



# ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

**‘Ηλεκτρονικό Ευρετήριο Φαρμακείων και Φαρμάκων στο  
μελλοντικό διαδίκτυο (Future Internet) με την χρήση  
τεχνολογιών προσανατολισμένων σε υπηρεσίες (service  
oriented).’**

**ΣΥΓΓΡΑΦΗ: ΜΑΝΤΖΙΩΡΗ ΕΛΕΝΗ (ΜΕ/08087)**  
**ΤΜΗΜΑ: ΔΙΚΤΥΟΚΕΝΤΡΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**  
**ΕΞΑΜΗΝΟ Δ΄: ΕΚΠΟΝΗΣΗ ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ**  
**ΕΠΙΒΛΕΨΗ: Δρ. ΒΕΡΑ – ΑΛΕΞΑΝΔΡΑ ΣΤΑΥΡΟΥΛΑΚΗ**

Μάρτιος 2012

## Τίτλος

*‘Ηλεκτρονικό Ευρετήριο Φαρμακείων και Φαρμάκων στο μελλοντικό διαδίκτυο (Future Internet) με την χρήση τεχνολογιών προσανατολισμένων σε υπηρεσίες (service oriented).’*



INTERIOR OF AN APOTHECARY'S SHOP.  
Late XIV. or Early XV. Century. Flemish.  
(From an Old Printing.)

Μάρτιος 2012

## Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια της διπλωματικής μου εργασίας κα. Βέρα Σταυρουλάκη, για την καθοδήγησή της και την πολύτιμη συμβολή της σε κάθε φάση της δημιουργίας της. Επίσης, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου και στους εργοδότες μου, που μου συμπαράσταθηκαν και με στήριξαν όχι μόνο κατά τη διάρκεια της εκπόνησης της πτυχιακής μου εργασίας αλλά και καθ' όλη τη διάρκεια των σπουδών μου στο Μεταπτυχιακό πρόγραμμα, έτσι ώστε να καταφέρω να επεκτείνω τις γνώσεις μου.

## Πρόλογος

Το βιβλίο αυτό αποτελεί την ανάλυση και την τεκμηρίωση της εφαρμογής Apothecary που αναπτύχθηκε στα πλαίσια της διπλωματικής εργασίας που υποχρεούται ο κάθε μεταπτυχιακός φοιτητής του τμήματος Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς να εκπονήσει.

Η εφαρμογή Apothecary αφορά την ανάπτυξη ενός πληροφοριακού συστήματος διαχείρισης και αναζήτησης φαρμακείου καθώς και φαρμάκου.

Το κείμενο αυτό απαρτίζεται από τέσσερα (4) κύρια μέρη. Στο πρώτο μέρος παρατίθενται αναφορές για όλες τις εφαρμογές που έχουν ήδη υλοποιηθεί και αφορούν το συγκεκριμένο αντικείμενο. Επιπλέον, έχουμε παραθέσει και όλες τις βελτιώσεις-επεκτάσεις που μπορούν να γίνουν ώστε να βελτιωθούν οι εν λόγω εφαρμογές, τις οποίες προσπαθήσαμε να τις υλοποιήσουμε στην εφαρμογή Apothecary (Κεφάλαιο 1).

Στο δεύτερο μέρος του βιβλίου αναλύονται οι βασικότερες τεχνολογίες αλλά και εργαλεία που χρησιμοποιήσαμε για να υλοποιήσουμε την εφαρμογή Apothecary (Κεφάλαια 2, 3).

Στο τρίτο μέρος παρέχεται η ανάλυση που έγινε πριν την υλοποίηση της εφαρμογής. Σε αυτό το μέρος παραθέτουμε τις απαιτήσεις, τα διαγράμματα χρήσης και όσα άλλα διαγράμματα μας βοήθησαν να αποφασίσουμε ποιες θα είναι οι προδιαγραφές της εφαρμογής που πρόκειται να αναπτύξουμε και ποια θα είναι η δομή της βάσης δεδομένων που θα χρησιμοποιήσουμε (Κεφάλαιο 4, 5).

Τέλος, στο τέταρτο μέρος παρουσιάζεται το αποτέλεσμα της υλοποίησης, δηλαδή, η τελική μορφή και δομή της εφαρμογής Apothecary. Στο σημείο αυτό παραθέτουμε όλα τα επιμέρους τμήματα της εφαρμογής που υλοποιήθηκαν καθώς και ένα αναλυτικό εγχειρίδιο χρήσης στο οποίο παρουσιάζονται οι βασικές οθόνες του πληροφοριακού συστήματος με οδηγίες χρήσης για κάθε μία από τις κατηγορίες χρηστών που υποστηρίζει η εφαρμογή (διαχειριστής συστήματος, Υπεύθυνος Φαρμακείου, Υπάλληλος Φαρμακείου). Επίσης γίνεται αναφορά στο πώς θα μπορούσαν να ενσωματωθούν RFID στην υλοποίηση και πόσο θα βοηθούσε αυτό στην καλύτερη εφαρμογή του Apothecary στην αγορά. (Κεφάλαιο 6, 7, 8).

# Περιεχόμενα

Στόχος και αντικείμενο της εργασίας - Περίληψη .....	6
1. Όσα υπάρχουν στην αγορά και όσα υλοποιήθηκαν στο Apothecary .....	7
2. RFID τεχνολογία .....	9
3. Τεχνολογίες Υλοποίησης .....	12
3.1. WEB Services .....	12
3.2. JavaScript.....	18
3.3. JQuery .....	18
3.4. jqGrid .....	19
3.5. AJAX .....	19
3.6. GoogleMaps (JavaScript V3)[14] .....	20
3.7. Google Chart API.....	20
3.8. C# .NET .....	21
4. Σχεδιασμός Μοντελοποίηση Λύσης (Use cases – οντοτήτων συσχετίσεων – message sequence charts) ....	22
4.1. Use Case και Activity Diagrams .....	22
4.1.1. Βασικοί Πρωταγωνιστές του Συστήματος .....	22
4.1.2. Πρότυπο Σεναρίου Χρήσης.....	25
4.1.3. Λειτουργικές Απαιτήσεις Συστήματος.....	26
4.1.4. Λειτουργίες Διαχειριστή συστήματος .....	29
4.1.5. Λειτουργίες Υπεύθυνου Φαρμακείου .....	40
4.1.6. Λειτουργίες Υπαλλήλου Φαρμακείου .....	43
4.1.7. Λειτουργίες Ασθενή – Αναζητητή (Εφαρμογή Αναζήτησης Μέσω Διαδικτύου) .....	46
5. Εγχειρίδιο Βάσης.....	53
5.1 Περιγραφή προβλήματος .....	53
5.2 Κατάλογος Περιγραφής Οντοτήτων .....	53
5.3 Χαρακτηριστικά Οντοτήτων.....	58
6. Υλοποίηση (Screenshots).....	66
6.1 WS1 : PharmacyWebService1 .....	66
6.1.1 Μέθοδος 1: Login.....	66
6.1.2 Μέθοδος 2: InsertPharmacy.....	66
6.1.3 Μέθοδος 3: InsertEmployee.....	66
6.1.4 Μέθοδος 4: InsertMedicine.....	66
6.1.5 Μέθοδος 5: GetEmployeeCharacteristics.....	67
6.1.6 Μέθοδος 6: GetMedicine .....	67
6.1.7 Μέθοδος 7: ReceiptOfGoodsInStore.....	68
6.2 AvailablePharmacyWS.....	69

6.2.1	Μέθοδος 1: GetAvailablePharmacies.....	69
6.2.2	Μέθοδος 2: GetListOfMatchMedicines.....	70
6.3	ControllerEmployees.....	70
6.3.1	Μέθοδος 1: GetEmployeeList.....	70
6.3.2	Μέθοδος 2: DeleteEmployee.....	70
6.4	ControllerMedicines.....	70
6.4.1	Μέθοδος 1: GetMedicineList.....	70
6.4.2	Μέθοδος 2: DeleteMedicine.....	72
6.5	ControllerPharmacies.....	72
6.5.1	Μέθοδος 1: GetPharmacyList.....	72
6.5.2	Μέθοδος 2: DeletePharmacy.....	72
6.6	PharmacyController.....	72
6.6.1	Μέθοδος 1: GetOpenPharmacies.....	72
6.6.2	Μέθοδος 2: GetListOfMatchMedicines.....	72
6.6.3	Μέθοδος 3: MedicineAvail.....	72
6.7.	Τεχνικές κλήσης των Web service που χρησιμοποιήθηκαν.....	74
6.8.	Υλοποίηση Google Charts στην εφαρμογή Apothecary.....	76
7.	Εκτίμηση απόδοσης της λύσης (πλεονεκτήματα).....	78
7.1	Οφέλη του πληροφοριακού συστήματος αναζήτησης φαρμάκων.....	78
7.1.1	Για τον πελάτη:.....	78
7.1.2	Για τον ιδιοκτήτη του φαρμακείου ο οποίος είναι καταχωρημένος και χρησιμοποιεί το πληροφοριακό σύστημα:.....	78
7.2	Οφέλη RFID readers για το φαρμακείο.....	78
8.	Εγχειρίδιο χρήσης.....	80
8.1	Εργαλείο φαρμακοποιών και διαχειριστή του συστήματος.....	80
8.1.1	Εισαγωγή στο σύστημα (Login).....	80
8.1.2	Διαχειριστής του Συστήματος.....	81
8.1.3	Υπεύθυνος Φαρμακείου.....	84
8.1.4	Υπάλληλος Φαρμακείου.....	85
8.2	Εφαρμογή Αναζήτησης Φαρμάκου και Φαρμακείου Μέσω Διαδικτύου.....	85
9.	Εφαρμογή των RFIDs στο πληροφοριακό σύστημα Apothecary.....	89
10.	Παραδοτέα.....	91
11.	Συντομογραφίες.....	92
12.	Αναφορές.....	93

## Στόχος και αντικείμενο της εργασίας - Περίληψη

Το πιο σημαντικό θέμα στη ζωή κάθε ανθρώπου είναι η υγεία του καθώς και των αγαπημένων του προσώπων. Δυστυχώς όμως, η καθημερινότητα έχει γίνει πολύ δύσκολη και ο ελεύθερος χρόνος που έχει καθένας μας είναι πολύ περιορισμένος. Απόρροια όλων αυτών είναι να αφιερώνει ολοένα και λιγότερο χρόνο στα θέματα της υγείας του.

Ακόμη όμως και αν βρεθεί ο απαραίτητος χρόνος, διάφορες συγκυρίες, κυρίως στη χώρα μας π.χ. το περιορισμένο ωράριο των φαρμακείων, κάνουν την αναζήτηση και την αγορά των φαρμάκων δύσκολη. Η αναζήτηση φαρμακείων γίνεται μέσω αρκετών ιστοσελίδων, μέσω εφημερίδων καθώς και από τηλεφώνου. Αντίθετα όμως, η αναζήτηση διαθεσιμότητας των φαρμάκων στα φαρμακεία είναι πολύ πιο δύσκολη.

Το παραπάνω πρόβλημα μας οδήγησε στην σύλληψη της ιδέας Apothecary. Η εφαρμογή αυτή σχεδιάστηκε για να κάνει πιο εύκολη και γρήγορη την αναζήτηση των ζητούμενων φαρμάκων και των φαρμακείων στα οποία αυτά είναι διαθέσιμα μια δεδομένη ώρα σε μια δεδομένη περιοχή.

Για να λειτουργήσει η εφαρμογή Apothecary, θα πρέπει οι φαρμακοποιοί να καταχωρούν κάθε εισαγωγή προϊόντων στην αποθήκη τους καθώς και κάθε εξαγωγή προκειμένου τα διαθέσιμα αποθέματα να εμφανίζονται στους ασθενείς/πελάτες. Με την παραπάνω διαδικασία, όχι μόνο θα αυξήσουν τις πωλήσεις τους, αλλά θα μπορούν να ελέγχουν και το απόθεμα κάθε προϊόντος, ώστε όταν αυτό βρίσκεται σε έλλειψη ή είναι μικρό το απόθεμα του (π.χ. 5 εναπομείναντα τεμάχια), να ενημερώνεται για να προχωρήσει σε παραγγελία.

Για να γίνει πιο εύκολη η καταχώρηση των φαρμάκων στην εφαρμογή, δίνεται η δυνατότητα στους φαρμακοποιούς να χρησιμοποιήσουν RFIDs (Radio Frequency Identification), αλλιώς στα ελληνικά «ταυτοποίηση μέσω ραδιοσυχνοτήτων», μέσω των οποίων αναγνωρίζονται αυτόματα όλα τα προϊόντα που εισάγονται στην αποθήκη, και καταχωρούνται στην βάση δεδομένων της εφαρμογής.

Η ονομασία Apothecary, η οποία, επιλέχτηκε για αυτό το πληροφοριακό σύστημα, είναι ιστορικής σημασίας, καθώς με τον όρο αυτό χαρακτηρίζαν στο παρελθόν τον άνθρωπο του ιατρικού κλάδου, ο οποίος, παρασκεύαζε και διένειμε φαρμακευτικά είδη σε γιατρούς, χειρουργούς και ασθενείς. Ο όρος αυτός έχει πλέον καταργηθεί και το ρόλο του τον έχουν πάρει οι φαρμακοποιοί.

# 1. Όσα υπάρχουν στην αγορά και όσα υλοποιήθηκαν στο Apothecary.

Με την ανάπτυξη του διαδικτύου, όλο και περισσότερες εφαρμογές δημιουργούνται και δημοσιεύονται σε αυτό που αφορούν την αναζήτηση φαρμακείων. Στο διαδίκτυο είναι αναρτημένες αρκετές εφαρμογές για την αναζήτηση διαθέσιμου φαρμακείου και κυρίως για τις εφημερίες αυτών. Κάποιες από αυτές εμφανίζουν τα σημεία ενδιαφέροντος αυτά στον χάρτη, ενώ άλλες δίνουν χρήσιμες συμβουλές υγείας. Αυτό που δεν υπάρχει ακόμη, τουλάχιστον για τα φαρμακεία της Ελλάδας, είναι να μπορεί ο ασθενής να ενημερωθεί για την διαθεσιμότητα κάποιου επιθυμητού φαρμάκου στα φαρμακεία αυτά. Για να γίνει αυτό πρέπει να συνδεθεί η αποθήκη των φαρμακείων στο σύστημα έτσι ώστε να είναι διαθέσιμη και η πληροφορία του αποθέματος αυτών.

Στην εφαρμογή Apothecary λοιπόν, υλοποιήθηκε και αυτή η πρόσθετη παροχή προς τον ασθενή. Ο χρήστης λοιπόν θα μπορεί να βλέπει διαθεσιμότητα για όλα τα φαρμακεία της Αττικής καθώς και τη διαθεσιμότητα συγκεκριμένου φαρμάκου, μόνο όμως σε όσα από τα φαρμακεία έχουν συνδέσει την αποθήκη τους με την εφαρμογή. Η σύνδεση αυτών των φαρμακείων στην εφαρμογή δίνει και κάποιες πρόσθετες παροχές σε αυτό. Ο υπεύθυνος του φαρμακείου μπορεί να έχει πρόσβαση στο σύστημα διαχείρισης για να καταχωρεί τις νέες παραλαβές καθώς και να δει κάποια στατιστικά που παρέχονται από το σύστημα και έχουν να κάνουν με τις αναζητήσεις για το φαρμακείο αυτό αλλά και για τα φάρμακα. Για παράδειγμα μπορεί να δει τα δημοφιλέστερα φάρμακα έτσι ώστε να προετοιμαστεί κατάλληλα σύμφωνα με την ζήτηση.

Η εφαρμογή Apothecary έχει τη δυνατότητα να επεκταθεί και να συνδεθεί με λογιστικά ή άλλα προγράμματα που ήδη έχουν τα φαρμακεία έτσι ώστε να γίνει όλο και πιο εύκολη η καταχώρηση νέων εμπορευμάτων καθώς και η μείωση αυτών μέσω της πώλησης.

Τα τελευταία χρόνια, η φαρμακοβιομηχανίες χρησιμοποιούν και ασχολούνται με τη νέα γενιά των γραμμωτών κωδικών (barcodes), την τεχνολογία RFID, ή τεχνολογία αυτόματης αναγνώρισης προϊόντων μέσω ραδιοσυχνότητας. Η τεχνολογία RFID αναγνωρίζει αυτόματα τα προϊόντα που φέρουν αυτόν τον τύπο κωδικών, χωρίς να απαιτείται οπτική επαφή της συσκευής με το προϊόν. Παράλληλα, επιτρέπει την ταυτόχρονη αναγνώριση πολλών διαφορετικών προϊόντων, με αποτέλεσμα την επιτάχυνση όλων των διαδικασιών στην εφοδιαστική αλυσίδα. (Τριανταφυλλάκης, 2005).



1.1 Η εφοδιαστική αλυσίδα των φαρμάκων (*"The Global Traceability Standard Supporting Visibility, Quality & Safety In The Supply Chain"*)[2]



Η καταχώρηση, επίσης, RFID reader στο ταμείο καθώς και στην αποθήκη του φαρμακείου, όπως προτείνεται από την εργασία, θα διευκόλυνε όχι μόνο τις πωλήσεις αλλά και των έλεγχο και την ενημέρωση της αποθήκης και γενικά τον έλεγχο διακίνησης του εμπορεύματος. Οι readers, επιτρέπουν την ταυτόχρονη αναγνώριση πολλών προϊόντων, με αποτέλεσμα την επιτάχυνση όλων των διαδικασιών στην εφοδιαστική αλυσίδα.

Για την υλοποίηση της εφαρμογής Apothecary χρησιμοποιήθηκαν νέες σύγχρονες τεχνικές και προγραμματιστικά εργαλεία τα οποία σκοπεύουν αύξηση των επιδόσεων όπως Web Services, JQuery, web services, Google Maps και άλλα. Οι νέες τεχνολογίες που χρησιμοποιήθηκαν έχουν ως αποτέλεσμα πιο γρήγορες σελίδες, συνεπώς και πιο γρήγορες αναζητήσεις.

## 2. RFID τεχνολογία

**Radio Frequency Identification (RFID)** είναι τεχνολογία που επιτρέπει την αυτόματη αναγνώριση αντικειμένων. Βασίζεται στην αποθήκευση και ασύρματη ανάκτηση δεδομένων έμψυχων και άψυχων αντικειμένων μέσω μικροσκοπικών συσκευών, τα ονομαζόμενα **RFID tags** ή **transponders**.

Η αναγνώριση μέσω ραδιοσυχνοτήτων (συχνά αναφέρεται και ως «ραδιοσυχνική ταυτοποίηση») εξελίσσεται στις μέρες μας σε μια σημαντική τεχνολογία που καθιστά εφικτή την αναγνώριση, την ταυτοποίηση και την ιχνηλασιμότητα φυσικών αντικειμένων, όπως προϊόντων ή άλλων περιουσιακών στοιχείων.

Τοποθετούνται σε προϊόντα, ζώα ή ακόμη και σε ανθρώπους. Περιλαμβάνουν μικροσκοπικά chips με κεραίες, τα οποία τους επιτρέπουν να λαμβάνουν σήματα και να ανταποκρίνονται σε ραδιοσυχνότητες, οι οποίες εκπέμπονται από έναν RFID πομποδέκτη.

Με ταυτόχρονη βελτίωση των αντίστοιχων διαδικασιών, δίνεται η δυνατότητα σε επιχειρήσεις για συνεχή και αξιόπιστη παρακολούθηση της θέσης, της κατάστασης αλλά και του ιστορικού των προϊόντων της. Ένα άλλο σημαντικό στοιχείο τους είναι ότι μπορούν να συνεργαστούν με είδη υπάρχοντα πληροφοριακά συστήματα και τελικά να προσφέρουν πλεονεκτήματα σε όλο το εύρος της εφοδιαστικής αλυσίδας καθώς και σε άλλες δραστηριότητες.[1]

### Ιστορικά

Η RFID τεχνολογία δεν είναι καινούρια. Το 1945 ο Leon Theremin ανακάλυψε ένα κατασκοπευτικό εργαλείο για τη Σοβιετική κυβέρνηση (έναν τύπο κοριού). Η συσκευή εξέπεμπε ραδιοκύματα με πληροφορίες συνομιλιών (audio information). Παρόλο που η συσκευή ήταν μία κρυφή παθητική συσκευή υποκλοπών και όχι ένα tag αναγνώρισης, θεωρείται ως ο πρόδρομος της τεχνολογίας RFID. Άλλοι αναφέρουν ότι τα πρώτα RFID συστήματα σχετίζονται με την τεχνολογία των Radars για εντοπισμό εχθρικών αεροπλάνων και ήταν γνωστά από το 1920.

Στον εμπορικό τομέα φαίνεται ότι πρωτοεμφανίστηκε το 1960 όπου εμφανίστηκαν και οι πρώτες αντίστοιχες εφαρμογές.

Το 1998 άρχισε μια νέα εποχή για τα RFIDs αφού τότε άρχισαν να γίνονται έρευνες στην παρακολούθηση και την αναγνώριση αντικειμένων καθώς αυτά κινούνται μεταξύ διάφορων φυσικών θέσεων. Η έρευνα αυτή επικεντρώθηκε σε ασύρματες τεχνολογίες.

Στις μέρες μας πλέον, υπάρχει ραγδαία εξέλιξη με παράλληλη μείωση κόστους του εξοπλισμού και την αύξηση της ασφάλειας.[2]

## Πως λειτουργεί

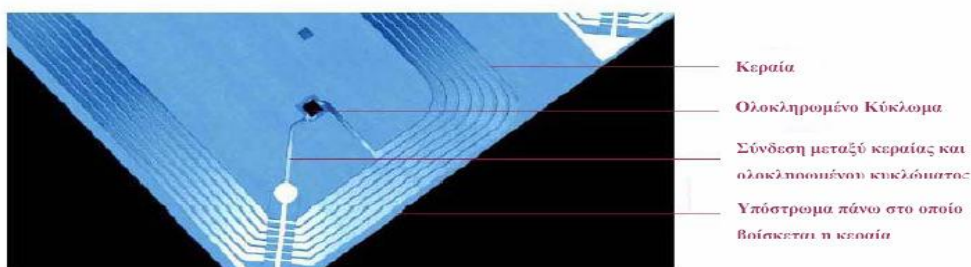
Ένα RFID αποτελείται από τον πομποδέκτη και τον αναγνώστη. Η πληροφορία ανταλλάσσεται ασύρματα μεταξύ ενός αντικειμένου που φέρει έναν πομποδέκτη και μια συσκευή ανάγνωσης/εγγραφής. Στην πραγματικότητα, ένα σύστημα RFID αποτελείται από έναν ή περισσότερους πομποδέκτες (tags), μια ή περισσότερες συσκευές ανάγνωσης (readers) και ένα εξειδικευμένο λογισμικό (middleware) εγκατεστημένο σε έναν κεντρικό υπολογιστή.

Ο αναγνώστης στέλνει σήμα (εκπέμπει ραδιοσυχνότητες) σε όσα Tag βρίσκονται στην εμβέλειά του. Αυτά το λαμβάνουν και εκπέμπουν πίσω τα δεδομένα που έχουν αποθηκευμένα. Ο reader τα παραλαμβάνει, τα αποκωδικοποιεί και τα στέλνει, μέσω καλωδίου ή ασύρματα, στο κεντρικό μηχάνημα με το λογισμικό όπου και αξιοποιούνται ανάλογα. Ο reader μπορεί, σε κάποιες περιπτώσεις, να αλλάξει τα αποθηκευμένα δεδομένα στα tag.[2]

## Τα επιμέρους κομμάτια

Το tag είναι ένα microchip με κεραία και συνήθως εμφανίζεται ως αυτοκόλλητο στο οποίο αναγράφονται κάποιες πληροφορίες (π.χ. barcode). Σε κάποιες περιπτώσεις, όπου απαιτείται αντοχή σε αντίξοες συνθήκες, είναι καλυμμένα με πλαστικό περίβλημα για μεγαλύτερη ανθεκτικότητα. Η κεραία του είναι συντονισμένη να λαμβάνει τα ηλεκτρομαγνητικά κύματα που εκπέμπονται από τον αναγνώστη (Reader). Όταν το παθητικό tag λάβει ένα κατάλληλο κύμα διεγείρεται και απαντά στέλνοντας μία απάντηση, η οποία λαμβάνεται από τον αναγνώστη (RFID reader).

Τα tags μπορεί να είναι Read only, Read – write, ή Write once – read many (WORM). Στην περίπτωση Read – write, τα δεδομένα του μπορούν να αλλάξουν με αντίστοιχη εντολή από τον reader που βρίσκεται στην εμβέλεια του. Στην περίπτωση write – once tag, τα περιεχόμενα αυτού γράφονται μια φορά και δεν μπορούν να αλλάξουν κατά την διάρκεια της ζωής τους.



2.1 Σύσταση Αυτοκόλλητου RFID Tag

Τα tags χωρίζονται σε παθητικά, ημιπαθητικά και σε ενεργητικά.

Τα παθητικά δεν περιέχουν στο εσωτερικό τους μπαταρία και για αυτό έχουν θεωρητικά άπειρο χρόνο ζωής, αφού δεν τελειώνει η μπαταρία. Επίσης, είναι και πιο μικρά από τα υπόλοιπα.

Τα ημιπαθητικά μοιάζουν με τα παθητικά, με μόνη διαφορά ότι περιέχουν και μια μικρή μπαταρία που τα κάνει να είναι πιο εύκολα αναγνώσιμα.

Τα ενεργητικά έχουν την δική τους μπαταρία οπότε έχουν και το δικό τους σήμα. Είναι μεγαλύτερα σε διάσταση από τα άλλα δύο, είναι ακριβότερα ενώ έχουν μεγαλύτερα μήμη και εμβέλεια.



2.2. Βασικά στοιχεία ενός RFID συστήματος [3]

Οι readers, είναι εξειδικευμένοι πομποδέκτες οι οποίοι χρησιμοποιούνται για την συλλογή των δεδομένων από τα RFID tags και την αποστολή τους στο κεντρικό μηχάνημα. Ανάλογα με την εφαρμογή που θέλουμε να εξυπηρετούν, μπορεί να είναι σταθεροί ή κινητοί. Η κεραία του μπορεί να είναι εσωτερική, εξωτερική ή να συνδέεται με εξωτερικές κεραίες, ανάλογα με την εμβέλεια που πρέπει να καλύψει κάθε φορά. Στην περίπτωση που ο αναγνώστης είναι σταθερός, μπορεί να συνδεθεί και ενσύρματα με το κεντρικό λογισμικό εκμετάλλευσης των δεδομένων. Αλλιώς μπορεί να επικοινωνεί ασύρματα μέσω WiFi, Bluetooth, ή μέσω κινητής τηλεφωνίας GSM\GPRS\3G κ.λ.π.



