρχει διαθέσιμη για την κάθε κατηγορία σε κάθε ομάδα αίματος (0+, 0-, AB+, AB-, A+,A-,B+,B-). Ο SuppliesController είναι ο χειριστής της όλης διαδικασίας. Μόλις λάβουμε ένα αίτημα GET :

```
1
2   var supplies = require( '../ controllers / SuppliesController ' );
3
4   router . get( '/ supplies ' , supplies . SuppliesForm ) ;
```

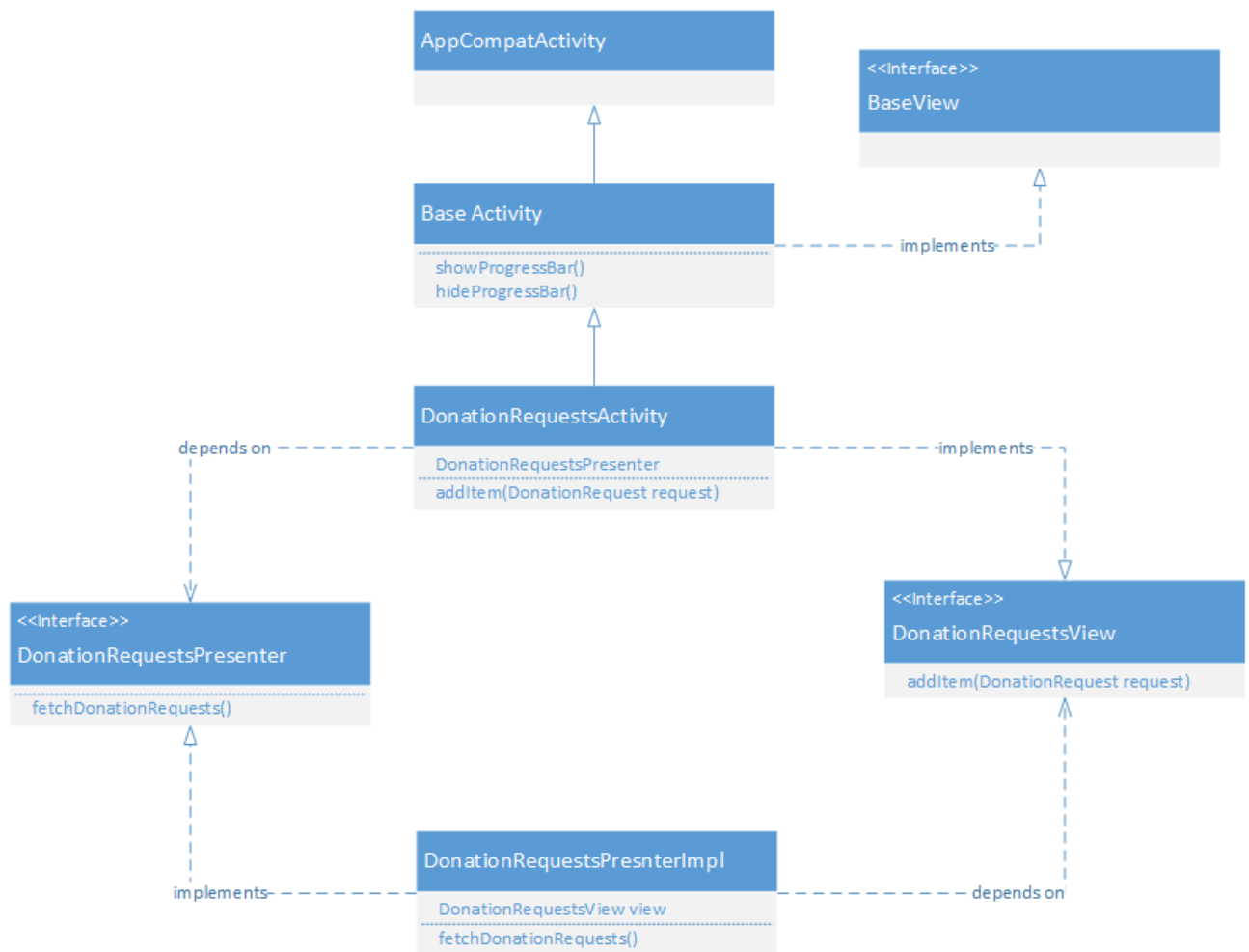
μεταβαίνουμε στον SuppliesController ο οποίος χειρίζεται το αίτημα. Με την κλήση κατάλληλης συνάρτησης της αντίστοιχης κλάσης του μοντέλου, λαμβάνει τα δεδομένα των διαθέσιμων ποσοτήτων, και στην συνέχεια καλεί το SuppliesView , περνώντας τα ζητούμενα δεδομένα σαν όρισμα.

```
1
2       res . render( ' SuppliesView ' , {
3           title : ' Supplies Management ' ,
4           data : results ,
5       })
```

## 5.4 Mobile App

Κατά την υλοποίηση της εφαρμογής για έξυπνα κινητά με λειτουργικό σύστημα android χρησιμοποιήθηκε το αρχιτεκτονικό πρότυπο MVP (βλ. 4.2.1). Κάθε Activity της εφαρμογής μας κληρονομεί από τη κλάση BaseActivity η οποία περιέχει τις βασικές κοινές λειτουργίες (για παράδειγμα εμφάνιση και απόκριση μπάρας φόρτωσης). Για να γίνει πιο εύκολα κατανοητή η αρχιτεκτονική της εφαρμογής μας, στο σχήμα 5.3 βλέπουμε ένα ενδεικτικό διάγραμμα κλάσεων.

Σε αυτό το σημείο πρέπει να σημειωθεί ότι για το dependency injection που φαίνεται στο σχήμα 5.3 έγινε χρήση του dependency injector dagger[150].



Σχήμα 5.3: Παράδειγμα διαγράμματος κλάσεων - MVP Android.

### Σύνδεση στην εφαρμογή - Login

Η οθόνη σύνδεσης στο σύστημα υλοποιείται μέσω ενός LoginActivity και ενός LoginPresenter ο οποίος το συνοδεύει. Όπως δείξαμε και στο σχήμα 5.3, ο LoginPresenter γίνεται inject στο LoginActivity και το LoginView (το οποίο υλοποιείται από το LoginActivity) γίνεται inject στον LoginPresenter αντίστοιχα. Μόλις ο χρήστης συμπληρώσει τα στοιχεία του και κάνει tap στο κουμπί σύνδεσης, το LoginActivity “περνάει” τα διαπιστευτήρια στον LoginPresenter ο οποίος αρχικώς τα ελέγχει για να εξακριβώσει την ορθότητά τους. Σε περίπτωση προβλήματος ως προς την ορθότητα ο LoginPresenter λέει στο LoginActivity τι μήνυμα λάθους θα πρέπει να δείξει στον χρήστη. Σε περίπτωση που δεν υπάρχει πρόβλημα ορθότητας, επικοινωνεί με το μοντέλο για να κάνει ταυτοποίηση του χρήστη. Τέλος ο LoginPresenter ενημερώνει το LoginActivity το οποίο με την σειρά του ενημερώνει τον χρήστη με κατάλληλο μήνυμα.

Παρακάτω βλέπουμε ενδεικτικά την κλήση του Presenter από το Activity:

```

1  protected void attemptLogin ()
2  {
3      resetErrors ();
4
5      // Store values at the time of the login attempt.
6      String email = mEmailView.getText (). toString (). trim ();
7      String password = mPasswordView.getText (). toString (). trim ();
8
9      loginPresenter . authenticateUser ( email , password );
10
11 }

```

Παρακάτω βλέπουμε ενδεικτικά πως ο Presenter λέει στο Activity να δείξει μήνυμα λάθους:

```

1  public void onUsernameError (String anError)
2  {
3      view . setUsernameError ( anError );
4      view . setUsernameFocus ();
5      view . hideProgressBar ();
6      cancel = true ;
7  }

```

#### 5.4.1 Εγγραφή στην εφαρμογή - Sign Up

Η οθόνη εγγραφής στο σύστημα υλοποιείται μέσω ενός RegisterActivity και ενός RegisterPresenter ο οποίος το συνοδεύει. Ο RegisterPresenter γίνεται inject στο RegisterActivity και το RegisterView (το οποίο υλοποιείται από το RegisterActivity) γίνεται inject στον RegisterPresenter αντίστοιχα. Μόλις ο χρήστης συμπληρώσει τα απαραίτητα για την εγγραφή του στοιχεία (ΑΜΚΑ) και κάνει tap στο κουμπί εγγραφής, το RegisterActivity καλεί τον RegisterPresenter ο οποίος αρχικώς τα ελέγχει για να εξακριβώσει την ορθότητά τους. Με βάση τα στοιχεία εγγραφής, ανοίγει ασφαλή διάυλο επικοινωνίας με τη βάση δεδομένων της ΗΔΙΚΑ και "τραβάει" τα στοιχεία του χρήστη και στην συνέχεια επικοινωνεί με το κατάλληλο model, το οποίο θα περάσει τα στοιχεία στην βάση δεδομένων στο υπολογιστικό νέφος (cloud computing). Τέλος, ο RegisterPresenter λέει στο RegisterActivity να εμφανίσει το κατάλληλο μήνυμα στον χρήστη.

Παρακάτω βλέπουμε ενδεικτικά την κλήση του Presenter από το Activity:

```

1
2     @OnClick(R.id.register_button)
3     protected void attemptRegister ()
4     {
5         resetErrors ();
6         String AMKA = mAMKAView.getText (). toString (). trim ();
7         registerPresenter . registerUser (AMKA);
8     }

```

Παρακάτω βλέπουμε ενδεικτικά πως ο Presenter λέει στο Activity να δείξει ότι είχαμε επιτυχία:

```

1
2
3     public void onSuccess ()
4     {
5         view . navigateToFirstSignUp ();
6     }

```

### 5.4.2 Δημιουργία νέου ραντεβού

Η οθόνη δημιουργίας ενός νέου ραντεβού στο σύστημα υλοποιείται μέσω ενός `CreateAppointmentActivity` και ενός `CreateAppointmentPresenter` ο οποίος το συνοδεύει. Ο `CreateAppointmentPresenter` γίνεται inject στο `CreateAppointmentActivity` και το `CreateAppointmentView` (το οποίο υλοποιείται από το `CreateAppointmentActivity`) γίνεται inject στον `CreateAppointmentPresenter` αντίστοιχα. Ο χρήστης βλέπει τα διαθέσιμα ραντεβού για ένα συγκεκριμένο γεγονός αιμοδοσίας και κάνοντας tap στην ώρα και ημέρα που επιθυμεί, του εμφανίζεται ένα μήνυμα επιβεβαίωσης της κράτησης του ραντεβού μέσω dialog fragment στο οποίο επιβεβαιώνει ή απορρίπτει την κράτηση του ραντεβού. Τα στοιχεία του ραντεβού μεταφέρονται από τον `CreateAppointmentActivity` στον `CreateAppointmentPresenter` ο οποίος αναλαμβάνει την κατοχύρωση του ραντεβού στο αιμοδοτικό κέντρο. Τέλος, ο `CreateAppointmentPresenter` λέει στο `CreateAppointmentActivity` να εμφανίσει το κατάλληλο μήνυμα στον χρήστη.

Παρακάτω βλέπουμε ενδεικτικά την κλήση του Presenter από το Activity:

```

1
2     @OnClick(R.id.create_appointment_button)

```

```

3     protected void attemptCreateAppointment()
4     {
5         resetErrors();
6         Date date = formatter.parse( date.getText().toString().trim() )↵
7         ;
8         Institution institution = Event.getInstitution();
9         createAppointmentPresenter.createAppointment( institution , date , ↵
        event);
    }

```

### 5.4.3 Εμφάνιση ιστορικού αιμοδοσιών

Η οθόνη εμφάνισης του ιστορικού αιμοδοσίας υλοποιείται μέσω ενός `DonationsHistoryActivity` και ενός `DonationsHistoryPresenter` ο οποίος το συνοδεύει. Ο χρήστης βλέπει μια λίστα από τις αιμοδοσίες τις οποίες έχει συμμετάσχει και έχει τη δυνατότητα με `tap` να δει επιπλέον πληροφορίες.

Παρακάτω βλέπουμε ενδεικτικά την κλήση του `Presenter` από το `Activity` καθώς και την συνάρτηση που καλείται όταν ο χρήστης κάνει `tap` σε κάποια αιμοδοσία:

```

1     donationRequestsPresenter.fetchDonationRequests(status);
2
3     list.setOnItemClickListener(new AdapterView.OnItemClickListener↵
4         ()
5         {
6             @Override
7             public void onItemClick(AdapterView<?> parent, View view, ↵
8                 int position, long id)
9             {
10                Log.d(TAG, "Donation Request tapped, let's see the ↵
11                    details");
12                DonationRequest donationRequest = adapter.getItem(↵
                    position);
                openDonationRequestDetailView(donationRequest);
            }
        });

```

## 5.4.4 Λήψη ειδοποίησης(push notification)

Ο χρήστης λαμβάνει την ειδοποίηση (push notification), την ανοίγει και ξεκινά το DonationRequestDetailsActivity. Παρακάτω βλέπουμε ενδεικτικά την συνάρτηση που παίρνει τα δεδομένα από το push notification για να τα εμφανίσει στο DonationRequestActivity:

```
1
2 public void updateView( final DonationRequest donationRequest)
3 {
4     if (donationRequest != null)
5     {
6         Event event = donationRequest.getEvent();
7         String fullAddress = event.getStreet() + " " + event.↵
            getStreetNo();
8         String fullBloodType = event.getBloodType() + event.↵
            getRhesus();
9         String fullQuantity = event.getQuantity() + "/" + event.↵
            getQuantityRequired();
10
11         venue.setText( event.getCity() );
12         date.setText(event.getDate());
13         address.setText(fullAddress);
14         bloodType.setText(fullBloodType);
15         quantity.setText(fullQuantity);
16     }
17 }
```

## 5.5 Testing and Tools

### 5.5.1 Android Application Testing

Στο παρόν κεφάλαιο θα αναφερθούμε αρχικά στο περιβάλλον ελέγχου που μας παρέχει το Android καθώς και στους διαφορετικού τύπους ελέγχου που μπορούμε να εκτελέσουμε. Στην συνέχεια θα αναφερθούμε στους ελέγχους που πραγματοποιήσαμε καθώς και στα αποτελέσματά τους.

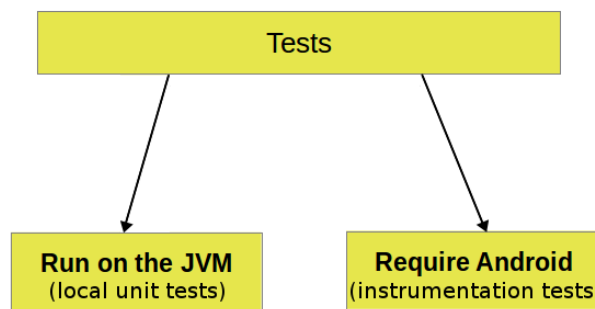
Η δυνατότητα αυτοματοποιημένου έλεγχου των εφαρμογών του λειτουργικού Android είναι ιδιαί-



τερα σημαντική λόγω του μεγάλου αριθμού διαφορετικών συσκευών που υπάρχουν στην αγορά. Είναι αδύνατον να δοκιμάσουμε για μία εφαρμογή όλες τις δυνατές ρυθμίσεις μιας συσκευής, για αυτό αποτελεί συνηθισμένη πρακτική η εκτέλεση δοκιμών με βάση τις πιο συνηθισμένες ρυθμίσεις. Στόχος είναι η επίτευξη ενός υψηλού ποσοστού κάλυψης δοκιμών (test coverage), έτσι ώστε να παράγουμε μια εφαρμογή απαλλαγμένη από σημαντικά σφάλματα αλλά και να διευκολύνουμε την συντήρηση και επέκτασή της στο μέλλον.

Οι δοκιμές στο λειτουργικό σύστημα Android βασίζονται στο JUnit, και μπορούν να χωριστούν σε δύο μεγάλες κατηγορίες όπως φαίνεται στο σχήμα 5.4 ανάλογα με το αν απαιτούν το λειτουργικό σύστημα android ή αν χρειάζονται μόνο την εικονική μηχανή της Java (JVM - Java Virtual Machine) κατά την διάρκεια της εκτέλεσής τους[10].

## Categories of Android tests



**Σχήμα 5.4:** Κατηγορίες δοκιμών στο Android

Εφόσον είναι εφικτό, είναι πάντα προτιμότερο να τρέχουμε τις δοκιμές στην εικονική μηχανή της Java, δεδομένου ότι απαιτεί πολύ λιγότερο χρόνο σε σχέση με το χρόνο που απαιτείται για την εγκατάσταση και εκτέλεση της εφαρμογής σε μία συσκευή Android. Στην συνέχεια της παρούσας ενότητας η εφαρμογή που δοκιμάζεται θα αποκαλείται *εφαρμογή υπό δοκιμή*.

### Unit Testing (Δοκιμή μονάδας)

Έλεγχοι τύπου unit testing χρησιμοποιούνται για την δοκιμή μικρών μεμονωμένων στοιχείων της εφαρμογής, χωρίς να εξετάζεται η αλληλεπίδραση τους με άλλα στοιχεία της εφαρμογής. Δηλαδή όπως δηλώνει και το όνομα τους, εξετάζουν τη συμπεριφορά ενός στοιχείου ως ξεχωριστή μονάδα. Για να γίνει καλύτερα αντιληπτό το παραπάνω, ας υποθέσουμε ότι θέλουμε να δοκιμάσουμε ένα κουμπί το οποίο αναμένουμε να μας μεταφέρει σε μια άλλη οθόνη. Στα πλαίσια της δοκιμής μονάδας θα

γίνει έλεγχος αν η πρόθεση (intent) για αλλαγή οθόνης δημιουργήθηκε σωστά και όχι αν μεταφερθήκαμε στην άλλη οθόνη. Σε περίπτωση όμως που εκτελούσαμε δοκιμές ενσωμάτωσης (instrumentation testing βλ. ενότητα 5.5.1) θα έπρεπε να γίνει έλεγχος ότι η επόμενη οθόνη ξεκίνησε όπως αναμέναμε.

Τα Unit tests χρησιμοποιούν το JUnit test framework και δεν πρέπει να χρησιμοποιούν λειτουργίες από το Android API έτσι ώστε να είναι σε θέση να τρέξουν στην εικονική μηχανή της Java (JVM) κάτι το οποίο απαιτεί πολύ λιγότερο χρόνο σε σχέση με τον χρόνο τον οποίο θα χρειάζονταν για να τρέξουν σε περιβάλλον Android. Κάθε εξάρτηση από το λειτουργικό Android θα πρέπει να αντικατασταθεί στον κώδικα που θέλουμε να δοκιμάσουμε με κάποιο mocking framework όπως είναι το Mockito.

Παρακάτω βλέπουμε ένα απλό JUnit test που φτιάξαμε με σκοπό να δοκιμάσουμε την λειτουργία της συνάρτησης που ελέγχει την εγκυρότητα του κωδικού χρήστη.

```
1 public class ValidatorTest
2 {
3     private final Validator validator = new Validator();
4
5     // Password Tests
6     @Test
7     public void password_validator_less_than_6 ()
8     {
9         String password = "12345";
10        boolean expected = false;
11
12        assertThat(validator.isPasswordValid(password), is(expected));
13
14    }
15
16    @Test
17    public void password_validator_more_than_6 ()
18    {
19        String password = "1234567";
20        boolean expected = true;
21
22        assertThat(validator.isPasswordValid(password), is(expected));
23
24    }
25 }
```

## Instrumentation Testing

Όταν θέλουμε να δοκιμάσουμε κλάσεις και λειτουργίες της εφαρμογής οι οποίες αλληλεπιδρούν με το λειτουργικό Android τότε δημιουργούμε τις λεγόμενες δοκιμές ενσωμάτωσης (instrumentation tests). Το Android API για δοκιμές παρέχει στον προγραμματιστή άγκιστρα (hooks) μέσω των οποίων μπορεί να αλληλεπιδράσει με τον κύκλο ζωής των στοιχείων λογισμικού και της εφαρμογής. Αυτά τα άγκιστρα απαρτίζουν το instrumentation API και επιτρέπουν στις δοκιμές να ελέγχουν γεγονότα από τον κύκλο ζωής (life cycle events) και την αλληλεπίδραση του χρήστη.

Υπό κανονικές συνθήκες ένα στοιχείο του Android ακολουθεί έναν κύκλο ζωής που έχει καθοριστεί από το σύστημα βάση της αλληλεπίδρασης του με τον χρήστη. Για παράδειγμα, όταν δημιουργείται μια πρόθεση (intent) για έναρξη κάποιου Activity, καλείται η μέθοδος onCreate() του εν λόγω Activity. Στην συνέχεια αν ο χρήστης ανοίξει κάποια άλλη εφαρμογή, καλείται η μέθοδος onDestroy() του Activity. Το instrumentation API αν και δεν επιτρέπει στον προγραμματιστή να καλέσει αυτές τις μεθόδους απευθείας, του επιτρέπει να εξομοιώσει πλήρως την συμπεριφορά ενός χρήστη η οποία προκαλεί τέτοια γεγονότα. Για παράδειγμα επιτρέπει στον προγραμματιστή να στείλει γεγονότα πληκτρών ή αφής, ελέγχοντας με αυτό τον τρόπο τον κύκλο ζωής της υπό δοκιμής εφαρμογής[23].

Τα instrumentation tests δεν γίνεται να τρέξουν στο JVM αλλά χρειάζονται περιβάλλον με λειτουργικό σύστημα Android. Μπορούν να τρέξουν είτε σε πραγματική συσκευή Android η οποία έχει συνδεθεί με τον υπολογιστή στον οποίο γίνεται η ανάπτυξη, είτε σε εικονική συσκευή Android. Δεδομένου του πλήθους των συσκευών που τρέχουν το λειτουργικό σύστημα Android, και συνυπολογίζοντας τις διάφορες εκδόσεις που μπορεί να τρέχει κάθε συσκευή, κάθε δοκιμή θα πρέπει να γίνεται με χρήση εξομοίωσης σε διάφορα επίπεδα API αλλά και μεγέθη οθονών. Για παράδειγμα αν η εφαρμογή υποστηρίζει επίπεδο API μεγαλύτερο του 14 θα πρέπει να γίνει έλεγχος για κάθε API μεγαλύτερο του 14 με τουλάχιστον 2 διαφορετικά μεγέθη οθόνης για κάθε API.