2.3 RFID Reader

Οι readers έχουν την δυνατότητα να διαβάσουν ταυτόχρονα μεγάλο αριθμό tags. Αυτό είναι πολύ χρήσιμο στην περίπτωση φόρτωσης – εκφόρτωσης, αφού μπορεί να διαβάσει, ανάλογα με το μέγεθός του, ακόμη και ολόκληρα container από προϊόντα με RFID tags. Το εξειδικευμένο ενδιάμεσο λογισμικό (middleware software), είναι το κομμάτι του RFID συστήματος το οποίο επεξεργάζεται τα δεδομένα που παίρνει από το reader και τα μετατρέπει με τον πιο

αποτελεσματικό τρόπο, ανάλογα με τις απαιτήσεις κάθε φορά της επιχείρησης. Το λογισμικό αυτό μπορεί να συνδέεται και με άλλα πληροφοριακά συστήματα στην επιχείρηση, έτσι ώστε να συμβάλει στην πλήρη αυτοματοποίησή της.[3]

## 3. Τεχνολογίες Υλοποίησης

### 3.1. WEB Services

Η τεχνολογία web services επιτρέπει, σε διαφορετικές εφαρμογές, να επικοινωνούν μεταξύ τους, ανεξάρτητα την πλατφόρμα ή τη γλώσσα προγραμματισμού με την οποία έχουν δημιουργηθεί.

Ως web service χαρακτηρίζεται μια διεπαφή λογισμικού (software interface), η οποία περιγράφει μια συλλογή από λειτουργίες που μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων XML.

Η τεχνολογία αυτή χρησιμοποιεί μηνύματα, σε γλώσσα XML, για να περιγράψει μία λειτουργία προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή. Μια ομάδα από web services, οι οποίες αλληλεπιδρούν μεταξύ τους, καθορίζει μια εφαρμογή web services.

#### **Βασικά πλεονεκτήματα και οφέλη**

- Εύκολος χειρισμός δεδομένων
- Απλό πρωτόκολλο επικοινωνίας
- Απλή υποδομή
- Μικρό κόστος δημιουργίας και χρήσης
- Ευκολία στην επικοινωνία
- Διαλειτουργικότητα και ευκολία ανάπτυξης νέων εφαρμογών
- Αλληλεπίδραση μεταξύ υπηρεσιών σε οποιαδήποτε πλατφόρμα και οποιαδήποτε γλώσσα προγραμματισμού.
- Χαλαρή συνδεσιμότητα μεταξύ εφαρμογών.
- Προσαρμογή ήδη υπάρχουσών εφαρμογών στις μεταβαλλόμενες επιχειρησιακές συνθήκες και ανάγκες των πελατών.

#### **Βασικά βήματα για την υλοποίηση ενός Web Service**

Από τον πάροχο της Υπηρεσίας :

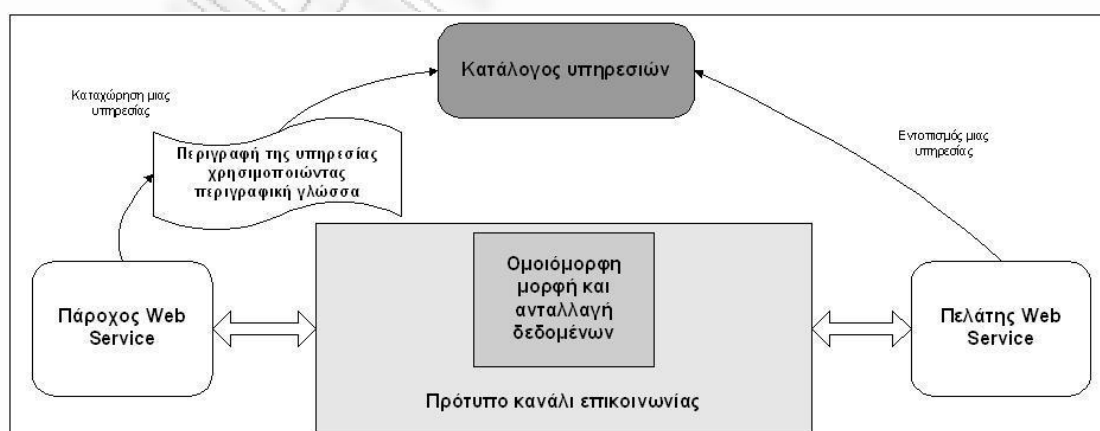
- Καθορισμός των υπηρεσιών που θα παρέχονται
- Υλοποίηση της λειτουργίας πίσω από την υπηρεσία
- Εγκατάσταση της εφαρμογής - παρόχου των υπηρεσιών
- Δημοσίευση των Web Services με την βοήθεια μιας υπηρεσίας καταλόγου
- Αναμονή για επεξεργασία αιτήσεων πελατών

Από τον Χρήστη της Υπηρεσίας :

- Προσδιορισμός των υπηρεσιών που θα απαιτηθούν
- Εντοπισμός του Web Service ρωτώντας μια υπηρεσία καταλόγου
- Αποστολή της αίτησης στην υπηρεσία
- Λήψη της απάντησης από την υπηρεσία

### Βασικές Τεχνολογίες των Web Services

Επίπεδο	Τεχνολογία
Ομοιόμορφη μορφή και ανταλλαγή δεδομένων	XML
Πρότυπο κανάλι επικοινωνίας	SOAP
Πρότυπη περιγραφική γλώσσα για την περιγραφή των παρεχόμενων υπηρεσιών	WSDL
Καταχώρηση και εντοπισμός των παρεχόμενων υπηρεσιών	UDDI



3.1 Αρχιτεκτονική των Web Services [5]

## eXtensible Markup Language (XML)

Ένα έγγραφο XML, στην πιο απλή του μορφή, είναι ένα αρχείο κειμένου το οποίο περιέχει δεδομένα μαζί με σήμανση η οποία καθορίζει τη δομή των δεδομένων

## Document Type Definition (DTD)

Το DTD είναι μία προδιαγραφή η οποία μπορεί να ακολουθηθεί, όταν δημιουργούμε ένα έγγραφο XML, για να εξασφαλίσουμε την ορθότητά του.

## Web Services Description Language (WSDL)

Οι εφαρμογές που παρέχουν Web Services διαφημίζουν τις υπηρεσίες που παρέχουν, χρησιμοποιώντας μια πρότυπη περιγραφική γλώσσα που ονομάζεται WSDL. Η WSDL βασίζεται στην XML και χρησιμοποιεί ένα σύνολο ετικετών για να περιγράψει ένα Web Service, τις υπηρεσίες που αυτό παρέχει (μεθόδους), το που μπορεί να εντοπιστεί για να καλεστεί και να χρησιμοποιηθεί (URL) κ.τ.λ. Έτσι οι προγραμματιστές, μπορούν να ενημερωθούν και να πάρουν πληροφορίες για ένα Web Service και πώς θα το καλέσουν πριν το χρησιμοποιήσουν σε μια εφαρμογή.

Ένα έγγραφο WSDL χρησιμοποιεί τα παρακάτω στοιχεία (elements) για τον ορισμό δικτυακών υπηρεσιών:

- **Types**
- **Message.**
- **Operation**
- **Port Type**
- **Binding**
- **Port**
- **Service**

## Universal Description, Discovery and Integration (UDDI)

Οι εφαρμογές που παρέχουν Web Services είναι διαθέσιμες και εμφανίζονται σε ένα κατάλογο από παρόχους υπηρεσιών, χρησιμοποιώντας το UDDI. Με τον ίδιο τρόπο οι προγραμματιστές - πελάτες εντοπίζουν τους παρόχους υπηρεσιών (Web Services) που επιθυμούν να χρησιμοποιήσουν χρησιμοποιώντας UDDI. Το UDDI βασίζεται και αυτό στην XML.

*“Τα web services έχουν λόγο ύπαρξης μόνο όταν, δυνητικοί χρήστες μπορούν να βρουν πληροφορίες ικανές ώστε να επιτρέψουν την εκτέλεσή τους.*

*Το Universal Description Discovery & Integration (UDDI) εστιάζει στον καθορισμό ενός συνόλου από υπηρεσίες που θα υποστηρίξουν την περιγραφή και την ανακάλυψη:*

- Των εταιριών, των οργανισμών και άλλων παρόχων web services
- Των web services, που είναι διαθέσιμες
- Και των τεχνικών διεπαφών οι οποίες μπορούν να χρησιμοποιηθούν ώστε να έχει κάποιος χρήστης πρόσβαση σε αυτές τις υπηρεσίες.

## Υπηρεσίες του UDDI

Πληροφορία	Λειτουργίες
<u>White pages</u> : Πληροφορίες όπως το όνομα, η διεύθυνση, το τηλέφωνο και άλλες πληροφορίες επικοινωνίας για μία επιχείρηση.	<u>Publish</u> : Πώς ο προμηθευτής ενός web service καταχωρεί των εαυτό του.
<u>Yellow Pages</u> : Πληροφορίες που κατηγοριοποιούν επιχειρήσεις. Βασίζονται σε υπάρχοντα πρότυπα κατηγοριοποίησης (μη ηλεκτρονικά).	<u>Find</u> : Πώς μία εφαρμογή βρίσκει ένα συγκεκριμένο web service.
<u>Green Pages</u> : Τεχνικές πληροφορίες για τα web services που παρέχονται από μία επιχείρηση.	<u>Bind</u> : Πώς μία εφαρμογή συνδέεται, και αλληλεπιδρά με ένα web service αφού αυτό βρεθεί.

Το SOAP είναι το κανάλι που χρησιμοποιείται για την επικοινωνία μεταξύ μιας εφαρμογής – Web Service και μιας εφαρμογής – πελάτη. Χαρακτηρίζεται από απλότητα γιατί, δεν καθορίζει κανένα νέο πρωτόκολλο μεταφοράς, αλλά στηρίζεται (επαναχρησιμοποιεί ) το HTTP ή το SMTP για την μεταφορά δεδομένων ως μηνύματα. Η χρήση αυτών των πρωτοκόλλων εξασφαλίζει ότι οι εφαρμογές – προμηθευτές με τις εφαρμογές – πελάτες μπορούν να επικοινωνήσουν μέσω του διαδικτύου.

*"Το SOAP στην έκδοση 1.2 είναι ένα ελαφρύ πρωτόκολλο προορισμένο για την ανταλλαγή δομημένων πληροφοριών σε ένα αποκεντρωμένο, διανεμημένο περιβάλλον. Χρησιμοποιεί τεχνολογίες XML για να καθορίσει ένα επεκτάσιμο πλαίσιο, παρέχοντας μια δομή μηνυμάτων η οποία μπορεί να ανταλλαχθεί πάνω από ποικίλα δικτυακά πρωτόκολλα. Το πλαίσιο έχει σχεδιαστεί να είναι ανεξάρτητο από οποιοδήποτε προγραμματιστικό μοντέλο και σημασιολογία υλοποίησης."*

### Χαρακτηριστικά του SOAP

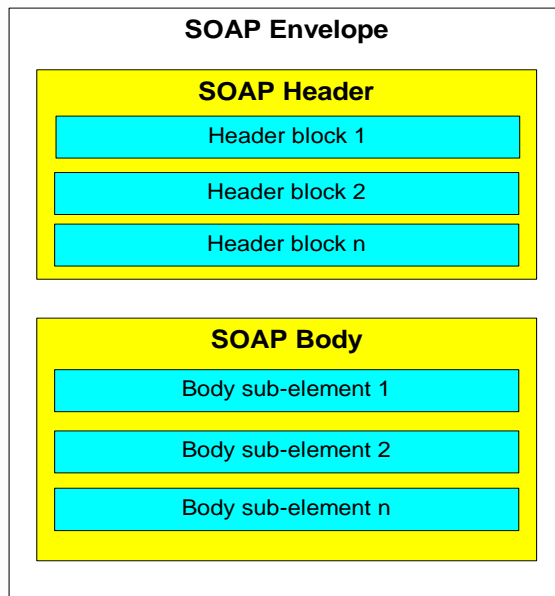
- **Απλότητα:** συνεπώς το κόστος και η πολυπλοκότητα υλοποίησης μειώνονται αισθητά.
- **Ανεξαρτησία:** Το SOAP είναι ανεξάρτητο από πλατφόρμα και γλώσσα προγραμματισμού, συνεπώς μπορεί να χρησιμοποιηθεί για επικοινωνία μεταξύ εφαρμογών γραμμένων για διαφορετικές πλατφόρμες και σε διαφορετικές γλώσσες προγραμματισμού.
- **Ευελιξία:** Χρησιμοποιεί πρότυπα πρωτόκολλα όπως το HTTP και το SMTP ως μέσα μεταφοράς, οπότε μπορεί να χρησιμοποιηθεί στο διαδίκτυο και να διαπερνά τείχη προστασίας χωρίς συμβιβασμούς στην ασφάλεια της υποδομής μιας επιχείρησης. Αυτό, αυτομάτως μειώνει, και σε ορισμένες περιπτώσεις εξαλείφει, το κόστος



υποδομής, αφού οι περισσότερες επιχειρήσεις σήμερα έχουν και τον εξοπλισμό και την τεχνογνωσία για τη χρήση του διαδικτύου.

- **Επεκτασιμότητα:** Αν και δεν προσφέρει τόσες πολλές λειτουργίες όσο άλλες τεχνολογίες, όπως το CORBA και το DCOM, επιτρέπει σε άλλα πρότυπα να το επεκτείνουν παρέχοντας υπηρεσίες που λείπουν από αυτό. Αυτό το χαρακτηριστικό αποδείχθηκε ίσως το σημαντικότερο, γιατί επάνω του βασίζονται πολλές αναπτυσσόμενες τεχνολογίες των web services, που προσφέρουν υπηρεσίες όπως «αξιοπιστία» (reliability), «δρομολόγηση» (routing) και «ασφάλεια» (security).

### Δομή ενός μηνύματος SOAP



3.2 Δομή μηνύματος SOAP[5]

### Παράδειγμα SOAP Request

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-
envelope" soap:encodingStyle=" http://www.w3.org/2001/12/soap-encoding">
  <soap:Body xmlns:m="http://www.example.org/stock">
    <m:GetStockPrice>
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  </soap:Body>
</soap:Envelope>
```

### Παράδειγμα SOAP Response

```
<?xml version="1.0"?>
```

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-
envelope" soap:encodingStyle=" http://www.w3.org/2001/12/soap-encoding">
  <soap:Body xmlns:m="http://www.example.org/stock">
    <m:GetStockPriceResponse>
      <m:Price>34.5</m:Price>
    </m:GetStockPriceResponse>
  </soap:Body>
</soap:Envelope>
```

### Τοπολογίες Web Services

**Απλές, πληροφοριακές, ή τύπου I**, δηλαδή Web Services που υποστηρίζουν μόνο απλές δράσεις (αίτημα/απόκριση). Περιμένουν πάντα κάποιο αίτημα, το επεξεργάζονται και αποκρίνονται

**Περίπλοκες ή τύπου II**, είναι τα Web Services που υλοποιούν κάποιας μορφής συντονισμό μεταξύ εισερχόμενων και εξερχόμενων εργασιών. Συνήθως περιλαμβάνουν ή συνθέτουν πολλές προϋπάρχουσες υπηρεσίες.

### Συμπεράσματα

Η ανάπτυξη των Web Services δεν είναι μια επανάσταση στα κατακεμημένα συστήματα. Αντίθετα, είναι μια φυσική εξέλιξη της εφαρμογής της XML από δομημένη αναπαράσταση πληροφορίας, σε δομημένη αναπαράσταση μηνυμάτων μεταξύ των εφαρμογών.

Τα Web Services παρέχουν ένα πολύ διαφορετικό τρόπο σκέψης για την ανάπτυξη λογισμικού.

Από το μοντέλο πελάτη-εξυπηρετητή, στα συστήματα n-επιπέδων και τα κατακεμημένα συστήματα, οι εφαρμογές Web Services αποτελούν το αποκορύφωμα κάθε μίας από αυτές τις αρχιτεκτονικές σε συνδυασμό με το διαδίκτυο.

## 3.2. JavaScript

Η **JavaScript** είναι μια γλώσσα προγραμματισμού η οποία, παράγει δυναμικό περιεχόμενο και εκτελεί κώδικα από την πλευρά του πελάτη (client-side) σε ιστοσελίδες. Το πρότυπο της γλώσσας κατά τον οργανισμό τυποποίησης ECMA ονομάζεται ECMAScript.

Η αρχική έκδοση της Javascript βασίστηκε στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας χαρακτηριστικά από νεότερες γλώσσες.

Αρχικά, χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), δηλαδή στον φυλλομετρητή (browser) του χρήστη, με αποτέλεσμα να χαρακτηριστεί σαν *client-side γλώσσα προγραμματισμού*. Αυτό σημαίνει ότι, η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί ακόμη και σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες προγραμματισμού, όπως η PHP, εκτελούνται στο διακομιστή και γι' αυτό χαρακτηρίζονται *server-side γλώσσες προγραμματισμού*.

Ο κώδικας Javascript μιας σελίδας περικλείεται από στις ετικέτες της HTML `<script type="text/javascript">` και `</script>`.

Για παράδειγμα, ο ακόλουθος κώδικας Javascript εμφανίζει ένα πλαίσιο διαλόγου με το κείμενο "Hello World".

```
<script type="text/javascript">
alert('Hello World');
</script>
```

Στις μέρες μας έχουν φτιαχτεί πολλές βιβλιοθήκες βασισμένες σε Javascript. Κάποιες από αυτές, όπως η JQuery που αναλύεται παρακάτω, χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής «Apothecary».

## 3.3. JQuery

Η JQuery είναι μια βιβλιοθήκη-framework της Javascript που απλοποιεί την αλληλεπίδραση με τα έγγραφα HTML, επιτρέποντας την διαχείριση events, την ανάπτυξη animations και προσθέτει δυναμικότητα, με την τεχνολογία AJAX, στις ιστοσελίδες. Η jQuery, όπως και άλλες βιβλιοθήκες, προσφέρει μια σειρά από λειτουργίες βασισμένες σε Javascript που διαφορετικά θα απαιτούσαν πολύ περισσότερο κώδικα. Αυτό σημαίνει ότι, με τις συναρτήσεις αυτής της βιβλιοθήκης, πραγματοποιούνται εντυπωσιακά αποτελέσματα σε λιγότερο χρόνο και χώρο. Επίσης, επιτρέπει την επιλογή και τροποποίηση στοιχείων του DOM καθώς και τη διαχείριση όλων των events των browsers.

Η βιβλιοθήκη αυτή είναι multibrowser, δηλαδή λειτουργεί με τις περισσότερες τελευταίες εκδόσεις σε όλους τους browsers (π.χ. Mozilla Firefox, Internet Explorer, Safari, Opera κ.λπ.)

Τέλος, επιτρέπει την διαμόρφωση του CSS, διαφόρων εφέ και animations, AJAX, και επιτρέπει επεκτάσεις, από τις οποίες χρησιμοποιήθηκαν αρκετές στην παρούσα εφαρμογή. Μια από τις επεκτάσεις της βιβλιοθήκης αυτής, που χρησιμοποιήθηκε κατά κόρον στην

εφαρμογή Apothecary, είναι το jqGrid με την χρήση της οποίας εμφανίζονται τα δεδομένα, όπου χρειάζεται, σε μορφή πίνακα.

### 3.4. jqGrid

Η jqgrid είναι μία επέκταση της βιβλιοθήκης JQuery που, μας επιτρέπει την αναπαράσταση δεδομένων με τη μορφή πινάκων δεδομένων (grids) χρησιμοποιώντας AJAX λειτουργικότητα. Οι δυνατότητες που μας προσφέρει το jqGrid είναι:

- Ομαδοποίηση των δεδομένων και σελιδοποίησή τους. Με την σελιδοποίηση των δεδομένων τα τελευταία εμφανίζονται σε σελίδες.
- Προσθήκη, τροποποίηση, διαγραφή και αναζήτηση εγγραφών
- Αποδοχή δεδομένων μέσω διαφόρων τύπων δεδομένων όπως XML, JSON, Array ή δεδομένων που εισάγει ο χρήστης.
- Πολλαπλή επιλογή γραμμών.
- Δημιουργία υποπινάκων και λεπτομερείς πίνακες.

Επειδή, ο πίνακας δεδομένων δημιουργείται και φορτώνει τα δεδομένα που προέρχονται από αιτήσεις AJAX στην πλευρά του πελάτη, είναι δυνατή η χρήση οποιασδήποτε γλώσσας προγραμματισμού από την πλευρά του εξυπηρετητή.

Αυτή η τεχνολογία, επίσης, μας επιτρέπει να περάσουμε αντικείμενα Javascript σε μεταβλητές JSON, οι οποίες είναι σειριοποιήσιμες και μπορούν να περάσουν ως δεδομένα στην δική μας επέκταση.[10]

### 3.5. AJAX

Το AJAX δεν είναι μια νέα γλώσσα προγραμματισμού, αλλά ένας νέος τρόπος διαχείρισης ήδη υπάρχοντων τεχνολογιών. Παρέχει έναν εύκολο τρόπο ανταλλαγής δεδομένων μεταξύ του Client και του Server, χωρίς να απαιτείται η φόρτωση όλης της σελίδας εκ νέου από το Server (post back). Αυτό έχει σαν αποτέλεσμα η σελίδες να είναι πιο γρήγορες.

Η τεχνολογία Ajax είναι βασισμένη σε κάποια Internet standards και χρησιμοποιεί έναν συνδυασμό από τις παρακάτω γλώσσες :

- **XMLHttpRequest object:** χρησιμοποιείται για την ασύγχρονη ανταλλαγή δεδομένων μεταξύ client και server.
- **JavaScript/DOM:** χρησιμοποιείται για την εμφάνιση των δεδομένων μέσω αμφίδρομης επικοινωνίας. Το DOM είναι μια προγραμματιστική διεπαφή (programming interface) για τα XML έγγραφα. Ορίζει τον τρόπο που μπορούμε να προσπελάσουμε και να χειριστούμε ένα XML έγγραφο. Εδώ γίνεται μέσω Javascript.
- **CSS:** για την μορφοποίηση της εμφάνισης των δεδομένων.
- **XML:** συχνά χρησιμοποιείται ως μορφή με την οποία μεταφέρονται τα δεδομένα.

Αυτή η τεχνολογία, χρησιμοποιήθηκε στην εφαρμογή Apothecary, σε πολλά σημεία, σε συνδυασμό κυρίως με τα Web Services. Τα Web Services καλούνται με Ajax Call εντολών μέσω JQuery Framework.

Την τεχνολογία Ajax τη χρησιμοποιούν και πολλές εφαρμογές που είναι πολύ διαδεδομένες σε όλο τον κόσμο. Μια από αυτές τις εφαρμογές είναι και το Goggle Maps, η οποία χρησιμοποιήθηκε και στην υλοποίηση της εφαρμογής.

### 3.6. GoogleMaps (JavaScript V3)[14]

Το GoogleMaps API είναι μια υπηρεσία που διατίθεται χωρίς χρέωση, με σκοπό να χρησιμοποιηθεί σε οποιαδήποτε ιστοσελίδα που παρέχει δωρεάν πληροφορίες σε όλους τους χρήστες του διαδικτύου, προσφέροντας μια, πολύ ισχυρή και φιλική προς το χρήστη, τεχνολογία χαρτογράφησης.

Η Google Maps μας παρέχει ένα μεγάλο αριθμό API, τα οποία μας επιτρέπουν να ενσωματώσουμε τους εν λόγω χάρτες αλλά και τη λειτουργικότητα της υπηρεσίας στη δική μας ιστοσελίδα, προκειμένου να εμφανίσουμε τα δικά μας δεδομένα πάνω στους χάρτες αυτούς.

Η Google Maps παρέχει ένα ευρύ φάσμα API που επιτρέπει στους προγραμματιστές να ενσωματώσουν την ισχυρή λειτουργικότητα και καθημερινή χρησιμότητα του Google Maps στις δικές τους δικτυακές εφαρμογές με ευκολία, χρησιμοποιώντας τα δεδομένα τους και εμφανίζοντας τα πάνω στους χάρτες της Google.

Για τη δική μας εφαρμογή χρησιμοποιήθηκε το JavaScript API V3. Η έκδοση αυτή είναι η τρίτη έκδοση του JavaScript API, η οποία κυκλοφόρησε τον Μάιο του 2009. Σύμφωνα με τα σχόλια των χρηστών, είναι μια εξελιγμένη έκδοση της προηγούμενης.

Για να αποκομίσουμε τα οφέλη της υπηρεσίας αυτής, έπρεπε να ενσωματώσουμε στην εφαρμογή μας τη βιβλιοθήκη JQuery αλλά και τη βιβλιοθήκη του Google Maps.

Για να μπορέσουμε να χρησιμοποιήσουμε τη βιβλιοθήκη JQuery έπρεπε, αρχικά να κατεβάσουμε από το διαδίκτυο τη τελευταία έκδοση της από την σελίδα του κατασκευαστή, και στη συνέχεια να την συμπεριλάβουμε στην HTML σελίδα τις εφαρμογής μας με τον ακόλουθο τρόπο:

```
<script type="text/javascript" src="js/jquery-1.7.1.min.js"> </span>
```

Για να χρησιμοποιήσουμε στην εφαρμογή μας την υπηρεσία Google Maps, ενσωματώσαμε στην HTML σελίδα της εφαρμογής μας την παρακάτω γραμμή:

```
<script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=false"> </span>
```

### 3.7. Google Chart API

Το Chart API της Google είναι ένα εργαλείο το οποίο επιτρέπει στον χρήστη, χρησιμοποιώντας το, να δημιουργήσει γραφήματα σύμφωνα με τα δεδομένα που έχει και κάτω από συγκεκριμένες συνθήκες. Στην συνέχεια τα γραφήματα αυτά, τα οποία είναι σε PNG μορφή, με την βοήθεια πάλι του Google API ενσωματώνονται στην σελίδα μέσω HTTP

αιτήματος. Οι τύποι των Charts τα οποία υποστηρίζονται είναι πολλά και μπορούν να ενσωματωθούν στην σελίδα εύκολα, κάνοντας ο χρήστης ένα αίτημα με συγκεκριμένο tag.