Όπως γίνεται εμφανές από τα παραπάνω το τρέξιμο των instrumentation tests αποτελεί μια ιδιαίτερα χρονοβόρα διαδικασία. Για την εξομάλυνση αυτού του προβλήματος υπάρχουν υπηρεσίες μέσω των οποίων οι δοκιμές μπορούν να γίνουν στο υπολογιστικό νέφος. Παρακάτω βλέπουμε ένα παράδειγμα δοκιμών instrumentation που φτιάξαμε για να ελέγξουμε την λειτουργία επαναφοράς κωδικού.

```
1 private static final String VALID_EMAIL = "name@email.com";
2 private static final String REAL_EMAIL = "i.fotopoulos42@gmail.com";
3
4 @Test
```

```

5  public void testResetPassword_noUserFound()
6  {
7      String no_user_found = "no user found with email "+VALID_EMAIL;
8
9      onView(withId(R.id.email)).perform(typeText(VALID_EMAIL), ←
      closeSoftKeyboard());
10     onView(withId(R.id.button_forgot_password)).perform(click());
11     onView(withText(no_user_found)).check(matches(isDisplayed()));
12 }
13
14 @Test
15 public void testResetPassword_success()
16 {
17     onView(withId(R.id.email)).perform(typeText(REAL_EMAIL), ←
      closeSoftKeyboard());
18     onView(withId(R.id.button_forgot_password)).perform(click());
19     onView(withText(res.getString(R.string.password_reset_success))←
      ).check(matches(isDisplayed()));
20 }

```

## 5.5.2 NodeJs Testing

Για τη συγγραφή και εκτέλεση τεστ για τον κώδικα σε JavaScript χρησιμοποιήσαμε τα εργαλεία Karma και Mocha. Το Mocha είναι μια βιβλιοθήκη που επιτρέπει τον ορισμό και την εκτέλεση δοκιμών σε ένα περιβάλλον Node.js χωρίς την ανάγκη ύπαρξης ενός φυλλομετρητή. Ορίζει περισσότερα από ένα στυλ συγγραφής τεστ και μας επιτρέπει να επιλέξουμε εκείνο που ταιριάζει περισσότερο στις ανάγκες του εκάστοτε τεστ. Στην περίπτωση που θέλουμε τα τεστ να εκτελούνται σε φυλλομετρητή το Mocha μας επιτρέπει να φορτώσουμε τα τεστ σε μία HTML σελίδα και να τα εκτελέσουμε στο φυλλομετρητή.

Το Chai είναι μία βιβλιοθήκη επικυρώσεων (assertion framework), η οποία συνεργάζεται άψογα με τη βιβλιοθήκη Mocha και δίνει τη δυνατότητα ορισμού των αναμενόμενων αποτελεσμάτων από κάποιο τεστ και ελέγχου κατά την εκτέλεσή του. Επίσης, όπως και το Mocha, μας επιτρέπει να διαλέξουμε από ένα σύνολο διαφορετικών στυλ για τον ορισμό των προσδοκιών. Παρατίθεται παράδειγμα τεστ γραμμένου με χρήση mocha και chai.

```
1 describe('JavaScript', function(){
2   it('should perform subtraction properly', function()
3     {
4       expect(2 - 0).to.equal(2);
5       expect(5 - 2).to.equal(3);
6       expect(7 - 3).not.to.equal(4);
7     });
8   });
```

Τέλος, το Karma είναι ένα εργαλείο το οποίο ενώνει τις δυνατότητες των ανωτέρω βιβλιοθηκών σε μία σελίδα HTML, εκτελεί όλα τα τεστ και επιστρέφει τα αποτελέσματα από την εκτέλεσή τους. Το Karma, βασίζεται σε ένα αρχείο ρυθμίσεων με βάση το οποίο γνωρίζει ποιες βιβλιοθήκες να χρησιμοποιήσει, που να βρει τα τεστ, σε τι πρόγραμμα περιήγησης να τα εκτελέσει, πώς να δημοσιεύσει τα αποτελέσματα και ενδεχομένως επιπλέον πληροφορίες σχετικές με την εκτέλεση των τεστ.



## Κεφάλαιο 6

### Διασυνδεσιμότητα

#### 6.1 Πρωτόκολλα επικοινωνίας

##### 6.1.1 Το πρότυπο Health Level Seven (HL7)

###### Ο Οργανισμός Health Level Seven (HL7)

Το πρότυπο HL7 έχει δημιουργηθεί από τον ομώνυμο οργανισμό Health Level Seven, ο οποίος αποτελεί μη κερδοσκοπικό οργανισμό με έτος ίδρυσης 1987 στις Ηνωμένες Πολιτείες της Αμερικής [141][21]. Ο οργανισμός HL7 συστάθηκε με στόχο τη δημιουργία παγκοσμίως αναγνωρισμένων και αξιόπιστων προτύπων για την ανταλλαγή ιατρικών δεδομένων μεταξύ ενδιαφερομένων φορέων στον τομέα της ιατρικής περίθαλψης. Τα δεδομένα αυτά αφορούν τόσο στην κλινική φροντίδα των ασθενών όσο και στο κομμάτι της διαχείρισης, οργάνωσης, αξιολόγησης και επέκτασης των υπηρεσιών Ιατρικής φροντίδας.

Ο μη κερδοσκοπικός οργανισμός Health Level Seven αποτελεί έναν από τους βασικότερους φορείς για τη διασφάλιση και διαλειτουργικότητα μεταξύ των συστημάτων παροχής υπηρεσιών Ηλεκτρονικής Ιατρικής Φροντίδας (e-Health). Αριθμεί πολυποίκιλα μέλη σε περισσότερες από 55 χώρες μεταξύ των οποίων δημόσιους και ιδιωτικούς φορείς Υγείας, εταιρείες παροχής Ιατρικού λογισμικού, εταιρείες παροχής Ιατρικού εξοπλισμού, ειδικούς συμβούλους, εμπειρογνώμονες, ασφαλιστικούς φορείς κ.τ.λ.[141]. Είναι δεδομένο άλλωστε ότι ο χώρος της Υγείας περιλαμβάνει αρκετούς ενδιαφερόμενους φορείς, καθιστώντας απαραίτητη τη χρήση προτύπων και αντίστοιχων αρχιτεκτονικών κατευθύνσεων ώστε να επιτευχθεί η διαλειτουργικότητα σε ένα ενοποιημένο σύστημα Ηλεκτρονικής Υγείας στο οποίο θα συμμετέχουν οι φορείς αυτοί, ο καθένας με το δικό του ρόλο. Επίσης αξίζει να σημειωθεί ότι έχει ο οργανισμός HL7 έχει τοπικά υποκαταστήματα σε όλες σχεδόν τις χώρες της Ευρώπης, στην Αυστραλία - Νέα Ζηλανδία, στην Ασία, στις Ηνωμένες Πολιτείες της Αμερικής και στη ζώνη

του Ειρηνικού. Στην Ελλάδα ιδρύθηκε και λειτουργεί από το 2003 το παράρτημα (μη κερδοσκοπικού χαρακτήρα) του διεθνούς οργανισμού Health Level Seven (HL7) με την επωνυμία "HL7 Hellas". Ο ιδρυτικός πυρήνας περιλαμβάνει δεκαπέντε (15) διακεκριμένα ονόματα φορέων τόσο από τον Πανεπιστημιακό όσο και από τον χώρο των εταιριών Ιατρικής Πληροφορικής και Τεχνολογίας[83].

## **Το πρότυπο HL7, έκδοση 2**

Το πιο ευρέως χρησιμοποιούμενο πρότυπο επικοινωνίας και διαλειτουργικότητας συστημάτων Ηλεκτρονικής Υγείας (healthcare interoperability standard) είναι η έκδοση 2 του προτύπου HL7 (HL7 v2). Χαρακτηριστικό παράδειγμα αποτελεί η περίπτωση των Ηνωμένων Πολιτειών της Αμερικής στις οποίες βρίσκει χρήση σε ποσοστό 95% των νοσοκομειακών συστημάτων[19].

Η έκδοση 2 του προτύπου HL7 εισήχθη τον Οκτώβριο του 1987, μόλις ένα χρόνο μετά την εμφάνιση της έκδοσης 1 και βρίσκεται υπό συνεχή ανάπτυξη εδώ και πάνω από 25 χρόνια. Τη στιγμή της συγγραφής της παρούσας διπλωματικής εργασίας, τελευταία έκδοση του HL7 v2 είναι η 2.7, η οποία έλαβε και έγκριση ως πρότυπο ANSI το έτος 2011[84]. Η έκδοση αυτή ορίζει διάφορους τύπους ηλεκτρονικών μηνυμάτων για την υποστήριξη διοικητικών, λογιστικών, οικονομικών, καθώς και ιατρικών διαδικασιών. Μία από τις βασικές απαιτήσεις κατά την εξέλιξη του HL7 v2 είναι η διασφάλιση ότι τα μηνύματα που έχουν δημιουργηθεί βάσει παλαιότερων εκδόσεων συνεχίζουν να είναι έγκυρα σε περιβάλλοντα που έχουν εφαρμόσει νεότερες εκδόσεις του προτύπου, οι οποίες έχουν σχεδιαστικά προσθετικά των προηγούμενων (backward compatibility). Για παράδειγμα μια εφαρμογή η οποία αποκωδικοποιεί μηνύματα της έκδοσης 2.5 πρέπει να είναι σε θέση να αποκωδικοποιήσει και μηνύματα της έκδοσης 2.4. Οι κύριοι λόγοι για τους οποίους οι παλαιότερες εκδόσεις του HL7 v2 χρησιμοποιούνται ακόμα σε τόσο μεγάλο βαθμό είναι οι παρακάτω:

1. Δεν υπάρχει ιδιαίτερο κίνητρο για αναβάθμιση δεδομένου ότι το όφελος από την επένδυση για αντικατάσταση της υπάρχουσας διεπαφής με κάποια καινούργια είναι αρκετά χαμηλό (low return on investment).
2. Δεδομένου της ευαίσθητης φύσης αλλά και της απαίτησης συνεχούς λειτουργίας (100% uptime) των συστημάτων υγείας, το ρίσκο αποτυχίας της αλλαγής λειτουργεί συνήθως ως αποτρεπτικός παράγοντας.

Αν και όπως προαναφέραμε το πρότυπο HL7 v2 χρήζει ευρείας αποδοχής δεν είναι χωρίς μειονεκτήματα, συγκεκριμένα[27]:



1. Δεν υπάρχει κάποιο σαφώς καθορισμένο μοντέλο αναφοράς της πληροφορίας που ανταλλάσσεται μέσω των μηνυμάτων αλλά ούτε και τρόπος να αναπαρασταθεί η σχέση μεταξύ των δεδομένων.
2. Χρησιμοποιεί μια εξαιρετικά ειδική σύνταξη στα μηνύματα με άμεσο αποτέλεσμα να είναι ιδιαίτερα δύσκολη η εκμάθηση αλλά και η υλοποίηση του προτύπου.
3. Αν και τα πολλά προαιρετικά χαρακτηριστικά που διαθέτει του παρέχουν μεγάλο βαθμό ευελιξίας, δημιουργούν προβλήματα ως προς τον έλεγχο της συμμόρφωσης προς το πρότυπο των διαφόρων υλοποιήσεων. Ως άμεσο αποτέλεσμα χρήζει μεγάλης προσπάθειας η εξασφάλιση ότι δύο εφαρμογές που θα επικοινωνήσουν μεταξύ τους, κάνουν χρήση των ίδιων χαρακτηριστικών.

Τα μηνύματα HL7 V2 αποτελούνται από μια ορισμένη σειρά τμημάτων (segments). Κάθε τμήμα περιέχει κάποια πεδία τα οποία είναι επίσης σε κάποια προκαθορισμένη σειρά. Κάθε πεδίο έχει προκαθορισμένους τύπους δεδομένων (data types), ο οποίος μπορεί να είναι οι ακόλουθοι[52]:

- Απλός τύπος μιας τιμής, παραδείγματος χάριν text ή date.
- Σύνθετος τύπος δεδομένων, ο οποίος αποτελείται από πολλαπλά επί μέρους συστατικά (components). Κάθε συστατικό έχει έναν δικό του προκαθορισμένο τύπο δεδομένων, ο οποίος με τη σειρά του μπορεί να είναι επίσης απλός ή σύνθετος τύπος δεδομένων. Καθεαυτόν τον τρόπο οδηγούμαστε στην δημιουργία υποσυστατικών (subcomponents).

Τα μηνύματα του προτύπου HL7 v2 μεταφέρονται με την πυροδότηση κάποιου γεγονότος. Κάθε μήνυμα αντιστοιχείται σε κάποια γενική κατηγορία, στην οποία ανήκει. Παραδείγματος χάριν, τα μηνύματα τα οποία έχουν σχέση με τη διαχείριση των ασθενών ανήκουν στην κατηγορία "ADT" (από τα αρχικά των λέξεων admission, discharge, transfer). Παρακάτω στον πίνακα 6.1 παρουσιάζονται κάποια παραδείγματα τύπων μηνυμάτων:

Τιμή	Περιγραφή
ACK	Μήνυμα επιβεβαίωσης (acknowledge)
ADT	Μήνυμα για τη διαχείριση των ασθενών
BTS	Μετάγγιση αίματος (Blood product transfusion)
PIN	Πληροφορίες ασφάλισης ασθενούς (Patient insurance information)

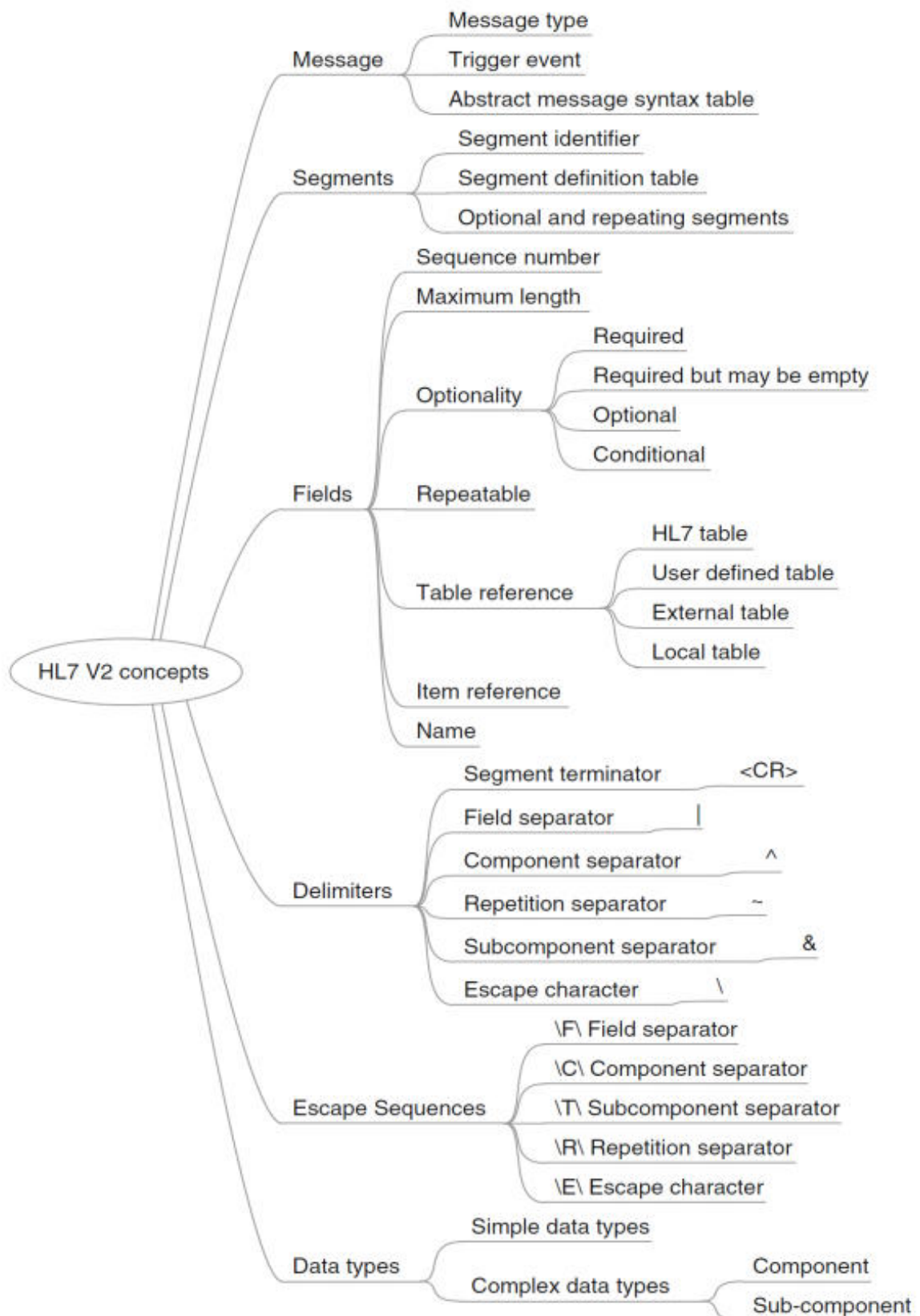
Πίνακας 6.1: Παραδείγματα τύπων μηνυμάτων HL7 V2

Όπως προαναφέρθηκε παραπάνω, πριν την δημιουργία κάποιου μηνύματος θα πρέπει να έχει προηγηθεί κάποιο γεγονός πυροδότησης. Κάθε γεγονός πυροδότησης ανήκει σε κάποιον τύπο μηνύματος. Για παράδειγμα στον τύπο μηνύματος ADT ανήκει το γεγονός A01, το οποίο αναφέρεται στην εισαγωγή ή επίσκεψη του ασθενή και το A02, το οποίο αναφέρεται στην μεταφορά του ασθενή. Για να επιτευχθεί ο πλήρης καθορισμός ενός μηνύματος θα πρέπει να γίνει συνδυασμός τόσο του τύπου μηνύματος όσο και του γεγονότος πυροδότησης. Βάση των παραδειγμάτων που έχουμε αναφέρει παραπάνω, ένα πλήρες όνομα μηνύματος θα ήταν το  $ADT^{A02}$ . Ο χαρακτήρας ”^” χρησιμοποιείται έτσι ώστε να διαχωρίσουμε τα διάφορα συστατικά ενός πεδίου. Δεδομένου της υψηλής σημασίας που παίζουν οι χαρακτήρες οριοθέτησης στην σωστή αποκωδικοποίηση και δημιουργία του μηνύματος, παρουσιάζουμε στον πίνακα 6.2 τους χαρακτήρες οριοθέτησης:

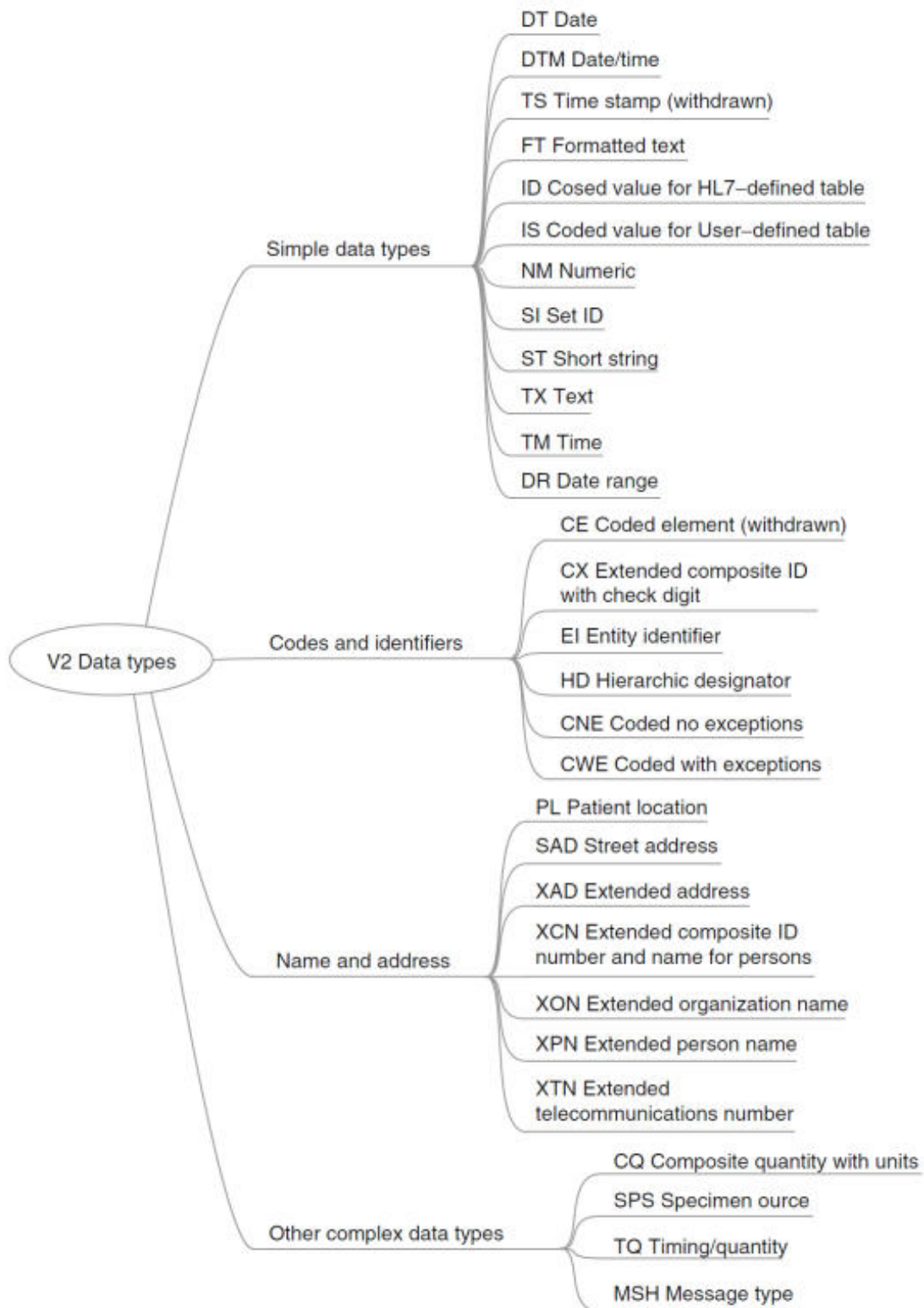
Χαρακτήρας	Χρήση
	Διαχωρισμός πεδίων
^	Διαχωρισμός συστατικών
~	Διαχωρισμός επαναλήψεων
\	Χαρακτήρας διαφυγής
&	Διαχωρισμός υποσυστατικών
< CR >	Τερματισμός τμημάτων

Πίνακας 6.2: Χαρακτήρες οριοθέτησης του προτύπου HL7 V2

Για να επιτευχθεί καλύτερη κατανόηση των παραπάνω κρίνεται σκόπιμο να συμπεριλάβουμε κάποια σχηματική αναπαράσταση. Στο σχήμα 6.1 βλέπουμε τα βασικά στοιχεία του προτύπου HL7 v2 ενώ στο σχήμα ?? βλέπουμε αναλυτικά τους τύπους δεδομένων που υποστηρίζει.



Σχήμα 6.1: Βασικές αρχές του προτύπου HL7 V2



**Σχήμα 6.2:** Τύποι δεδομένων του προτύπου HL7 V2

### Το πρότυπο HL7, έκδοση 3

Σκοπός της έκδοσης 3 του προτύπου HL7 είναι η υποστήριξη των ροών εργασίας σε όλο το φάσμα της Υγειονομικής Περίθαλψης. Η ανάπτυξη της ξεκίνησε το έτος 1995 και διήρκεσε 10 χρόνια, δεδομένου ότι μπήκε στην ‘αγορά’ εν έτη 2005. Υιοθέτησε μια διαφορετική προσέγγιση σε σχέση με τον προκάτοχο του, αφού βασίστηκε σε αντικειμενοστραφή λογική ενώ προσπάθησε παράλληλα και να περιορίσει την ευελιξία που υπήρχε στην έκδοση 2 με βασικό στόχο να εξαλείψει τα μειονεκτήματα τα οποία συνόδευαν την ευελιξία αυτή[85].

Η πιο βασική διαφοροποίηση της εν λόγω έκδοσης αποτελεί το μοντέλο RIM (Reference Information Model), το οποίο πηγάζει από την αντικειμενοστραφή λογική που έχει υιοθετηθεί[86]. Το μοντέλο RIM περιγράφει τις κλάσεις που ανήκουν στο RIM και μπορούν να αναπαραστήσουν το σύνολο των μηνυμάτων HL7 V3, τις σχέσεις μεταξύ τους, το μοντέλο καταστάσεων και τα πεδία της εκάστοτε κλάσης. Παρέχει μια ρητή αναπαράσταση των σημασιολογικών και λεκτικών συνδέσμων μεταξύ της πληροφορίας στα πεδία των μηνυμάτων του HL7. Σύμφωνα με τον οργανισμό HL7 το μοντέλο αυτό προσφέρει υψηλή ακρίβεια και βοηθάει σημαντικά στην μείωση του κόστους υλοποίησης. Τα βασικότερα πεδία που συναντάμε στις κλάσεις του RIM φαίνονται στον πίνακα 6.3.

Πεδίο	Λειτουργία
id	Ταυτοποίηση κάθε αντικειμένου (instance identifier)
classCode	Εξειδίκευση της κλάσης βάσει ορισμών του HL7 V3
code	Περιέχει το σχήμα κωδικοποίησης
statusCode	Δείχνει την τρέχουσα κατάσταση του αντικειμένου

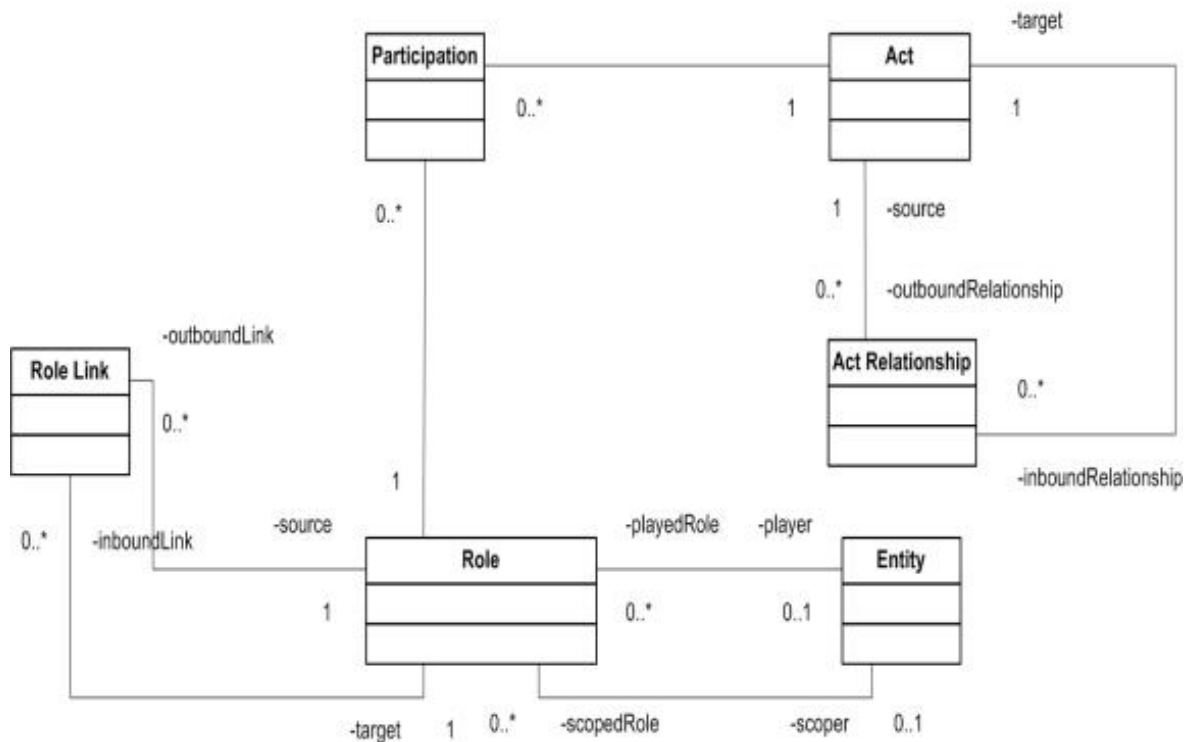
Πίνακας 6.3: Βασικά πεδία κλάσεων του μοντέλου RIM

Οι κύριες κλάσεις του μοντέλου RIM έχουν ως εξής[76]:

- **Act:** Κάθε μία πράξη ή ενέργεια που πραγματοποιείται είναι ένα Act (δηλαδή ανήκει στην κλάση act). Αποτελεί μια κλάση που περιλαμβάνει ένα τεράστιο εύρος ενεργειών, μερικά παραδείγματα είναι τα παρακάτω:
  - Παρατηρήσεις π.χ. ιατρικές διαγνώσεις, ευρήματα εξετάσεων
  - Γεγονότα π.χ. ραντεβού ασθενούς, εισαγωγή ασθενούς
- **ActRelationship:** Τα Acts συνδέονται μέσω της κλάσης ActRelationship.
- **Participation:** Ορίζει το περιβάλλον κάποιου Act (συγγραφέας, θέμα, τοποθεσία κ.τ.λ.)

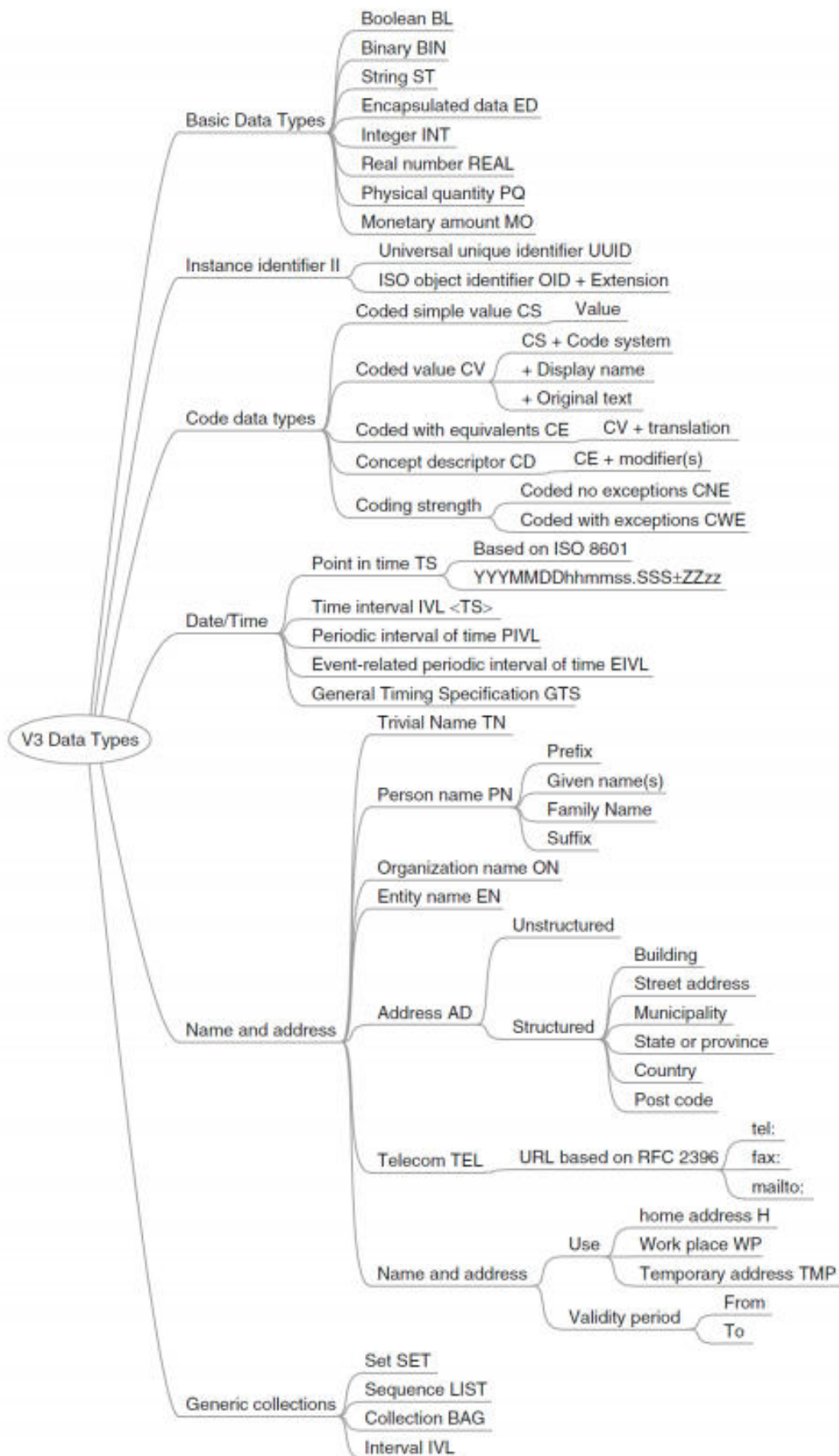
- **Entity:** Αποτελεί αναπαράσταση των φυσικών και μη οντοτήτων. Παραδείγματα οντοτήτων αποτελεί έναν ασθενής, ένας οργανισμός ή ακόμα και κάποιο νοσοκομείο.
- **Role:** Χρησιμοποιείται για να περιγράψει τους ρόλους που έχουν οι οντότητες σε κάποιο Act.
- **RoleLink:** Δείχνει τη σχέση που έχουν οι διάφοροι ρόλοι (Roles) μεταξύ τους.

Για να γίνουν πιο εύκολα κατανοητές οι παραπάνω κλάσεις καθώς και ο τρόπος με τον οποίο διασυνδέονται στο σχήμα 6.3 βλέπουμε το διάγραμμα κλάσεων[76].



**Σχήμα 6.3:** Διάγραμμα κλάσεων του μοντέλου RMI

Για λόγους πληρότητας στο σχήμα 6.4 βλέπουμε τους διάφορους τύπους δεδομένων που υποστηρίζονται από το πρωτόκολλο HL7 v3 (στο σχήμα 6.2 μπορείτε να δείτε τους τύπους για την έκδοση 2)



Σχήμα 6.4: Τύποι δεδομένων του προτύπου HL7 v3

## 6.1.2 CDA Documents

Το HL7 CDA (Clinical Document Architecture) αποτελεί ένα πρότυπο σήμανσης εγγράφου βασισμένο στο XML. Το CDA καθορίζει τη δομή και τη σημασιολογία των κλινικών εγγράφων με σκοπό την ανταλλαγή τους[26]. Ενώ το μοντέλο RIM του προτύπου HL7 V3 (βλ. 6.1.1) ορίζει τη δομή και την έννοια των επιμέρους συστατικών του εγγράφου προς ανταλλαγή. Ένα κλινικό έγγραφο είναι μια καταγραφή κλινικών εξετάσεων και υπηρεσιών με τα παρακάτω χαρακτηριστικά[26]:

- **Διάρκεια (Persistence):** ένα κλινικό έγγραφο “συνεχίζει να υπάρχει σε μια αναλλοίωτη κατάσταση για μία χρονική περίοδο που καθορίζεται από τοπικές και ρυθμιστικές ανάγκες”[52]. Οι πάροχοι υπηρεσιών υγείας υποχρεούνται βάση νόμων και κανονισμών να διατηρούν τα κλινικά έγγραφα για συγκεκριμένες χρονικές περιόδους. Για παράδειγμα κάποιο κλινικό έγγραφο μπορεί να χρειάζεται να διατηρείται για όλη την διάρκεια ζωής του ασθενούς και για κάποια επιπλέον χρόνια.
- **Υπευθυνότητα διαχείρισης (Stewardship):** ένα κλινικό έγγραφο διατηρείται από κάποιον οργανισμό στον οποίο και έχει ανατεθεί η φροντίδα του.
- **Δυνατότητα πιστοποίησης:** ένα κλινικό έγγραφο είναι μία συλλογή πληροφοριών που πρόκειται να επικυρωθεί νομικά.
- **Πλαίσιο (Context):** ένα κλινικό έγγραφο θέτει το προκαθορισμένο πλαίσιο για το περιεχόμενο του. Το πλαίσιο ενός κλινικού εγγράφου περιλαμβάνει:
  - Το αναγνωριστικό του εγγράφου,
  - ημερομηνίες και ώρες που σχετίζονται με το έγγραφο,
  - τον τύπου του εγγράφου,
  - τον ιδιοκτήτη του εγγράφου,
  - τον ασθενή (ή τους ασθενείς) στους οποίους αναφέρεται κ.λ.π.

Οι παραπάνω πληροφορίες είναι ενδεικτικές πληροφορίες πλαισίου που αποθηκεύονται στο header ενός CDA.

- **Αρτιότητα:** η επικύρωση ενός κλινικού εγγράφου εφαρμόζεται σε όλο και όχι σε τμήματα του εγγράφου χωρίς το πλήρες πλαίσιο του.
- **Αναγνωσιμότητα από ανθρώπους:** ένα κλινικό έγγραφο πρέπει να είναι αναγνώσιμο από ανθρώπους.



Ένα CDA έγγραφο είναι ένα καθορισμένο και πλήρες αντικείμενο πληροφοριών που μπορεί να περιέχει κείμενο, εικόνες, ήχους και άλλο περιεχόμενο πολυμέσων. Ένα έγγραφο CDA αποτελείται από δυο μέρη:

- **Επικεφαλίδα (header):** η οποία περιέχει ουσιαστικά το πλαίσιο (context) όπως περιγράφηκε παραπάνω.
- **Κυρίως σώμα (body):** περιέχει στοιχεία σχετικά με την εισαγωγή του ασθενή, τη διάγνωση, τα στοιχεία του ασθενή, τα φάρμακα κ.τ.λ. .

Παρακάτω βλέπουμε ένα παράδειγμα για το πεδίο author μιας επικεφαλίδας:

```
1 <author>
2   <time value="20000407"/>
3   <assignedAuthor>
4     <id extension="KP00017" root="2.16.840.1.113883.3.933"/>
5     <assignedPerson>
6       <name>
7         <given>Robert</ given>
8         <family>Dolin</ family>
9         <suffix>MD</ suffix>
10      </name>
11    </ assignedPerson>
12    <representedOrganization>
13      <id extension="M345" root="2.16.840.1.113883.3.933"/>
14    </ representedOrganization>
15  </ assignedAuthor>
16 </ author>
```

Παρακάτω βλέπουμε ένα παράδειγμα συστατικού (component) του κυρίου σώματος ενός εγγράφου CDA (ο κωδικός Loinc 10164-2 αναφέρεται στο ιστορικό της παρούσας ασθένειας του ασθενή):

```
1 <component>
2   <section>
3     <code code="10164-2" codeSystem="2.16.840.1.113883.6.1" ←
4       codeSystemName="LOINC"/>
5     <title>History of Present Illness</ title>
```

```
5 <text>
6   Henry Levin , the 7th is a 67 year old male referred for further ↔
   asthma management . Onset of asthma in his
7   <content revised="delete">twenties</content>
8   <content revised="insert">teens</content>
9   . He was hospitalized twice last year , and already twice this year .↔
   He has not been able to be weaned off steroids for the past ↔
   several months .
10 </text>
11 </section>
12 </component>
```

## 6.2 Δυνατότητες διασύνδεσης με άλλα υποσυστήματα

### 6.2.1 Ηλεκτρονική Συνταγογράφηση

Η διαδικασία της συνταγογράφησης φαρμάκων μπορεί να είναι ιδιαίτερα περίπλοκη και επιρρεπής σε λάθη, με δυσμενείς επιπτώσεις για την ασφάλεια των ασθενών. Η πρόληψη των επιπλοκών που μπορούν να προκύψουν λόγω ακατάλληλης συνταγογράφησης είναι μία πρόκληση που αντιμετωπίζει το σύστημα υγείας σε παγκόσμιο επίπεδο. Οι λανθασμένες φαρμακευτικές αγωγές, οι παρενέργειες των φαρμάκων και η αποτυχία σωστής συνταγογράφησης που να συντελέσει στην θεραπεία του ασθενούς εκτός του γεγονότος ότι αποτελεί απειλή για την ασφάλεια του, οδηγεί σε σημαντικές δαπάνες για τα συστήματα υγειονομική περίθαλψης. Οι πληροφορίες σχετικά με το ιστορικό του ασθενούς, τις αλλεργίες, τις παρελθοντικές και τις τρέχουσες φαρμακευτικές αγωγές καθώς και τα εργαστηριακά αποτελέσματα του είναι άκρως απαραίτητα ώστε οι κλινικοί ιατροί να συνταγογραφήσουν την κατάλληλη φαρμακευτική αγωγή, αλλά είναι συχνά δεν είναι διαθέσιμα. [48] Στην Ελλάδα ειδικότερα, υφίσταται το πρόβλημα της αλόγιστης ή πλασματικής συνταγογράφησης με αποτέλεσμα να χρειάζονται μέτρα για την επαναφορά της αξιοπιστίας της διαδικασίας. Χαρακτηριστικό παράδειγμα είναι τα στοιχεία του Υπουργείου Υγείας και Κοινωνικής Ασφάλισης, με βάση τα οποία το έτος 2009 εκτελεστήκαν στην Ελλάδα περίπου 100 εκατομμύρια ετησίως ενώ αντίστοιχα στην Δανία εκτελέστηκαν μόνο 15 εκατομμύρια συνταγές.

Με τον όρο ηλεκτρονική συνταγογράφηση (e-prescription) σύμφωνα με το Εθνικό Σύστημα Υγείας στην Αγγλία αναφερόμαστε στην αξιοποίηση των ηλεκτρονικών συστημάτων για την διευκόλυνση και την ενίσχυση της επικοινωνίας μίας ιατρικής εντολής ή συνταγής, βοηθώντας στην επιλογή, τη

διαχείριση και την προμήθεια ενός φαρμάκου μέσω της γνώσης και υποστήριξης αποφάσεων, παρέχοντας μια διαδρομή ελέγχου για το σύνολο της διαδικασίας χρήσης των φαρμάκων. Τα συστήματα ηλεκτρονικής συνταγογράφησης που προσφέρουν ηλεκτρονική υποστήριξη στους κλινικούς ιατρούς έχουν σχεδιαστεί για να βοηθήσουν στην περίπλοκη διαδικασία της συνταγογράφησης. [98] Επιτρέπουν σε έναν γιατρό, έναν φαρμακοποιό, μία νοσηλεύτρια να διαβιβάσει χωρίς σφάλματα, ακριβείς και κατανοητές συνταγές ηλεκτρονικά, μέσα από τον φορέα παροχής υπηρεσιών υγείας, στο φαρμακείο. [38]

Τα συστήματα ηλεκτρονικής συνταγογράφησης μπορεί να έχουν λειτουργικές δυνατότητες για την παροχή βασικής υποστήριξης αποφάσεων, όπως ο έλεγχος για τυχόν αλλεργίες στα φάρμακα, βασικές κατευθυντήριες γραμμές για τη δοσολογία, τεστ για πιθανές αντιδράσεις μεταξύ φαρμάκων κ.λ.π. Τα συστήματα αυτά είναι ιδιαίτερα χρήσιμα στο τεχνικό κομμάτι της συνταγογράφησης κατάλληλων φαρμάκων, καθώς προσφέρουν λειτουργίες όπως ο υπολογισμός της σωστής δόσης ή ο εντοπισμός αλληλεπιδράσεων μεταξύ των φαρμάκων. [96]

Τα συστήματα ηλεκτρονικής συνταγογράφησης χωρίζονται σε δύο κατηγορίες:

- Τα αυτόνομα συστήματα (stand alone systems). Πρόκειται για λειτουργικά συστήματα τα οποία είναι εγκατεστημένα στους ηλεκτρονικούς υπολογιστές και χρησιμοποιούνται είτε αυτόνομα είτε μέσω σύνδεσης στο διαδίκτυο. Τα συστήματα αυτά χρησιμοποιούνται κυρίως για τον έλεγχο θεμάτων ασφάλειας και γίνεται προσπάθεια αντικατάστασης τους από τα ολοκληρωμένα συστήματα συνταγογράφησης.
- Τα ολοκληρωμένα συστήματα συνταγογράφησης (Electronic Health Record, EHR Systems). Στα συστήματα αυτά ο ιατρός έχει στην διάθεση του όλο το ιστορικό του ασθενούς, τα αποτελέσματα των εξετάσεων του και τα χρησιμοποιεί για να βοηθηθεί στην επιλογή της κατάλληλης δραστικής ουσίας. Οι συναγερμοί ασφαλείας στην περίπτωση αυτή είναι πιο εξειδικευμένοι και ακριβείς.