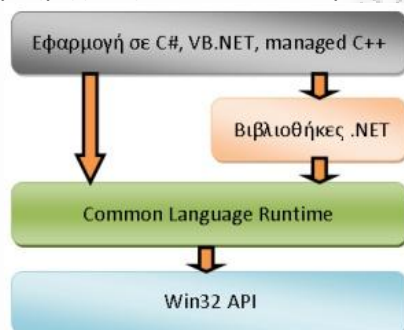
Αρχικά το εργαλείο αυτό ήταν εσωτερικό και χρησιμοποιούταν από την Google για την πιο εύκολη εμφάνιση των χαρτών. Στην συνέχεια δόθηκε στην διάθεση των Web Developers.

Στην εφαρμογή 'Apothecary' χρησιμοποιήθηκε για την εμφάνιση στατιστικών δεδομένων τα οποία μπορεί να δει ο υπεύθυνος του φαρμακείου και αφορούν, τις αναζητήσεις των φαρμακείων και των φαρμάκων που κάνουν οι χρήστες της εφαρμογής. Για τον σκοπό αυτό χρησιμοποιήθηκε το 'Vertical bar Chart' το οποίο παραμετροποιήθηκε αντίστοιχα με τις ανάγκες της εφαρμογής.[14]

### 3.8. C# .NET

Η C# είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού, η οποία δημιουργήθηκε από την εταιρεία Microsoft. Έχει πολλά κοινά στοιχεία και παρόμοια σύνταξη με την C++ και τη Java. Είναι γλώσσα ειδικά σχεδιασμένη για να υποστηρίζει το .NET framework της ίδιας της εταιρείας. Βασικό χαρακτηριστικό της είναι ότι, δεν παράγει απευθείας κώδικα μηχανής όπως η C++, αλλά ένα ενδιάμεσο κώδικα που στοχεύει το .NET. Το Visual Studio είναι το development εργαλείο μέσω του οποίου διαχειριζόμαστε στο .NET framework. Αυτό το εργαλείο, και συγκεκριμένα το Visual Studio 2008, χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής Apothecary.

Το .NET είναι μια πλατφόρμα η οποία διευκολύνει τον προγραμματιστή να επικεντρωθεί στον προγραμματισμό με ασφάλεια, αφού δεν χρειάζεται να ασχολείται με την διαχείριση μνήμης, την επικοινωνία μέσω δικτύου και άλλες τεχνικές λεπτομέρειες. Υποστηρίζει πληθώρα γλωσσών προγραμματισμού όπως η C++, η Visual Basic .NET και η C#, η οποία χρησιμοποιήθηκε και στην υλοποίηση της εφαρμογής.



3.1 Επίπεδα μεταξύ C# και Win32 API [13]

Η πλατφόρμα αυτή παρέχει μια πληθώρα από έτοιμες βιβλιοθήκες, τις λειτουργίες των οποίων μπορεί να χρησιμοποιήσει ο χρήστης στις εφαρμογές του. Η χρήση αυτών των βιβλιοθηκών (assemblies) κάνει πολύ εύκολη την επαναχρησιμοποίηση κώδικα, κάνοντας έτσι την ανάπτυξη οποιασδήποτε εφαρμογής γρήγορη και δομημένη.[13]

## 4. Σχεδιασμός Μοντελοποίηση Λύσης (Use cases - οντοτήτων συσχετίσεων - message sequence charts)

### 4.1. Use Case και Activity Diagrams

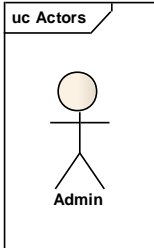
#### 4.1.1. Βασικοί Πρωταγωνιστές του Συστήματος

Η εφαρμογή που έχει αναπτυχθεί είναι χωρισμένη σε δύο μέρη. Το πρώτο μέρος αποτελείται από την εφαρμογή διαχείρισης φαρμακείων, την οποία χειρίζονται οι διαχειριστές του συστήματος, οι υπεύθυνοι των φαρμακείων και οι υπάλληλοι τους. Το δεύτερο μέρος είναι η μια εφαρμογή αναζήτησης με την οποία οποιοσδήποτε χρήστης μπορεί, μέσω του διαδικτύου, να αναζητήσει ένα ή περισσότερα φάρμακα ή φαρμακεία στην περιοχή που τον ενδιαφέρει, οποιαδήποτε μέρα.

##### 4.1.1.1. Administrator (Διαχειριστής Συστήματος - Χρήστης εφαρμογής διαχείρισης φαρμακείου)

Είναι ο υπεύθυνος για την καταχώρηση νέων φαρμάκων, νέων φαρμακείων, εφημεριών, την αλλαγή πληροφοριών όλων των παραπάνω, καθώς και για την συντήρηση γενικά του συστήματος. Κάθε νέο φάρμακο το οποίο βγαίνει σε κυκλοφορία καταχωρείται στην βάση μαζί με όλα τα χαρακτηριστικά του. Έτσι είναι διαθέσιμο στον υπεύθυνο κάθε φαρμακείου ώστε, σε τυχόν παραλαβή αυτού, να μπορεί να καταχωρήσει την ποσότητα της παραλαβής στο απόθεμα της αποθήκης, με σκοπό ο χρήστης – ασθενής που θα το αναζητήσει να το δει ως διαθέσιμο στο εν λόγω φαρμακείο.


Στοιχεία Πρωταγωνιστή	
Όνομα Actor	Διαχειριστής Συστήματος.
Σύντομη Περιγραφή	Είναι ένας έγκυρος χρήστης ο οποίος εγκαθιστά, διαχειρίζεται, και συντηρεί το σύστημα.
Κατηγορία	▪ Χρήστες



The diagram shows a UML Actor for 'Admin'. It consists of a circle representing the actor's head, a vertical line for the body, and three lines radiating from the bottom representing legs. The actor is enclosed in a rectangular frame with the label 'uc Actors' in the top-left corner and 'Admin' centered below the actor.


##### 4.1.1.2. Store Manager (Υπεύθυνος φαρμακείου - Χρήστης εφαρμογής διαχείρισης φαρμακείου)

Καταχωρεί νέα κομμάτια με βάση την παραλαβή καθώς έχει και όλες τις αρμοδιότητες ενός απλού υπάλληλου του φαρμακείου (βλέπε υπάλληλος φαρμακείου). Μπορεί να κάνει επίβλεψη της αποθήκης και να ενημερώνεται για την αριθμό των αναζητήσεων του φαρμακείου ανά μήνα και τις αναζητήσεις των δημοφιλέστερων φαρμάκων στην εφαρμογή, μέσω διαγραμμάτων (Charts).

Στοιχεία Πρωταγωνιστή		
Όνομα Actor	Υπεύθυνος φαρμακείου.	
Σύντομη Περιγραφή	Είναι ένας έγκυρος χρήστης ο οποίος είναι υπεύθυνος για τον έλεγχο και την ενημέρωση της αποθήκης του φαρμακείου.	
Κατηγορία	<ul style="list-style-type: none"> <li>▪ Χρήστες</li> </ul>	

#### 4.1.1.3. Υπάλληλος φαρμακείου (Χρήστης εφαρμογής διαχείρισης φαρμακείου)

Ο υπάλληλος του φαρμακείου μπορεί να χρησιμοποιεί το εργαλείο, πραγματοποιώντας αφαίρεση φαρμάκου από το απόθεμα της αποθήκης έμμεσα, μέσα από την πώληση. Όταν τελειώνει το απόθεμα ενός φαρμάκου, κατά την πώληση, ειδοποιείται με μήνυμα έτσι ώστε να ενημερωθεί έγκαιρα ο υπεύθυνος παραγγελιών του φαρμακείου για την παραγγελία του συγκεκριμένου φαρμάκου.

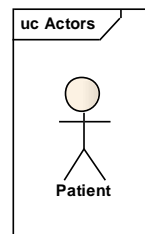
Στοιχεία Πρωταγωνιστή		
Όνομα Actor	Υπάλληλος φαρμακείου.	
Σύντομη Περιγραφή	Είναι ένας έγκυρος χρήστης ο οποίος είναι υπεύθυνος για την πώληση των φαρμάκων.	
Κατηγορία	<ul style="list-style-type: none"> <li>▪ Χρήστες</li> </ul>	

#### 4.1.1.4. Χρήστης εφαρμογής αναζήτησης μέσω διαδικτύου (Ασθενής - Αναζητητής)

Ο χρήστης-ασθενής μπορεί να αναζητήσει ένα φαρμακείο που είναι σε λειτουργία την ημέρα και ώρα που επιθυμεί στην περιοχή που τον ενδιαφέρει. Επίσης, μπορεί να αναζητήσει κάποιο φάρμακο σύμφωνα με τις απαιτήσεις του, και μόλις το επιλέξει μπορεί να δει σε ποια από τα ανοιχτά φαρμακεία στην περιοχή της επιλογής του, είναι διαθέσιμο. Η αναζήτηση γίνεται με τα στοιχεία τα οποία είναι διαθέσιμα κατά την στιγμή της αναζήτησης. Μπορεί δηλαδή την στιγμή κατά την οποία θα προσέλθει ο χρήστης-ασθενής στο φαρμακείο, το φάρμακο να μην είναι πλέον διαθέσιμο (Να έχει πωληθεί).



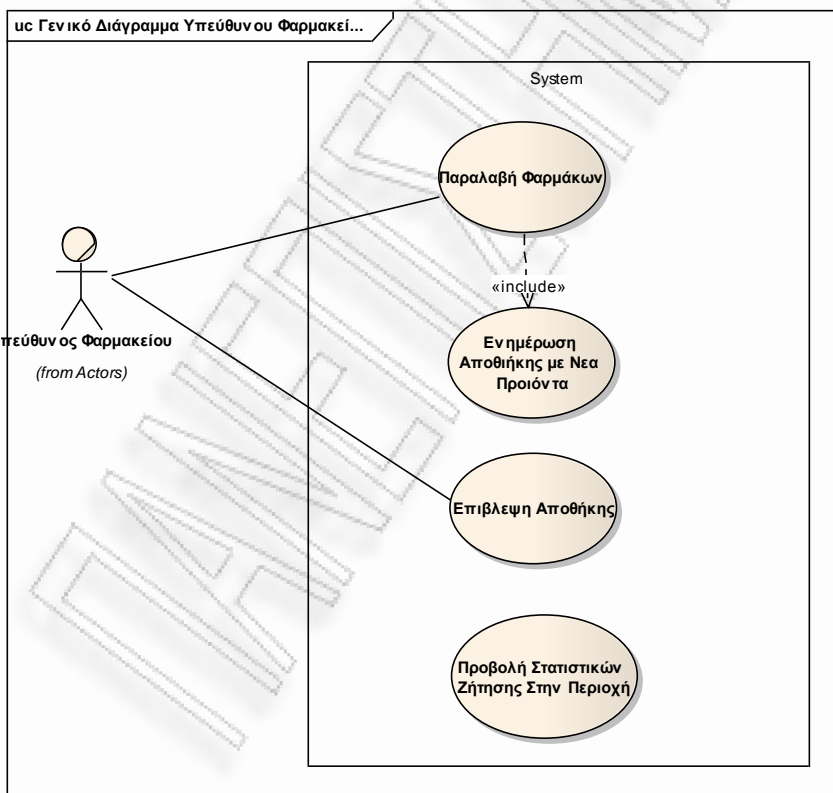
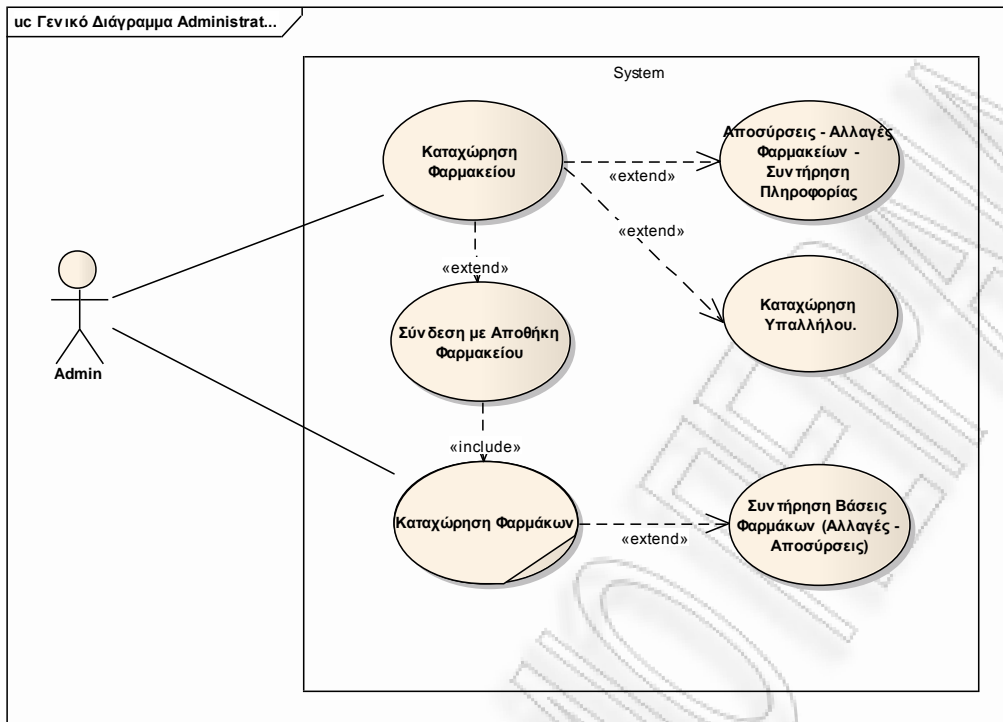
Στοιχεία Πρωταγωνιστή	
<b>Όνομα Actor</b>	Ασθενής - Αναζητητής.
<b>Σύντομη Περιγραφή</b>	<p>Είναι ένας έγκυρος χρήστης ο οποίος έχει την δυνατότητα να αναζητήσει διαθέσιμο φαρμακείο στην περιοχή που θέλει και την στιγμή που θέλει, καθώς και φάρμακο και σε ποιο από τα παραπάνω φαρμακεία είναι διαθέσιμο.</p>
<b>Κατηγορία</b>	<ul style="list-style-type: none"> <li>▪ Χρήστες</li> </ul>

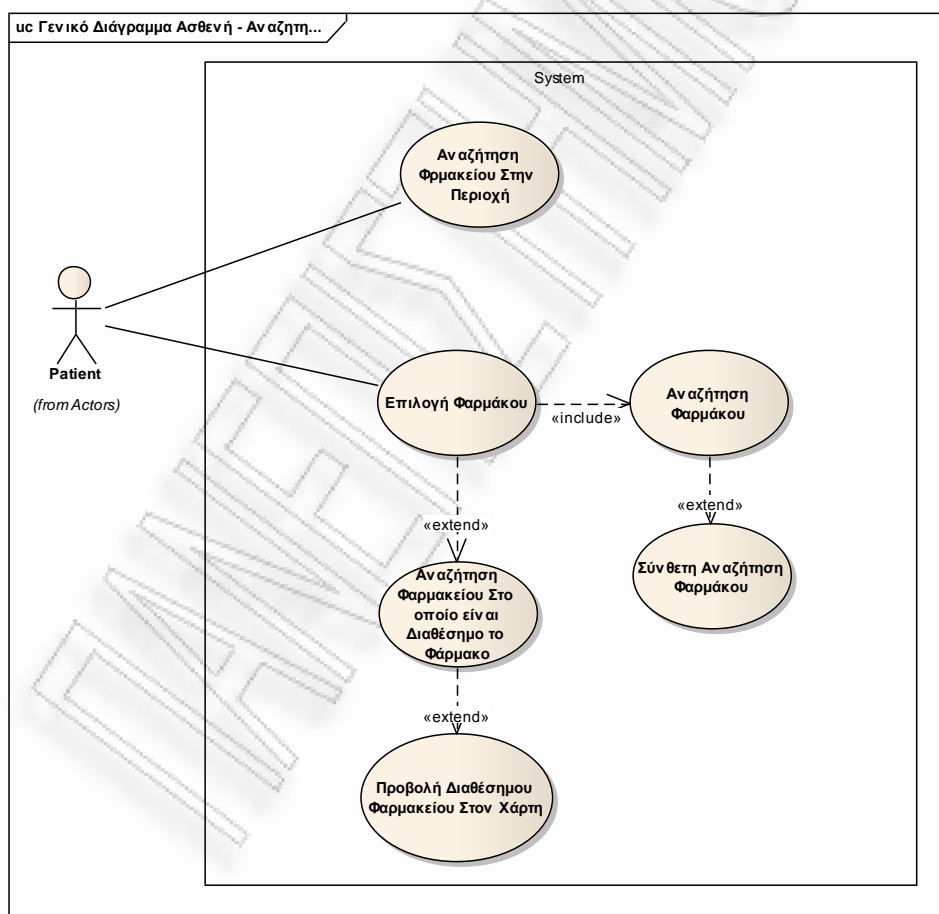
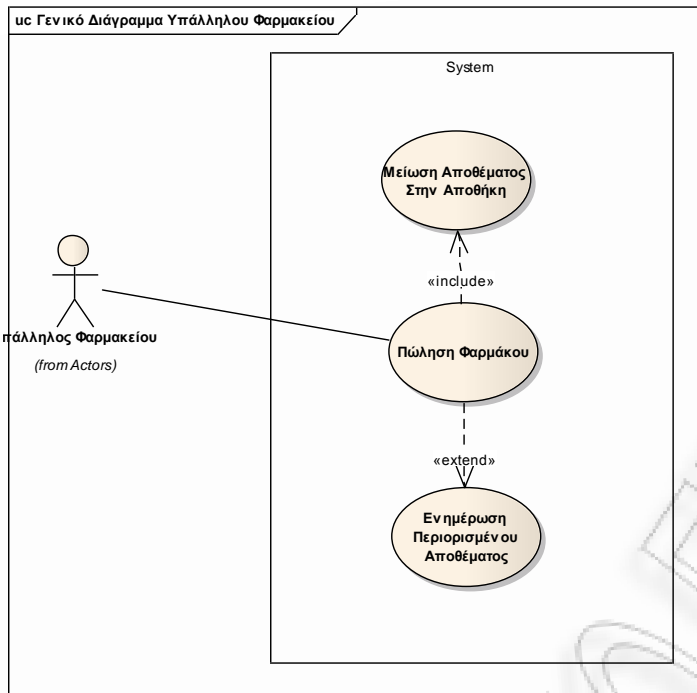


#### 4.1.2. Πρότυπο Σεναρίου Χρήσης

<b>Όνομα Σεναρίου Χρήσης</b>	<i>Όνομα του σεναρίου Χρήσης</i>	
<b>Συνοπτική Περιγραφή</b>	<i>Το τμήμα αυτό παρουσιάζει τον σκοπό του σεναρίου χρήσης και τι κάνει αυτή η λειτουργία εν συντομία.</i>	
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	<i>Ο/οι actor/s, που επιδρά/ούν με την περίπτωση χρήσης.</i>	
<b>Βασική Ροή Λειτουργίας</b>	<i>Πριγράφονται οι ενέργειες του actor ώστε να ενεργοποιηθεί η περίπτωση χρήσης, με την οποία έχει άμεση σχέση, και στη συνέχεια η απόκριση του συστήματος.</i>	
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<i>Αυτό το τμήμα περίπτωσης χρήσης περιγράφει μία εναλλακτική απόκριση του συστήματος, αναφερόμενοι σε ένα συγκεκριμένο βήμα της βασικής ροής.</i>
<b>Preconditions</b>	<b>1</b>	<i>Η Προ-Συνθήκη μίας περίπτωσης χρήσης είναι η κατάσταση που πρέπει να ισχύει πριν την εκτέλεση μίας περίπτωσης χρήσης.</i>
<b>Postconditions</b>	<b>1</b>	<i>Η Μετά-Συνθήκη μίας περίπτωσης χρήσης είναι μία λίστα από πιθανές καταστάσεις που μπορεί να ισχύσουν αμέσως μετά την ολοκλήρωση μίας περίπτωσης χρήσης.</i>
<b>Διάγραμμα Περίπτωσης Χρήσης</b>	<i>Το διάγραμμα περίπτωσης χρήσης (Use Case diagram) το οποίο περιγράφει τη λειτουργικότητα του συστήματος όπως αυτή γίνεται αντιληπτή από εξωτερικές οντότητες (Actors).</i>	
<b>Βοηθητικά Διαγράμματα</b>	<i>Αλλά διαγράμματα όπως Activity και Sequence Diagrams τα οποία ολοκληρώνουν την περιγραφή - απεικόνιση του συστήματος. Τα Activity παρουσιάζουν την ακολουθιακή ροή των Δραστηριοτήτων. Τα Sequence Diagram παρουσιάζουν μια αλληλεπίδραση αντικειμένων με έμφαση στην χρονική σειρά ανταλλαγής μηνυμάτων.</i>	

### 4.1.3. Λειτουργικές Απαιτήσεις Συστήματος





**1.1. Εισαγωγή Χρήστη στο Σύστημα (Login)**

<b>Όνομα Σεναρίου Χρήσης</b>	Εισαγωγή Χρήστη στο Σύστημα	
<b>Συνοπτική Περιγραφή</b>	Σκοπός αυτού του σεναρίου χρήσης είναι να επιτρέψει στον χρήστη να συνδεθεί στην εφαρμογή.	
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	Ο χρήστης, είτε έχει τον ρόλο υπεύθυνου φαρμακείου, είτε υπάλληλου φαρμακείου.	
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει να συνδεθεί στην εφαρμογή</li> <li>2. Το σύστημα παρέχει μία οθόνη στον χρήστη για να εισάγει το: <ul style="list-style-type: none"> <li>✓ Όνομα Χρήστη</li> <li>✓ Κωδικό Πρόσβασης</li> </ul> </li> <li>3. Ο χρήστης εισάγει τα στοιχεία</li> <li>4. Το σύστημα επικυρώνει τα στοιχεία σύνδεσης</li> <li>5. Τα στοιχεία σύνδεσης γίνονται δεκτά από το σύστημα και οδηγεί τον χρήστη ανάλογα με τον ρόλο του στην αρχική του οθόνη.</li> <li>6. Το σενάριο χρήσης τερματίζει</li> </ol>	
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<p>Τα στοιχεία σύνδεσης δεν είναι σωστά. Από το βήμα 4 της κύριας ροής:</p> <p style="margin-left: 40px;">5.b. Τα στοιχεία σύνδεσης δεν είναι σωστά</p> <p style="margin-left: 40px;">6.b. Το σύστημα ενημερώνει σχετικά τον χρήστη</p> <p style="margin-left: 40px;">7.b. Το σύστημα προτρέπει τον χρήστη να εισάγει τα στοιχεία σύνδεσης ξανά</p> <p>Το σενάριο χρήσης συνεχίζει από το βήμα 3 της κύριας ροής.</p>

<b>Preconditions</b>	<b>1</b>	Ο διαχειριστής της εφαρμογής θα πρέπει να έχει εισάγει τους χρήστες και να τους έχει ενημερώσει για το Username και το Password.
<b>Post conditions</b>		Μετά την επιτυχή έκβαση του σεναρίου ο χρήστης (διαχειριστής – υπάλληλος φαρμακείου – υπεύθυνος φαρμακείου) έχει επιτυχώς εισέλθει στο σύστημα.

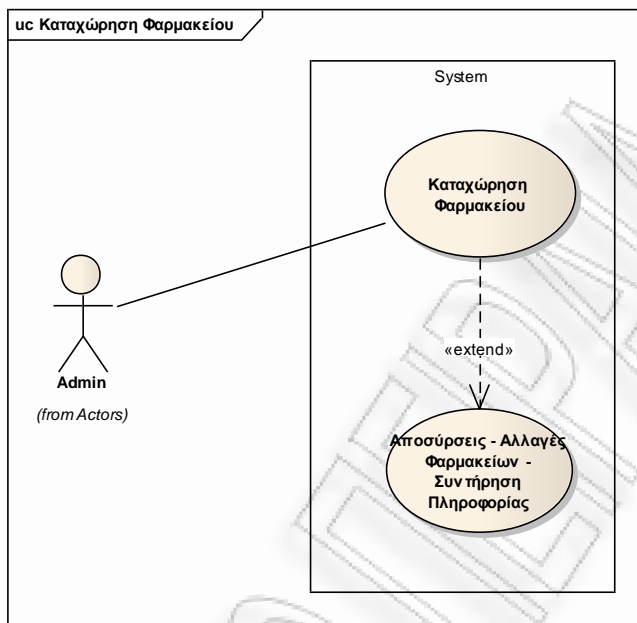
#### 4.1.4. Λειτουργίες Διαχειριστή συστήματος

##### 1.2. Καταχώρηση Φαρμακείων

<b>Όνομα Σεναρίου Χρήσης</b>	<i>Καταχώρηση Φαρμακείων</i>
<b>Συνοπτική Περιγραφή</b>	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ένας διαχειριστής του συστήματος ώστε να καταχωρίσει νέα Φαρμακεία στην βάση ή να αλλάξει τις πληροφορίες ήδη υπάρχοντων Φαρμακείων:
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	Ο διαχειριστής του Συστήματος.
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει την καρτέλα για καταχώρηση φαρμακείου όπου εμφανίζεται λίστα με τα είδη καταχωρημένα φαρμακεία και τα χαρακτηριστικά τους.</li> <li>2. Ο χρήστης πατάει το κουμπί «ADD» (επιλέγει προσθήκη).</li> <li>3. Το σύστημα εμφανίζει μια κενή εγγραφή για τη νέα καταχώρηση.</li> <li>4. Ο χρήστης συμπληρώνει όλα τα απαιτούμενα στοιχεία.</li> <li>5. Ο χρήστης επιλέγει «καταχώρηση».</li> <li>6. Το σύστημα ελέγχει αν έχουν συμπληρωθεί όλα τα απαιτούμενα στοιχεία για την επιτυχή ολοκλήρωση της καταχώρησης.</li> <li>7. Το φαρμακείο καταχωρείται.</li> <li>8. Το σενάριο τερματίζει.</li> </ol>

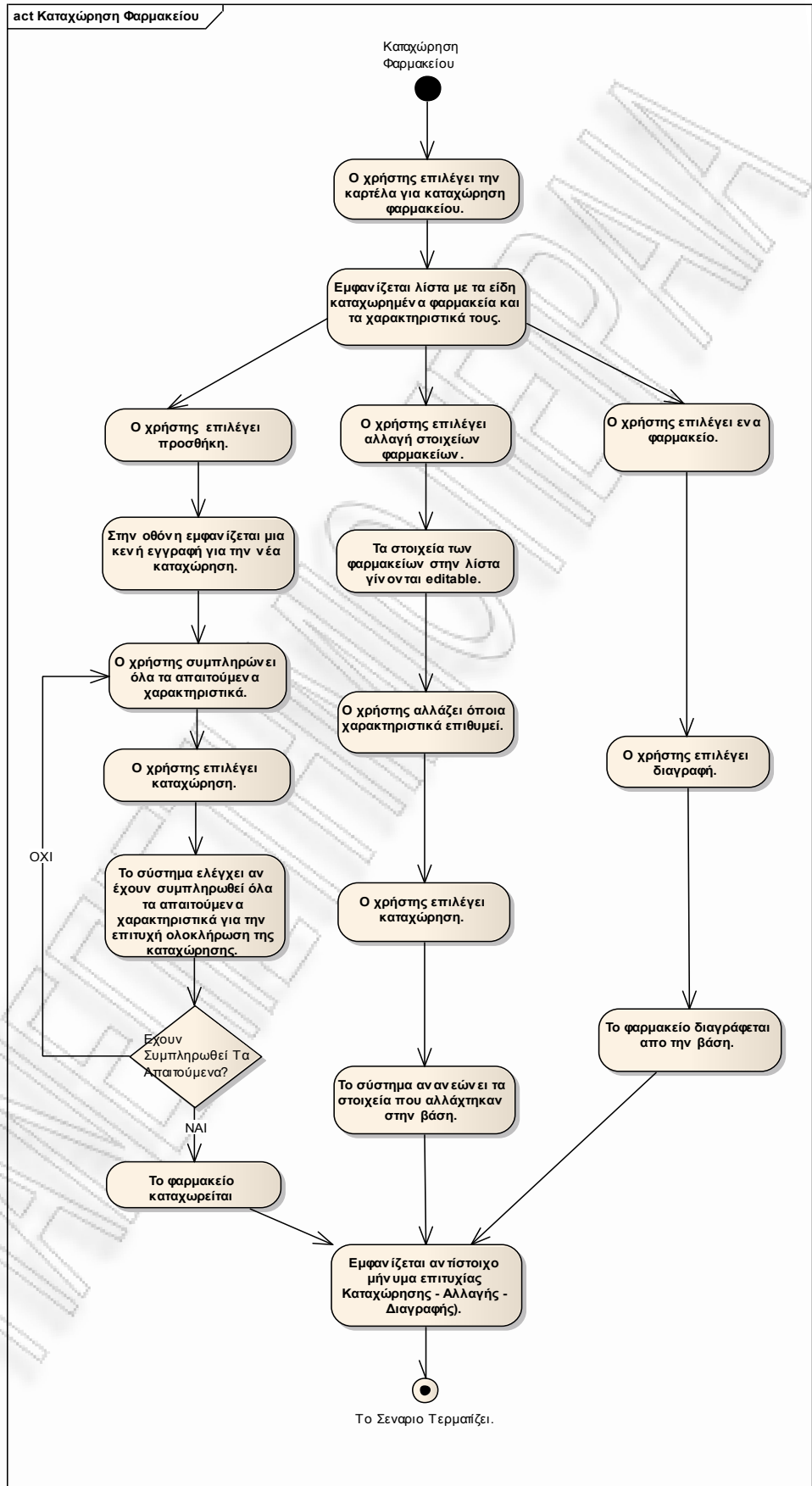
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<p>Μη επαρκή καταχωρημένα στοιχεία φαρμακείου στη φόρμα. Από το βήμα 6 της βασικής ροής:</p> <p>7.b. Το σύστημα επιστρέφει μήνυμα λάθους και προτρέπει τον χρήστη να καταχωρίσει όλα τα απαιτούμενα στοιχεία σωστά.</p> <p>Το σενάριο συνεχίζει από το βήμα 4 της βασικής ροής.</p>
	<b>2</b>	<p>Αλλαγή στοιχείων Φαρμακείου. Από το βήμα 1 της βασικής ροής:</p> <p>2.c. Ο χρήστης επιλέγει «EDIT» (επεξεργασία στοιχείων φαρμάκου).</p> <p>3.c. Τα στοιχεία των φαρμακείων στην λίστα γίνονται επεξεργάσιμα (editable).</p> <p>4.c. Ο χρήστης αλλάζει όποια στοιχεία επιθυμεί.</p> <p>5.c. Ο χρήστης επιλέγει «καταχώρηση».</p> <p>6.c. Το σύστημα ανανεώνει τα στοιχεία που αλλάχτηκαν στη βάση δεδομένων της εφαρμογής.</p> <p>Το σενάριο τερματίζει (8 βασική ροή).</p>
	<b>3</b>	<p>Διαγραφή Φαρμακείου. Από το βήμα 1 της βασικής ροής:</p> <p>2.d. Ο χρήστης επιλέγει το φαρμακείο το οποίο επιθυμεί να διαγράψει.</p> <p>3.d. Ο χρήστης επιλέγει διαγραφή «Delete».</p> <p>4.d. Το φαρμακείο διαγράφεται επιτυχώς από την βάση.</p> <p>Το σενάριο τερματίζει ( βήμα 8 βασική ροή).</p>
<b>Preconditions</b>	<b>1</b>	<p>Να έχει πρόσβαση ο διαχειριστής του συστήματος στην οθόνη καταχώρησης φαρμακείου.</p>
<b>Postconditions</b>	<b>1</b>	<p>Με την επιτυχή ολοκλήρωση του σεναρίου χρήσης ένα νέο φαρμακείο θα έχει εισαχθεί στο σύστημα ή θα έχουν αλλαχτεί τα στοιχεία κάποιου φαρμακείου στο σύστημα.</p>

**Διάγραμμα  
Περίπτωσης  
Χρήσης**





**Activity Diagram**

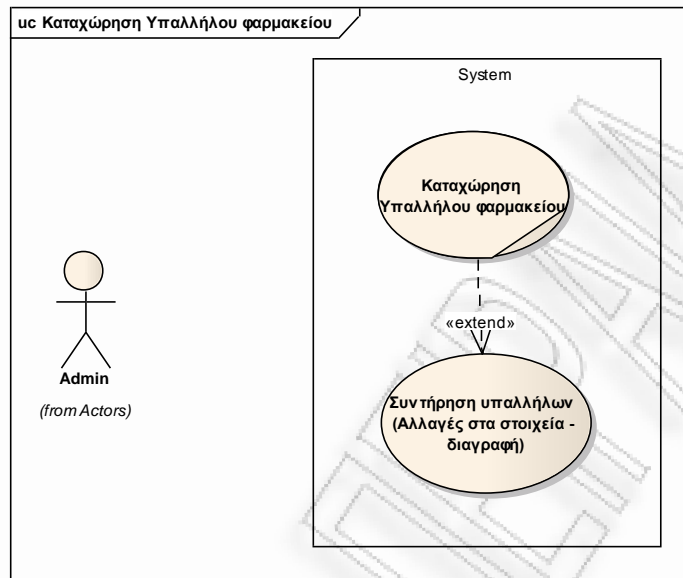


## 5.2 Καταχώρηση Υπαλλήλων

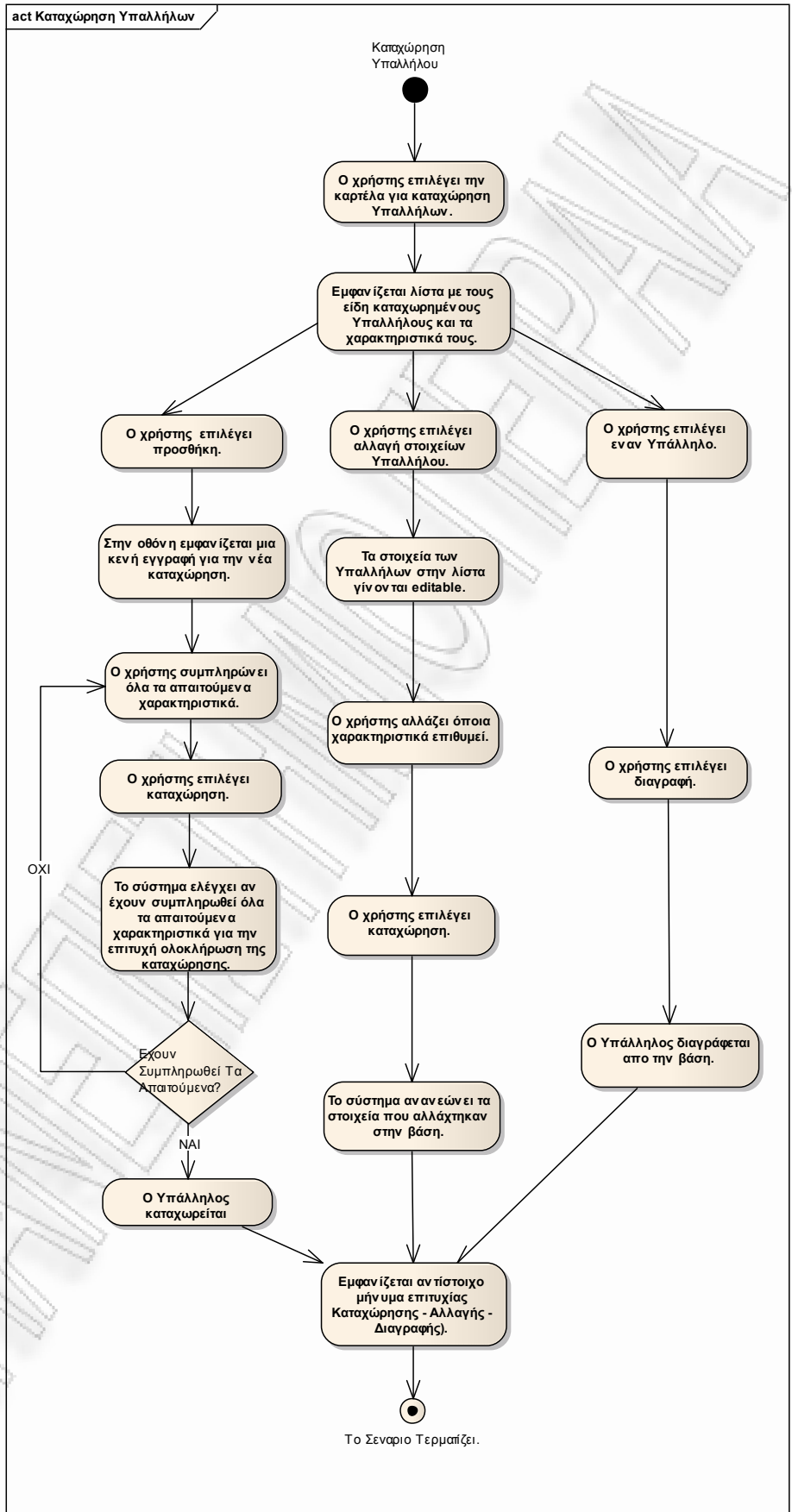
Όνομα Σεναρίου Χρήσης	<b>Καταχώρηση Υπαλλήλων</b>	
Συνοπτική Περιγραφή	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ο διαχειριστής του συστήματος ώστε να καταχωρίσει νέους υπαλλήλους στη βάση δεδομένων ή να αλλάξει τα στοιχεία ήδη υπάρχοντων Υπαλλήλων.	
Βασικός Πρωταγωνιστής (Primary Actor)	Ο διαχειριστής του Συστήματος.	
Βασική Ροή Λειτουργίας	<ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει την καρτέλα για καταχώρηση νέου υπαλλήλου όπου, εμφανίζεται λίστα με τους ήδη καταχωρημένους υπαλλήλους και τα στοιχεία τους.</li> <li>2. Ο χρήστης πατάει το κουμπί «ADD» (προσθήκη).</li> <li>3. Το σύστημα εμφανίζει μια κενή εγγραφή για τη νέα καταχώρηση.</li> <li>4. Ο χρήστης συμπληρώνει όλα τα απαιτούμενα στοιχεία.</li> <li>5. Ο χρήστης επιλέγει «καταχώρηση».</li> <li>6. Το σύστημα ελέγχει αν έχουν συμπληρωθεί όλα τα απαιτούμενα στοιχεία για την επιτυχή ολοκλήρωση της καταχώρησης.</li> <li>7. Ο υπάλληλος καταχωρείται.</li> <li>8. Το σενάριο τερματίζει.</li> </ol>	
Εναλλακτικές Ροές	<b>1</b>	<p>Μη επαρκή καταχωρημένα στοιχεία υπαλλήλου στην φόρμα. Από το βήμα 6 της βασικής ροής:</p> <p>7.b. Το σύστημα επιστρέφει μήνυμα λάθους και προτρέπει τον χρήστη να καταχωρίσει όλα τα απαιτούμενα στοιχεία σωστά.</p> <p>Το σενάριο συνεχίζει από το βήμα 4 της βασικής ροής.</p>

	<b>2</b>	<p>Αλλαγή στοιχείων Υπαλλήλου. Από το βήμα 1 της βασικής ροής:</p> <p>2.c. Ο χρήστης επιλέγει «EDIT» (επεξεργασία στοιχείων υπαλλήλου).</p> <p>3.c. Τα στοιχεία των υπαλλήλων στην λίστα γίνονται επεξεργάσιμα (editable).</p> <p>4.c. Ο χρήστης αλλάζει όποια στοιχεία επιθυμεί.</p> <p>5.c. Ο χρήστης επιλέγει «καταχώρηση».</p> <p>6.c. Το σύστημα ανανεώνει τα στοιχεία που αλλάχτηκαν στη βάση δεδομένων της εφαρμογής.</p> <p>Το σενάριο τερματίζει (8 βασική ροή).</p>
	<b>3</b>	<p>Διαγραφή Υπαλλήλων. Από το βήμα 1 της βασικής ροής:</p> <p>2.d. Ο χρήστης επιλέγει τον υπάλληλο το οποίο επιθυμεί να διαγράψει.</p> <p>3.d. Ο χρήστης επιλέγει διαγραφή «Delete».</p> <p>4.d. Ο υπάλληλος διαγράφεται επιτυχώς από την βάση.</p> <p>Το σενάριο τερματίζει (8 βασική ροή).</p>
<b>Preconditions</b>	<b>1</b>	Να έχει πρόσβαση ο διαχειριστής του συστήματος στην οθόνη καταχώρησης φαρμακείου.
	<b>2</b>	Να έχει καταχωρηθεί πρώτα το φαρμακείο από τον διαχειριστή στο οποίο εργάζεται ο υπάλληλος έτσι ώστε να μπορεί να το επιλέξει για να το συνδέσει.
<b>Postconditions</b>	<b>1</b>	Με την επιτυχή ολοκλήρωση του σεναρίου χρήσης ο υπάλληλος θα έχει εισαχθεί στο σύστημα ή θα έχουν αλλαχτεί τα στοιχεία του

**Διάγραμμα  
Περίπτωσης  
Χρήσης**



# Activity Diagram

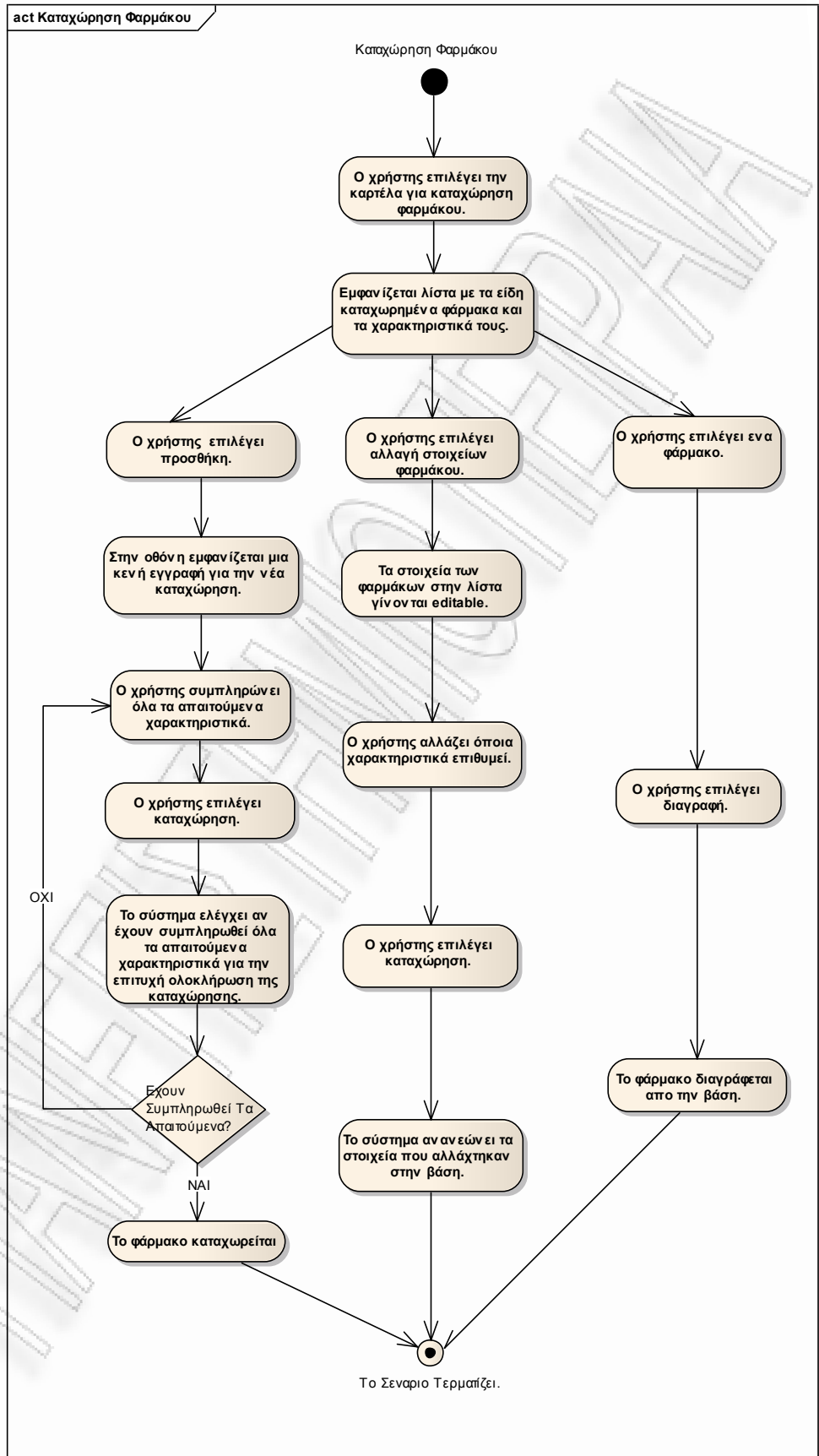


### 5.3 Καταχώρηση Φαρμάκων

<b>Όνομα Σεναρίου Χρήσης</b>	<b>Καταχώρηση Φαρμάκων</b>	
<b>Συνοπτική Περιγραφή</b>	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ο διαχειριστής του συστήματος ώστε να καταχωρήσει νέα φάρμακα στην βάση ή να αλλάξει τα στοιχεία ήδη υπαρχόντων Φαρμάκων:	
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	Ο διαχειριστής του Συστήματος.	
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει την καρτέλα για καταχώρηση φαρμάκου όπου εμφανίζεται λίστα με τα είδη καταχωρημένα φάρμακα και τα χαρακτηριστικά τους.</li> <li>2. Ο χρήστης πατάει το «ADD» button (επιλέγει προσθήκη).</li> <li>3. Το σύστημα εμφανίζει μια κενή εγγραφή για την νέα καταχώρηση.</li> <li>4. Ο χρήστης συμπληρώνει όλα τα απαιτούμενα χαρακτηριστικά.</li> <li>5. Ο χρήστης επιλέγει καταχώρηση.</li> <li>6. Το σύστημα ελέγχει αν έχουν συμπληρωθεί όλα τα απαιτούμενα χαρακτηριστικά για την επιτυχή ολοκλήρωση της καταχώρησης.</li> <li>7. Το φάρμακο καταχωρείται.</li> <li>8. Το σενάριο τερματίζει.</li> </ol>	
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<p>Μη επαρκή καταχωρημένα στοιχεία φαρμάκου στην φόρμα. Από το βήμα 6 της βασικής ροής.</p> <p>7.b. Το σύστημα επιστρέφει μήνυμα λάθους και προτρέπει τον χρήστη να καταχωρίσει όλα τα απαιτούμενα στοιχεία σωστά.</p> <p>Το σενάριο συνεχίζει από το βήμα 4 της βασικής ροής.</p>

	<b>2</b>	<p>Αλλαγή στοιχείων Φαρμάκου. Από το βήμα 1 της βασικής ροής.</p> <p>2.c. Ο χρήστης επιλέγει «EDIT» (επεξεργασία στοιχείων φαρμάκου).</p> <p>3.c. Τα στοιχεία των φαρμάκων στην λίστα γίνονται επεξεργάσιμα (editable).</p> <p>4.c. Ο χρήστης αλλάζει όποια στοιχεία επιθυμεί.</p> <p>5.c. Ο χρήστης επιλέγει «καταχώρηση».</p> <p>6.c. Το σύστημα ανανεώνει τα στοιχεία που αλλάχτηκαν στη βάση δεδομένων της εφαρμογής.</p> <p>Το σενάριο τερματίζει (8 βασική ροή).</p>
	<b>3</b>	<p>Διαγραφή Φαρμάκων. Από το βήμα 1 της βασικής ροής:</p> <p>2.d. Ο χρήστης επιλέγει το φάρμακο το οποίο επιθυμεί να διαγράψει.</p> <p>3.d. Ο χρήστης επιλέγει διαγραφή «Delete».</p> <p>4.d. Το φάρμακο διαγράφεται επιτυχώς από την βάση.</p> <p>Το σενάριο τερματίζει (8 βασική ροή).</p>
<b>Preconditions</b>	<b>1</b>	<p>Να έχει πρόσβαση ο διαχειριστής του συστήματος στην οθόνη καταχώρησης φαρμάκου.</p>
<b>Postconditions</b>	<b>1</b>	<p>Με την επιτυχή ολοκλήρωση του σεναρίου χρήσης ένα νέο φάρμακο θα έχει εισαχθεί στο σύστημα ή θα έχουν αλλαχτεί τα στοιχεία κάποιου φαρμάκου στο σύστημα.</p>
<b>Διάγραμμα Περίπτωσης Χρήσης</b>	<pre> graph TD     subgraph uc_Katachorisi_Farmakou [uc Καταχώρηση Φαρμάκου]         direction TB         UC1((Καταχώρηση Φαρμάκων))         UC2((Συντήρηση Βάσεις Φαρμάκων (Αλλαγές - Αποσύρσεις)))         UC1 -.-&gt; «extend»  UC2     end     Admin[Admin (from Actors)] --- UC1   </pre>	

**Activity Diagram**





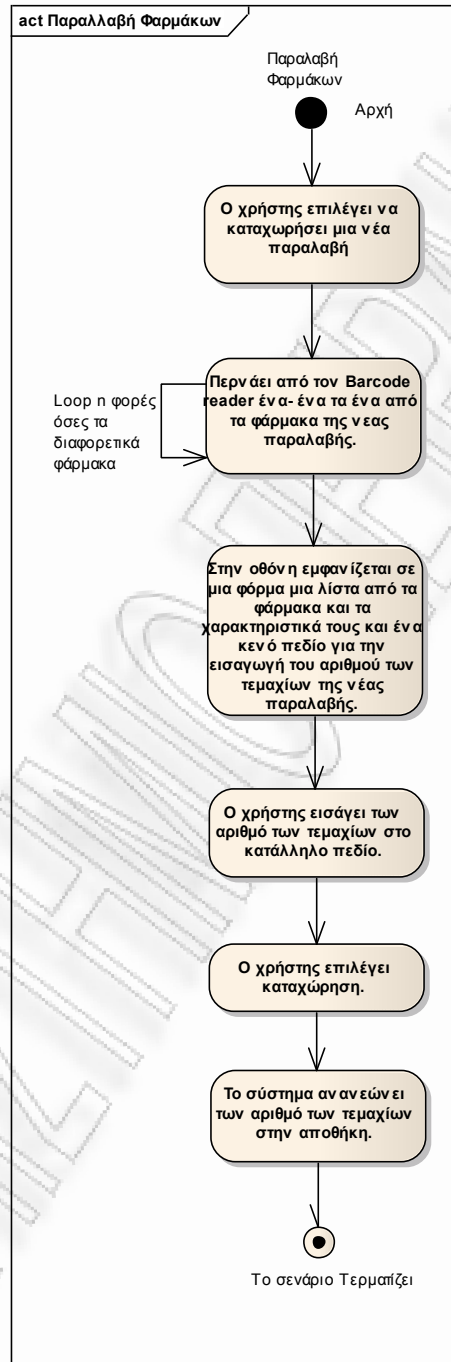
#### 4.1.5. Λειτουργίες Υπεύθυνου Φαρμακείου

##### 6.1 Παραλαβή Φαρμάκων

<b>Όνομα Σεναρίου Χρήσης</b>	<b>Παραλαβή Φαρμάκων</b>		
<b>Συνοπτική Περιγραφή</b>	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ο υπεύθυνος ενός φαρμακείου ώστε να καταχωρήσει μια νέα παραλαβή φαρμάκου στο σύστημα.		
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	Ο Υπεύθυνος του Φαρμακείου.		
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"><li>1. Ο χρήστης επιλέγει να καταχωρήσει μια νέα παραλαβή.</li><li>2. Περνάει από τον Barcode reader ένα- ένα τα φάρμακα της νέας παραλαβής.</li><li>3. Στην οθόνη εμφανίζεται, σε μια φόρμα, μια λίστα από τα φάρμακα και τα στοιχεία του καθώς και ένα κενό πεδίο για την εισαγωγή της ποσότητας της νέας παραλαβής.</li><li>4. Ο χρήστης εισάγει την ποσότητα του κάθε φαρμάκου στο κατάλληλο πεδίο.</li><li>5. Ο χρήστης επιλέγει «καταχώρηση».</li><li>6. Το σύστημα ανανεώνει την ποσότητα του φαρμάκου, που είναι διαθέσιμο στην αποθήκη, στη βάση δεδομένων.</li><li>7. Το σενάριο τερματίζει.</li></ol>		
<b>Εναλλακτικές Ροές</b>	<table border="1"><tr><td><b>1</b></td><td>Μη καταχωρημένο Φάρμακο. Από το βήμα 2 της βασικής ροής.  3.b. Το σύστημα επιστρέφει μήνυμα λάθους.  Ο χρήστης συνεχίζει με το πέρασμα του επόμενου φαρμάκου. Βήμα 2 της βασικής ροής.</td></tr></table>	<b>1</b>	Μη καταχωρημένο Φάρμακο. Από το βήμα 2 της βασικής ροής.  3.b. Το σύστημα επιστρέφει μήνυμα λάθους.  Ο χρήστης συνεχίζει με το πέρασμα του επόμενου φαρμάκου. Βήμα 2 της βασικής ροής.
<b>1</b>	Μη καταχωρημένο Φάρμακο. Από το βήμα 2 της βασικής ροής.  3.b. Το σύστημα επιστρέφει μήνυμα λάθους.  Ο χρήστης συνεχίζει με το πέρασμα του επόμενου φαρμάκου. Βήμα 2 της βασικής ροής.		