Συνοπτικά τα συστατικά της ηλεκτρονικής συνταγογράφησης μπορούν να κατηγοριοποιηθούν σε βασικές δυνατότητες συνταγογράφησης, πληροφορίες για το πλάνο υγείας και τις κλινικές ειδοποιήσεις (clinical alerts). Οι δυνατότητες συνταγογράφησης περιλαμβάνουν μια λίστα φαρμάκων, οδηγίες για τους ασθενείς, τον αριθμό των εγκεκριμένων ποσοτήτων, σχόλια του γιατρού που συνταγογραφεί προς τον φαρμακοποιό, καθώς και το πεδίο PRN. Οι πληροφορίες για το πλάνο υγείας του ασθενούς περιλαμβάνουν την ιατρική ασφάλεια που έχει και το ιστορικό των φαρμακευτικών αγωγών. Τέλος, οι κλινικές ειδοποιήσεις βασίζονται στα δημογραφικά στοιχεία και στα στοιχεία του ιατρικού ιστορικού

του ασθενούς και περιλαμβάνουν τις αντιδράσεις μεταξύ των φαρμάκων, τις αλλεργίες, τις προειδοποιήσεις για συγκεκριμένες ηλικιακές ομάδες και την κατάλληλη προσαρμογή της δόσης με βάση το βάρος του ασθενούς. [132]

Τα βασικά συστατικά ενός συστήματος ηλεκτρονικής συνταγογράφησης είναι [72]:

- Ο παραπέμπων ιατρός. Ο θεράπων ιατρός είναι ο κύριος χρήστης του συστήματος. Για την σύνδεση του στο σύστημα ακολουθείται μια διαδικασία επαλήθευσης ώστε να επιβεβαιωθεί η ταυτότητά του. Οι αναζητήσεις του θεράποντα ιατρού στην βάση δεδομένων που περιέχει τους φακέλους των ασθενών πραγματοποιείται με τη χρήση ειδικών πληροφοριών για τον ασθενή, συνήθως το ΑΜΚΑ για Ελλάδα ή τον αριθμό κοινωνικής ασφάλισης (social security number) στον εξωτερικό. Μόλις προσπελαστεί το σωστό αρχείο του ασθενούς, ο ιατρός εξετάζει τις ιατρικές πληροφορίες που περιέχει και προσθέτει ή ενημερώνει μία συνταγή στον ιατρικό φάκελο.
- Ο κόμβος συναλλαγών. Αποτελεί τον κοινό σύνδεσμο μεταξύ όλων των φορέων (παραπέμπων ιατρός και φαρμακείο). Αποθηκεύει και διατηρεί ένα κύριο κατάλογο των ασθενών για να υπάρχει η δυνατότητα γρήγορης πρόσβασης στις ιατρικές πληροφορίες των ασθενών, καθώς και στην λίστα των φαρμακείων. Όταν ο παραπέμπων ιατρός ανεβάσει κάποια νέα συνταγή στον φάκελο του ασθενούς, τότε αυτή μεταφέρεται αυτόματα και στον κόμβο συναλλαγών. Ο κόμβος συναλλαγών θα στείλει αυτόματα τις πληροφορίες με στο κεντρικό σύστημα διαχείρισης το οποίο θα απαντήσει με πληροφορίες σχετικά με την καταλληλότητα του ασθενή και το ιστορικό των φαρμακευτικών αγωγών του. Ο κόμβος συναλλαγών έπειτα στέλνει τις πληροφορίες στον ιατρό έτσι ώστε να έχει τις απαραίτητες πληροφορίες που του χρειάζονται και να ολοκληρώσει και να εγκρίνει την συνταγή.
- Το κεντρικό σύστημα διαχείρισης δεδομένων, στο οποίο βρίσκονται όλα τα στοιχεία ιατρικού φακέλου που ελέγχει την καταλληλότητα του ασθενή καθώς και της συνταγής. Το κεντρικό σύστημα διαχείρισης δεδομένων επικοινωνεί με τον ιατρό και το φαρμακείο.
- Το φαρμακείο που έχει εγκατεστημένο το λογισμικό της ηλεκτρονικής συνταγογράφησης. Το φαρμακείο ανακτά την συνταγή από το κεντρικό σύστημα διαχείρισης συναλλαγών μέσω του κόμβου συναλλαγών και έχει επιπλέον την ικανότητα να επικοινωνήσει με τον ιατρό και να τον ενημερώσει ότι η παραγγελία εκτελέστηκε.

## Ηλεκτρονική Συνταγογράφηση στην Ελλάδα

Η ηλεκτρονική συνταγογράφηση στην Ελλάδα βασίζεται στις οδηγίες του νόμου 3892/2010 (ΦΕΚ 189 Α) “Ηλεκτρονική καταχώριση και εκτέλεση ιατρικών συνταγών και παραπεμπτικών ιατρικών εξετάσεων”. Σε αυτόν το νόμο καταγράφονται τα ζητήματα ηλεκτρονικής συνταγογράφησης και τα ζητήματα των ηλεκτρονικών παραπεμπτικών.

Ο υπεύθυνος φορέας στην Ελλάδα είναι η Γενική Γραμματεία Κοινωνικών Ασφαλίσεων του Υπουργείου Εργασίας, Κοινωνικής Ασφάλισης και Πρόνοιας. Το σύστημα υποστηρίζεται ηλεκτρονικά από την Ηλεκτρονική Διακυβέρνηση Κοινωνικής Ασφάλισης (ΗΔΙΚΑ Α.Ε.). Η ΗΔΙΚΑ αποτελεί μια ανώνυμη, μη κερδοσκοπικού χαρακτήρα του δημοσίου την οποία αποζημιώνουν οι εξυπηρετούμενοι φορείς για τις υπηρεσίες του παρέχει, όπως για παράδειγμα το έργο μισθοδοσίας των φορέων[89].

Με τον όρο ηλεκτρονική συνταγογράφηση αναφερόμαστε στη παραγωγή, στην διακίνηση και στον έλεγχο των ιατρικών συνταγών και των παραπεμπτικών για ιατρικές πράξεις, με τη χρήση τεχνολογίας ηλεκτρονικών υπολογιστών και επικοινωνιών, έτσι ώστε να διασφαλίζεται η ασφάλεια, η εγκυρότητα και η διαφάνεια στις πληροφορίες που διακινούνται. Ουσιαστικά, το σύστημα περιλαμβάνει ένα αριθμό από διαδικασίες που σχετίζονται με τη δημιουργία, την εκτέλεση, τη διαχείριση, τον έλεγχο, την εκκαθάριση και την εξόφληση μιας φαρμακευτικής συνταγής ή ενός παραπεμπτικού για ιατρικές πράξεις. Διέπει όλα τα συστήματα και τις τοποθεσίες που εμπλέκονται όπως τα τακτικά ιατρεία των νοσοκομείων ή τα ιδιωτικά ιατρεία, τα κέντρα υγείας, τις κλινικές, τα διαγνωστικά κέντρα, τα φαρμακεία και τα ασφαλιστικά ταμεία.

Το ολοκληρωμένο σύστημα ηλεκτρονικής συνταγογράφησης στην Ελλάδα εξυπηρετεί τους εξής σκοπούς:

- Βοηθά στον αποτελεσματικό εκσυγχρονισμό του συστήματος της φαρμακευτικής περίθαλψης. Οι χειρόγραφες συνταγές έχουν αντικατασταθεί από τις ηλεκτρονικές και η διαδικασία έχει αυτοματοποιηθεί.
- Συντελεί στην ταυτοποίηση και τον έλεγχο των εμπλεκόμενων για να διασφαλίσει την ακεραιότητα της διαδικασίας και την ευρεία και επιτυχημένη επιχειρησιακή λειτουργία της ηλεκτρονικής συνταγογράφησης.
- Βελτιώνει την ασφάλεια και την ποιότητα της φροντίδας του ασθενούς. Με την αύξηση του όγκου των φαρμάκων και της πολυπλοκότητα των ιατρικών αναγκών των ασθενών, υφίσταται αυξημένος κίνδυνος σφαλμάτων και παρενεργειών. Η ηλεκτρονική συνταγογράφηση μπορεί

να βελτιώσει την ασφάλεια και την ποιότητα της φροντίδας του ασθενούς και να μειώσει τον αριθμό των λαθών

- Εισάγει και αξιοποιεί τις λειτουργίες της ηλεκτρονικής συνταγογράφησης στην καθημερινή πρακτική.

## **E-prescription**

Το ελληνικό σύστημα ηλεκτρονικής συνταγογράφησης, το επονομαζόμενο e-prescription, καθιερώθηκε το 2013. Πλέον καλύπτει περισσότερο από το 95% του συνόλου των συνταγών που γράφονται. Το e-prescription δημιουργήθηκε με σκοπό να παρέχει τις εξής λειτουργίες: την ηλεκτρονική καταχώρηση και διαχείριση των συνταγών, την ενίσχυση της γνώσης μέσω της δυνατότητας άμεσης πρόσβασης στις πληροφορίες των φαρμάκων, την υποστήριξη λήψης αποφάσεων καθώς βοηθά την επιλογή των φαρμάκων με διάφορες ειδοποιήσεις, την υποστήριξη κατά την διοίκηση, την δημιουργία ηλεκτρονικών συνδέσμων μεταξύ των νοσοκομείων και των φαρμακείων, τις βελτιώσεις στις ήδη υπάρχουσες εργασιακές διαδικασίες, και τέλος την ανάπτυξη μιας διαδρομής ελέγχου για το σύνολο της χρήσης των φαρμάκων. [117]

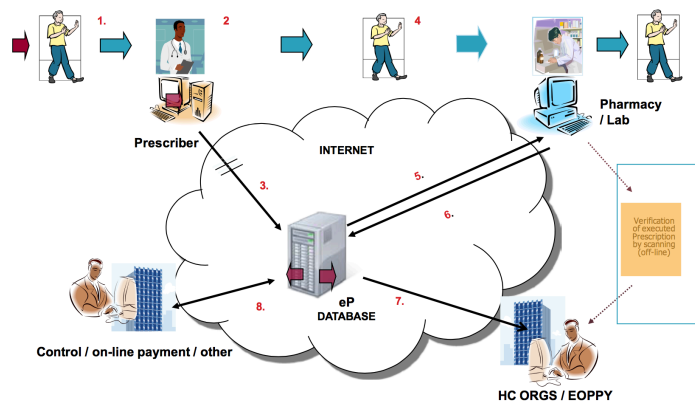
Το e-prescription πραγματοποιεί την ηλεκτρονική επεξεργασία των συνταγών για όλους τους ασθενείς, οι οποίοι είναι ασφαλισμένοι σε κάποιον από τους εθνικούς φορείς ασφαλίσεων. Το σύστημα εστιάζει ιδιαίτερα στην βελτίωση της ασφάλειας των ασθενών, την καλύτερη αξιοποίηση των πόρων και στην διαλειτουργικότητα με άλλα εθνικά ή ευρωπαϊκά συστήματα. Είναι διαθέσιμο μέσω του διαδικτύου (internet) και αποτελεί ένα αυτόνομο ηλεκτρονικό σύστημα εισόδου στο οποίο κάθε υποβολή συνταγής είναι ασφαλής και αναγνωρίζεται από έναν μοναδικό αριθμό. Στο σχήμα 6.5 απεικονίζεται η συνολική δομή του συστήματος [63].

Ο θεράπων ιατρός δημιουργεί τις συνταγές οι οποίες αποθηκεύονται στην εθνική βάση του συστήματος e-prescription, η οποία βρίσκεται στην ΗΔΙΚΑ [89]. Όταν ένα φαρμακείο θέλει να εκτελέσει μία συνταγή την "διαβάζει" από την κεντρική βάση. Οι πληροφορίες ου βρίσκονται στην βάση δεδομένων είναι διαθέσιμες (μέσω ασφαλούς on-line πρόσβασης, ή εξειδικευμένων περιοδικών αναφορών) σε όλα τα σχετικά και ενδιαφερόμενα μέρη (Οργανισμοί Ασφάλισης Υγείας, Υπουργείο Υγείας, εποπτικές αρχές, κλπ). Οι συνταγές που έχουν εκτελεστεί από τα φαρμακεία ελέγχονται από τους οργανισμούς ασφάλισης, ενώ το σύστημα e-prescription χρησιμοποιείται για να διασταυρώνονται οι έλεγχοι [126].

Η εφαρμογή του e-prescription είναι μία διαδικτυακή εφαρμογή. Στο σύστημα έχουν πρόσβαση μόνο εξουσιοδοτημένοι χρήστες (ιατροί, φαρμακοποιοί, κλπ), οι οποίοι αναγνωρίζονται και αποκτούν

άδεια πρόσβασης μετά την χρήση κατάλληλων διαπιστευτηρίων. για χωροθέτηση και αδειοδότηση.

Ο θεράπων ιατρός δημιουργεί συνταγές, οι οποίες παρέχουν όλες τις απαραίτητες πληροφορίες τόσο για τους ιατρικούς όσο και για τους διοικητικούς σκοπούς. Στις πληροφορίες αυτές συμπεριλαμβάνεται ο αριθμός κοινωνικής ασφάλισης του ασθενούς και του ιατρού, ο κωδικός της διάγνωσης (που κωδικοποιείται στο ICD-10), τα στοιχεία του καθορισμένου φαρμάκου (ποσότητα, δοσολογία, κ.λ.π.), το ποσοστό συμμετοχής του ασθενή στην πληρωμή του φαρμάκου και άλλα στοιχεία. Αυτές οι πληροφορίες εκτός από την βοήθεια που προσφέρουν στους ιατρούς, συντελούν στην γνώση των λεπτομερειών της διαδικασίας συνταγογράφησης εύκολα και σε πραγματικό χρόνο. Η δυνατότητα αυτή καθιστά το e-prescription σύστημα ένα ισχυρό εργαλείο όχι μόνο για την ενίσχυση της παρακολούθησης της δημόσιας υγείας και τον σχεδιασμό της, αλλά και για την βελτίωση του διοικητικού έλεγχου και τη διαφάνεια των δαπανών των φαρμάκων. Τέλος, το σύστημα e-prescription παρέχει στους ασθενείς την δυνατότητα να επιλέξουν, το φάρμακο που επιθυμούν ή μπορούν να αντεπεξέλθουν οικονομικά από μια λίστα ισοδυνάμων φαρμάκων στην οποία, συμπεριλαμβάνονται και τα γενόσημα φάρμακα.



Σχήμα 6.5: Το σύστημα E-prescription στην Ελλάδα.

## 6.2.2 Ηλεκτρονικός Ιατρικός Φάκελος (EMR)

Στην σύστημα υγείας η πληρότητα και η διαθεσιμότητα των δεδομένων και της πληροφορίας είναι ζωτικής σημασίας. Ημιτελείς πληροφορίες μπορούν να έχουν ως αποτέλεσμα κακή διάγνωση και θεραπεία, σπατάλη χρημάτων και πόρων ακόμα και να επιφέρουν καταστάσεις οι οποίες να απειλούν τη ζωή. Ο όγκος των πληροφοριών που σχετίζονται με την φροντίδα του ασθενούς έχει πολλαπλασιαστεί τα τελευταία χρόνια, γεγονός που οφείλεται σε μεγάλο ποσοστό στην ενσωμάτωση αυξημένου αριθμού εργαστηριακών και παρακλινικών εξετάσεων στους φακέλους των ασθενών.

Ο ηλεκτρονικός ιατρικός φάκελος είναι ένα σύστημα σχεδιασμένο ώστε να υποστηρίξει την από-

λυτη διαθεσιμότητα και την ακρίβεια ιατρικών ή άλλων πληροφοριών με σκοπό την παροχή ιατρικής περίθαλψης. Ο ηλεκτρονικός ιατρικός φάκελος ασθενούς είναι μία διαρκώς εξελισσόμενη έννοια και καθορίζεται ως μια συλλογή από ιατρικές πληροφορίες ιδιωτών ή πληθυσμών. Είναι αποθηκευμένος σε ψηφιακή μορφή, έχει την ικανότητα να διαμοιράζεται μεταξύ διαφορετικών ιατρικών λογισμικών και μπορεί να μεταφερθεί μέσω του δικτύου. Ένας ηλεκτρονικός φάκελος μπορεί να περιλαμβάνει μεγάλο πλήθος δεδομένων σε πλήρη ή περιληπτική μορφή και πρέπει κάθε χρονική στιγμή να περιέχει την επίσκεψη του ασθενούς, το ιατρικό ιστορικό του, τη διάγνωση, τη νοσηλεία (συνταγογράφηση, αποτελέσματα εργαστηριακών εξετάσεων), τα δημογραφικά στοιχεία του (Όνομα, ΑΜΚΑ, Ασφαλιστικός φορέας, ομάδα αίματος κ.τ.λ.), προσωπικά στοιχεία όπως βάρος, ύψος, φύλο και αλλεργίες.

Σε αυτό το σημείο θα πρέπει να κάνουμε τον διαχωρισμό του ηλεκτρονικού ιατρικού φακέλου (EMR) και του ηλεκτρονικού φακέλου υγείας (EHR). Ο ηλεκτρονικός φάκελος υγείας (EHR) είναι μια εξελισσόμενη έννοια που αποθηκεύει ψηφιακά ένα υποσύνολο δεδομένων ή όλα τα δεδομένα σχετικά με τις ιατρικές πράξεις που έγιναν κατά τη διάρκεια της ζωής ενός ατόμου με σκοπό την υποστήριξη της ποιοτικής, προσβάσιμης και αποτελεσματικής συνέχειας στην παροχή υπηρεσιών υγείας. Πρόκειται δηλαδή, για ένα υποσύνολο του EMR κάθε οργανισμού παροχής περίθαλψης, που είναι στην ιδιοκτησία του ασθενούς και περιέχει τις εισαγωγές του ασθενούς. Ο ηλεκτρονικός ιατρικός φάκελος σε αντίθεση ορίζεται ως το αρχείο του ασθενούς που δημιουργήθηκε από τους παρόχους υγείας για συγκεκριμένες επισκέψεις του ασθενή σε νοσοκομεία ή ιατρεία, και μπορεί να χρησιμεύσει ως πηγή δεδομένων για τον ηλεκτρονικό φάκελο υγείας (EHR). Ο ηλεκτρονικός φάκελος υγείας (EHR) δημιουργείται και διατηρείται εντός ενός θεσμικού οργάνου, όπως ένα νοσοκομείο, ένα ολοκληρωμένο δίκτυο διανομής, μία κλινική, ή ένα γραφείο ιατρού για να δώσει σε όλους τους εμπλεκόμενους στην ιατρική φροντίδα πρόσβαση στο ιατρικό ιστορικό του ασθενούς [75]. Ο EMR είναι η νόμιμη καταγραφή που δημιουργείται στα νοσοκομεία και αποτελεί την πηγή των δεδομένων για τον EHR. Ο EHR αντιπροσωπεύει την ικανότητα στο να μοιράζονται εύκολα τις ιατρικές πληροφορίες οι “άμεσοι ενδιαφερόμενοι” και οι πληροφορίες που αφορούν τους ασθενείς να τους ακολουθούν και να μπορούν να ανακτηθούν από οποιαδήποτε μονάδα υγείας.

Το λογισμικό του ηλεκτρονικού φακέλου υγείας ουσιαστικά είναι ένα σύστημα διαχείρισης ιατρικών φακέλων που βασίζεται σε ηλεκτρονικούς υπολογιστές. Το γεγονός αυτό έχει ως αποτέλεσμα να αποθηκεύονται και να ανακτώνται δεδομένα γρήγορα και με ασφάλεια. Επιπλέον, τα δεδομένα επεξεργάζονται εύκολα και γρήγορα και μπορούν να μεταφερθούν σε οποιαδήποτε σύστημα, σε οποιαδήποτε απόσταση. Το σύστημα καταγραφής των δεδομένων γίνεται πιο αποτελεσματικό και περισσότερο πλήρες. Ο ηλεκτρονικός ιατρικός φάκελος εμπεριέχει μία πληθώρα δεδομένων, τα οποία έχουν διαφορετική μορφή [111]. Το ιστορικό, η κλινική εξέταση και τα αποτελέσματα εργαστηριακών εξε-



τάσεων, είναι σε μορφή κειμένου, οι εξετάσεις του ασθενούς (ακτινογραφίες, τομογραφίες - αξονικές, μαγνητικές, απλές - υπέρηχοι κ.α.) είναι σε μορφή στατικών εικόνων, τα ηλεκτροκαρδιογραφήματα είναι σε μορφή βιοσημάτων, τα αποτελέσματα των ενδοσκοπικών εξετάσεων (γαστροσκόπηση κ.α.) είναι σε μορφή βίντεο κ.λ.π. Στον ηλεκτρονικό ιατρικό φάκελο, όλα τα δεδομένα ενσωματώνονται στον φάκελο του ασθενούς χωρίς να παίζει σημαντικό ρόλο η μορφή τους. Σε διάφορα σημεία του κειμένου του ιστορικού και της κλινικής εξέτασεως ενσωματώνονται ακτινολογικές ή βιοχημικές εξετάσεις, πράγμα που κάνει αμέσως εμφανή την συσχέτιση των εν λόγω εξετάσεων με την γενικότερη κατάσταση του ασθενούς.

Μερικά από τα πιο ουσιαστικά πλεονεκτήματα της χρήσης ενός συστήματος ηλεκτρονικού ιατρικού φακέλου είναι τα εξής[115][114]:

- Ο ηλεκτρονικός ιατρικός φάκελος ασθενούς είναι μία τεχνολογική καινοτομία με μεγάλο όφελος και αποτελεσματικότητα στην παροχή φροντίδας υγείας. Οι πληροφορίες που αφορούν έναν συγκεκριμένο ασθενή έχουν συγκεντρωθεί και έχουν αποθηκευτεί στον ηλεκτρονικό υπολογιστή μειώνοντας στο ελάχιστο τον κίνδυνο να χαθούν ή να παραποιηθούν τα δεδομένα, ενώ μπορούν να χρησιμοποιηθούν παράλληλα από αρκετούς χρήστες των υπηρεσιών υγείας γρήγορα και άμεσα.
- Κέρδος χρόνου κατά την αναζήτηση και τη συμπλήρωση των στοιχείων του ιατρικού φακέλου. Η διείσδυση των τεχνολογιών αιχμής στον ιατρικό κόσμο εκσυγχρονίζει την διαδικασία, καθώς κρατάει όλα τα στοιχεία συγκεντρωμένα και παρέχει δυνατότητα άμεσης πρόσβασης - αναζήτησης, με αποτέλεσμα να επιταχύνει τις διαδικασίες.
- Η φροντίδα οργανώνεται καλύτερα, σωστότερα, και ταχύτερα γύρω από τον ασθενή. Ανάλογα με την κατάσταση της υγείας του ασθενούς οργανώνεται και προγραμματίζεται η θεραπεία και οι διαγνωστικές ή άλλου είδους θεραπευτικές παρεμβάσεις που πρέπει να γίνουν.
- Στους ηλεκτρονικούς φακέλους το ενδεχόμενο και η πιθανότητα λάθους ή παράλειψης ελαχιστοποιείται και έτσι δημιουργείται εμπιστοσύνη προς το σύστημα.