	2	<p>Στην περίπτωση που το φαρμακείο παρέχει RFID reader. Από το βήμα 1 της βασικής ροής.</p> <p>2.c. Ο υπεύθυνος περνάει από τον reader όλο το κιβώτιο με τη νέα παραλαβή.</p> <p>3.c. Στην οθόνη εμφανίζονται τα χαρακτηριστικά των φαρμάκων και οι ποσότητες που περιέχει το κιβώτιο.</p> <p>4.c. Ο χρήστης επιλέγει «παραλαβή».</p> <p>5.c. Η ποσότητες των φαρμάκων, που είναι διαθέσιμα στην αποθήκη του φαρμακείου καταχωρούνται στη βάση δεδομένων της εφαρμογής.</p> <p>Το σενάριο τερματίζει. Βήμα 7 της βασικής ροής.</p>
<b>Preconditions</b>	1	Να έχει πρόσβαση ο υπεύθυνος στην οθόνη παραλαβής φαρμάκων.
	2	Να έχει καταχωρηθεί το φάρμακο από τον διαχειριστή του συστήματος.
	3	Να έχει συνδεθεί η αποθήκη του φαρμακείου με το φαρμακείο από τον διαχειριστή του συστήματος.
<b>Postconditions</b>	1	Με την επιτυχή ολοκλήρωση του σεναρίου χρήσης θα έχει ενημερωθεί η αποθήκη του φαρμακείου με τα νέα αποθέματα.
<b>Διάγραμμα Περίπτωσης Χρήσης</b>	<pre> graph TD     subgraph System         UC1((Καταχώρηση Φαρμάκων))         UC2((Συντήρηση Βάσεις Φαρμάκων (Αλλαγές - Αποσύρσεις)))         UC1 -.-&gt; «extend»  UC2     end     Admin[Admin (from Actors)] --- UC1   </pre>	

## Activity Diagram



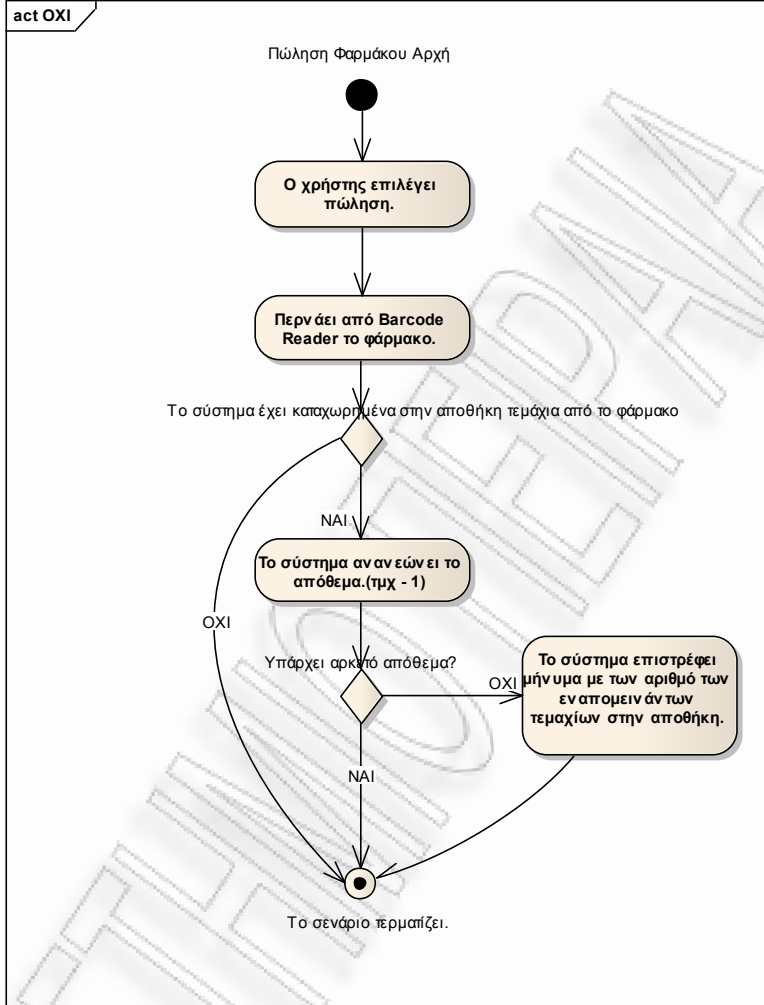
#### 4.1.6. Λειτουργίες Υπάλληλου Φαρμακείου

##### Πώληση φαρμάκου.

<b>Όνομα Σεναρίου Χρήσης</b>	<b>Πώληση φαρμάκου</b>	
<b>Συνοπτική Περιγραφή</b>	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ένας υπάλληλος φαρμακείου ώστε να πουλήσει ένα φάρμακο (στην πραγματικότητα, εμμέσως, να ανανεώσει το απόθεμα της αποθήκης).	
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	Ο Υπάλληλος του Φαρμακείου.	
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"> <li>1. Ο χρήστης επιλέγει «πώληση».</li> <li>2. Περνάει από Barcode Reader ή RFID reader το φάρμακο.</li> <li>3. Το σύστημα, αν έχει καταχωρημένη στην αποθήκη ποσότητα μεγαλύτερη του μηδενός από το συγκεκριμένο φάρμακο, ανανεώνει το απόθεμα μειώνοντάς το κατά μία μονάδα (τμχ = τμχ - 1),</li> <li>4. Το σενάριο τερματίζει.</li> </ol>	
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<p>Μικρό απόθεμα στην αποθήκη μετά την πώληση. Από το βήμα 3 της βασικής ροής :</p> <p>4.b. Το σύστημα επιστρέφει μήνυμα με τον αριθμό των εναπομεινάντων τεμαχίων στην αποθήκη.</p> <p>5.b. Ο χρήστης επιλέγει «οκ».</p> <p>Το σενάριο τερματίζει. Βήμα 4 της βασικής ροής.</p>
<b>Preconditions</b>	<b>1</b>	Να έχει πρόσβαση ο υπάλληλος του φαρμακείου στην οθόνη πώλησης φαρμάκου.
	<b>2</b>	Να έχει γίνει η παραλαβή από τον υπεύθυνο του φαρμακείου.

<b>Postconditions</b>	<b>1</b> Με την επιτυχή ολοκλήρωση του σεναρίου χρήσης θα έχει ενημερωθεί η αποθήκη του φαρμακείου με τα νέα αποθέματα. Θα έχουν αφαιρεθεί τα πουλημένα τεμάχια.
<b>Διάγραμμα Περίπτωσης Χρήσης</b>	<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: right; margin-right: 10px;">uc Πώληση Φαρμάκου</p> <pre> graph TD     Actor[πάλληλος Φαρμακείου (from Actors)] --- UC1((Πώληση Φαρμάκου))     UC1 -.-&gt; «include»  UC2((Μείωση Αποθέματος Στην Αποθήκη))     UC1 -.-&gt; «extend»  UC3((Ενημέρωση Περιορισμένου Αποθέματος))     </pre> </div>

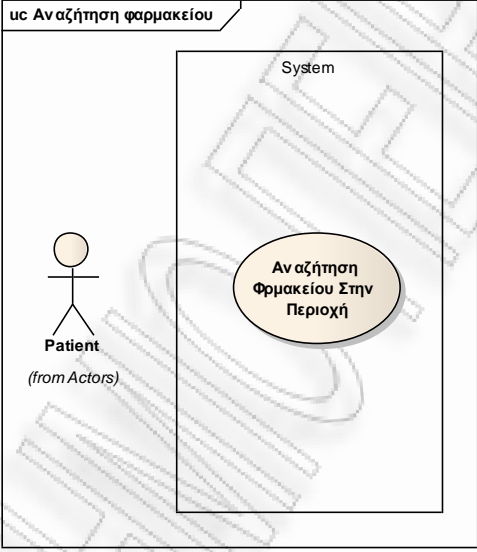
## Activity Diagram



#### 4.1.7. Λειτουργίες Ασθενή – Αναζητητή (Εφαρμογή Αναζήτησης Μέσω Διαδικτύου)

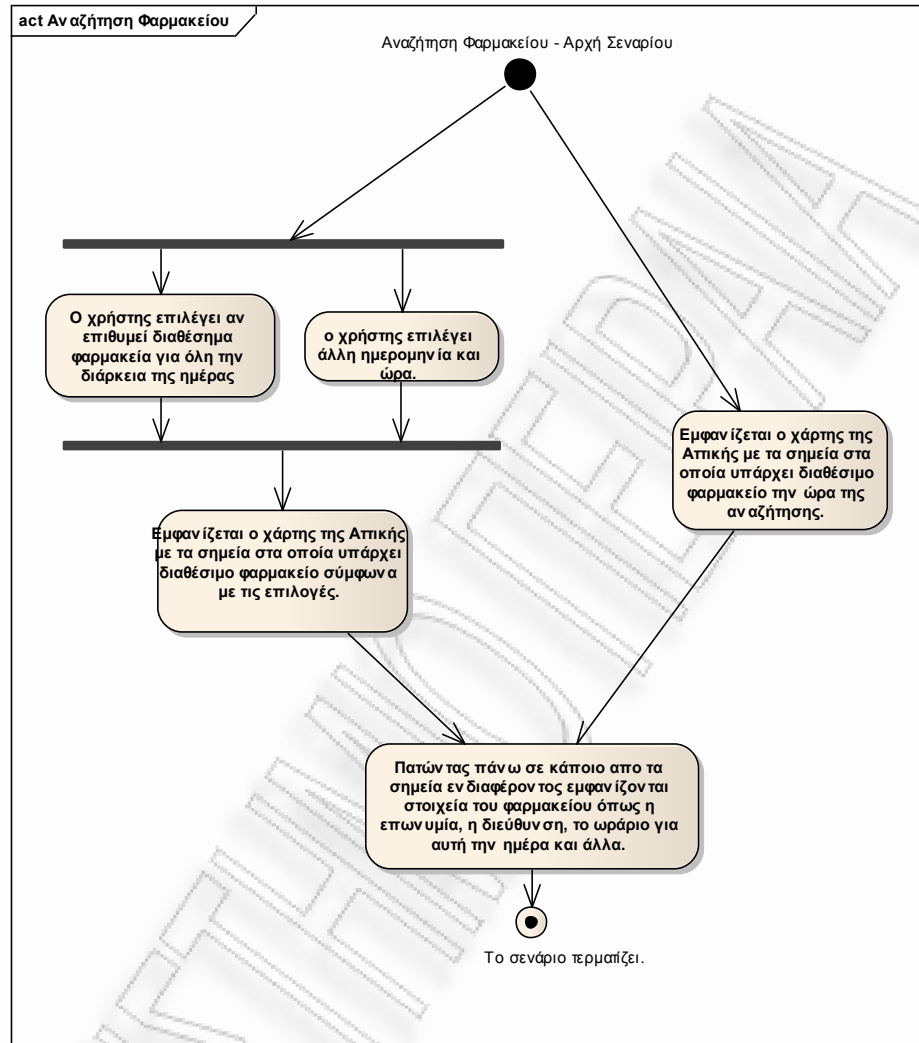
##### Αναζήτηση Φαρμακείου

Όνομα Σεναρίου Χρήσης	<b>Αναζήτηση Φαρμακείου</b>	
Συνοπτική Περιγραφή	Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ένας ασθενής – αναζητητής ώστε να βρει το φαρμακείο που επιθυμεί.	
Βασικός Πρωταγωνιστής (Primary Actor)	Ο Ασθενής.	
Βασική Ροή Λειτουργίας	<ol style="list-style-type: none"> <li>1. Ο χρήστης επισκέπτεται την ιστοσελίδα της εφαρμογής Apothecary.</li> <li>2. Εμφανίζεται ο χάρτης της Αττικής με τα σημεία στα οποία υπάρχει διαθέσιμο φαρμακείο την ώρα της αναζήτησης.</li> <li>3. Ο χρήστης μπορεί να περιηγηθεί στον χάρτη για να εντοπίσει το σημείο που τον ενδιαφέρει.</li> <li>4. Πατώντας πάνω στο σημείο ενδιαφέροντος εμφανίζονται τα στοιχεία του φαρμακείου όπως η επωνυμία, η διεύθυνση, το ωράριο για αυτή την ημέρα και άλλα.</li> <li>5. Το σενάριο τερματίζει.</li> </ol>	
Εναλλακτικές Ροές	<b>1</b>	<p>Ο χρήστης επιθυμεί να αναζητήσει διαθέσιμα φαρμακεία για άλλη μέρα και ώρα. Από το βήμα 2 της βασικής ροής :</p> <p>3.b. Ο χρήστης επιλέγει «άλλη ημερομηνία και ώρα». Επιλέγει «Αναζήτηση».</p> <p>4.b. Το σύστημα εμφανίζει στον χάρτη τα διαθέσιμα φαρμακεία για την μέρα και ώρα την οποία έχει ορίσει ο χρήστης.</p> <p>Το σενάριο συνεχίζει από το βήμα 4 της βασικής ροής.</p>
		<p>Αν επιθυμεί να αναζητήσει τα διαθέσιμα φαρμακεία όλη την μέρα επιλέγει στην ώρα όλη την ημέρα «All Day».</p>

<b>Preconditions</b>	<b>1</b>	Να έχουν γίνει οι απαιτούμενες ενέργειες από τον Διαχειριστή του συστήματος. Να έχει καταχωρήσει τα φαρμακεία.
<b>Postconditions</b>	<b>1</b>	Με την επιτυχή ολοκλήρωση του σεναρίου ο χρήστης θα έχει ενημερωθεί με τα διαθέσιμα φαρμακεία την μέρα την ώρα και στην περιοχή που τον ενδιαφέρει.
<b>Διάγραμμα Περίπτωσης Χρήσης</b>	 <p>The diagram is a UML Use Case Diagram titled 'uc Αναζήτηση φαρμακείου'. It features an actor labeled 'Patient (from Actors)' on the left. To the right is a rectangular system boundary labeled 'System'. Inside the system boundary is an oval use case labeled 'Αναζήτηση Φαρμακείου Στην Περιοχή'.</p>	



**Activity Diagram**



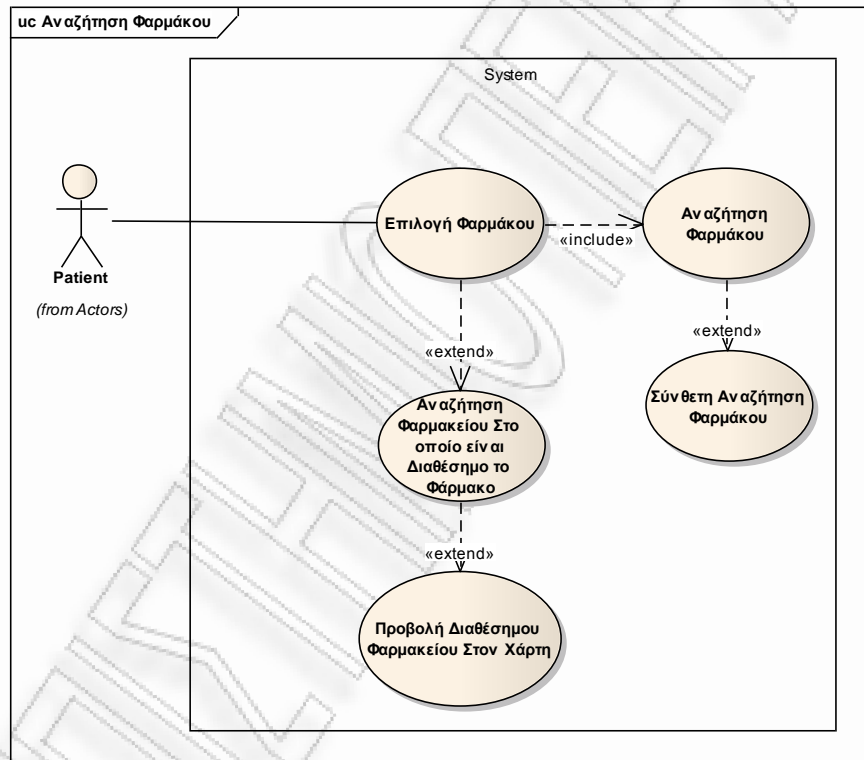
Actor Action		System Response	
1	Ο χρήστης επιλέγει αν επιθυμεί συγκεκριμένη ημερομηνία και ώρα.	2	Το σύστημα εμφανίζει στον χάρτη τα διαθέσιμα φαρμακεία.
3	Ο χρήστης επιλέγει ένα από αυτά τα φαρμακεία.	4	Το σύστημα εμφανίζει τα στοιχεία του φαρμακείου.

#### 4.2. Αναζήτηση Φαρμάκου

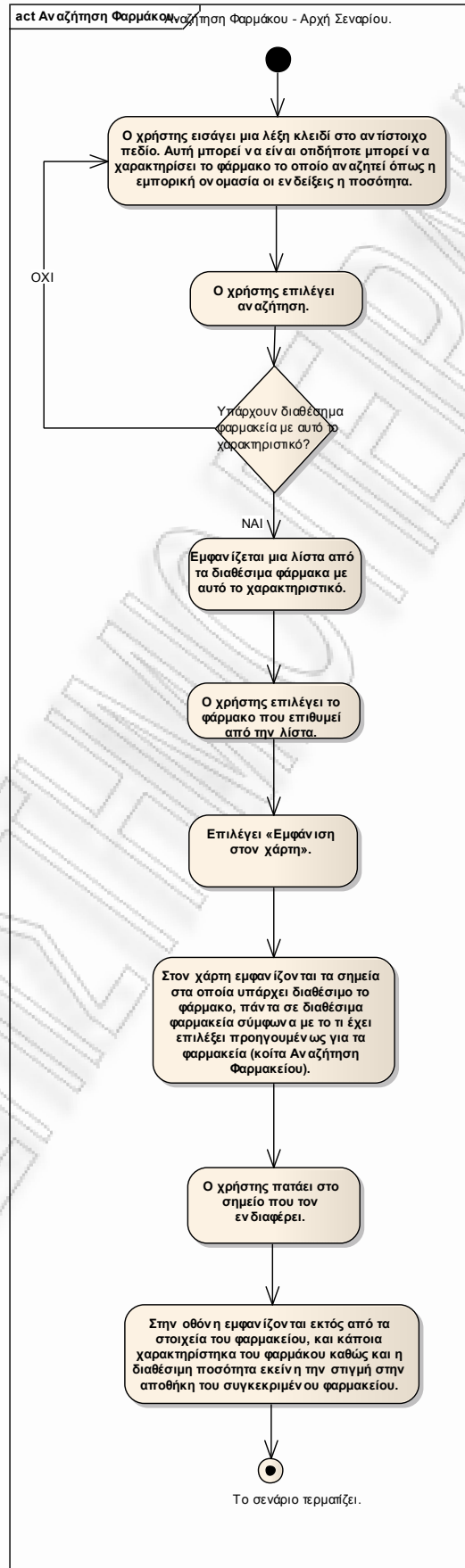
<b>Όνομα Σεναρίου Χρήσης</b>	<b>Αναζήτηση Φαρμάκου</b>	
<b>Συνοπτική Περιγραφή</b>	<p>Αυτό το σενάριο χρήσης περιγράφει τα βήματα που ακολουθεί ένας ασθενής έτσι ώστε να βρει το φάρμακο που επιθυμεί και να εντοπίσει στην συνέχεια, αν επιθυμεί, σε ποιο φαρμακείο στην περιοχή που τον ενδιαφέρει είναι διαθέσιμο.</p>	
<b>Βασικός Πρωταγωνιστής (Primary Actor)</b>	<p>Ο Ασθενής.</p>	
<b>Βασική Ροή Λειτουργίας</b>	<ol style="list-style-type: none"> <li>1 Ο χρήστης επισκέπτεται την ιστοσελίδα αναζήτησης φαρμακείου και φαρμάκου.</li> <li>2 Εισάγει μια λέξη κλειδί στο αντίστοιχο πεδίο. Αυτή μπορεί να είναι οτιδήποτε μπορεί να χαρακτηρίσει το φάρμακο το οποίο αναζητεί όπως, η εμπορική ονομασία, οι ενδείξεις και άλλα.</li> <li>3 Επιλέγει «αναζήτηση».</li> <li>4 Στην οθόνη εμφανίζεται μια λίστα με τα φάρμακα που αντιστοιχούν στο κριτήριο αναζήτησης του χρήστη.</li> <li>5 Ο χρήστης επιλέγει το φάρμακο που επιθυμεί από τη λίστα.</li> <li>6 Στον χάρτη εμφανίζονται τα σημεία στα οποία είναι διαθέσιμο το φάρμακο, σύμφωνα με τα κριτήρια που έχει επιλέξει προηγουμένως για την αναζήτηση φαρμακείου (βλέπε Αναζήτηση Φαρμακείου).</li> <li>7 Όταν ο χρήστης πατήσει στο σημείο ενδιαφέροντος της επιλογής του, εκτός από τα στοιχεία του φαρμακείου, εμφανίζονται και κάποια στοιχεία; του φαρμάκου, καθώς και η διαθέσιμη ποσότητα στην αποθήκη του συγκεκριμένου φαρμακείου τη δεδομένη στιγμή.</li> <li>8 Το σενάριο τερματίζει.</li> </ol>	
<b>Εναλλακτικές Ροές</b>	<b>1</b>	<p>Στο βήμα 4 αν δεν βρεθεί κάποιο φάρμακο με αυτά τα χαρακτηριστικά ή αν δεν είναι κάποιο από αυτά που εμφανίζονται στη λίστα αυτό που ο χρήστης επιθυμεί, μπορεί να επαναλάβει την αναζήτηση με άλλη λέξη κλειδί. Συνεχίζει λοιπόν από το βήμα 3 της βασικής ροής.</p>

<b>Preconditions</b>	<b>1</b>	Να έχουν γίνει οι απαιτούμενες ενέργειες από τον Διαχειριστή του συστήματος. Να έχει καταχωρήσει τα φάρμακα και τα φαρμακεία.
<b>Postconditions</b>	<b>1</b>	Με την επιτυχή ολοκλήρωση του σεναρίου ο χρήστης θα έχει ενημερωθεί με τα διαθέσιμα φάρμακα και θα έχει ενημερωθεί με το που θα τα βρει σύμφωνα με την μέρα, την ώρα, και την περιοχή που τον ενδιαφέρει.

**Διάγραμμα  
Περίπτωσης  
Χρήσης**



**Activity Diagram**



Actor Action		System Response	
1	Επισκέπτεται τη σελίδα αναζήτησης φαρμακείου και φαρμάκου.	2	Το σύστημα εμφανίζει τα ανοιχτά φαρμακεία στην περιοχή.
3	Εισάγει μια λέξη κλειδί στο αντίστοιχο πεδίο που βρίσκεται στο πάνω μέρος της σελίδας. Η λέξη αναζήτησης μπορεί να είναι οτιδήποτε μπορεί να χαρακτηρίσει το φάρμακο το οποίο αναζητεί όπως: η εμπορική ονομασία, οι ενδείξεις, η ποσότητα.	4	Στην οθόνη εμφανίζεται μια λίστα από τα διαθέσιμα φάρμακα με αυτό το χαρακτηριστικό.
5	Ο χρήστης επιλέγει το φάρμακο που επιθυμεί από αυτά.	6	Στον χάρτη εμφανίζονται τα σημεία στα οποία είναι διαθέσιμο το φάρμακο, σύμφωνα με τα κριτήρια που έχει επιλέξει προηγουμένως για την αναζήτηση φαρμακείου (βλέπε Αναζήτηση Φαρμακείου).
7	Ο χρήστης πατάει στο σημείο που τον ενδιαφέρει	8	Στην οθόνη εμφανίζονται, εκτός από τα στοιχεία του φαρμακείου, και κάποια χαρακτηριστικά του φαρμάκου καθώς και η διαθέσιμη ποσότητα στην αποθήκη του συγκεκριμένου φαρμακείου τη δεδομένη στιγμή.

## 5. Εγχειρίδιο Βάσης

### 5.1 Περιγραφή προβλήματος

Το μοντέλο περιγράφει τη διαδικασία αναζήτησης και διαχείρισης φαρμακείων και των φαρμάκων τα οποία διαθέτουν στην αποθήκη τους.

Τα φαρμακεία, έχουν εφημερίες. Επίσης, έχουν αποθέματα φαρμάκων στην αποθήκη τους. Με την πώληση κάποιου φαρμάκου τα αποθέματα αυτά μειώνονται και με την παραλαβή αυξάνονται. Τα φάρμακα ανήκουν σε διάφορες κατηγορίες και έχουν διάφορα χαρακτηριστικά όπως ενδείξεις, τρόπο χρήσης, είδος συσκευασίας. Κάποια από αυτά συνταγογραφούνται ενώ κάποια άλλα όχι, κάποια πληρώνονται, πλήρως ή μερικώς, από το ταμείο ασφάλισης του ασθενή-χρήστη, ενώ κάποια άλλα είναι υποχρεωμένος να τα πληρώσει ο ίδιος.