### **Ηλεκτρονικός Ιατρικός Φάκελος στην Ελλάδα**

Η εφαρμογή της Πληροφορικής στον δημόσιο τομέα στην Ελλάδα καθυστέρησε πολύ. Οι ιατρικοί φάκελοι στα περισσότερα Δημόσια νοσοκομεία ακόμα και σήμερα παραμένουν χειρόγραφοι, ογκώδεις, ασαφείς, δυσεύρετοι, δυσανάγνωστοι ενώ πολλές φορές χάνονται, φθείρονται και αλλοιώνονται.

Η αναζήτηση ιστορικών και κλινικών δεδομένων είναι πολύ δύσκολη, ενώ η εξαγωγή στατιστικών συμπερασμάτων εντελώς αβέβαιη και πολύπλοκη. Στις περιπτώσεις που υπάρχει ατομικός ηλεκτρονικός φάκελος, τα περιεχόμενα δεδομένα δεν μπορούν να επικοινωνήσουν ακόμα και με το εσωτερικό δίκτυο του ιδίου νοσοκομείου, με κυριότερη αιτία το ότι ο ηλεκτρονικός φάκελος και το πληροφοριακό διαχειριστικό σύστημα δεν έχουν ούτε την κατάλληλη διασύνδεση ούτε την απαραίτητη διαλειτουργικότητα.

Η κλινική πληροφορία για τους εξωτερικούς ασθενείς καταγράφεται χειρόγραφα από τους εφημερεύοντες γιατρούς σε πράσινες καρτέλες, και όταν ο ασθενής επισκεφθεί ξανά το νοσοκομείο πολλές φορές αναγκάζεται να αποκτήσει νέα κάρτα είτε γιατί η προηγούμενη χάθηκε, είτε γιατί καταχωρήθηκε με διαφορετικό όνομα αποκτώντας άλλο αριθμό μητρώου. Ακόμα κι αν ο ασθενής έχει καταχωρηθεί στο Διαχειριστικό Πληροφοριακό σύστημα αποκτώντας αυτόματα έναν μοναδικό Αριθμό Μητρώου από το Γραφείο Κίνησης ή τη Γραμματεία Εξωτερικών ιατρείων, οι γιατροί συνεχίζουν να αναζητούν την χειρόγραφη καρτέλα για να καταγράψουν τη διάγνωση και το θεραπευτικό σχήμα, αρνούμενοι την καταγραφή όχι μόνο στο τερματικό τους, αλλά ακόμα και στο εκτυπωμένο έντυπο νοσηλείας. Απλές προγραμματισμένες επεμβάσεις ενώ είναι ήδη καταχωρημένες στο Διαχειριστικό Πληροφοριακό σύστημα και απαιτούν την χωρίς πολυπλοκότητα χειρισμού ηλεκτρονική επιβεβαίωση από το αντίστοιχο ιατρικό / νοσηλευτικό προσωπικό, εξακολουθούν να καταγράφονται στις πράσινες καρτέλες.

Η διαδικασία συμπλήρωσης του ιατρικού φακέλου για τους νοσηλευόμενους ασθενείς δεν είναι ιδιαίτερα σαφής ως προς τη διάγνωση και την πορεία της νόσου, ενώ όσον αφορά στη συνταγογράφηση τα πράγματα είναι οργανωμένα, καθώς στα περισσότερα νοσοκομεία εφαρμόζεται το σύστημα e-prescription. Η ασάφεια αυτή είναι ιδιαίτερα εμφανής στις περιπτώσεις της διάγνωσης στο ηλεκτρονικό εξιτήριο, στο οποίο οι διοικητικοί υπάλληλοι αντιγράφοντας στον ηλεκτρονικό υπολογιστή τη χειρόγραφη διάγνωση εξόδου αντιμετωπίζουν προβλήματα με το χειρόγραφο κείμενο, καθώς μπορεί να είναι πολύ γενικό ή δυσδιάκριτο. Για παράδειγμα η χειρόγραφη διάγνωση “σχιζοφρένεια, σχιζοτυπικές διαταραχές” στο νοσοκομείο Α αποτελεί έναν γενικό χαρακτηρισμό νόσου, ενώ σύμφωνα με τη διεθνή κωδικοποίηση ICD-10 μπορεί να σημαίνει 9 διαφορετικές μορφές νόσου (F20 Σχιζοφρένεια, F21 Σχιζοτυπική διαταραχή, F22 Εμμένουσες παραληρηματικές διαταραχές, F23 Οξείες και παροδικές ψυχωσικές διαταραχές, F24 Επαγωγική παραληρηματική διαταραχή, F25 Σχιζοσυναισθηματικές διαταραχές, F28 Άλλες μη οργανικές ψυχωσικές διαταραχές, F29 Διάφορες μορφές μη οργανικής ψύχωσης) [64]. Μεγάλο επίσης πρόβλημα αποτελεί η ηλεκτρονική καταγραφή χειρουργικών επεμβάσεων και η μετεγχειρητική κατάσταση, που απαιτούν μορφή ελεύθερου κειμένου, με συνέπεια την απομόνωση της κλινικής κατάστασης του ασθενή σε κάποιον φθαρμένο ογκώδη φάκελο, στοι-

χείο του Αρχείου κάθε νοσοκομείου. Οι γενικεύσεις στις διαγνώσεις εισόδου ή εξόδου ενός ασθενούς παρεμποδίζουν την στοιχειώδη πληροφορία για την κατάσταση υγείας. Για παράδειγμα ο ανωτέρω ασθενής σε περίπτωση μετέπειτα εισαγωγής σε ένα διαφορετικό νοσοκομείο συνοδευόμενος από τη γενική διάγνωση “σχιζοφρένεια, σχιζοτυπικές διαταραχές”, υπόκειται σε πληθώρα περιττών και επαναλαμβανόμενων εξετάσεων

Το 2014 ξεκίνησε η εφαρμογή του ηλεκτρονικού Φακέλου Υγείας των ασφαλισμένων του ΕΟΠΥΥ. φάκελος αποτελεί την ηλεκτρονική αποτύπωση του βιβλιαρίου υγείας του κάθε ασφαλισμένου του ΕΟΠΥΥ και περιέχει πληροφορίες σχετικά με διαγνώσεις, νοσηλείες, επισκέψεις, παροχές, ιατρικές πράξεις, αιμοκαθάρσεις και φάρμακα, αλλά δεν ταυτίζεται με τον ηλεκτρονικό ιατρικό φάκελο. Ο Φάκελος Ασφάλισης Υγείας -που ουσιαστικά συνδέει όλο το σύστημα δηλαδή την ΗΔΙΚΑ, e-dary, ΚΜΕΣ με τον ΕΟΠΥΥ και σε δεύτερο χρόνο τα νοσοκομεία- αποτελεί ένα εργαλείο προκειμένου να ελέγχονται όσοι γιατροί πραγματοποιούν υπέρ-συνταγογράφηση είτε φαρμάκων, είτε διαγνωστικών πράξεων. Σύμφωνα με την διοίκηση του ΕΟΠΥΥ, ο προσωπικός Φάκελος Υγείας κάθε ασφαλισμένου περιέχει στοιχεία για το σύνολο των νοσηλειών, των υπηρεσιών υγείας, των υλικών, των ασθενειών και των διαγνώσεων που υπάρχουν καταγεγραμμένες στο μηχανογραφικό σύστημά του. Στον Φάκελο Ασφάλισης Υγείας έχει δυνατότητα πρόσβασης μόνον ο ασφαλισμένος με τους κωδικούς που διαθέτει καθώς αφορά προσωπικά του στοιχεία, τα οποία έχει δυνατότητα να τα γνωστοποιήσει με προσωπική του ευθύνη.

### 6.2.3 epSOS

Η ανάγκη διασύνδεσης των συστημάτων υγείας και των πολιτικών που ακολουθούνται σε κάθε κράτος σχετικά με την υγεία στην Ευρωπαϊκή Ένωση, γίνεται όλο και πιο πειστική, λόγω της αυξημένης κινητικότητας των ασθενών και των επαγγελματιών του ιατρικού τομέα και της διάδοσης των νέων ιατρικών τεχνολογιών και τεχνικών που αφορούν την τεχνολογία της πληροφορίας. Η διασύνδεση των διάφορων συστημάτων υγείας είναι απαραίτητη για την καλύτερη ποιότητα αλλά και την άμεση πρόσβαση των πολιτών στη διασυνοριακή περίθαλψη. Πολλά κράτη μέλη της Ευρωπαϊκής Ένωσης έχουν αναπτύξει ηλεκτρονικά συστήματα σχετικά με τα αρχεία των ασθενών. Ο ηλεκτρονικός ιατρικός φάκελος του ασθενούς και η ηλεκτρονική συνταγογράφηση είναι δύο πολύ σημαντικά συστήματα τα οποία προάγουν την ασφαλή και αξιόπιστη ιατροφαρμακευτική περίθαλψη.

Η Ευρωπαϊκή Ένωση στα πλαίσια του στόχου της επικοινωνίας και της συνεργασίας των χωρών που εντάσσονται σε αυτή και την ύπαρξη κοινών οδηγιών στον τομέα της υγείας, τον Ιούλιο του 2008 σε συνεργασία με τους δημόσιους φορείς που έχουν αρμοδιότητα τις ηλεκτρονικές υπηρεσίες υγείας

(ηλεκτρονικός ιατρικός φάκελος, ηλεκτρονικής συνταγογράφησης) ξεκίνησε ένα σημαντικό έργο, τον ευρωπαϊκό πιλότο epSOS (Εξυπνες Ανοιχτές Ηλεκτρονικές Υπηρεσίες για τους Ευρωπαίους Ασθενείς). Ο κύριος στόχος του epSOS είναι η ανάπτυξη ενός πρακτικού πλαισίου ηλεκτρονικής υγείας και κατάλληλων υποδομών στον τομέα της Πληροφορικής και των Επικοινωνιών που θα επιτρέπουν την ασφαλή πρόσβαση των διάφορων μη εθνικών ευρωπαϊκών συστημάτων υγειονομικής περίθαλψης στις πληροφορίες αναφορικά με την υγεία του ασθενούς. Το epSOS είναι ένα έργο για την προώθηση της δια-λειτουργικότητας της ηλεκτρονικής υγείας, το οποίο χρηματοδοτεί η Ευρωπαϊκή Ένωση και θέλει να χτίσει μια υποδομή υπηρεσιών, που θα επιτρέπει τη διασυνοριακή διαλειτουργικότητα των συστημάτων ηλεκτρονικών μητρώων υγείας στην Ευρώπη, χωρίς να καταπατά όμως νομοθετικές ρυθμίσεις ή να υπερβαίνει τα ήδη υπάρχουσα εθνικά συστήματα[51].

Το epSOS αποτελείται από δύο χωριστές υπηρεσίες ηλεκτρονικής υγείας, τον ιατρικό φάκελο ασθενούς (patient medical record) και την ηλεκτρονική συνταγογράφηση (e-prescription), για τις οποίες αναζητούνται δια-λειτουργικές μέθοδοι στη διασυνοριακή επικοινωνία. Ειδικότερα, ο ιατρικός φάκελος ασθενούς (patient medical record) στα πλαίσια του epSOS αποτελείται από στοιχεία τα οποία εμπεριέχουν γενικές πληροφορίες για τον ασθενή (π.χ. όνομα, ηλικία, γένος), σημαντικά κλινικά δεδομένα (π.χ. αλλεργίες, χρόνιες ασθένειες, χειρουργικές επεμβάσεις), καθώς και τα τρέχοντα αλλά και παλαιότερα φαρμακευτικά σκευάσματα που λάμβανε. Ο φάκελος ασθενούς παρέχει τις βασικές πληροφορίες, οι οποίες βοηθούν τους επαγγελματίες του τομέα της υγείας να λάβουν σωστότερες και πιο ασφαλείς αποφάσεις σχετικά με την θεραπεία και τα φάρμακα που θα ακολουθήσουν οι ασθενείς. Οι πληροφορίες αυτές πρέπει να παρέχονται στην γλώσσα των εκάστοτε γιατρών, ώστε να μην προκύπτουν γλωσσικά εμπόδια. Η ηλεκτρονική συνταγογράφηση αφορά τη συνταγογράφηση των φαρμάκων με χρήση κατάλληλου λογισμικού και την ηλεκτρονική διαβίβασή τους. Στην ηλεκτρονική συνταγογράφηση η διαλειτουργικότητα μεταξύ των εθνικών συστημάτων είναι απαραίτητη όταν έχουμε έναν ασθενή ο οποίος χρειάζεται κάποιο φάρμακο, το οποίο έχει ήδη συνταγογραφηθεί σε κάποια άλλη χώρα. Ο φαρμακοποιός θα πρέπει να αποκτήσει ηλεκτρονική πρόσβαση στη συνταγή, και όταν το φάρμακο αγοραστεί, το ηλεκτρονικό σύστημα θα πρέπει να ενημερώσει σχετικά με τα διανεμημένα φάρμακα το σύστημα υγειονομικής περίθαλψης της χώρας του ασθενούς[58].

Το epSOS επιτυγχάνει την ασφαλή πρόσβαση των ασθενών σε πληροφορίες σχετικά με την υγεία τους για τα διάφορα ευρωπαϊκά συστήματα υγείας. Το σύστημα είναι φτιαγμένο για να εξυπηρετήσει έναν πολίτη της Ευρωπαϊκής Ένωσης που ενώ είναι κατοικεί σε μια συγκεκριμένη χώρα δέχεται τακτικά υπηρεσίες υγείας σε μια άλλη χώρα εξαιτίας του καθώς εργάζεται σε αυτή τη χώρα ή βρίσκεται προσωρινά σε μία άλλη χώρα (π.χ. για τουρισμό) και είναι ανάγκη να εκτελέσει μία ιατρική συνταγή της χώρας του ή χρειάζεται επείγουσα ιατρική περίθαλψη. Το epSOS παρέχει την κοινοτική οδηγία

για τη διασυννοριακή ιατρική περίθαλψη, η οποία θα πρέπει να ενσωματωθεί στο εθνικό δίκαιο των κρατών μελών τα επόμενα χρόνια και επιπροσθέτως προτείνει τεχνικές λύσεις σε όλα τα επίπεδα διαλειτουργικότητας, όπως και σε θέματα πολιτικής και σε νομικά ζητήματα. επεκτείνεται πέρα από την έκδοση συστάσεων, μοντέλων οργάνωσης, και εργαλείων λογισμικού στην δοκιμή των αποτελέσματα αυτών μέσω πραγματικών πιλοτικών εφαρμογών σε πολλές ευρωπαϊκές περιφέρειες και χώρες.

## 6.3 Eligibility as a Service

### 6.3.1 Motivation

Η πρωταρχική ευθύνη της υπηρεσίας μετάγγισης αίματος είναι να παρέχει ένα ασφαλές και επαρκές απόθεμα αίματος και προϊόντων που παράγονται από το αίμα. Θα πρέπει να διασφαλίζεται ότι τα προϊόντα που προέρχονται από δωρεές αίματος παρουσιάζουν ελάχιστο κίνδυνο ύπαρξης οποιασδήποτε μόλυνσης, που θα μπορούσε να μεταδοθεί μέσω μετάγγισης. Επίσης πρέπει να εξασφαλίζει ότι η υγεία του αιμοδότη δεν θα επιβαρυνθεί με οποιοδήποτε από τη δωρεά αίματος. Όλοι οι υποψήφιοι αιμοδότες αξιολογούνται για την καταλληλότητά τους να δωρίσουν αίμα, σε κάθε περίπτωση δωρεάς.

Οι βασικές φάσεις ελέγχου του αίματος είναι οι εξής:

- Φάση 1: Συμπλήρωση χειρόγραφου ερωτηματολογίου σχετικά με το ιστορικό υγείας του ασθενούς (βλ. παράρτημα Α).
- Φάση 2: Εξέταση του εθελοντή αιμοδότη από ιατρό και βασικές εξετάσεις (μέτρηση πίεσης, έλεγχος αιμοσφαιρίνης) και ερωτήσεις σχετικά με το ιατρικό ιστορικό του αιμοδότη.
- Φάση 3: Μετά την αιμοληψία, το αίμα περνάει μοριακούς και μικροβιολογικούς ελέγχους.

Οι πληροφορίες που παρέχονται από 164 χώρες για την παγκόσμια Βάση Δεδομένων για την ασφάλεια αίματος δείχνουν ότι πραγματοποιούνται περισσότερες από 92 εκατομμύρια αιμοδοσίες ετησίως[125]. Επιπλέον, τουλάχιστον 13 εκατομμύρια υποψήφιοι δότες απορρίπτονται πριν προλάβουν να φτάσουν στην αιμοληψία, δηλαδή στις φάσεις 1 και 2 όπως περιγράφηκαν παραπάνω[152]. Οι λόγοι απόρριψης κατά τις φάσεις αυτές, οι οποίοι και θα εξεταστούν στην παρούσα διπλωματική, έχουν να κάνουν με το ιατρικό ιστορικό του ασθενή (χρόνιες ασθένειες, αλλεργίες κ.λ.π.), με υπάρχουσες ιατρικές καταστάσεις (λοιμώξεις, λήψη αντιβίωσης, λήψη φαρμάκων που σε καθιστούν ακατάλληλο για αιμοδοσία κ.λ.π.) και με καταστάσεις της προσωπικής ζωής του εθελοντή (π.χ. ομοφυλοφιλία). Το υπερβολικά μεγάλο ποσοστό απορρίψεων 14.13% τονίζει την ανάγκη να μειωθεί ο

αριθμός των περιπτώσεων της προσωρινής απόρριψης (με τον όρο προσωρινή απόρριψη, αναφερόμαστε σε προσωρινή αναβολή της αιμοδοσίας που συμβαίνει στις φάσεις 1, 2 λόγω προσωρινής ακαταλληλότητας των δοτών).

Το ποσοστό αυτό, γίνεται ιδιαίτερος σημαντικό και ζημιογόνο οικονομικά αν κάνουμε μία ανάλυση των επιμέρους κοστών της αιμοδοσίας, δεδομένου του γενικότερου υψηλού κόστους κατά τη διαδικασία της αιμοδοσίας. Στη συνέχεια θα παρουσιάσουμε την μέση τιμή αυτών των κοστών, όπως υπολογίστηκε από το ελληνικό Υπουργείο Υγείας, το 2011 [60]. Στα κόστη περιλαμβάνεται το κόστος μηχανογράφησης (απογραφή και συντήρηση) των κέντρων αίματος, η τιμή των διαφόρων τύπων εξοπλισμού καθώς και το υπόλοιπο του κύκλου ζωής (που μετράται σε έτη) για τον ιατρικό εξοπλισμό. Επίσης, μετρώνται τα πάγια κόστη των μονάδων τα οποία σχετίζονται με την ηλεκτρική ενέργεια, τον καθαρισμό, την ασφάλεια, την διαχείριση και όλες τις άλλες υποστηρικτικές υπηρεσίες που διατίθενται για τα κέντρα αίματος. Στον πίνακα 6.4 αναφέρονται αναλυτικά τα κόστη, των φάσεων 1 και 2 για την συλλογή 1 μονάδας αίματος στην Ελλάδα.

Περιγραφή	Μέσο κόστος
Προσωπικό	46.86
Γενικά έξοδα	2.88
Αναλώσιμα	1.09
Μηχανογράφηση	0.92
Σακούλα αίματος	12.30
Εξοπλισμός	0.04
Έμμεσο κόστος	34.00
Συνολικό κόστος	98.07

Πίνακας 6.4: Κόστος για την συλλογή 1 μονάδας αίματος στην Ελλάδα.

Παρατηρούμε ότι το συνολικό κόστος για τις φάσεις 1 και 2 είναι 98.07 € ανά μονάδα αίματος. Με βάση το ποσοστό απόρριψης 14.13% βλέπουμε ότι για κάθε 100 αιμοδοσίες έχουμε 1,385.73 € έξοδα για περιπτώσεις στις οποίες δεν θα πραγματοποιηθεί ποτέ αιμοληψία. Σε αυτό το σημείο πρέπει να τονίσουμε, ότι δεν έχουν συμπεριληφθεί όλα τα κόστη στην μελέτη μας, παρά μόνο τα απολύτως διακριτά και ξεκάθαρα μετρήσιμα.

Πέραν του οικονομικού μέρους, μια άλλη πολύ σημαντική παράμετρος είναι η άσκοπη κατανάλωση χρόνου τόσο του ιατρικού προσωπικού του αιμοδοτικού κέντρου, όσο και των εθελοντών αιμοδοτών. Οι εθελοντές αιμοδότες καταναλώνουν αρκετό χρόνο για τον προγραμματισμό, την μετάβαση στο αιμοδοτικό κέντρο, την αναμονή και για την διαδικασία αιμοδοσίας. Δεδομένου ότι το

μεγαλύτερο ποσοστό είναι εργαζόμενοι, ο χρόνος αυτός είναι σημαντικός και είναι επιθυμητό να αποφευχθεί η περιττή ταλαιπωρία που προκύπτει για τους δωρητές λόγω προσωρινής τους απόρριψης. Η αυτοματοποιημένη και πλήρως μηχανογραφημένη προσωρινή απόρριψη ενός εθελοντή αιμοδότη, θα εξασφαλίσει μείωση της σπατάλης του χρόνου αυτού.

Μία επιπλέον πολύ σημαντική παράμετρος αποτελεί το ποσοστό επιστροφής των εθελοντών αιμοδοτών μετά από μία προσωρινή απόρριψη. Μελέτες έχουν δείξει ότι οι περισσότεροι εθελοντές δεν επιστρέφουν αν απορριφθούν μία φορά. Πιο συγκεκριμένα, μόλις το 11% επιστρέφει μετά την απαιτούμενη περίοδο για να πραγματοποιήσει δωρεά αίματος[79].