### 5.2 Κατάλογος Περιγραφής Οντοτήτων

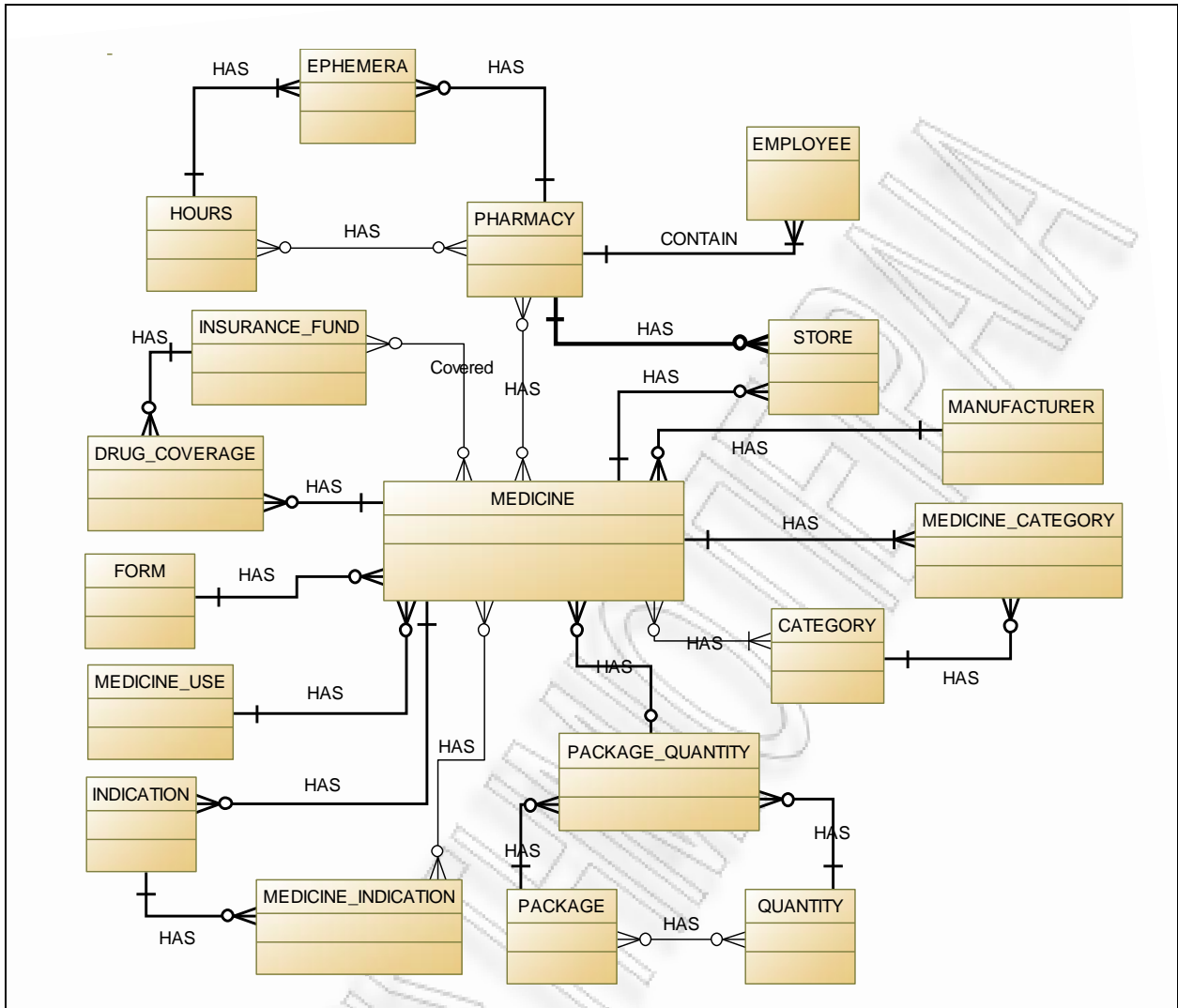
Παρακάτω παρατίθεται ο αναλυτικός κατάλογος των οντοτήτων – πινάκων της **εφαρμογής αναζήτησης φαρμάκου σε διαθέσιμο φαρμακείο και διαχείριση φαρμάκων στην αποθήκη**, όπου φαίνεται ο χαρακτηρισμός κάθε οντότητας καθώς και το διάγραμμα οντοτήτων – σχέσεων.

ΟΝΤΟΤΗΤΑ	ΠΕΡΙΓΡΑΦΗ ΟΝΤΟΤΗΤΑΣ	ΣΧΟΛΙΑ / ΕΠΙΣΗΜΑΝΣΕΙΣ
<b>MEDICINE</b>	Φάρμακο ή γενικά οποιοδήποτε προϊόντα διαθέτει ένα φαρμακείο προς πώληση. (φαρμακευτικά, καλλυντικά σκευάσματα)	Για να ξεχωρίσουμε τα φάρμακα από τα υπόλοιπα σκευάσματα χρησιμοποιούμε την οντότητα ΚΑΤΗΓΟΡΙΑ φαρμάκου.
<b>PHARMACY</b>	Σημεία πώλησης φαρμάκων. Τα καταστήματα.	
<b>STORE</b>	Ενδιάμεση οντότητα η οποία συνδέει συγκεκριμένο φάρμακο με συγκεκριμένο σημείο πώλησης του (φαρμακείο).	Σε κάθε συγκεκριμένο φάρμακο συγκεκριμένης αποθήκης φαρμακείου αντιστοιχεί μια τιμή που αποτελεί το stock του φαρμάκου (διαθέσιμη ποσότητα).
<b>HOURS</b>	Ωράρια καταστημάτων.	Θεωρούμε ότι κάθε μέρα είναι χωρισμένη σε έξη τετράωρα.

ΟΝΤΟΤΗΤΑ	ΠΕΡΙΓΡΑΦΗ ΟΝΤΟΤΗΤΑΣ	ΣΧΟΛΙΑ / ΕΠΙΣΗΜΑΝΣΕΙΣ
<b>EPHEMERA</b>	Οντότητα που συνδέει κάποια ωράρια με φαρμακεία. Αντιπροσωπεύει τις ώρες λειτουργίας ενός καταστήματος ανά μέρα.	Κάθε φαρμακείο μπορεί να λειτουργεί από κανένα έως πολλά ωράρια και σε κάθε ωράριο μπορεί να λειτουργούν πολλά καταστήματα. Γι' αυτό χρειαστήκαμε ενδιάμεσο πίνακα.
<b>CATEGORY</b>	Αυτή η οντότητα περιέχει τις διαθέσιμες κατηγορίες φαρμάκων.	Αναλγητικό, καλλυντικό...
<b>MEDICINE_CATEGORY</b>	Ενδιάμεσος πίνακας που συνδέει ένα φάρμακο με την κατηγορία στην οποία ανήκει.	Κάθε φάρμακο ανήκει σε μια ή περισσότερες κατηγορίες και σε κάθε κατηγορία ανήκουν παραπάνω από ένα φάρμακα.
<b>INDICATION</b>	Οντότητα με τις διαθέσιμες ενδείξεις φαρμάκων.	Αϋπνία, πονοκέφαλο...
<b>MEDICINE_INDICATION</b>	Ενδιάμεση οντότητα για τις ενδείξεις που έχει το κάθε φάρμακο.	Κάθε φάρμακο μπορεί να έχει πολλές ενδείξεις και κάθε ένδειξη μπορεί να την έχουν πολλά φάρμακα.
<b>MANUFACTURER</b>	Οντότητα με τους κατασκευαστές των φαρμάκων.	
<b>MEDICINE_USE</b>	Οντότητα με χρήσεις που μπορεί να έχουν τα φάρμακα.	
<b>INSURANCE_FUND</b>	Οντότητα με τα ασφαλιστικά ταμεία.	
<b>DRUG_COVERAGE</b>	Ενδιάμεση οντότητα η οποία παρουσιάζει την κάλυψη του φαρμάκου από κάποια ταμεία.	Κάθε ασφαλιστικό ταμείο μπορεί να δικαιολογεί σε κάποιο ποσοστό περισσότερα από ένα φάρμακα. Κάθε φάρμακο μπορεί να καλύπτεται από περισσότερα

ΟΝΤΟΤΗΤΑ	ΠΕΡΙΓΡΑΦΗ ΟΝΤΟΤΗΤΑΣ	ΣΧΟΛΙΑ / ΕΠΙΣΗΜΑΝΣΕΙΣ
		από ένα ταμεία.
<b>FORM</b>	Οντότητα με τις μορφές που μπορεί να έχει ένα φάρμακο.	Αλοιφή, χάπι...
<b>PACKAGE</b>	Οντότητα με τις συσκευασίες που μπορεί να έχει ένα φάρμακο.	Μπουκάλι, σωληνάριο ...
<b>QUANTITY</b>	Διαθέσιμες ποσότητες φαρμάκων.	Περιέχει και την μονάδα μέτρησης. Π.χ. ml, τμ, χάπια...
<b>PACKAGE_QUANTITY</b>	Ποσότητα ανά συσκευασία.	Κάθε φάρμακο έχει μια ποσότητα ανά συσκευασία. Κάθε ποσότητα ανά συσκευασία μπορεί να την έχουν πάνω από ένα φάρμακα.
<b>EMPLOYEE</b>	Εργαζόμενοι στο φαρμακείο.	Κάθε φαρμακείο έχει πάνω από έναν υπάλληλο κάθε υπάλληλος θεωρούμε ότι δουλεύει σε ένα μόνο φαρμακείο.
<b>STATISTICS</b>	Στατιστικά αναζήτησης.	Κάθε γραμμή σε αυτό τον πίνακα μας δίνει την πληροφορία για την αναζήτηση φαρμακείου ή και φαρμάκου το οποίο αναζητήθηκε σε αυτό.





<b>ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ</b>	<b>ΔΙΑΔΙΚΑΣΙΕΣ ΕΥΡΕΣΗΣ ΦΑΡΜΑΚΟΥ ΣΕ ΔΙΑΘΕΣΙΜΟ ΦΑΡΜΑΚΕΙΟ</b>	<b>18/09/2011</b>	<b>V 1.0</b>
<b>ΜΟΝΤΕΛΟ ΟΝΤΟΤΗΤΩΝ ΣΥΣΧΕΤΙΣΕΩΝ – ER</b>		<b>©UNIPi</b>	

- Το μοντέλο περιγράφει την διαδικασία αναζήτησης και διαχείρισης φαρμακείων και των φαρμάκων τα οποία διαθέτουν στην αποθήκη τους.
- Το φαρμακείο μπορεί να είναι ανοικτό εκτός από τα σπάνια ωράρια και σε άλλες βάρδιες στις οποίες εφημερεύει. Θεωρούμε ότι υπάρχουν κάποια standard ωράρια κάθε μέρα χωρισμένα ανά τετράωρα .
- Σε ίδια ωράρια μπορεί λειτουργούν περισσότερα από ένα φαρμακεία καθώς και κάθε φαρμακείο μπορεί να έχει πολλά ωράρια.
- Το φαρμακείο μπορεί να έχει από κανένα έως πολλά τεμάχια από το ίδιο φάρμακο.
- Κάθε φάρμακο μπορεί να υπάρχει διαθέσιμο από κανένα έως σε πολλά φαρμακεία.
- Κάθε φάρμακο έχει από καμία έως πολλές ενδείξεις.
- Κάθε ένδειξη μπορεί να υπάρχει από κανένα έως σε πολλά φάρμακα.
- Κάθε είδος συσκευασίας μπορεί να περιέχει διάφορες ποσότητες.
- Κάθε ποσότητα μπορεί να είναι διαθέσιμη σε παραπάνω από ένα φάρμακο.
- Θεωρούμε ότι κάθε φάρμακο μπορεί να έχει μια χρήση.
- Κάθε χρήση μπορεί να γίνεται από πολλά φάρμακα.
- Κάθε φάρμακο μπορεί να μην δικαιολογείται από κανένα ταμείο ή να δικαιολογείται από πολλά ταμεία.
- Κάθε ταμείο μπορεί να δικαιολογεί παραπάνω από ένα φάρμακο.
- Κάθε φαρμακείο έχει πάνω από έναν υπάλληλο και κάθε υπάλληλος θεωρούμε ότι δουλεύει σε ένα μόνο φαρμακείο.

### 5.3 Χαρακτηριστικά Οντοτήτων

ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
MEDICINE				
	MedicineId	Κωδικός φαρμάκου.	VARCHAR (20)	PK
	MedicineName	Ονομασία φαρμάκου.	VARCHAR (45)	
	Description	Περιγραφή φαρμάκου.	VARCHAR (50)	
	Barcode	Κωδικός Barcode.	VARCHAR (20)	
	RFIDCode	Κωδικός RFID (EPC code).	VARCHAR (20)	
	PartNumber	Παρτίδα παραγωγής φαρμάκου.		
	MinItems	Είναι η ποσότητα αποθέματος του φαρμάκου στην αποθήκη για την οποία θεωρείται ότι υπάρχει έλλειψη και ενημερώνεται ο φαρμακοποιός.	INTEGER	
	NeedPrescriptionFlag	Flag για την απαίτηση ιατρικής συνταγής. Χρησιμοποιείται η τιμή '1' για το true και '0' για το false.	BIT	
	MedicineUsed	Κωδικός Χρήσης Φαρμάκου.	INTEGER	FK
	FormId	Κωδικός μορφής φαρμάκου.	INTEGER	FK

ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
	PackageQuantityId	Κωδικός για τον συνδυασμό συσκευασίας και ποσότητας.	VARCHAR (20)	FK
	ManufacturerId	Κωδικός κατασκευαστή φαρμάκου.	BIGINT	FK
<b>PHARMACY</b>				
	PharmacyId	Κωδικός φαρμακείου.	VARCHAR (20)	PK
	PharmacyName	Επωνυμία φαρμακείου.	VARCHAR (45)	
	Address	Διεύθυνση.	VARCHAR (100)	
	Telephone	Τηλέφωνο επικοινωνίας φαρμακείου.	VARCHAR (20)	
	ManagerName	Όνομα υπεύθυνου φαρμακείου.	VARCHAR (45)	
	Longitude	Γεωμετρικό μήκος.		
	Latitude	Γεωμετρικό πλάτος.		
<b>STORE</b>				
	StoreId	Κωδικός αποθήκης.	INTEGER	PK
	MedicineId	Κωδικός φαρμάκου.	VARCHAR (45)	
	PharmacyId	Κωδικός φαρμακείου.	VARCHAR (45)	
	stock	Απόθεμα του φαρμάκου στη	INTEGER	

ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
		συγκεκριμένη αποθήκη.		
<b>HOURS</b>				
	HoursId	Κωδικός Ωραρίου.	BIGINT	PK
	BusinessDate	Ημερομηνία Ωραρίου.	VARCHAR (10)	
	TimeStart	Ώρα έναρξης ωραρίου.	TIMESTAMP	
	TimeEnd	Ώρα λήξης ωραρίου.	TIMESTAMP	
<b>EPHEMERA</b>				
	EphemeralId	Κωδικός Εφημερίας. Συγκεκριμένου ωραρίου συγκεκριμένου φαρμακείου για μια συγκεκριμένη μέρα.	VARCHAR (20)	PK
	HoursId	Κωδικός Ωραρίου.	BIGINT	FK
	PharmacyId	Κωδικός Φαρμακείου.	VARCHAR (20)	FK
<b>CATEGORY</b>				
	CategoryId	Κωδικός κατηγορίας.	INTEGER	PK
	Description	Περιγραφή κατηγορίας φαρμάκου.	VARCHAR (50)	
<b>MEDICINE_CATEGORY</b>				
	MedicineCategoryId	Κωδικός κατηγορίας συγκεκριμένου	VARCHAR (20)	PK

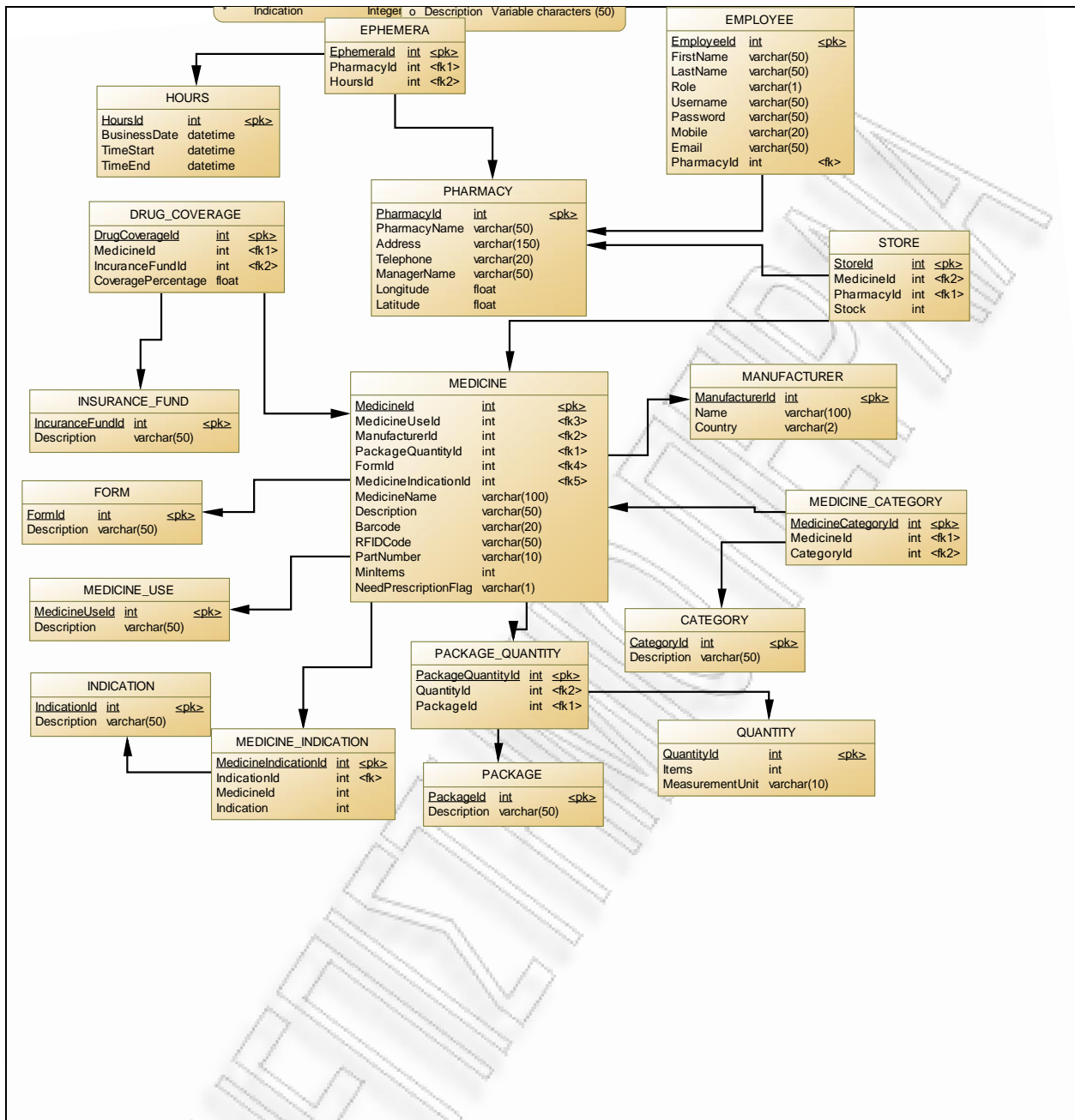
ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
		φαρμάκου.		
	MedicineId	Κωδικός φαρμάκου.	VARCHAR (20)	FK
	CategoryId	Κωδικός κατηγορίας.	INTEGER	FK
<b>INDICATION</b>				
	IndicationId	Κωδικός ένδειξης.	INTEGER	PK
	Description	Περιγραφή ένδειξης φαρμάκων.	VARCHAR (50)	
<b>MEDICINE_INDICATION</b>				
	MedicineIndicationId	Κωδικός ένδειξης ανά φάρμακο.	VARCHAR (20)	PK
	MedicineId	Κωδικός φαρμάκου.	VARCHAR (20)	FK
	IndicationId	Κωδικός ένδειξης.	INTEGER	FK
<b>MANUFACTURER</b>				
	ManufacturerId	Κωδικός κατασκευαστή φαρμάκου.	BIGINT	PK
	Name	Επωνυμία κατασκευαστή.	VARCHAR (50)	
	Country	Χώρα που εδρεύει ο κατασκευαστής.	VARCHAR (20)	
<b>MEDICINE_USE</b>				
	MedicineUseId	Κωδικός Χρήσης Φαρμάκου.	INTEGER	PK
	Description	Περιγραφή χρήσης	VARCHAR	

ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
		του φαρμάκου.	(20)	
<b>INSURANCE_FUND</b>				
	IncuranceFundId	Κωδικός ασφαλιστικού ταμείου.	INTEGER	PK
	Description	Περιγραφή ασφαλιστικού ταμείου.	VARCHAR (20)	
<b>DRUG_COVERAGE</b>				
	DrugCoverageId	Κωδικός συγκεκριμένου ταμείου που καλύπτει συγκεκριμένο φάρμακο.	VARCHAR (20)	PK
	MedicineId	Κωδικός φαρμάκου.	INTEGER	FK
	IncuranceFundId	Κωδικός ασφαλιστικού ταμείου.	INTEGER	FK
	CoveragePercentage	Ποσοστό κάλυψης (%) στην τιμή του φαρμάκου.		
<b>FORM</b>				
	FormId	Κωδικός φαρμακοτεχνικής μορφής φαρμάκου.	INTEGER	PK
	Description	Περιγραφή φαρμακοτεχνικής μορφής φαρμάκου και περιεκτικότητας. Π.χ.tab 100mg	VARCHAR (50)	
<b>PACKAGE</b>				

ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
	Packageld	Κωδικός συσκευασίας.	INTEGER	PK
	Description	Περιγραφή συσκευασίας.	VARCHAR (50)	
<b>QUANTITY</b>				
	QuantityId	Κωδικός ποσότητας	VARCHAR (20)	PK
	Items	Ποσότητα.	INTEGER	
	MeasurementUnit	Μετρήσιμη μονάδα. Π.χ. τμχ, χάπια, ml...	VARCHAR (10)	
<b>PACKAGE_QUANTITY</b>				
	PackageQuantityId	Κωδικός για τον συνδυασμό συσκευασίας και ποσότητας.	VARCHAR (20)	PK
	Packageld	Κωδικός συσκευασίας.	INTEGER	FK
	QuantityId	Κωδικός ποσότητας.	VARCHAR (20)	FK
<b>EMPLOYEE</b>				
	EmployeeId	Κωδικός υπαλλήλου φαρμακείου.		PK
	FirstName	Το όνομα του υπαλλήλου.		
	LastName	Το επώνυμο του υπαλλήλου.		
	Role	Ο ρόλος του υπαλλήλου. Χρησιμοποιούμε 1 για		



ΟΝΤΟΤΗΤΑ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΠΕΡΙΓΡΑΦΗ	ΤΥΠΟΣ	ΚΛΕΙΔΙ
		Manager και 2 για απλό υπάλληλο του φαρμακείου. Εδώ εισάγεται και ο διαχειριστής του συστήματος με Role = 3.		
	UserName	Το Username του υπαλλήλου.		
	Password	Το Password του υπαλλήλου.		
	PharmacyId	Ο κωδικός του φαρμάκου.		FK
<b>STATISTICS</b>				
	StatisticsId	Ο μοναδικός κωδικός στοιχείων αναζήτησης. Από κάθε αναζήτηση ασθενή στην σελίδα αναζήτησης έχουμε μια εγγραφή σε αυτόν τον πίνακα.	INT	PK
	SearchDate	Η ημερομηνία αναζήτησης.	Datetime	
	PharmacyId	Ο κωδικός του φαρμακείου.	INT	FK
	MedicineId	Ο κωδικός του φαρμάκου.	INT	FK
	IP	Η IP από όπου έγινε η αναζήτηση.	VARCHAR (20)	



<b>ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ</b>	<b>ΔΙΑΔΙΚΑΣΙΕΣ ΕΥΡΕΣΗΣ ΦΑΡΜΑΚΟΥ ΣΕ ΔΙΑΘΕΣΗΜΟ ΦΑΡΜΑΚΕΙΟ</b>	<b>06/12/2011</b>	<b>V 1.0</b>
<b>ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ RM - 1</b>		<b>©UNIPi</b>	

## 6. Υλοποίηση (Screenshots)

Η εφαρμογή χωρίζεται σε δύο μέρη (πρόγραμμα φαρμακείου – διαχειριστή και WEB εφαρμογή αναζήτησης από τους ασθενείς)

### 6.1 WS1 : PharmacyWebService1

Αυτό το web Service δημιουργήθηκε και χρησιμοποιήθηκε κυρίως για την εξυπηρέτηση του προγράμματος διαχείρισης του φαρμακείου.

Οι μέθοδοι που περιέχει είναι οι εξής :

#### 6.1.1 Μέθοδος 1: Login

```
public object[] Login(string Username, string Password)
```

Αυτή η μέθοδος χρησιμοποιείται για την πιστοποίηση του χρήστη και την εισαγωγή του στο σύστημα. Σαν παραμέτρους απαιτεί το Username και το Password του χρήστη. Σε επιτυχία επιστρέφει και κάποια άλλα στοιχεία του χρήστη όπως των ρόλο του και τον κωδικό που έχει αυτός στο σύστημα.

#### 6.1.2 Μέθοδος 2: InsertPharmacy

```
public bool InsertPharmacy(String employeeId, String pharmacyName, String address, String telephone, String managerName, String longitude, String latitude)
```

Μέθοδος για την εισαγωγή Φαρμακείων στο σύστημα. Παίρνει σαν παραμέτρους στοιχεία του φαρμακείου όπως την διεύθυνση, το γεωγραφικό μήκος και πλάτος και άλλα. Επίσης, απαιτείται για ασφάλεια του συστήματος και η εισαγωγή του κωδικού του Administrator. Τον κωδικό αυτό τον παίρνει το σύστημα από το session το οποίο το έχει αποκτήσει από το Login, οπότε αν έχει κλείσει το session ή αν δεν έχει προηγηθεί επιτυχής είσοδος του χρήστη ο οποίος επιχειρεί εισαγωγή, δεν επιτρέπεται η εισαγωγή από αυτόν.

#### 6.1.3 Μέθοδος 3: InsertEmployee

```
public bool InsertEmployee(String employeeId, String firstName, String lastName, String role, String empUsername, String empPassword, String mobile, String email, int pharmacyId)
```

Μέθοδος για την εισαγωγή χρηστών στο σύστημα. Χρειάζεται ως είσοδο κάποια χαρακτηριστικά του χρήστη. Με το επιτυχή κάλεσμα αυτής την μεθόδου ένας ακόμη χρήστης του συστήματος έχει εισαχθεί στο σύστημα.

#### 6.1.4 Μέθοδος 4: InsertMedicine

```
public bool InsertMedicine(String employeeId, String medicineName, String description, String barcode, String rfidCode, String
```

```
partNumber, String minItems, String needPrescriptionFlag, int
medicineUseId, int formId, int manufacturerId, int packageQuantityId)
```

Μέθοδος για την εισαγωγή φαρμάκων στο σύστημα. Χρειάζεται ως είσοδο κάποια στοιχεία του φαρμάκου όπως την εμπορική ονομασία, αν δικαιολογείται από κάποια ταμείο, την ποσότητα και άλλα. Με το επιτυχή κάλεσμα αυτής την μεθόδου ένας ακόμη χρήστης του συστήματος έχει εισαχθεί στο σύστημα.