### **6.3.2 Decision Support Systems & Clinical Decision Support**

Τα συστήματα υποστήριξης αποφάσεων (DSS - Decision Support Systems) μπορούν να οριστούν ως εφαρμογές λογισμικού οι οποίες παρέχουν υποστήριξη κατά την διαδικασία λήψης αποφάσεων, παρέχοντας βοήθεια στους χρήστες να κατανοήσουν τις επιπτώσεις των δράσεών τους[61]. Τα DSS βρίσκουν χρήση σ' ένα ευρύ φάσμα εφαρμογών και εξυπηρετούν τη διαχείριση, τη λειτουργία αλλά και τα επίπεδα σχεδιασμού ενός συστήματος ή οργανισμού. Τα συστήματα υποστήριξης λήψης αποφάσεων μπορεί να είναι είτε πλήρως μηχανογραφημένα, είτε να διαχειρίζονται από ανθρώπινο δυναμικό ή να αποτελούν συνδυασμό και των δύο (υβριδικά) [117].

Τα συστήματα υποστήριξης λήψης κλινικών αποφάσεων (CDS) παρέχουν στους κλινικούς γιατρούς, στο προσωπικό, στους ασθενείς, και σε άλλα άτομα με γνώσεις σχετικές με αντικείμενα του τομέα υγείας, υποστήριξη στα κλινικά καθήκοντα τα οποία είναι συσχετισμένα με την διαδικασία απόφασης, με σκοπό την ενίσχυση της υγείας και της υγειονομικής περίθαλψης[97]. Ένα σύστημα υποστήριξης κλινικών αποφάσεων, ορίζεται ως ένα “ενεργό σύστημα γνώσης, το οποίο χρησιμοποιεί δύο ή περισσότερες κατηγορίες δεδομένων του ασθενούς για να δημιουργήσει εξειδικευμένες συμβουλές”. Ένα CDS σύστημα είναι απλά ένα σύστημα υποστήριξης αποφάσεων που επικεντρώνεται στην διαχείριση της γνώσης με τέτοιο τρόπο ώστε να παρέχει αξιόπιστες κλινικές συμβουλές. Σε αυτό το σημείο κρίνουμε απαραίτητο να τονίσουμε ότι το CDS δεν αποσκοπεί σε καμία περίπτωση σε αντικατάσταση του γιατρού αλλά έχει καθαρά επικουρικό ρόλο, με τον γιατρό να έχει πάντα τον τελευταίο λόγο[117].

Στην παρούσα διπλωματική μας ενδιαφέρουν τα CDS συστήματα, τα οποία στηρίζονται σε κάποια βάση γνώσης (Knowledge-based CDSS) .Τα συστήματα αυτά αποτελούνται από τρία μέρη: τη βάση των γνώσεων, έναν μηχανισμό εξαγωγής συμπερασμάτων, και ένα μηχανισμό επικοινωνίας. Η βάση γνώσης περιέχει τους κανόνες και τις συσχετίσεις των στοιχείων που συγκεντρώνονται σε αυτήν,

που συνήθως παίρνουν τη μορφή του κανόνες IF-THEN. Το σύστημα συμπερασμού συνδυάζει τους κανόνες από τη βάση της γνώσης με τα δεδομένα. Ο μηχανισμός επικοινωνίας επιτρέπει στο σύστημα να δείχνει τα αποτελέσματα που προκύπτουν στον αιτούντα.

## 6.4 Υλοποίηση

Στόχος είναι η υλοποίηση μίας υπηρεσίας, η οποία θα τρέχει σε υπολογιστικό νέφος ιδιωτικού τύπου (private cloud) και θα κάνει έναν πρωταρχικό έλεγχο σχετικά με το αν ο εθελοντής επιτρέπεται να αιμοδοτήσει ή όχι. Η υπηρεσία εξετάζει την καταλληλότητα του εθελοντή, με βάση το ιατρικό ιστορικό του. Αντλεί δεδομένα από τον ηλεκτρονικό ιατρικό του φάκελο τους ασθενούς και την ηλεκτρονική συνταγογράφηση με σκοπό να εξετάσει αν υπάρχουν διαγνώσεις ή συνταγογραφήσεις, οι οποίες παραβιάζουν τα κριτήρια επιλεξιμότητας. Η υπηρεσία ανακτά τα δεδομένα που χρειάζεται για τους ελέγχους της με σύνδεση στο σύστημα epSOS, κάνοντας χρήση του AMKA του συγκεκριμένου εθελοντή.

Τα κριτήρια επιλεξιμότητας, τα οποία προκύπτουν με βάση τις οδηγίες του WHO, [152] που ελέγχει το σύστημα μας είναι τα εξής.

- Αντιβιοτικά: Υποχρεωτική αποχή του αιμοδότη για 7 ημέρες, μετά την λήξη της θεραπείας.
- Αιμορραγίες: Αν υπάρχει ιστορικό αιμορραγιών ή μη φυσιολογικής πήξης του αίματος, ο εθελοντής αποκλείεται.
- Μετάγγιση αίματος: Υποχρεωτική αποχή 12 μηνών από αιμοδοσία.
- Καρκίνος: Υποχρεωτική αποχή 12 μηνών από αιμοδοσία.
- Ειδικές περιπτώσεις καρκίνου: Λευχαιμία, λέμφωμα, νόσος του Hodgkin και άλλοι καρκίνοι του αίματος συνεπάγονται τον οριστικό αποκλεισμό του αιμοδότη.
- Γρίπη, κρουολόγημα: Υποχρεωτική αποχή του αιμοδότη για 7 ημέρες.
- Ο εθελοντής έχει πάρει ανθρώπινη αυξητική ορμόνη: Μόνιμος αποκλεισμός του εθελοντή.
- Εξαγωγή δοντιού ή οδοντική εγχείριση: Υποχρεωτική αποχή 3 ημερών
- Καρδιακά προβλήματα: υποχρεωτική αποχή 6 μηνών μετά από ένα επεισόδιο στηθάγχης, υποχρεωτική αποχή 6 μηνών μετά από μια καρδιακή προσβολή, υποχρεωτική αποχή 6 μηνών μετά από εγχείριση bypass ή αγγειοπλαστική, υποχρεωτική αποχή 6 μηνών μετά από αλλαγή , που είχε ως αποτέλεσμα την αλλαγή στα φάρμακά σας.



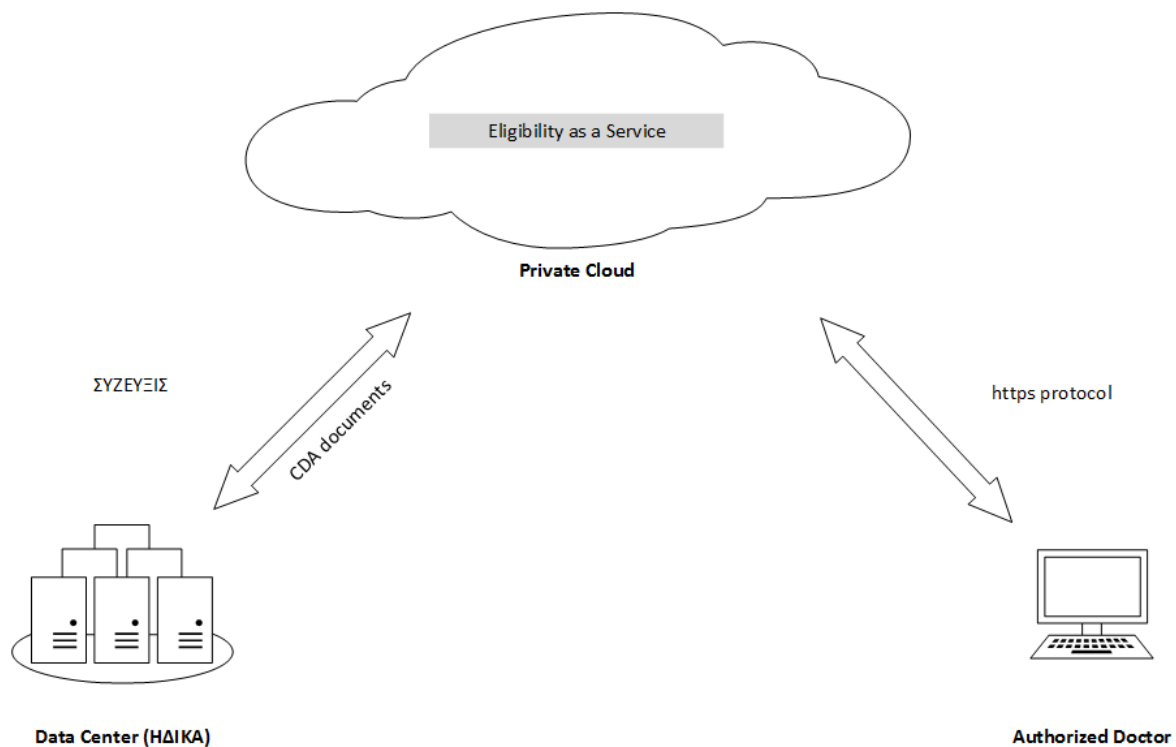
- Αιμοχρωμάτωση: Μόνιμος αποκλεισμός του εθελοντή.
- Ηπατίτιδα: Μόνιμος αποκλεισμός του εθελοντή.
- HIV, AIDS: Μόνιμος αποκλεισμός του εθελοντή.
- Ερυθρά, ιλαρά, παρωτιτιδά: Υποχρεωτική αποχή 4 εβδομάδων.
- Ελονοσία: υποχρεωτική αποχή 3 ετών μετά την θεραπεία με ελονοσία.
- Σύφιλη ή γονόρροια: υποχρεωτική αποχή ενός χρόνου μετά την ολοκλήρωση της θεραπείας.
- Δρεπανοκυτταρική αναιμία: Μόνιμος αποκλεισμός του εθελοντή.
- Φυματίωση: Μόνιμος αποκλεισμός του εθελοντή.
- Accutane, Amnesteem, Claravis ή Sotret, Proscar, and Propecia: Υποχρεωτική αποχή 4 εβδομάδων μετά την τελευταία δόση.
- Avodart, Jalyn: Υποχρεωτική αποχή 6 εβδομάδων μετά την τελευταία δόση.
- Ασπιρίνη: Υποχρεωτική αποχή 48 ωρών.
- Effient: Υποχρεωτική αποχή 14 ημερών εβδομάδων.
- Coumadin , heparin, Pradaxa, Xarelto , Lovenox: Υποχρεωτική αποχή.
- Plavix: Υποχρεωτική αποχή 14 ημερών μετά την τελευταία δόση.
- Soriatane: Υποχρεωτική αποχή 3 χρόνων μετά την τελευταία δόση.
- Tegison: Μόνιμος αποκλεισμός του εθελοντή.
- Ticlid:Υποχρεωτική αποχή 14 ημερών μετά την τελευταία δόση.

Με βάση τα άνωθεν, το ιατρικό ιστορικό του ασθενούς και την κατάλληλη παραμετροποίηση, ο αλγόριθμος που τρέχει αποφαινεται για το αν ο ασθενής είναι κατάλληλος ή όχι για αιμοδοσία. Όπως είδαμε για τα CDS (Clinical Decision Support) συστήματα, το σύστημα επιστρέφει την πρόταση του σχετικά με το αν πρέπει να απορριφθεί ή όχι ο ασθενής και την αναφορά των ευρημάτων του αλλά ο γιατρός είναι αυτός ο οποίος καταλήγει στην τελική απόφαση έγκρισης ή απόρριψης του ασθενούς. Το σύστημα χρησιμοποιεί προκαθορισμένους κανόνες, ενώ προσαρμόζει την βάση γνώσης στα ιατρικά δεδομένα που αντλεί για τον συγκεκριμένο ασθενή που εξετάζεται.

## Αρχιτεκτονική και Σχεδιασμός

Τα στοιχεία και συστήματα που χρησιμοποιούνται στην δημιουργία του Eligibility as a Service είναι τα εξής:

- το CDS σύστημα, το οποίο βρίσκεται σε ένα ιδιωτικό νέφος (cloud).
- Το “ΣΥΖΕΥΞΙΣ”, το οποίο είναι ένα έργο του Υπουργείου Διοικητικής Μεταρρύθμισης & Ηλεκτρονικής Διακυβέρνησης (ΥΔΜΗΔ), με το οποίο επιδιώκεται η ανάπτυξη και ο εκσυγχρονισμός της τηλεπικοινωνιακής υποδομής του Δημόσιου Τομέα. Αποτελεί ένα δίκτυο πρόσβασης και κορμού για τους φορείς του Δημοσίου, με σκοπό να καλύψει όλες τις ανάγκες για τη μεταξύ τους επικοινωνία με Τηλεφωνία (τηλεφωνική επικοινωνία ανάμεσα στους φορείς). Θα χρησιμοποιηθεί για να εξασφαλίσουμε την ασφαλή πρόσβαση στα ιατρικά δεδομένα των ασθενών.
- Το κέντρο δεδομένων (data center), της ΗΔΙΚΑ (Ηλεκτρονική Διακυβέρνηση Κοινωνικής Ασφάλισης Α.Ε.), στο οποίο στεγάζονται τα δεδομένα του e-prescription και το patient record.
- CDA(ersos) πρωτόκολλο επικοινωνίας, για την μεταφορά των ιατρικών απαιτούμενων δεδομένων από την ΗΔΙΚΑ στο CDS σύστημα.



**Σχήμα 6.6:** Αρχιτεκτονική του συστήματος EaaS (Eligibility as a Service).

Τα δεδομένα της ηλεκτρονικής συνταγογράφησης και του ηλεκτρονικού ιατρικού φακέλου του ασθενούς βρίσκονται στο κέντρο δεδομένων (data center) της ΗΔΙΚΑ. Για να προσπελαστούν τα δεδομένα με ασφάλεια, χρησιμοποιούμε ασφαλή πρόσβαση μέσω του συστήματος ΣΥΖΕΥΞΙΣ. Τα δεδομένα που στέλνονται από το patient record και το e-prescription μορφοποιούνται σε κατάλληλα CDA documents, με τις επιπλέον προτυποποιήσεις που προτείνει το epSOS για να αποσταλούν στο CDS σύστημα.

Το επιτυχές σενάριο χρήσης είναι το εξής: Ο εγκεκριμένος γιατρός εκκινεί το Eligibility as a Service. Πληκτρολογεί στο σύστημα το ΑΜΚΑ του εθελοντή και ανακατευθύνεται ώστε να συνδεθεί, με χρήση των κωδικών του και μέσω της ασφαλούς πρόσβασης που εξασφαλίζει το ΣΥΖΕΥΞΙΣ, στο e-prescription και στο patient record. Το σύστημα “τραβά” τα δεδομένα που χρειάζεται και με βάση τους κανόνες παραγωγής καταλήγει αν ο αιμοδότης τηρεί τα κριτήρια επιλεξιμότητας ή όχι. Η σύνδεση με το e-prescription και το patient record τερματίζεται, αντίγραφο των δεδομένων δεν διατηρείται και το σύστημα επιστρέφει στον γιατρό την απόφαση του μαζί με την κατάλληλη αναφορά. Στο σημείο αυτό, να τονίσουμε ότι το σύστημα συμπεριφέρεται σαν “μαύρο κουτί”, γεγονός που μας επιτρέπει να χρησιμοποιήσουμε τα ευαίσθητα ιατρικά δεδομένα χωρίς να χρειάζεται να εφαρμόσουμε ανωνυμοποίηση τους (anonymization) και χωρίς να παραβιάζουμε την Αρχή Προστασίας Δεδομένων Προσωπικού Χαρακτήρα.



## Κεφάλαιο 7

### Επίλογος

Στο πλαίσιο της παρούσας εργασίας μελετήσαμε, σχεδιάσαμε και υλοποιήσαμε ένα ολοκληρωμένο σύστημα διαχείρισης αιμοδοσιών. Αφού εντοπίστηκαν τα κύρια τεχνολογικά ζητήματα εφαρμογής του συστήματος, προτείνουμε τις τεχνολογικές προσεγγίσεις για την αντιμετώπιση τους. Το σύστημα μας σχεδιάστηκε με σκοπό την προσέλκυση, την αύξηση και την διατήρηση του απαραίτητου αριθμού ενεργών εθελοντών, ώστε να καλύπτονται οι ανάγκες της χώρας. Αρκετή βαρύτητα δόθηκε στην αρχιτεκτονική πρόταση υλοποίησης του ενιαίου συστήματος και ακόμα περισσότερο στην αρχιτεκτονική και την σχεδίαση της εφαρμογής, η οποία εκμεταλλεύεται τις σύγχρονες τεχνολογίες cloud. Επιπλέον, σχεδιάσαμε το σύστημα διαχείρισης των αιμοδοσιών από την πλευρά των κέντρων αιμοδοσίας, με κριτήρια την αξιοπιστία, την ευχρηστία και την αποτελεσματικότητα. Παρουσιάστηκε ένα ενδεικτικό σχήμα της βάσης δεδομένων, η οποία είναι σχεδιασμένη ώστε εξυπηρετεί το πλήρες σύστημα και να είναι εύκολα επεκτάσιμη.

Μελετήσαμε τα ζητήματα διαλειτουργικότητας και ασφαλούς επικοινωνίας με τρίτα συστήματα και προτάθηκαν μηχανισμοί χρήσης των πρωτοκόλλων αυτών και επίτευξης της επιθυμητής διαλειτουργικότητας. Τέλος, αναπτύξαμε ένα Clinical Decision Support System ως μία ανεξάρτητη Software as a Service υπηρεσία, η οποία είναι πολύτιμη για τις υπηρεσίες αιμοδοσίας και προλαμβάνει ανεπιθύμητα γεγονότα, βοηθώντας ουσιαστικά στην βελτιστοποίηση των υπηρεσιών αιμοδοσίας και στην πρόληψη της μετατροπής των ενεργών αιμοδοτών σε ανενεργούς. Το σύστημα αυτό διασυνδέθηκε πλήρως με τα υπόλοιπα σύστημα αιμοδοσίας, και το δεύτερο “ενσωμάτωσε” ουσιαστικά την υπηρεσία.

## 7.1 Μελλοντική δουλειά

### 7.1.1 Εφαρμογή έξυπνου ρολογιού - SmartWatch

Τα τελευταία χρόνια τα ρολόγια έχουν εξελιχθεί και αυτά σε έξυπνα (Smartwatch). Η κύρια λειτουργία τους έχει εξελιχθεί και παραπέρα της απλής ενημέρωσης της ώρας. Ενώ τα πρώτα μοντέλα των έξυπνων ρολογιών εκτελούσαν βασικές λειτουργίες, όπως υπολογισμούς, μετατροπές και κάποιες φορές εκτέλεση παιχνιδιών, τα πιο πρόσφατα έξυπνα ρολόγια είναι σε γενικές γραμμές υπολογιστές που φοριούνται. Δίνεται η δυνατότητα σύνδεσής τους με κάποιο smartphone, tablet ή phablet μέσω Bluetooth. Με τον τρόπο αυτό μπορούν να εμφανίσουν εισερχόμενες κλήσεις, να προκαλέσουν εξερχόμενες κλήσεις, για αυτό και μερικά έξυπνα ρολόγια χαρακτηρίζονται και ως “watchphones ή ρολόγια-τηλέφωνα”, και να παρέχουν στην άλλη συσκευή πληροφορίες από τους αισθητήρες τους. Μπορούν, επίσης, όπως όλοι οι υπολογιστές, να διαθέτουν κάμερα, θερμομέτρο, πυξίδα, οθόνη αφής, πλοήγηση μέσω GPS, ακουστικά, κάρτες SD.

Μια μελλοντική επέκταση του συστήματος αποτελεί η δημιουργία εφαρμογής για Smartwatches. Στην εφαρμογή αυτή, θα περιλαμβάνονται οι τρεις πιο βασικές λειτουργικότητες: λήψη ειδοποιήσεων (push notifications) για γεγονότα αιμοδοσίας και διαχείριση τους, δηλαδή αποδοχή, απόρριψη ή αναβολή, το κλείσιμο ραντεβού για αιμοδοσία καθώς και οι υπενθυμίσεις. Ένα από τα πολύ σημαντικά πλεονεκτήματα των smartwatch, το οποίο θέλουμε να εκμεταλλευτούμε είναι η δυνατότητα άμεσης ενημέρωσης για σημαντικά γεγονότα, οποιαδήποτε χρονική στιγμή και σε όποια τοποθεσία βρισκόμαστε χωρίς να είναι απαραίτητη η συνεχής ενασχόληση και χρήση του κινητού τηλεφώνου. Ως εκ τούτου, ειδοποιήσεις που συγκαταλέγονται σε αυτές με αυξημένη βαρύτητα μπορούν να σταλούν στο smartwatch για άμεση επαφή με τον χρήστη.

### 7.1.2 Υποσύστημα πρόβλεψης αιμοδοσιών

Στις μελλοντικές επεκτάσεις περιλαμβάνεται μετά την έναρξη της πιλοτικής εφαρμογής του συστήματος, πέραν από τα οφέλη που έχουμε προαναφέρει στην παρούσα διπλωματική, να πραγματοποιηθεί συλλογή δεδομένων. Τα δεδομένα αυτά θα περάσουν αρχικά από διαδικασία ανωνυμοποίησης (anonymization) και στη συνέχεια θα τα επεξεργαστούμε με σκοπό να βγάλουμε τις παράγωγες μεταβλητές που μας ενδιαφέρουν και μπορούν να βοηθήσουν στην μελέτη μας. Θα γίνει πλήρη ανάλυση των δεδομένων. Θα γίνει εξέταση των δεδομένων, με σκοπό να βρεθούν χαρακτηριστικά τα οποία πιθανόν να έχουν αξία στην πρόβλεψη των μελλοντικών αιμοδοσιών ενός εθελοντή.

Προηγούμενες μελέτες έχουν δείξει ότι οι διάφορες πτυχές του ιστορικού των δωρεών του κάθε δότη συμβάλλουν στην ερμηνεία της μετέπειτα συμπεριφοράς του (επιστροφή για αιμοδοσία ή όχι). Σύμφωνα με τα δεδομένα των James και Matthews το ποσοστό της επιστροφής των δωτών αυξάνεται γραμμικά σε σχέση με τον αριθμό των προηγούμενων αιμοδοσιών του εθελοντή [92].

Πολλές από τις παραγόμενες μεταβλητές που έχουμε σκοπό να εξάγουμε συσχετίζονται εγγενώς. Αυτό είναι αποδεκτό σε ένα μοντέλο σχεδιασμένο για προβλέψεις, εφ' όσον τέτοιες συσχετίσεις υπάρχουν στον ίδιο βαθμό στο σύνολο των δεδομένων που χρησιμοποιούνται για την εκτίμηση όσο και το σύνολο των δεδομένων που χρησιμοποιούνται για την πρόβλεψη. Θα πρέπει, ωστόσο, να υπάρξει ιδιαίτερη προσοχή στην ερμηνεία της σχετικής σημασίας των μεταβλητών που θα χρησιμοποιηθούν και θα συσχετιστούν, καθώς αν μια μεταβλητή παραλειφθεί από το μοντέλο, η σημασία των συσχετισμένων μεταβλητών θα αυξηθεί. Στόχος μας είναι, να γίνει πρόβλεψη του μελλοντικού όγκου αιμοδοσιών καθώς και αναγνώριση μοτίβων και τάσεων αιμοδοσίας με στατιστικά μοντέλα.





## Βιβλιογραφία

- [1] D. Abadi, S. Madden, and M. Ferreira. Integrating compression and execution in column-oriented database systems. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 671–682. ACM, 2006.
- [2] D. J. Abadi, P. A. Boncz, and S. Harizopoulos. Column-oriented database systems. *Proceedings of the VLDB Endowment*, 2(2):1664–1665, 2009.
- [3] P. B. G. Abraham Silberschatz. *Operating System Concepts*. Wiley, 2005.
- [4] Android. Android application fundamentals. <http://developer.android.com/guide/components/fundamentals.html>
- [5] Android. Android broadcast receiver. <http://developer.android.com/reference/android/content/BroadcastReceiver>
- [6] Android. Android content providers. <http://developer.android.com/guide/topics/providers/content-providers.html>.
- [7] Android. Android services. <http://developer.android.com/guide/components/services.html>.
- [8] Android. The android source code: Governance philosophy. <https://source.android.com/source/>.
- [9] Android. Get started with publishing. <http://developer.android.com/distribute/googleplay/start.html>.
- [10] Android. Testing fundamentals. <http://developer.android.com/tools/testing/testingandroid.html>.
- [11] AngularJS. Angularjs: Conceptual overview. <http://docs.angularjs.org/guide/>.
- [12] J. Anuzzi. *Advanced Android Application Development*. Addison-Wesley Professional, 2014.
- [13] Apple. Apple developer program. <https://developer.apple.com/programs/>.
- [14] P. Avery. Research indicates increase in cloud computing. *IT Business Edge*, 2009.
- [15] K. Banker. *MongoDB in action*. Manning Publications Co., 2011.

- [16] Bannerman. Cloud computing adoption risks: State of play. *Software Engineering Conference Cloud Workshop*, 2010.
- [17] D. Basu and R. Kulkarni. Overview of blood components and their preparation. *Indian Journal of Anaesthesia*, 58(5):529–537, 2014.
- [18] Y. Benkler. *The Wealth of Networks*. Yale University Press, 2006.
- [19] T. Benson. *Principles of health interoperability HL7 and SNOMED*. Springer Science, 2012.
- [20] E. Bertino, B. C. Ooi, R. Sacks-Davis, K.-L. Tan, J. Zobel, B. Shidlovsky, and B. Catania. Object-oriented databases. In *Indexing Techniques for Advanced Database Systems*, pages 1–38. Springer, 1997.
- [21] B. Blobel and P. Pharow. HL7—more than a communications standard. *Advanced Health Telematics and Telemedicine: The Magdeburg Expert Summit Textbook*, 96:266, 2003.
- [22] D. Blumenthal. Stimulating the adoption of health information technology. *N Engl J Med*, 360(15):1477–1479, Apr. 2009.
- [23] P. Blundell. *Learning Android Application Testing*. Packt Publishing, 2015.
- [24] P. H. B. Bolton-Maggs. Transfusion safety in 2012: main messages from the shot annual report for 2012. *Transfusion Med*, 23(4):217–218, 2013.
- [25] G. Booch, J. Rumbaugh, and I. Jacobson. *Unified Modeling Language User Guide, The (2Nd Edition) (Addison-Wesley Object Technology Series)*. Addison-Wesley Professional, 2005.
- [26] K. W. Boone. *The CDA Tm Book*. Springer Science & Business Media, 2011.
- [27] K. Bourquard and L. Gall. Standards for interoperability in digital health: Selection and implementation in an ehealth project. In *Requirements Engineering for Digital Health*, pages 95–115. Springer, 2015.
- [28] Bowers. What are white blood cells.
- [29] N. Bowers. What is plasma.
- [30] S. Brahler. Analysis of the android architecture. *karlsruhe institute of technology*, 2010.
- [31] G. Briscoe and A. Marinos. Digital ecosystems in the clouds: Towards community cloud computing. *CoRR*, abs/0903.0694, 2009.
- [32] R. Cattell. Scalable sql and nosql data stores. *SIGMOD Rec.*, 39(4):12–27, May 2011.

- [33] J. Cechanowicz, C. Gutwin, B. Brownell, and L. Goodfellow. Effects of gamification on participation and data quality in a real-world market research domain. In *Gamification 2013*, pages 58–65, Stratford Ontario, 2013.
- [34] M. Clinic. Blood donation.
- [35] A. Cockburn. *Writing Effective Use Cases*. Addison-Wesley Professional, 2000.
- [36] E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387, June 1970.
- [37] C. Collins. *Android in Practice*. Manning Publications, 2011.
- [38] I. o. M. Committee on Quality of Health Care in America. *To Err Is Human:: Building a Safer Health System*. National Academies Press, 2000.
- [39] X. G. Condori. Community cloud computing. *Revista de Información Teconología Y Sociedad*, 2013.
- [40] B. Custer, A. Chinn, N. V. Hirschler, M. P. Busch, and E. L. Murphy. The consequences of temporary deferral on future whole blood donation. *Transfusion*, 47(8):1514–1523, 2007.
- [41] B. Custer, K. S. Schlumpf, D. Wright, T. L. Simon, S. Wilkinson, P. M. Ness, and for the NHLBI Retrovirus Epidemiology Donor Study-II. Donor return after temporary deferral. *Transfusion*, 51(6):1188–1196, 2011.
- [42] Dan. Cloud models images. <http://www.techiestuffs.com/3-types-of-cloud-computing-services-for-businesses/>.
- [43] Danek. Cloud computing and the canadian environment. *Public Works Government Services*, 2009.
- [44] R. Dawkins. *How a Scientist Changed the Way We Think*. Oxford University Press, 2007.
- [45] R. R. P. de Vries, J.-C. Faber, P. F. W. Strengers, and M. of the Board of the International Haemovigilance Network. Haemovigilance: an effective tool for improving transfusion practice. *Vox Sanguinis*, 100(1):60–67, 2011.
- [46] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels. Dynamo: Amazon’s highly available key-value store. *SIGOPS Oper. Syst. Rev.*, 41(6):205–220, Oct. 2007.

- [47] S. Deterding, D. Dixon, R. Khaled, and L. Nacke. From game design elements to gamefulness: Defining "gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, MindTrek '11, pages 9–15, New York, NY, USA, 2011. ACM.
- [48] L. N. Devine EB, Wilson-Norton JL. Characterization of prescribing errors in an internal medicine clinic. *Am J Health Syst Pharm*, 2007 May.
- [49] D. N. Dhingra. Blood safety in the developing world and who initiatives. *Vox Sanguinis*, 83:173–177, 2002.
- [50] F. Doelitzscher, A. Sulistio, C. Reich, H. Kuijs, and D. Wolf. Private cloud for collaboration and e-learning services: from iaas to saas. *Computing*, 91(1):23–42, 2011.
- [51] A. Dogac. Interoperability in ehealth systems. *Proc. VLDB Endow.*, 5(12):2026–2027, Aug. 2012.
- [52] R. H. Dolin, L. Alschuler, and Beebe. The hl7 clinical document architecture. *Journal of the American Medical Informatics Association*, 8(6):552–569, 2001.
- [53] J. T. Dudley, Y. Pouliot, R. Chen, A. A. Morgan, and A. J. Butte. Translational bioinformatics in the cloud: an affordable alternative. *Genome Medicine*, 2(8):51–51, Aug. 2010.
- [54] Edlich. List of nosql databases. <http://nosql-database.org/>, 2015.
- [55] E.KE.A. Blood donation statistics.
- [56] E.KE.A. Donor questionnaire.
- [57] E.KE.A. Διαδικασία Αιμοδοσίας.
- [58] epSOS. epsos services.
- [59] E. Feingold. The gift relationship: From human blood to social policy. *Medical Care*, 11(3):pp. 261–263, 1973.
- [60] V. Fragoulakis, K. Stamoulis, E. Grouzi, and N. Maniadakis. The cost of blood collection in greece: An economic analysis. *Clinical Therapeutics*, 36(7):1028–1036.e5, 2014.
- [61] French. Decision analysis and decision support systems. *The University of Manchester*, 2000.
- [62] Y. C. Fu. The game of life: Designing a gamification system to increase current volunteer participation and retention in volunteer-based nonprofit organizations. *Undergraduate Student Research Awards*, 2011.

- [63] D. A. G. Pangalos. The new greek national e-prescription system: A powerful ehealth tool for improving services to citizens and administrative control. *New Horizons in Industry and Education*, 2013.
- [64] Galinos. Icd-10.
- [65] Gartner. Gartner says by 2015, more than 50 percent of organizations that manage innovation processes will gamify those processes, 2011.
- [66] L. Goldman. *Textbook of Medicine*. Saunders, 2000.
- [67] C. Gong. The characteristics of cloud computing. *39th International Conference on Parallel Processing Workshops*, 2010.
- [68] S. Goyal. Public vs private vs hybrid vs community - cloud computing: A critical review. *International Journal of Computer Network & Information Security;Feb2014, Vol. 6 Issue 3, p20*, February 2014.
- [69] P. M. T. Grance. The nist definition of cloud computing. *Special Publication 800-145*, 2011.
- [70] F. GRAZIOLI. Mapping relational databases into a distributed nosql. *Politecnico di Milano*, 2014.
- [71] A. J. Grindon. Safe and good use of blood in surgery (sanguis). girolamo sirchia, anna maria giovanetti, d.b.l. mclelland, giovanni n. fracchia. *Transfusion*, 36(1):88–88, 1996.
- [72] J. M. Grossman, D. A. Cross, E. R. Boukus, and G. R. Cohen. Transmitting and processing electronic prescriptions: experiences of physician practices and pharmacies. *Journal of the American Medical Informatics Association*, 19(3):353–359, 2012.
- [73] R. L. Grossman. The case for cloud computing. *IT Professional*, 11(2):23–27, 2009.
- [74] M. Guidry. Components of an android application. <http://eagle.phys.utk.edu/guidry/android/applicationFundamentals.html>.
- [75] T. D. Gunter and N. P. Terry. The emergence of national electronic health record architectures in the united states and australia: Models, costs, and questions. *Journal of Medical Internet Research*, 7(1):e3–, Feb. 2005.
- [76] C. N. M. Gunther SCHADOW. The hl7 reference information model under scrutiny. In *Ubiquity: technologies for better health in aging societies: proceedings of MIE2006*, volume 124, page 151. IOS Press, 2006.

- [77] R. Gupta. Backbone.js vs angularjs : Demystifying the myths. <http://www.nebithi.com/2012/12/27/backbone-and-angular-demystifying-the-myths/>, 27.12.2012.
- [78] M. Haghghat, S. Zonouz, and M. Abdel-Mottaleb. Cloudid: Trustworthy cloud-based and cross-enterprise biometric identification. *Expert Systems with Applications*, 42(21):7905 – 7916, 2015.
- [79] D. Halperin, J. Baetens, and B. Newman. The effect of short-term, temporary deferral on future blood donation. *Transfusion*, 38(2):181–183, 1998.
- [80] J. Hamari, J. Koivisto, and H. Sarsa. Does gamification work? – a literature review of empirical studies on gamification. In *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, pages 3025–3034, Jan 2014.
- [81] J. Han, E. Haihong, G. Le, and J. Du. Survey on nosql database. In *Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on*, pages 363–366, Oct 2011.
- [82] H. He and A. K. Singh. Graphs-at-a-time: query language and access methods for graph databases. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 405–418. ACM, 2008.
- [83] H. Hellas. Who we are. <http://www.hl7.org.gr/en/content/who-we-are>.
- [84] HL7. HL7 messaging standard version 2.7.
- [85] HL7. HL7 version 3 product suite.
- [86] HL7. HL7 version 3: Reference information model (rim).
- [87] A. J. I. Jacobson, M. Ericsson. *The Object Advantage: Business Process Reengineering with Object Technology*. Addison-Wesley Professional, 1994.
- [88] IDC. Smartphone os market share, 2015.
- [89] IDIKA. Idika s.a. <http://www.idika.gr>.
- [90] Interoute. Cloud characteristics.
- [91] J. M. L. Ivo Blohm. Gamification - design of it-based enhancing services for motivational support and behavioral change. *Business & Information Systems Engineering*, 2013.
- [92] M. D. James RC. Analysis of blood donor return behaviour using survival regression methods. *Transfusion Med*, 1996.

- [93] J. Kabachinski. What's the forecast for cloud computing in healthcare? *Biomedical Instrumentation & Technology*, 45(2):146–150, Mar. 2011.
- [94] G. Kanagaraj and A. Sumathi. Proposal of an open-source cloud computing system for exchanging medical images of a hospital information system. In *Trendz in Information Sciences and Computing (TISC), 2011 3rd International Conference on*, pages 144–149, Dec 2011.
- [95] A. M. Kaplan and M. Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, 53(1):59 – 68, 2010.
- [96] F. Kart, L. Moser, and P. Melliar-Smith. Building a distributed e-healthcare system using soa. *IT Professional*, 10(2):24–30, March 2008.
- [97] C. A. H. Kensaku Kawamoto. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. 2005.
- [98] P. Kierkegaard. E-prescription across europe. *Health and Technology*, 3(3):205–219, 2013.
- [99] W. Kim. *Introduction to object-oriented databases*, volume 90. MIT press Cambridge, MA, 1990.
- [100] W. Kim, J. Nicolas, and S. Nishio. The object-oriented database system. In *Deductive and Object-Oriented Databases: Proceedings of the First International Conference on Deductive and Object-Oriented Databases (DOOD89) Kyoto Research Park, Kyoto, Japan, 4-6 December 1989*, page 223. Elsevier, 2014.
- [101] K. Landsteiner. Zur kenntnis der antifermentativen, lytischen und agglutinierenden wirkungen des blutserums und der lympe. *Zentralblatt Bakteriologie*, 1900.
- [102] K. Landsteiner and A. S. Wiener. An agglutinable factor in human blood recognized by immune sera for rhesus blood. *Experimental Biology and Medicine*, 43(1):223, 1940.
- [103] N. Leavitt. Whatever happened to object-oriented databases? *Computer*, 33(8):16–19, 2000.
- [104] L. Lee. Volume of blood in a human, 1998.
- [105] K. Lemmens, C. Abraham, T. Hoekstra, R. Ruiter, W. De Kort, J. Brug, and H. Schaalma. Why don't young people volunteer to give blood? an investigation of the correlates of donation intentions among young nondonors. *Transfusion*, 45(6):945–955, 2005.
- [106] K. P. H. Lemmens, C. Abraham, R. A. C. Ruiter, I. J. T. Veldhuizen, A. E. R. Bos, and H. P. Schaalma. Identifying blood donors willing to help with recruitment. *Vox Sanguinis*, 95(3):211–217, 2008.

- [107] T. C. Liu, P. Kuperan, C. Tan, T. K. Lin, and S. H. Lee. Structuring a safer donor-replacement program. *Transfusion*, 38(4):368–372, 1998.
- [108] Y. Liu, T. Alexandrova, and T. Nakajima. Gamifying intelligent environments. In *Proceedings of the 2011 International ACM Workshop on Ubiquitous Meta User Interfaces*, Ubi-MUI '11, pages 7–12, New York, NY, USA, 2011. ACM.
- [109] G. Lucassen and S. Jansen. Gamification in consumer marketing - future or fallacy? *Procedia - Social and Behavioral Sciences*, 148:194 – 202, 2014. 2nd International Conference on Strategic Innovative Marketing.
- [110] J. S. M. Missbach. Private cloud infrastructures for sap. *Springer Berlin Heidelberg*, 2013.
- [111] G. Makoul, R. H. Curry, and P. C. Tang. The use of electronic medical records: Communication patterns in outpatient encounters. *Journal of the American Medical Informatics Association : JAMIA*, 8(6):610–615, July 2001.
- [112] O. Marantidou, L. Loukopoulou, E. Zervou, G. Martinis, A. Egglezou, P. Fountouli, P. Dimoxenus, M. Parara, M. Gavalaki, and A. Maniatis. Factors that motivate and hinder blood donation in greece. *Transfusion Medicine (Oxford, England)*, 17(6):443–450, Aug. 2007.
- [113] A. Mardan. To engineers who tried to use jade template engine and can't get started.
- [114] S. McLANE. Designing an emr planning process based on staff attitudes toward and opinions about computers in healthcare. *Computers Informatics Nursing*, 23(2):–, 2005.
- [115] N. Menachemi N FAU Menachemi and T. H. Collum TH FAU Collum. Benefits and drawbacks of electronic health record systems.
- [116] M. H. Mike Cantelon. *Node.js in Action*. Manning Publications, 2013.
- [117] J. K. Miller RA1, Gardner RM. Clinical decision support and electronic prescribing systems: a time for responsible thought and action. *Journal of the American Medical Informatics Association*, 2005 Jul-Aug.
- [118] A. H. Misje, V. Bosnes, O. Gåsdal, and H. E. Heier. Motivation, recruitment and retention of voluntary non-remunerated blood donors: a survey-based questionnaire study. *Vox Sanguinis*, 89(4):236–244, 2005.
- [119] S. Mohan. *Node.js Essentials*. Packt Publishing, Sep 9, 2014.



- [120] L. S.-T. Oana-Sorina Lupșe, Mihaela Marcella Vida. Cloud computing and interoperability in healthcare information systems. *INTELLI 2012 : The First International Conference on Intelligent Systems and Applications*, 2012.
- [121] G. of Greece. Ministerial directive Y4γ-OIK.121672-2009. *Greek Government Gazette*, 2009.
- [122] A. S. of Hemology. Blood basics.
- [123] B. of Wisconsin. Blood 101 (all about blood).
- [124] W. H. Organization. *Blood donor selection*. World Health Organization, 2012.
- [125] W. H. Organization. More voluntary blood donations essential, 2012.
- [126] I. Papala. The new greek national e-prescription system: a framework for a more productive and effective implementation. *National School of public administration, Division of health economics, Athens*, 2012.
- [127] J. Peng, X. Zhang, Z. Lei, B. Zhang, W. Zhang, and Q. Li. Comparison of several cloud computing platforms. In *Information Science and Engineering (ISISE), 2009 Second International Symposium on*, pages 23–27, Dec 2009.
- [128] Politis. Blood donation systems as an integral part of the health system. *Arch Hellen Med*, 2000.
- [129] M. Potel. Mvp: Model-view-presenter. *Taligent*, 1996.
- [130] A. D. Rajnikant Palwe, Gurudatt Kulkarni. A new approach to hybrid cloud. *International Journal of Computer Science and Engineering Research and Development*, 2015.
- [131] E. Redmond and J. R. Wilson. *Seven databases in seven weeks: a guide to modern databases and the NoSQL movement*. Pragmatic Bookshelf, 2012.
- [132] A. C. O. Rheumatology. E-prescribing. <http://www.rheumatology.org/practice/office/hit/erx.asp>.
- [133] H. Robin, L. Marc, and Z. Robert. Mda: A formal approach to game design and game research. *AAAI workshop*, 2004.
- [134] I. Robinson, J. Webber, and E. Eifrem. *Graph databases*. ” O’Reilly Media, Inc.”, 2013.
- [135] D. Rojas, B. Kapralos, and A. Dubrowski. The missing piece in the gamification puzzle. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, Gamification ’13, pages 135–138, New York, NY, USA, 2013. ACM.

- [136] J. J. S. Bhardwaj and S. Jain. Cloud computing: A study of infrastructure as a service. *International Journal of engineering and information Technology*, 2010.
- [137] J. Schindler. I/o characteristics of nosql databases. *Proceedings of the VLDB Endowment*, 5(12):2020–2021, 2012.
- [138] E. J. Schweitzer. Reconciliation of the cloud computing model with us federal electronic health record regulations. *Journal of the American Medical Informatics Association*, 19(2):161–165, Mar. 2012.
- [139] M. Seeger and S. Ultra-Large-Sites. Key-value stores: a practical overview. *Computer Science and Media, Stuttgart*, 2009.
- [140] C. Sessa. *50 Android Hacks*. Manning Publications, 2013.
- [141] H. L. Seven. Health level seven international. <http://www.hl7.org/>.
- [142] D. Shaver. Impact of the social media on consumer behavior in the united. *University of Central Florida*, 2007.
- [143] J. W. . Sons. *Professional Node.js: Building Javascript Based Scalable Software*. 01-Oct-2012.
- [144] D. Stainsby, S. MacLennan, and P. J. Hamilton. Management of massive blood loss: a template guideline. *British Journal of Anaesthesia*, 85(3):487–491, 2000.
- [145] M. Stonebraker. Sql databases v. nosql databases. *Communications of the ACM*, 53(4):10–11, 2010.
- [146] C. Tidy. *Thalassaemia*, 2015.
- [147] S. Tilkov and S. Vinoski. Node.js: Using javascript to build high-performance network programs. *IEEE Internet Computing*, 14(6):80–83, 2010.
- [148] B. Tudorica and C. Bucur. A comparison between several nosql databases with comments and notes. In *Roedunet International Conference (RoEduNet), 2011 10th*, pages 1–5, June 2011.
- [149] S. Ward. Consumer socialization. *Journal of Consumer Research*, 1(2):pp. 1–14, 1974.
- [150] J. Wharton. Dagger - a fast dependency injector for android. <http://square.github.io/dagger/>.
- [151] S. Whittemore. *The Circulatory System*. Chelsea House Pub, 2009.
- [152] WHO. Blood safety - eligibility requirements.