### 6.1.5 Μέθοδος 5: GetEmployeeCharacteristics

```
public object[] GetEmployeeCharacteristics(String employeeId)
```

Η μέθοδος αυτή επιστρέφει χαρακτηριστικά ενός χρήστη. Παίρνει ως παράμετρο τον κωδικό του και επιστρέφει όλα τα χαρακτηριστικά του, όπως όνομα, φαρμακείο στο οποίο ανήκει, το ρόλο του και άλλα.

Parameters		
	employeeId	
Response - Results	Success	
		result[0] = firstname result[1] = lastname result[2] = Pharmacy Name result[3] = Address result[4] = Telephone
	Fail	
		result[0] = null; result[1] = null; result[2] = null; result[3] = null; result[4] = null;

### 6.1.6 Μέθοδος 6: GetMedicine

```
public object[] GetMedicine(String employeeId, String barcode)
```

Μέθοδος η οποία παίρνει ως παράμετρο τον κωδικό (barcode) του φαρμάκου και επιστρέφει τα χαρακτηριστικά του όπως εμπορική ονομασία, μορφή, ποσότητα, είδος συσκευασίας και άλλα.

Parameters		
	employeeId	
	barcode	
Response - Results	Success	
		result[0] = MedicineId result[1] = MedicineName result[2] = Description result[3] = Barcode result[4] = RFIDCode result[5] = PartNumber result[6] = NeedPrescriptionFlag result[7] = Pharmacy Use Description result[8] = Medicine Form Description result[9] = Medicine Factory Name
	Fail	
		result[0] = null; result[1] = null; result[2] = null; result[3] = null; result[4] = null; result[5] = null; result[6] = null; result[7] = null; result[8] = null; result[9] = null;

### 6.1.7 Μέθοδος 7: ReceiptOfGoodsInStore

```
public bool ReceiptOfGoodsInStore(String employeeId, String
medicineId, int itemsForSupply)
```

Μέθοδος με την οποία γίνεται η παραλαβή φαρμάκων – σκευασμάτων στο φαρμακείο. Είσοδος είναι ο μοναδικός κωδικός του φαρμάκου και τα τεμάχια της παραλαβής. Υστερα

από επιτυχή κάλεσμα της μεθόδου αυτής η αποθήκη του φαρμακείου στην οποία ανήκει ο χρήστης που προκαλεί την παραλαβή ανανεώνεται με τα τεμάχια αυτά.

## 6.2 AvailablePharmacyWS

Αυτή η υπηρεσία δημιουργήθηκε κυρίως για την εξημέρωση της εφαρμογής αναζήτησης Φαρμάκου και Φαρμακείου και αποτελείται από της παρακάτω κύριες μεθόδους :

### 6.2.1 Μέθοδος 1: GetAvailablePharmacies

```
public object[] GetAvailablePharmacies(String open, String date, String time)
```

Αυτή η μέθοδος επιστρέφει τα διαθέσιμα φαρμακεία στην Αττική την ημέρα και την ώρα που επιθυμείται. Η παράμετρος open όταν είναι ίση με 1 (open=1) τότε αγνοείται η ώρα και επιστρέφονται τα διαθέσιμα φαρμακεία για όλη την διάρκεια της ημέρας, ανεξαρτήτως ώρας. Σε επιτυχή κάλεσμα επιστρέφεται λίστα με χαρακτηριστικά των φαρμακείων όπως επωνυμία, διεύθυνση, γεωγραφικό μήκων και πλάτος, ωράριο και άλλα.

Parameters		
	Open	Is open all the day?
	date	Date that you need to find available pharmacies.
	time	Time that you need to find available pharmacies.
Response - Results	Success	
		result[0]= PharmacyId result[1]= PharmacyName result[2]= Address result[3]= Telephone result[4]= Longitude result[5]= Latitude result[6]= OpenFrom result[7]= OpenTo
	Fail	
		result[0]= null result[1]= null

		<pre>result[2]= null result[3]= null result[4]= null result[5]= null result[6]= null result[7]= null</pre>
--	--	--

## 6.2.2 Μέθοδος 2: GetListOfMatchMedicines

```
public object[] GetListOfMatchMedicines (String searchKey)
```

Επιστρέφει λίστα με διαθέσιμα φάρμακα των οποίων κάποιο χαρακτηριστικό τους ταιριάζει με το 'searchKey' που δόθηκε ως παράμετρος. Σε επιτυχή κάλεση, επιστρέφεται λίστα με πλήθος χαρακτηριστικών του κάθε φαρμάκου.

Οι δύο παραπάνω μέθοδοι των υπηρεσιών μπορούν να χρησιμοποιηθούν από οποιονδήποτε χρήστη γνωρίζοντας το URL της υπηρεσίας και τις απαιτήσεις της κάθε μεθόδου που αυτά περιέχουν και επιθυμούν να χρησιμοποιήσουν. Εκτός αυτών των δύο υπηρεσιών, δημιουργήθηκαν και δύο WEB Services για την εξυπηρέτησης μόνο σκοπών της συγκεκριμένης εργασίας και έχουν αναπτυχθεί εσωτερικά στα αντίστοιχα solution. Αυτές οι υπηρεσίες είναι οι παρακάτω :

Στο solution της διαχείρισης φαρμακείου :

## 6.3 ControllerEmployees

### 6.3.1 Μέθοδος 1: GetEmployeeList

```
public XmlDocument GetEmployeeList ()
```

Αυτή η μέθοδος επιστρέφει μια λίστα με όλους τους χρήστες του συστήματος και τα χαρακτηριστικά τους.

### 6.3.2 Μέθοδος 2: DeleteEmployee

```
public Boolean DeleteEmployee (int employeeId)
```

Μέθοδος για την διαγραφή συγκεκριμένου χρήστη του συστήματος.

## 6.4 ControllerMedicines

### 6.4.1 Μέθοδος 1: GetMedicineList

```
public XmlDocument GetMedicineList ()
```

Αυτή η μέθοδος επιστρέφει μια λίστα με όλα τα φάρμακα τα οποία είναι αποθηκευμένα στο σύστημα και τα χαρακτηριστικά τους.

Parameters		
	employeeld	
	barcode	
Response - Results	Success	
		Result[0]= Medicineld Result[1]= MedicineName Result[2]= Medicine Description Result[3]= Barcode Result[4]= RFIDCode Result[5]= PartNumber Result[6]= MinItems Result[7]= NeedPrescriptionFlag Result[8]= Medicine Use Description Medicine Form Description Result[9]= Package Description + Quantity ItemsQuantity MeasurementUnit Result[10]= Medicine Manufacturer Name
	Fail	
		result[0] = null; result[1] = null; result[2] = null; result[3] = null; result[4] = null; result[5] = null; result[6] = null; result[7] = null; result[8] = null; result[9] = null; result[10] = null;



## 6.4.2 Μέθοδος 2: DeleteMedicine

```
public Boolean DeleteMedicine(int medicineId)
```

Μέθοδος για την διαγραφή συγκεκριμένου φαρμάκου από την βάση του συστήματος.

## 6.5 ControllerPharmacies

### 6.5.1 Μέθοδος 1: GetPharmacyList

```
public XmlDocument GetPharmacyList()
```

Αυτή η μέθοδος επιστρέφει μια λίστα με όλα τα φαρμακεία τα οποία είναι αποθηκευμένα στο σύστημα και τα βασικά χαρακτηριστικά τους.

### 6.5.2 Μέθοδος 2: DeletePharmacy

```
public Boolean DeletePharmacy(int pharmacyId)
```

Μέθοδος για την διαγραφή συγκεκριμένου φαρμακείου από την βάση του συστήματος.

## 6.6 PharmacyController

### 6.6.1 Μέθοδος 1: GetOpenPharmacies

```
public XmlDocument GetOpenPharmacies(string isOpen, string date, string time)
```

Αυτή η μέθοδος χρησιμοποιεί την μέθοδο **GetAvailablePharmacies** της remote υπηρεσίας **AvailablePharmacyWS** για να επιστρέψει τα διαθέσιμα φαρμακεία.

### 6.6.2 Μέθοδος 2: GetListOfMatchMedicines

```
public string GetListOfMatchMedicines(string searchKey)
```

Αυτή η μέθοδος επιστρέφει λίστα με διαθέσιμα φάρμακα των οποίων ταιριάζει κάποιο χαρακτηριστικό τους με το 'searchKey' που δίνεται ως παράμετρος. Σε επιτυχή κάλεση, επιστρέφεται λίστα με πλήθος χαρακτηριστικών του κάθε φαρμάκου. Τα αποτελέσματα σε αυτή την περίπτωση είναι dynamic HTML μορφή. Κατασκευάζεται από πριν πίνακας σε HTML με τα φάρμακα και επιστρέφεται σε αυτή τη μορφή το αποτέλεσμα, έτοιμο για εμφάνιση στην σελίδα.

### 6.6.3 Μέθοδος 3: MedicineAvail

```
public XmlDocument MedicineAvail(string isOpen, string date, string time, int medicineId)
```

Η συγκεκριμένη μέθοδος επιστρέφει τα διαθέσιμα φαρμακεία στα οποία υπάρχει διαθέσιμο ένα συγκεκριμένο φάρμακο. Επιστρέφει χαρακτηριστικά τόσο των φαρμακείων όσο και του φαρμάκου το οποίο αναζητείται.

Parameters		
	MedicineId	Medicine that we search for.
	Open	Is open all the day?
	date	Date that you need to find available pharmacies.
	time	Time that you need to find available pharmacies.
Response - Results	Success	
		result[0]= PharmacyId result[1]= PharmacyName result[2]= Address result[3]= Telephone result[4]= Longitude result[5]= Latitude result[6]= OpenFrom result[7]= OpenTo result[8]= MedicineId result[9]= MedicineName result[10]= MedicineDescription result[11]= ManufacturerId result[12]= Manufacturer result[13]= PackageId result[14]= PackageDescription result[15]= MedicineUsed result[16]= MedicineUse result[17]= QuantityId result[18]= PackageQuantityId result[19]= IndicationId result[20]= Indication result[21]= PackageQuantityDescr

		result[22]= Stock
	Fail	
		result[0]= null result[1]= null result[2]= null result[3]= null result[4]= null result[5]= null result[6]= null result[7]= null result[8]= null result[9]= null result[10]= null result[11]= null result[12]= null result[13]= null result[14]= null result[15]= null result[16]= null result[17]= null result[18]= null result[19]= null result[20]= null result[21]= null result[22]= null

## 6.7. Τεχνικές κλήσης των Web service που χρησιμοποιήθηκαν

Τα Web Services αναπτύχθηκαν στο περιβάλλον ASP.NET μέσω του εργαλείου Visual Studio. Όλες οι κλήσεις των υπηρεσιών ολοκληρώνονται με την εκτέλεση μιας procedure ή ενός SQL Query στον SQL Server έτσι ώστε να επιστραφούν με επιτυχία τα αποτελέσματα.

Η κλήση των υπηρεσιών πραγματοποιείται κυρίως μέσω Ajax Calls με τη χρήση JavaScript κώδικα περνώντας, στο URL που βρίσκεται η υπηρεσία, τις αναγκαίες παραμέτρους. Παρακάτω παρουσιάζεται ένα παράδειγμα από την κλήση μιας υπηρεσίας η οποία υλοποιήθηκε στο ίδιο το solution από όπου και καλείται. Είναι η μέθοδος GetListOfMatchMedicines που περιέχεται στο Web Service PharmacyController και η παράμετρος που απαιτείται για την κλήση της είναι η λέξη κλειδί της αναζήτησης του φαρμάκου. Μετά το επιτυχές κάλεσμα επιστρέφεται μια λίστα από φάρμακα και τα χαρακτηριστικά τους τα οποία έχουν κάποιο χαρακτηριστικό που σχετίζεται με την λέξη κλειδί. Σε επιτυχή κλήση το αποτέλεσμα επιστρέφεται σε μορφή JSON (JavaScript Object Notation), στην function successResultMatchedMedicines η οποία εμφανίζει τα αποτελέσματα σε html table.

```

function GetListOfMatchMedicines() {
    $.ajax({
        type: "POST",
        url:
"http://localhost:1589/PharmacyClientWebSite/WebServices/PharmacyCont
roller.asmx/GetListOfMatchMedicines",
        data: { searchKey: $('#txtSearch').val() },
        datatype: "json",
        success: successResultMatchedMedicines,
        error: function (e) {
            alert("error");
        }
    });
}

```

Παρακάτω παρουσιάζεται η κλήση μιας υπηρεσίας η οποία έχει δημιουργηθεί με το Visual Studio η οποία τοποθετείται και τρέχει σε έναν απομακρυσμένο Server. Στο web site προστέθηκε ένα Web Reference, χρησιμοποιώντας το URL του Web Service, με την ονομασία PharmacyService. Από δω και πέρα το PharmacyService namespace είναι αυτό που αντιστοιχεί στο όνομα του remote server στον οποίο τρέχει η υπηρεσία PharmacyWebService1.

Οι παρακάτω γραμμές κώδικα γράφτηκαν για την κλήση της μεθόδου – υπηρεσίας Login. Η μέθοδος Login παίρνει σαν παραμέτρους το Username και το Password του χρήστη και μετά από μια επιτυχή κλήση επιστρέφει ένα xml με τα στοιχεία του χρήστη.

```

PharmacyService.Service1 Serv = new PharmacyService.Service1();

object[] result = Serv.Login(txtUsername.Text.ToString(),
txtPassword.Text.ToString());

```

## 6.8. Υλοποίηση Google Charts στην εφαρμογή Apothecary

Για την υλοποίηση των δύο γραφημάτων (charts) με τα στατιστικά των αναζητήσεων χρησιμοποιήθηκε το Chart API της Google. Με το παρακάτω html tag μεταφέρεται ο κώδικας του γραφήματος στον browser του χρήστη όταν φορτώνεται η σελίδα.

```
<script type="text/javascript" src="https://www.google.com/jsapi"></script>
```

Κάθε φορά που ο χρήστης επισκέπτεται τη σελίδα που παρουσιάζονται τα στατιστικά στοιχεία αναζήτησης, καλείται μια υπηρεσία η οποία επιστρέφει τα, απαραίτητα για τη δημιουργία του γραφήματος, στατιστικά δεδομένα.

```
$.ajax({
  type: "POST",
  url: "http://localhost:1177/WebServices/ControllerPharmacies.aspx/GetStatistics",
  data: { stat:askStat,
          pharmacyId: pharId
        },
  datatype: "json",
  success: StatResult,
  error: function (e) {
    alert("error");
  }
});
```

Το αποτέλεσμα του παραπάνω τμήματος κώδικα, είναι ένα αντικείμενο τύπου JSON που περιέχει ένα xml. Το xml το μετατρέπουμε σε δύο πίνακες με τη χρήση της παρακάτω function. Ο ένας πίνακας αντιπροσωπεύει τις αναζητήσεις κάθε φαρμακείου ('Αναζητήσεις ανα μήνα') και ο άλλος τις αναζητήσεις σε φάρμακα ('Τα πιο δημοφιλή φάρμακα').

Με την εντολή drawMeds(), στην οποία περνάμε τις ρυθμίσεις και τα δεδομένα του πίνακα, και με την εντολή draw() του Chart API εμφανίζουμε τελικά τα δύο γραφήματα.

```
function StatResult(result) {

var xmlDoc = result;
var stat = xmlDoc.getElementsByTagName('stat');
var xAxis = xmlDoc.getElementsByTagName('xAxis');
var yAxis = xmlDoc.getElementsByTagName('yAxis');
var dat=[]

for (i = 0; i < stat.length; i++) {
  dat.push([xAxis[i].childNodes[0].nodeValue,
            parseInt(yAxis[i].childNodes[0].nodeValue)]);
}

if (stat[0].childNodes[0].nodeValue == "clicks") {
```

```

var optionsClicks = {
  title: 'Αναζητήσεις ανά μήνα',
  hAxis: { title: 'Μήνες', titleTextStyle: { color: 'red' } }
};

drawClicks(optionsClicks, dat);

}
else {
  var optionsMeds = {
    title: 'Πιο δημοφιλή φάρμακα',
    hAxis: { title: 'Φάρμακα', titleTextStyle: { color: 'red' } }
  };

  drawMeds(optionsMeds, dat);
}
}

function drawClicks(optionsClicks, statClicks) {

var dataClicks = new google.visualization.DataTable();

  dataClicks.addColumn('string', 'Μήνες');
  dataClicks.addColumn('number', 'Clicks');
  dataClicks.addRows(statClicks);

  var options = optionsClicks;

  var chartCl = new
google.visualization.ColumnChart(document.getElementById('chartClicks'));
  chartCl.draw(dataClicks, options);
}

function drawMeds(optionsMeds, statMeds) {
  var dataMedicines = new google.visualization.DataTable();
  dataMedicines.addColumn('string', 'Φάρμακα');
  dataMedicines.addColumn('number', 'Αναζητήσεις');
  dataMedicines.addRows(statMeds);
  var optionsMed = optionsMeds;
  var chartMed = new
google.visualization.ColumnChart(document.getElementById('chartMedicines'));
  chartMed.draw(dataMedicines, optionsMed);
}

```

## 7. Εκτίμηση απόδοσης της λύσης (πλεονεκτήματα)

### 7.1 Οφέλη του πληροφοριακού συστήματος αναζήτησης φαρμάκων.

#### 7.1.1 Για τον πελάτη:

- Εύκολη και γρήγορη αναζήτηση φαρμάκου
- Εύκολη και γρήγορη αναζήτηση φαρμακείου
- Γρήγορη επικοινωνία με το φαρμακείο, αφού θα υπάρχουν καταχωρημένα τα στοιχεία του.
- Δυνατότητα πληροφόρησης για τα φάρμακα
- Εντοπισμός κατάλληλου φαρμάκου – σκευάσματος σύμφωνα με την ανάγκη του.

#### 7.1.2 Για τον ιδιοκτήτη του φαρμακείου ο οποίος είναι καταχωρημένος και χρησιμοποιεί το πληροφοριακό σύστημα:

- Καλύτερο, ευκολότερο και ταχύτερο έλεγχο της αποθήκης
- Γρηγορότερη διαχείριση του φαρμακείου
- Καλύτερο έλεγχο του φαρμακείου
- Αύξηση πωλήσεων
- Αναγνωρισιμότητα μέσω του διαδικτύου – αύξηση πελατείας
- Ενημέρωση για την ζήτηση στην περιοχή (στατιστικά)

### 7.2 Οφέλη RFID readers για το φαρμακείο

Στην περίπτωση που η παρούσα εφαρμογή επεκταθεί χρησιμοποιώντας RFID readers στο φαρμακείο, θα υπάρξουν κάποια πρόσθετα οφέλη τα οποία αναφέρονται παρακάτω. Απαραίτητη προϋπόθεση για να είναι όλο και περισσότερα τα οφέλη αυτά, είναι όλο και περισσότερα φάρμακα να έχουν RFID Tags.

Το βασικότερο όφελος που θα αποκομίσουμε με τη χρήση των RFID readers είναι η γρήγορη καταχώρηση των νέων ποσοτήτων που εισάγονται στην αποθήκη, αφού κάποιος Readers έχουν την δυνατότητα να διαβάζουν, χωρίς οπτική επαφή, μεγάλο αριθμό ετικετών RFID. Μπορεί λοιπόν, ο υπεύθυνος του φαρμακείου που διαθέτει έναν τέτοιο reader, να μην κάνει τη καταμέτρηση σε μια παραλαβή, αρκεί να περνάει ολόκληρο το εμπόρευμα που παρέλαβε από τον reader και, αν είναι συνδεδεμένο με το σύστημά του (με ενδιάμεσο λογισμικό Middleware), να προστίθεται αυτόματα στη αποθήκη.

Επιπλέον, με τη χρήση των RFID readers, μπορεί να πιστοποιηθεί η προέλευση των φαρμάκων, αφού σε μία τέτοια ετικέτα μπορεί να περιέχεται η πληροφορία για τον κατασκευαστή του φαρμάκου, τον κωδικό του (SKU) καθώς και έναν αύξοντα σειριακό αριθμό ο οποίος χαρακτηρίζει μοναδικά το προϊόν. Με τον τρόπο αυτό αποφεύγετε και η

πειρατεία που έχει σαν αποτέλεσμα την διαφυγή κερδών για τις φαρμακευτικές αλλά και, σε ορισμένες περιπτώσεις, κίνδυνο για την υγεία του ασθενούς.

Όλες οι ετικέτες RFID έχουν τη δυνατότητα να περιέχουν την ημερομηνία παραγωγής του φαρμάκου. Αυτό μπορεί να συντελέσει και στην καλύτερη διαχείριση της αποθήκης καθώς οι εργαζόμενοι του φαρμακείου μπορούν να προωθούν αρχικά τα φάρμακα που εισήχθηκαν παλιότερα στην αποθήκη και κατόπιν τα νεότερα (First in – first out).

Οι ετικέτες RFID έχουν μεγαλύτερη διάρκεια ζωής και είναι αναγνώσιμες σε δυσμενείς συνθήκες, αφού είναι πολύ πιο δύσκολο να καταστραφούν από τις απλές ετικέτες Barcode. Έτσι δίνουν την δυνατότητα να αναγνωστούν ύστερα από κάποια καταστροφή π.χ. υγρασία, πλημύρα, καύσιμα κ.α. Επίσης, είναι εύκολα αναγνώσιμες και ύστερα από πολύ καιρό.

Ένα παραπάνω και σημαντικό χαρακτηριστικό που προσδίδει η εν λόγω τεχνολογία είναι η εύκολη ανάκληση παρτίδων σκευάσματος που για κάποιο λόγο έχουν καταστεί επικίνδυνες.

Τέλος, είναι πολύ εύκολη και η αλλαγή τιμής κάθε προϊόντος στην περίπτωση που η ετικέτα είναι επανεγγράψιμη (Read - Write). Αποτέλεσμα αυτού είναι η διασφάλιση των φαρμακείων ότι θα πουληθούν τα προϊόντα στη σωστή τιμή.



## 8. Εγχειρίδιο χρήσης

### 8.1 Εργαλείο φαρμακοποιών και διαχειριστή του συστήματος

Το πληροφοριακό σύστημα Apothecary αποτελεί ένα εργαλείο που βοηθάει τους φαρμακοποιούς και τους υπαλλήλους του φαρμακείου να ελέγχουν την αποθήκη του φαρμακείου. Επιπρόσθετα βασικός σκοπός αυτού του εργαλείου είναι η ένταξη όσο το δυνατόν περισσότερων φαρμακείων στο ηλεκτρονικό σύστημα αναζήτησης διαθέσιμου φαρμακείου και φαρμάκου στην περιοχή και την ημερομηνία που ενδιαφέρει τον αναζητητή. Το εργαλείο που έχει αναπτυχθεί και του οποίου η λειτουργία περιγράφεται παρακάτω αφορά τριών ειδών χρήστες, τον/τους διαχειριστή/ές, τους υπεύθυνους των φαρμακείων και τους υπαλλήλους των φαρμακείων. Έτσι κάποιες οθόνες χρησιμοποιούνται από τους υπεύθυνους του φαρμακείου, κάποιες άλλες από τους υπάλληλους και τέλος κάποιες οθόνες βοηθούν τον διαχειριστή του συστήματος να καταχωρεί να παραγοντοποιεί και να συντηρεί ότι απαιτεί το σύστημα για να λειτουργεί σωστά.

#### 8.1.1 Εισαγωγή στο σύστημα (Login)

Ο χρήστης (διαχειριστής, υπεύθυνος φαρμακείου, υπάλληλος φαρμακείου) εισάγει το Username και το Password που του έχει δοθεί από τον διαχειριστή του συστήματος. Στην συνέχεια επιλέγει 'Είσοδος' στο σύστημα.

Πληροφοριακό Σύστημα  
Υπηρεσία Διαχείρισης Φαρμακείων

Είσοδος Στο Σύστημα

Username

Password

[Επιστροφή](#)

Copyright © 2011 - Πανεπιστήμιο Πειραιώς  
Σχεδιασμός-Ανάπτυξη υπηρεσίας: Μαντζιώρη Ελένη ΜΕ/08087

8.1 Είσοδος στην εφαρμογή Apothecary.

Εάν το Username και το Password είναι έγκυρα, τότε στην οθόνη εμφανίζεται η αρχική σελίδα του χρήστη. Αυτή η σελίδα διαφέρει ανάλογα με το είδος του χρήστη.

Στην περίπτωση που το Username και το Password δεν είναι έγκυρα, τότε το σύστημα εμφανίζει στην οθόνη το μήνυμα 'Λάθος Username ή Password' και προτρέπει τον χρήστη να ξαναπεράσει τα στοιχεία του και να ξαναδοκιμάσει εισαγωγή.

## 8.1.2 Διαχειριστής του Συστήματος

### Αρχική Σελίδα

Αφού ο διαχειριστής του συστήματος εισαχθεί επιτυχώς στο σύστημα (βλέπε 'Εισαγωγή στο Σύστημα'), εμφανίζεται αρχικά μια οθόνη με την περιοχή του διαχειριστή στην οποία φαίνονται τα στοιχεία του.

Πληροφοριακό Σύστημα  
Υπηρεσία Διαχείρισης Φαρμακείων

Αρχική Καταχώρηση Φαρμακείων Καταχώρηση Φαρμάκου Καταχώρηση Υπαλλήλων

Περιοχή Διαχειριστή

Καλωσήρθατε στην Υπηρεσία Διαχείρισης φαρμακείου.

Όνομα: Ελένη  
Επώνυμο: Μαντζώρη

Αποσύνδεση

Copyright © 2011 - Πανεπιστήμιο Πειραιώς  
Σχεδιασμός-Ανάπτυξη υπηρεσίας: Μαντζώρη Ελένη ME/08087

### 8.2 Αρχική σελίδα διαχειριστή συστήματος Apothecary

Με το κουμπί αποσύνδεση, ο χρήστης μπορεί να αποσυνδεθεί από το σύστημα.

Ο διαχειριστής του συστήματος έχει τις παρακάτω δυνατότητες από τις οθόνες του συστήματος.

1. Καταχώρηση, Αλλαγή Στοιχείων και Διαγραφή Φαρμακείων.
2. Καταχώρηση, Αλλαγή Στοιχείων και Διαγραφή Φαρμάκων.
3. Καταχώρηση, Αλλαγή Στοιχείων και Διαγραφή Υπαλλήλων Φαρμακείων.

Για να επιλέξει κάποια από αυτές τις ενέργειες αρκεί να επιλέξει το αντίστοιχο Tab από αυτά που βρίσκονται στην κορυφή της σελίδας.

## Καταχώρηση Φαρμακείων

Ο χρήστης επιλέγει την καρτέλα (Καταχώρηση Φαρμακείων). Στην οθόνη εμφανίζεται μια λίστα με τα φαρμακεία που είναι καταχωρημένα στο σύστημα.