- [153] Wikipedia. Bootstrap.
- [154] Wikipedia. Software framework.
- [155] Z. Zhou, H. Zhang, X. Du, P. Li, and X. Yu. Prometheus: Privacy-aware data retrieval on hybrid cloud. In *INFOCOM, 2013 Proceedings IEEE*, pages 2643–2651, April 2013.
- [156] G. Zichermann. *Gamification by design*. O’Reilly Media, 2011.
- [157] O. Zuckerman and A. Gal-Oz. Deconstructing gamification: evaluating the effectiveness of continuous measurement, virtual rewards, and social comparison for promoting physical activity. *Personal and Ubiquitous Computing*, 18(7):1705–1719, 2014.
- [158] ΚΕ.ΕΛ.Π.ΝΟ. Ενημερωτικό Δελτίο, September 2015.
- [159] Καραβαγγέλη-Βλάτσα. Εθελοντική Αιμοδοσία. *Μικροβιολογικά Χρονικά*, 2005.
- [160] C. R. Κωνσταντίνα Πολίτη. ΔΕΛΤΙΟ ΕΠΙΔΗΜΙΟΛΟΓΙΚΗΣ ΕΠΙΤΗΡΗΣΗΣ ΛΟΙΜΩΞΕΩΝ ΠΟΥ ΜΕΤΑΔΙΔΟΝΤΑΙ ΜΕ ΤΟ ΑΙΜΑ (1996-2011) ΕΠΙΤΗΡΗΣΗΣ ΑΝΕΠΙΘΥΜΗΤΩΝ ΑΝΤΙΔΡΑΣΕΩΝ ΚΑΙ ΣΥΜΒΑΝΤΩΝ ΣΧΕΤΙΚΑ ΜΕ ΤΗ ΜΕΤΑΓΓΙΣΗ ΑΙΜΑΤΟΣ (1997-2011) ΕΠΙΤΗΡΗΣΗΣ ΑΝΕΠΙΘΥΜΗΤΩΝ ΑΝΤΙΔΡΑΣΕΩΝ ΚΑΙ ΣΥΜΒΑΝΤΩΝ ΚΑΤΑ ΚΑΙ ΜΕΤΑ ΤΗΝ ΑΙΜΟΔΟΣΙΑ (2003-2011). Συντονιστικό Κέντρο Αιμοεπαγρύπνησης (ΣΚΑΕ), 2012.



## **Παράρτημα Α**

### **Ερωτηματολόγιο Αιμοδότη**



Ημ/νία:

Αρ. Μητρ. Αιμοδότη:

## ΙΣΤΟΡΙΚΟ ΑΙΜΟΔΟΤΗ

### ΣΤΟΙΧΕΙΑ ΑΙΜΟΔΟΤΗ (Συμπληρώνεται από τον αιμοδότη)

Επώνυμο:	<input type="text"/>	Όνομα:	<input type="text"/>	Πατρώνυμο:	<input type="text"/>
Έτος γέννησης:	<input type="text"/>	Τόπος γέννησης:	<input type="text"/>		
Επάγγελμα:	<input type="text"/>	Αριθμ. Ασπ. Ταυτότ.:	<input type="text"/>		
Διεύθυνση:	<input type="text"/>	T.K. - Πόλη:	<input type="text"/>	Τηλ.:	<input type="text"/>

### (Συμπληρώνεται από τον ιατρό)

Αρτηριακή Πίεση (συστ./διασπ):	<input type="text"/> mm/Hg	Σφύξεις:	<input type="text"/> /min	Θερμοκρασία:	<input type="text"/> °C	Βάρος:	<input type="text"/> Kg
Αιματοκρίτης:	<input type="text"/> %	Αιμοσφαιρίνη:	<input type="text"/> gr/dl	Ομάδα Αίματος:	<input type="text"/> Rh <input type="text"/>	Ύψος:	<input type="text"/> m
Γενική Εντύπωση:	<input type="text"/>						

ΚΑΤΗΓΟΡΙΑ ΑΙΜΟΔΟΤΗ: Εθελοντής  Εθελοντής ενόπλων δυνάμεων  Συγγενής ασθενή

### ΙΑΤΡΙΚΟ ΙΣΤΟΡΙΚΟ

Έχετε δώσει αίμα άλλη φορά; ΝΑΙ:  ΟΧΙ:  Πότε δώσατε τελευταία φορά;

	ΝΑΙ	ΟΧΙ		ΝΑΙ	ΟΧΙ
Έχετε ποτέ αποκλειστεί από αιμοδοσία;	<input type="checkbox"/>	<input type="checkbox"/>	Μεταδοτικό νόσημα στο περιβάλλον σας	<input type="checkbox"/>	<input type="checkbox"/>
Κάνετε επικίνδυνο επάγγελμα ή χόμπι;	<input type="checkbox"/>	<input type="checkbox"/>	Λήψη φαρμάκων;	<input type="checkbox"/>	<input type="checkbox"/>
Είχατε προβλήματα υγείας παλαιότερα;	<input type="checkbox"/>	<input type="checkbox"/>	Τις τελευταίες 5 ημέρες ήσαστε ασπιρίνη;	<input type="checkbox"/>	<input type="checkbox"/>
<b>Είχατε ποτέ:</b> Ήπτερο ή Ηπατίτιδα	<input type="checkbox"/>	<input type="checkbox"/>	Γονιθθήκατε ή ζήσατε ή ταξιδέματε στο εξωτερικό;	<input type="checkbox"/>	<input type="checkbox"/>
Σύφιλη	<input type="checkbox"/>	<input type="checkbox"/>	Χάσατε βάρος - έχετε πυρετό ή αδένες διογκωμένους;	<input type="checkbox"/>	<input type="checkbox"/>
Ελονοσία	<input type="checkbox"/>	<input type="checkbox"/>	Έχετε υποστεί ποτέ μεταμόσχευση κερατοειδούς ή σκληρού χιτώνα στο μάτι σας;	<input type="checkbox"/>	<input type="checkbox"/>
Φυματίωση	<input type="checkbox"/>	<input type="checkbox"/>	Έχετε ακούσει ότι στην οικογένεια σας υπάρχει κίνδυνος εμφάνισης της νόσου Creutzfeldt - Jakob (νόσος τρελών αγελάδων);	<input type="checkbox"/>	<input type="checkbox"/>
Ρευματοειδή αρθρίτιδα	<input type="checkbox"/>	<input type="checkbox"/>	Λήψη αχολυμάτων αυξητικής ορμόνης	<input type="checkbox"/>	<input type="checkbox"/>
Καρδιοπάθεια	<input type="checkbox"/>	<input type="checkbox"/>	<b>ΚΑΝΑΤΕ:</b>		
Προκαρδιαούς πόντους	<input type="checkbox"/>	<input type="checkbox"/>	Εξουληγή ή θεραπεία δοντού την τελευταία εβδομάδα;	<input type="checkbox"/>	<input type="checkbox"/>
Υπέρταση	<input type="checkbox"/>	<input type="checkbox"/>	Εμβόλια την τελευταία εβδομάδα;	<input type="checkbox"/>	<input type="checkbox"/>
Σπασμούς (ως ενήλικας)	<input type="checkbox"/>	<input type="checkbox"/>	Εγχείρηση ή ιατρικός εξτόσας το χρόνο που πήρατε;	<input type="checkbox"/>	<input type="checkbox"/>
Αποθυμίες	<input type="checkbox"/>	<input type="checkbox"/>	Μεταγγιση αίματος ή παραγώγων αίματος τον τελευταίο χρόνο;	<input type="checkbox"/>	<input type="checkbox"/>
Παθήσας στομάχου	<input type="checkbox"/>	<input type="checkbox"/>	Τατουάζ ή τρύπημα αυτών ή βελονισμό;	<input type="checkbox"/>	<input type="checkbox"/>
Έλκος	<input type="checkbox"/>	<input type="checkbox"/>	Τρυπηθήκατε από βελόνα σύριγγας;	<input type="checkbox"/>	<input type="checkbox"/>
Άλλες συχαρήσας	<input type="checkbox"/>	<input type="checkbox"/>	Κάποια πληγή ή αμυγή του δέρματός σας ή βλεονόγος του στόματός σας ήρθε σε επαφή με ξένο αίμα;	<input type="checkbox"/>	<input type="checkbox"/>
Παθήσας των νεφρών	<input type="checkbox"/>	<input type="checkbox"/>	Μήνας μέσα στο χρόνο που πήρατε ήσαστε έγκυος;	<input type="checkbox"/>	<input type="checkbox"/>
Διαβήτη	<input type="checkbox"/>	<input type="checkbox"/>			
Αλλεργία	<input type="checkbox"/>	<input type="checkbox"/>			
Αναμία	<input type="checkbox"/>	<input type="checkbox"/>			
Άλλα νοσήματα	<input type="checkbox"/>	<input type="checkbox"/>			

### ΠΡΟΣΟΧΗ! - ΠΟΙΟΣ ΔΕΝ ΠΡΕΠΕΙ ΝΑ ΔΩΣΕΙ ΑΙΜΑ

- Όποιος έχει έστω και μία ομοφυλοφιλική σχέση από το 1977.
  - Όποιος έχει κάνει κάποια ενδοφλέβια χρήση ναρκωτικών.
  - Όποιος είχε πολλούς ερωτικούς συντρόφους χωρίς τη συστηματική χρήση προφυλακτικών μέσων, κατά τη τελευταία δεκαετία.
  - Ερωτικοί σύντροφοι πολλαπλασιάζομενων στόμων.
  - Όποιος είχε τους τελευταίους 12 μήνες σεξουαλική επαφή με σύντροφο που πληρωνόταν είτε με χρίματα είτε με ναρκωτικό για να κάνει σεξ.
  - Γενικό όποιος νομίζει ότι υπάρχει πιθανότητα να έχει εκτεθεί στον ιο που προκαλεί AIDS.
- Αν κάτι από τα παραπάνω σας απασχολεί, μπορείτε να το συζητήσετε με τον ιατρό της Αιμοδοσίας, **ΜΗ ΓΙΝΕΤΕ ΟΜΩΣ ΑΙΜΟΔΟΤΗΣ**

Διόρθωσι την αιμολεμμία, απάντηση ειλικρινά στις ερωτήσεις και έλαβο γνώση των αναφερομένων πληροφοριών. Βεβαιώνω ότι η αίμα μου, είναι στη διάθεση της Υπηρεσίας Αιμοδοσίας για κάθε ασθενή που θα το χραιζέται.

Ο/Η Αιμοδότης

Ο/Η ενεργήσας την εξέταση Ιατρός

## **Παράρτημα Β**

### **Διασύνδεση Κέντρων Αίματος και Νοσοκομειακών Υπηρεσιών Αιμοδοσίας**



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΥΠΟΥΡΓΕΙΟ ΥΓΕΙΑΣ  
ΓΕΝΙΚΗ Δ/ΝΣΗ ΥΠΗΡΕΣΙΩΝ ΥΓΕΙΑΣ  
ΔΙΕΥΘΥΝΣΗ ΑΝΑΠΤΥΞΗΣ ΜΟΝ. ΥΓΕΙΑΣ  
ΤΜΗΜΑ Γ΄

Ταχ. Δ/νση: Αριστοτέλους 17  
Ταχ. Κώδικας: 101 87  
Τηλέφωνο : 2105236476  
FAX: 2105233434

ΑΝΑΡΤΗΤΕΟ ΣΤΟ ΔΙΑΔΙΚΤΥΟ

Αθήνα 26.11.2013  
Αριθμ. Πρωτ. Υ4γ/Γ.Π.οικ.109074

Προς: Όπως ο Π.Δ.

Θέμα: Διασύνδεση Κέντρων Αίματος και Νοσοκομειακών Υπηρεσιών Αιμοδοσίας.

- Σχετ.: 1. Το αριθμ. ΓΠ/11/3300/12-11-12 έγγραφο του Εθνικού Κέντρου Αιμοδοσίας.  
2. Η αριθμ. Υ4γ/Γ.Π.οικ.120993/10-12-12 απόφαση έγκρισης σκοπιμότητας για την προμήθεια αντιδραστηρίων αιμοδοσίας για το μοριακό έλεγχο του αίματος.  
3. Η αριθμ. Υ4γ/Γ.Π.οικ.120990/10-12-12 απόφαση έγκρισης σκοπιμότητας για την μεταφορά αίματος, προϊόντων αίματος και των συνοδευτικών τους δειγμάτων.  
4. Το αριθμ. Υ4γ/Γ.Π.οικ.106270/18-11-2013 έγγραφο του Υπουργείου μας.

Κατόπιν των ανωτέρω σχετικών και δεδομένου ότι ο μοριακός έλεγχος του αίματος θα πραγματοποιείται σε τέσσερα (4) πλέον Κέντρα Αίματος της χώρας ήτοι: α) Ε.ΚΕ.Α., β) Π.Γ.Ν.Θ. «ΑΧΕΠΑ», γ) Π.Γ.Ν. Πατρών «Παναγία η Βοήθεια» και δ) Π.Γ.Ν. ΗΡΑΚΛΕΙΟΥ-Γ.Ν. «ΒΕΝΙΖΕΛΕΙΟ» (νοσοκομειακή μονάδα «ΒΕΝΙΖΕΛΕΙΟ»), η διασύνδεση των Νοσοκομειακών Υπηρεσιών Αιμοδοσίας με τα εν λόγω κέντρα έχει ως ακολούθως:

#### **Εθνικό Κέντρο Αιμοδοσίας (Ε. ΚΕ.Α)**

1. Γ.Ν Νίκαιας Πειραιά «ΑΓΙΟΣ ΠΑΝΤΕΛΕΗΜΩΝ»-Γ.Ν.Δυτ. Αττικής «Η ΑΓΙΑ ΒΑΡΒΑΡΑ»
2. Γ. Αντικαρκινικό Νοσοκομείο Πειραιά «ΜΕΤΑΞΑ»
3. Γ.Ν. Πειραιά «ΤΖΑΝΕΙΟ»
4. Γ.Ν. Ελευσίνας «ΘΡΙΑΣΙΟ»
5. Π.Γ.Ν. «ΑΤΤΙΚΟΝ»
6. Γ.Ν. Κορίνθου
7. Γ.Ν.Χαλκίδας-Γ.Ν.Κ.Υ. Καρύστου-Γ.Ν.Κ.Υ. Κύμης
8. Γ.Ν.Λιβαδειάς-Γ.Ν.Θηβών
9. Γ.Ν. Μυτιλήνης «ΒΟΣΤΑΝΕΙΟ»-Γ.Ν.Κ.Υ. Λήμνου
10. Γ.Ν. Χίου «ΣΚΥΛΙΤΣΕΙΟ».
11. Γ.Ν. Σάμου «ΑΓ. ΠΑΝΤΕΛΕΗΜΩΝ»-Γ.Ν.Κ.Υ. Ικαρίας
12. Γ.Ν. «ΚΟΡΓΙΑΛΕΝΕΙΟ - ΜΠΕΝΑΚΕΙΟ» Ε.Ε.Σ.
13. Π.Γ.Ν.Α. «Αρεταίειο»
14. 417 Νοσηλευτικό Ιδρυμα Μ.Τ.Σ. (ΝΙΜΤΣ)
15. 401 Γεν. Στρατιωτικό Νοσοκομείο Αθηνών
16. Ναυτικό Νοσοκομείο Αθηνών
17. 251 Γενικό Νοσοκομείο Αεροπορίας



18. Γ.Ν.Α. «Λαϊκό»
19. Γ.Ν.Α. «Γ.Γεννηματάς»
20. Γ.Ν. Αττικής «ΣΙΣΜΑΝΟΓΛΕΙΟ-ΑΜΑΛΙΑ ΦΛΕΜΙΓΚ»
21. Γ.Ν. Ν. Ιωνίας «ΚΩΝΣΤΑΝΤΟΠΟΥΛΕΙΟ»-Πατησίων
22. Γ.Ν.Παίδων Αθηνών «ΑΠΑ ΣΟΦΙΑ»- Παίδων Πεντέλης
23. Γ.Ν. Παίδων Αθηνών «Π. & Α. ΚΥΡΙΑΚΟΥ»
24. Γ.Ν. «ΕΛ. ΒΕΝΙΖΕΛΟΥ-ΑΛΕΞΑΝΔΡΑ»
25. Γ.Ν.Α. «Ο ΕΥΑΓΓΕΛΙΣΜΟΣ»-Οφθαλμιατρείο Αθηνών - Πολυκλινική
26. Γ.Ν. Αθηνών «ΙΠΠΟΚΡΑΤΕΙΟ»
27. Γ.Ν. Αθηνών «Η ΕΛΠΙΣ»
28. Γ.Ν. Αθηνών «Η ΠΑΜΜΑΚΑΡΙΣΤΟΣ»
29. Γ.Ν.Αττικής «ΚΑΤ»-Εθνικό Κέντρο Αποκατάστασης
30. Γ.Ν. Νοσημάτων Θώρακος Αθηνών «ΣΩΤΗΡΙΑ»
31. Γ. Αντικαρκινικό Νοσοκομείο Αθηνών «ΑΓΙΟΣ ΣΑΒΒΑΣ»
32. Γ.Ν. «ΑΣΚΛΗΠΕΙΟ ΒΟΥΛΑΣ»
33. Γ. Ογκολογικό Νοσοκομείο Κηφισιάς «ΟΙ ΑΓΙΟΙ ΑΝΑΡΓΥΡΟΙ»
34. Γ.Ν. Σύρου «ΒΑΡΔΑΚΕΙΟ & ΠΡΩΪΟ»-Γ.Ν.Κ.Υ. Νάξου
35. Γ.Ν. Ρόδου-Γ.Ν.Κ.Υ. Κώ-Γ.Ν.Κ.Υ. Καλύμνου
36. Γ.Ν. Κέρκυρας
37. Γ.Ν. Καρπενησίου
38. Γ.Ν. Αμφισσας
39. Γ.Ν.Ζακύνθου
40. Γ.Ν.Κεφαλληνίας
41. Γ.Ν. Λαμίας

**Πανεπιστημιακό Γ.Ν. Θεσσαλονίκης «ΑΧΕΠΑ»**

1. Γ.Ν.Θ. «ΙΠΠΟΚΡΑΤΕΙΟ»
2. Αντικαρκινικό Νοσοκομείο Θεσσαλονίκης «ΘΕΑΓΕΝΕΙΟ»
3. Γ.Ν. Θεσσαλονίκης «Γ. ΠΑΠΑΝΙΚΟΛΑΟΥ»
4. Πρώτο Γ.Ν. Θεσσαλονίκης «ΑΓΙΟΣ ΠΑΥΛΟΣ»
5. Γ.Ν. Χαλκιδικής
6. Γ.Ν. Φλώρινας
7. Γ.Ν. Καστοριάς
8. Γ.Ν. «Μαμάτσειο»-«Μποδοσάκειο»
9. Γ.Ν. Ημαθίας
10. Γ.Ν. Κιλκίς
11. Γ.Ν. Πέλλας
12. Γ.Ν. Θεσσαλονίκης «Γ.Γεννηματάς»-«Ο Αγ. Δημήτριος»
13. Γ.Ν. Ξάνθης
14. Π.Γ.Ν. Εβρου
15. Γ.Ν Σερρών
16. Γ.Ν. Καβάλας
17. Γ. Ν. Δράμας
18. Γ.Ν. Κομοτηνής
19. Π.Γ.Ν. Λάρισας-Γ.Ν.Λάρισας “Κουτλιμπάνειο & Τριανταφύλλειο»
20. Γ.Ν. Βόλου «Αχιλλοπούλειο»
21. Γ.Ν. Τρικάλων
22. Γ.Ν. Καρδίτσας

23. Γ.Ν. Κατερίνης
24. Γ.Ν. Γρεβενών

**Πανεπιστημιακό Γ.Ν. Πατρών «Παναγία η Βοήθεια»**

1. Γ.Ν Πατρών
2. Γ.Ν. Ανατολικής Αχαΐας
3. Γ.Ν. Ηλείας
4. Γ.Παναρκαδικό Ν. Τρίπολης «Η Ευαγγελίστρια»
5. Γ.Ν. Λακωνίας
6. Γ.Ν. Μεσσηνίας
7. Γ.Ν.Αργολίδας
8. Γ.Ν. Αιτωλοακαρνανίας
9. Γ.Ν. Ιωαννίνων “Γ.ΧΑΤΖΗΚΩΣΤΑ”
10. Π.Γ.Ν. Ιωαννίνων
11. Γ.Ν. Αρτας
12. Γ.Ν. Πρέβεζας
13. Γ.Ν.-Κ.Υ. Φιλιατών
14. Γ.Ν. Λευκάδας

**Πανεπιστημιακό Γ.Ν. Ηρακλείου - Γ.Ν. «ΒΕΝΙΖΕΛΕΙΟ» (Νοσοκομειακή Μονάδα Γ.Ν. “ΒΕΝΙΖΕΛΕΙΟ”**

1. Π.Γ.Ν. Ηρακλείου-Γ.Ν. Ηρακλείου «ΒΕΝΙΖΕΛΕΙΟ» (Νοσοκομειακή μονάδα Π.Γ.Ν. Ηρακλείου)
2. Γ.Ν. Χανίων
3. Γ.Ν. Ρεθύμνης
4. Γ.Ν. Αγ. Νικολάου-Γ.Ν.Κ.Υ. Ιεράπετρας-Γ.Ν.Κ.Υ.Σητείας-Γ.Ν.Κ.Υ.Νεαπόλεως»

**ΠΙΝΑΚΑΣ ΔΙΑΝΟΜΗΣ**

- |  |              |
|--|--------------|
| 1. Όλες τις Υγειονομικές Περιφέρειες της χώρας           | Ο ΥΦΥΠΟΥΡΓΟΣ |
| 2. Όλα τα Νοσηλευτικά Ιδρύματα της χώρας                 |              |
| 3. Ε.ΚΕ.Α.<br>Ολυμπιονίκη Χρ. Μάντικα 7<br>13671 Αχαρνάι | ΑΝΤ. ΜΠΕΖΑΣ  |
| 4. Αρεταίειο Νοσοκομείο                                  |              |
| 5. 417 Νοσηλευτικό Ίδρυμα ΜΤΣ (ΝΙΜΤΣ)                    |              |
| 6. 401 Γ. Στρατιωτικό Νοσοκομείο Αθηνών                  |              |
| 7. Ναυτικό Νοσοκομείο Αθηνών                             |              |
| 8. 251 Γενικό Νοσοκομείο Αεροπορίας                      |              |

**ΕΣΩΤΕΡΙΚΗ ΔΙΑΝΟΜΗ**

- 1.Γραφ. κ. Υπουργού
- 2.Γραφ. κας Υφυπουργού
- 3.Γραφ. κ. Γεν. Γραμματέα
- 4.Γραφ. κ. Γεν. Δ/ντή Υπ. Υγείας
- 5.Δ/νση Ανάπτυξης Μονάδων Υγείας