The screenshot shows the 'Πληροφοριακό Σύστημα Υπηρεσία Διαχείρισης Φαρμακείων' interface. It features a navigation bar with buttons for 'Αρχική', 'Καταχώρηση Φαρμακείων', 'Καταχώρηση Φαρμάκου', and 'Καταχώρηση Υπαλλήλων'. Below the navigation bar, there is a section titled 'Pharmacies' with a table listing various pharmacies. The table has columns for Pharmacy Name, Address, Telephone, ManagerName, Longitude, and Latitude. Below the table, there is a pagination control showing 'Page 1 of 2' and 'Displaying 1 to 15 of 17 items'. At the bottom of the page, there is a copyright notice: 'Copyright © 2011 - Πανεπιστήμιο Πειραιώς Σχεδιασμός-Ανάπτυξη υπηρεσίας: Μαντζιώρη Ελένη ΜΕ/08087'.

Pharmacy Name	Address	Telephone	ManagerName	Longitude	Latitude
Anna Micalopoulou	Kadmou 12	2102345346	Άννα Μιχαλοπούλου	23.1234	37.6321
ΑΝΤΩΝΑΚΗ ΜΑΡΙΑ	ΤΡΙΦΥΛΙΑΣ 37-39-ΛΑΡΙΣΗΣ, ΑΜΠΕ	2106911362	ΑΝΤΩΝΑΚΗ ΜΑΡΙΑ	23.7614	37.9943
ΒΑΝΔΩΡΟΣ ΣΠΥΡΙΔΩΝ	ΜΑΡΙΝΑΠΑΤΜΟΥ ΜΗΔΕΙΑΣ 45 ΚΑΙ	2102776552	ΒΑΝΔΩΡΟΣ ΣΠΥΡΙΔΩΝ	23.7607	38.0328
ΒΑΣΙΛΟΠΟΥΛΟΥ ΕΛΕΝΗ	Δ.ΑΜΑΡΕΩΣ 56 - ΑΓ.ΦΑΝΟΥΡΙΟΥ,	2107519496	ΒΑΣΙΛΟΠΟΥΛΟΥ ΕΛΕΝΗ	23.7499	37.9664
ΓΙΑΝΝΙΑ ΠΑΡΑΣΚΕΥΗ	ΑΓ.ΤΡΙΑΔΟΣ 13, ΑΓ.ΠΑΡΑΣΚΕΥΗ	2106390988	ΓΙΑΝΝΙΑ ΠΑΡΑΣΚΕΥΗ	23.8354	38.2056
ΔΟΥΜΟΥΛΑΚΗ ΘΕΩΝΗ	ΠΑΛ.ΚΑΒΑΛΑΣ 116 (ΟΠΙΣΘΕΝ ΑΓ.	2105985931	ΔΟΥΜΟΥΛΑΚΗ ΘΕΩΝΗ	24.4175	40.949
ΔΡΑΝΔΑΚΗΣ ΝΙΚΟΛΑΟΣ	ΑΓ.ΜΕΛΕΤΙΟΥ 2, ΚΥΨΕΛΗ	2108223348	ΔΡΑΝΔΑΚΗΣ ΝΙΚΟΛΑΟΣ	23.7388	37.9995
ΞΕΡΧΟΣ ΓΡΗΓΟΡΙΟΣ	ΔΟΥΚ.ΠΑΛΑΚΕΝΤΙΑΣ 36, ΜΕΛΙΣΣΙΑ	2106137562	ΞΕΡΧΟΣ ΓΡΗΓΟΡΙΟΣ	23.8334	38.0248

### 8.3 Οθόνη καταχώρησης Φαρμακείων

Στο κάτω σημείο της σελίδας βρίσκεται μια μπάρα με διάφορες δυνατότητες.

The screenshot shows the bottom navigation bar of the interface. It includes a search icon, a dropdown menu with '15' selected, navigation arrows, 'Page 1 of 1', and 'Displaying 1 to 3 of 3 items'.

Με την βοήθεια αυτής της μπάρας ο χρήστης μπορεί να επιλέξει το πλήθος των φαρμακείων που επιθυμεί να εμφανίζονται ανά σελίδα καθώς και να αναζητήσει (search) ή να πλοηγηθεί ώστε να εντοπίσει το φαρμακείο που επιθυμεί.

Για να αναζητήσει ο χρήστης ένα συγκεκριμένο φαρμακείο θα πρέπει να γνωρίζει την επωνυμία του ή τον κωδικό του. Πατώντας τον φακό που υπάρχει στην μπάρα, εμφανίζεται ακριβώς από πάνω της μια νέα μπάρα αναζήτησης, όπως φαίνεται παρακάτω.

The screenshot shows the search bar interface. It includes a 'Quick Search' input field, a dropdown menu with 'PharmacyName' selected, and a 'Clear' button. Below the search bar, there is a pagination control showing 'Page 1 of 1' and 'Displaying 1 to 3 of 3 items'.

Ο χρήστης μπορεί να επιλέξει από το Drop Down List βάση ποιού στοιχείου του Φαρμακείου επιθυμεί να κάνει την αναζήτηση, ενώ στο πλαίσιο διαλόγου, στα αριστερά της μπάρας, πληκτρολογεί την λέξη για την οποία θα κάνει την αναζήτησή του.

Μια επιπλέον δυνατότητα που υπάρχει στην μπάρα αναζήτησης είναι ο καθαρισμός του φίλτρου με το κουμπί 'Clear'.

Με τη βοήθεια της μπάρας ο χρήστης μπορεί και να πλοηγηθεί σε προηγούμενες και επόμενες σελίδες αποτελεσμάτων, καθώς να κάνει και Ανανέωση (Refresh) της σελίδας.

### Προσθήκη Νέου Φαρμακείου

Ο χρήστης επιλέγει add και στην οθόνη εμφανίζεται η παρακάτω φόρμα καταχώρησης.

Πληροφοριακό Σύστημα

Add Pharmacy

Όνομα Φαρμακείου : Sikalias  
Διεύθυνση : Μουσών 40, Περιστερί  
Τηλέφωνο : 210-5724543  
Υπεύθυνος Καταστήματος : Μάρρας Πέτρος  
Γεωγραφικό μήκος : 23.342567  
Γεωγραφικό πλάτος : 37.984567

Καταχώρηση

8.4 Καρτέλα καταχώρησης στοιχείων νέου φαρμακείου.

Ο χρήστης πληκτρολογεί ή επιλέγει τα πεδία και πατάει το κουμπί 'Καταχώρηση'. Αν το φαρμακείο καταχωρηθεί επιτυχώς στο σύστημα τότε εμφανίζεται αντίστοιχο μήνυμα στην οθόνη.

### Διαγραφή Φαρμακείου

Ο χρήστης επιλέγει τα φαρμακεία τα οποία επιθυμεί να διαγράψει και επιλέγει διαγραφή 'delete'. Μετά την επιτυχή διαγραφή εμφανίζεται ένα σχετικό μήνυμα. Το φαρμακείο έχει διαγραφεί.

Αρχική Καταχώρηση Φαρμακείων Καταχώρηση Φαρμάκου Καταχώρηση Υπαλλήλων

Pharmacies

add edit delete

PharmacyName	Address	Name	Longitude	Latitude
Αλφια Μιχαλοπούλου	Καδίου 12	Μιχαλοπούλου	23.1234	37.6321
Sikalias	Μουσών 40, Περιστερί	Πέτρος	23.3426	37.9846
Ζήσης	Κόβου 13, Περιστερί	Μεσάνδρος	23.7032	37.9034
Κόλλας	Καρέλης 9	Βανάσης	23.6932	38.0043
Παπάς	Θαλή 12	Ελένη	23.6532	37.9543

Pharmacy Deleted

OK

8.5 Παράθυρο επιτυχούς διαγραφής φαρμακείου.

### Καταχώρηση Φαρμάκων

Ο χρήστης επιλέγει την καρτέλα 'Καταχώρηση Φαρμάκου'. Στην συνέχεια ακολουθεί την ίδια διαδικασία με την καταχώρηση, αλλαγή και διαγραφή φαρμακείου.

## Καταχώρηση Υπαλλήλων

Ο χρήστης επιλέγει την καρτέλα 'Καταχώρηση Υπαλλήλων'. Στην συνέχεια ακολουθεί την ίδια διαδικασία με την καταχώρηση, αλλαγή και διαγραφή φαρμακείου.

### 8.1.3 Υπεύθυνος Φαρμακείου

Ο υπεύθυνος του φαρμακείου καταχωρεί νέα προϊόντα από την παραλαβή καθώς έχει και όλες τις αρμοδιότητες ενός απλού υπάλληλου του φαρμακείου (Βλέπε υπάλληλος φαρμακείου). Μπορεί να κάνει επίβλεψη της αποθήκης και να ενημερώνεται για την συχνότητα των αναζητήσεων του φαρμακείου ανά μήνα και τις αναζητήσεις των δημοφιλέστερων φαρμάκων που έγιναν μέσω της εφαρμογής Apothecary μέσω γραφημάτων (Charts).

#### Παραλαβή

Ο χρήστης επιλέγει το δεύτερο Tab ('Παραλαβή'). Περνάει από τον RFID Reader τα φάρμακα ένα - ένα ή και σε ποσότητα (όλο το κιβώτιο). Στην οθόνη εμφανίζονται τα χαρακτηριστικά του φαρμάκου και η ποσότητα τους. Ο χρήστης επιλέγει 'Καταχώρηση στην αποθήκη'.

Τα τεμάχια έχουν προστεθεί στο απόθεμα της αποθήκης.

Πληροφοριακό Σύστημα  
Υπηρεσία Διαχείρισης Φαρμακείων

Αρχική Παραλαβή Προβολή Στατιστικών

Περιοχή Υπεύθυνου Φαρμακείου

503928715

Aspirine  
Aspirine 20mg  
503928715  
123456789  
EA123  
0  
Ενέσημο  
Χάπι  
Bayer GR

20 tms

Καταχώρηση τεμαχίων στην αποθήκη  
Read Barcode  
Αποσύνδεση

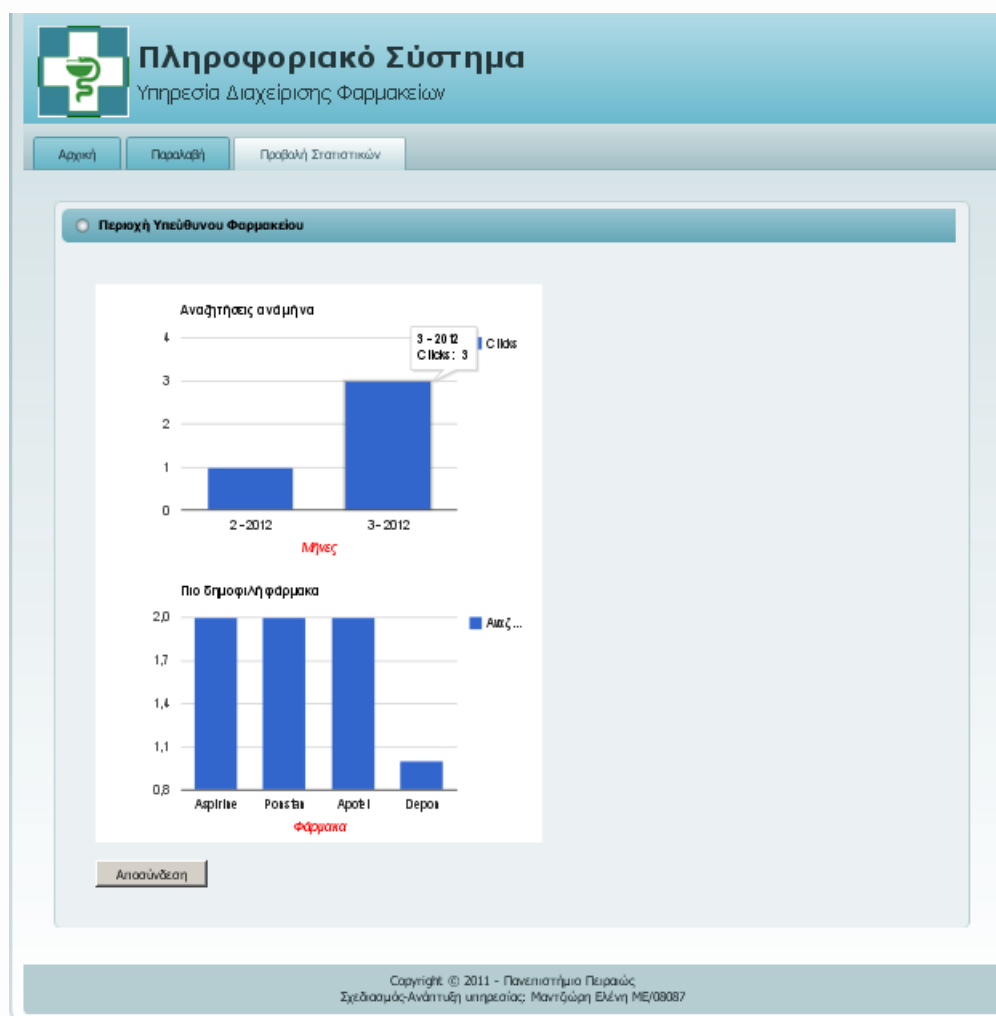
Copyright © 2011 - Πανεπιστήμιο Πειραιώς  
Σχεδιασμός-Ανάπτυξη υπηρεσίας: Μαντζιώρη Ελένη ME/08087

8.6 Παραλαβή φαρμακευτικών εμπορευμάτων.

#### Προβολή Στατιστικών

Ο χρήστης επιλέγει το τρίτο tab ('Προβολή Στατιστικών'). Στην οθόνη εμφανίζονται δύο διαγράμματα. Το ένα παρουσιάζει το πλήθος των αναζητήσεων ανά μήνα που έχουν γίνει μέσω της διαδικτυακής εφαρμογής για το συγκεκριμένο φαρμακείο. Πόσοι, δηλαδή, ασθενής επέλεξαν στον χάρτη το φαρμακείο για να προβάλουν περισσότερα στοιχεία.

Το δεύτερο εμφανίζει τα πιο δημοφιλή φάρμακα σε αναζητήσεις στην περιοχή.



#### 8.7 Προβολή στατιστικών αναζητήσεων

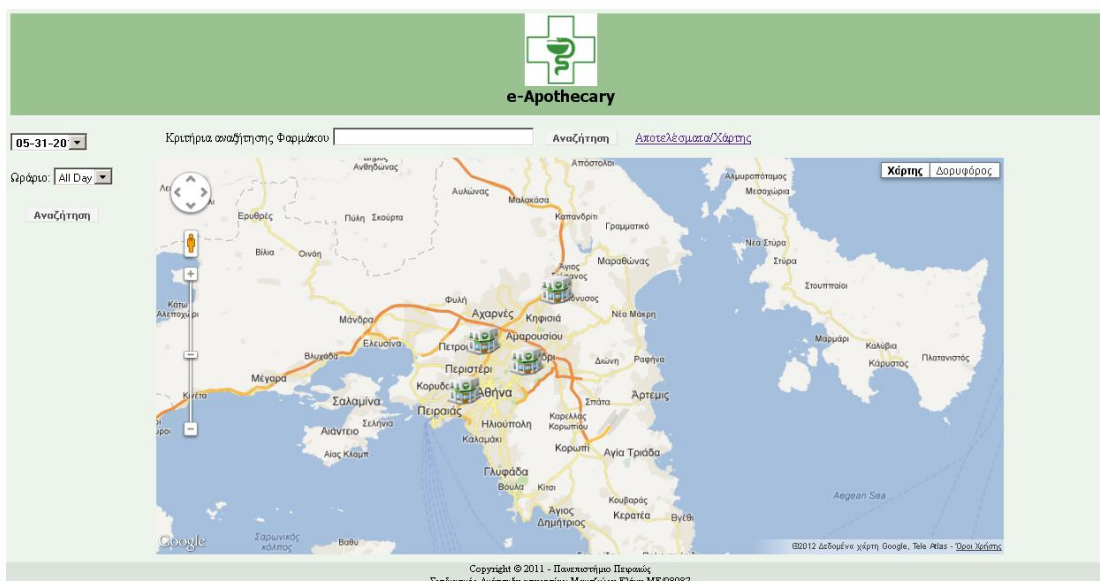
### 8.1.4 Υπάλληλος Φαρμακείου

#### Πώληση

Ο χρήστης επιλέγει το δεύτερο tab ('Πώληση'). Περνάει το φάρμακο που πρόκειται να πωληθεί από τον Reader. Στην οθόνη εμφανίζονται τα χαρακτηριστικά του φαρμάκου. Πατώντας το κουμπί 'Πώληση', ολοκληρώνεται η πώληση και το φάρμακο αφαιρείται από το απόθεμα της αποθήκης.

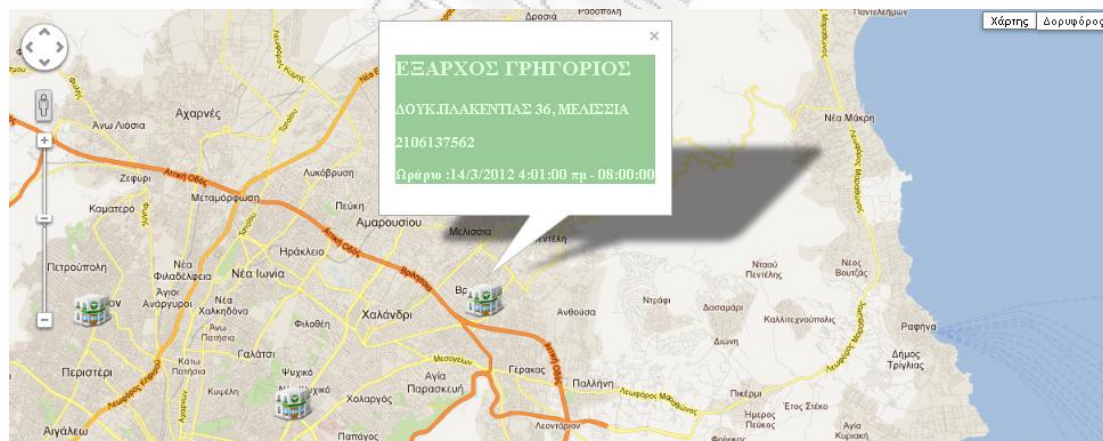
### 8.2 Εφαρμογή Αναζήτησης Φαρμάκου και Φαρμακείου Μέσω Διαδικτύου

Μπαίνοντας αρχικά στην διαδικτυακή εφαρμογή αναζήτησης φαρμάκου και φαρμακείου ο ασθενής – αναζητητής, παρουσιάζεται ένας χάρτης της Αττικής με σημεία στα οποία υπάρχουν φαρμακεία που λειτουργούν κατά την ημέρα της αναζήτησης.



8.8 Αρχική σελίδα αναζήτησης φαρμάκου και φαρμακείου. Εμφάνιση διαθέσιμων φαρμακείων την ημέρα της αναζήτησης.

Ο χρήστης μπορεί να περιηγηθεί στον χάρτη και να κάνει zootη έτσι ώστε να δει καλύτερα τα διαθέσιμα φαρμακεία. Περνώντας το ποντίκι πάνω από κάθε σημείο ενδιαφέροντος εμφανίζεται η επωνυμία του φαρμακείου ενώ, πατώντας πάνω σε αυτό παρουσιάζονται σε ένα info window παραπάνω πληροφορίες για το φαρμακείο όπως, η διεύθυνση, το τηλέφωνο και το ωράριο λειτουργίας της συγκεκριμένης ημέρας.



8.9 Εμφάνιση περισσότερων στοιχείων του σημείου ενδιαφέροντος - φαρμακείου.

Στο αριστερό μέρος της οθόνης ο χρήστης μπορεί να διαλέξει μια άλλη ημερομηνία από αυτές για τις οποίες έχουν δηλωθεί στο σύστημα ωράρια καθώς και την ώρα για την οποία θέλει να πραγματοποιήσει την αναζήτηση. Επιπλέον, υπάρχει ένα check box στο οποίο δηλώνει ο χρήστης αν επιθυμεί την αναζήτηση για διαθέσιμα φαρμακεία όλη την ημέρα ή να λαμβάνεται υπ' όψιν η δηλωμένη ώρα.

Στο πάνω μέρος της οθόνης, στο κέντρο, υπάρχει ένα πλαίσιο στο οποίο ο χρήστης μπορεί να εισάγει χαρακτηριστικά για κάποιο φαρμακευτικό σκεύασμα για το οποίο ενδιαφέρεται.

Το λεκτικό μπορεί να είναι είτε η ονομασία του φαρμάκου, είτε μια ένδειξη-χρήση ή κάποιο άλλο χαρακτηριστικό του φαρμάκου. Αν υπάρχει κάποιο φάρμακο στο σύστημα το οποίο να έχει αυτό το χαρακτηριστικό, τότε πατώντας ο χρήστης το κουμπί 'Αναζήτηση' εμφανίζεται μια λίστα με τα διαθέσιμα φάρμακα τα οποία έχουν αυτό το χαρακτηριστικό.

Φάρμακο	Περιγραφή	Μορφή	Χρήση	Ενδείξεις	Κατηγορία	Ταμείο
Aspirine	Aspirine 20mg	Χάπι	Κατάπιση	ΠΟΝΟΚΕΦΑΛΟΣ, ΠΥΡΕΤΟΣ	ΑΝΤΙΠΥΡΕΤΙΚΑ	ΙΚΑ, ΤΕΒΕ
Ponstan	Ponstan 200mg	Χάπι	Κατάπιση	ΠΟΝΟΣ ΣΤΗΝ ΚΟΙΛΙΑ, ΚΟΙΛΟΠΟΝΟΣ, ΠΟΝΟΣ ΣΤΟ ΔΟΝΤΙ, ΠΟΝΟΔΟΝΤΟΣ, ΠΥΡΕΤΟΣ	ΑΝΑΙΣΘΗΤΙΚΑ	ΙΚΑ
Arotel	Arotel 100 mg	Υγρό	Ενέσιμο	ΠΥΡΕΤΟΣ	ΑΝΤΙΠΥΡΕΤΙΚΑ	ΙΚΑ
Deron	Deron 100 mg	Υγρό	Κατάπιση	ΠΥΡΕΤΟΣ	ΑΝΤΙΠΥΡΕΤΙΚΑ	ΤΕΒΕ

8.10 Λίστα με διαθέσιμα φάρμακα.

Ο χρήστης μπορεί να διαλέξει ένα από αυτά και αυτόματα εμφανίζεται ο χάρτης με τα φαρμακεία τα οποία έχουν διαθέσιμο το φάρμακο στην αποθήκη τους, λαμβάνοντας υπ' όψιν και τα κριτήρια αναζήτησης για τα φαρμακεία, τα οποία πιθανόν έχει επιλέξει ο χρήστης (ημερομηνία, ώρα). Πατώντας τώρα πάνω σε κάθε φαρμακείο εμφανίζεται info window το οποίο εκτός από της πληροφορίες για το φαρμακείο παρουσιάζει και πληροφορίες για το φάρμακο και για την ποσότητα που υπάρχουν αυτή την στιγμή στην αποθήκη.



8.11 Εμφάνιση επιπλέον στοιχείων φαρμακείου και φαρμάκου που επιλέχθηκε.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑΣ

## 9. Εφαρμογή των RFIDs στο πληροφοριακό σύστημα Apothecary

Με την τοποθέτηση RFID readers στα φαρμακεία που διαθέτουν την εφαρμογή Apothecary θα μπορούσαν να αυξηθούν τα οφέλη του συστήματος. Αν στο φαρμακείο έχουν τοποθετηθεί RFID readers θα μπορεί ο υπεύθυνος του φαρμακείου πολύ γρήγορα να κάνει την καταχώρηση των νέων φαρμάκων στην αποθήκη αφού, από απόσταση μπορούν να διαβαστούν μεγάλες ποσότητες ταυτόχρονα. Επίσης, μπορεί να ελέγχει γενικά τι εισάγεται και τι εξάγεται από εμπόρευμα του φαρμακείου του. Αρά, ενημερώνεται σε τυχόν κλοπή φαρμάκου. Επίσης, τοποθετώντας και έναν RFID reader στο ταμείο και συνδέοντας τον με το πρόγραμμα πώλησης των φαρμάκων μπορεί, περνώντας το φάρμακο από αυτόν, να αυτοματοποιήσει την πώληση. Επίσης, αν το πρόγραμμα μπορεί να εμφανίσει όλα τα δεδομένα του tag θα μπορεί ανά πάσα στιγμή να δώσει πληροφορίες για τον φάρμακο στον ασθενή. Όλα αυτά βέβαια προϋποθέτουν εκτός από την τοποθέτηση reader στο φαρμακείο, ότι τα φάρμακα επίσης θα διαθέτουν RFID πομπό.

### Πρόσθετα οφέλη RFID readers για την εφαρμογή Apothecary

- Γρήγορη καταχώρηση των νέων τεμαχίων στην αποθήκη αφού κάποιοι Readers έχουν την δυνατότητα να διαβάζουν χωρίς οπτική επαφή μεγάλο αριθμό RFIDs . Μπορεί λοιπόν ο Υπεύθυνος του φαρμακείου που διαθέτει έναν τέτοιο reader να γλυτώσει την καταμέτρηση σε μια παραλαβή. Αρκεί να περνάει ολόκληρο το εμπόρευμα που παρέλαβε από τον reader και αν είναι συνδεδεμένο με το σύστημά του (με ενδιάμεσο λογισμικό Middleware) προστίθεται αυτόματα στη αποθήκη. Έτσι πετυχαίνει την μείωση λαθών στην καταχώρηση καθώς απολαμβάνει την αυτοματοποίηση και ταχύτητα διαδικασιών.
- Πιστοποίηση των φαρμάκων αφού περιέχει την πληροφορία για τον κατασκευαστή του φαρμάκου των κωδικό του (SKU) καθώς και έναν αύξοντα σειριακό αριθμό ο οποίος χαρακτηρίζει μοναδικά το προϊόν. Οπότε αποφεύγετε και η πειρατεία που έχει σαν αποτέλεσμα την διαφυγή κερδών για τις φαρμακευτικές καθώς και σε ορισμένες περιπτώσεις κίνδυνο για την υγεία του ασθενούς. Το φαρμακείο διασφαλίζει λοιπόν την ποιότητα των φαρμάκων και των παρεχόμενων υπηρεσιών.
- Βέλτιστη και έγκυρη διαχείριση της αποθήκης (First in – first out)
- Οι ετικέτες έχουν μεγαλύτερη διάρκεια ζωής και είναι αναγνώσιμες σε δυσμενής συνθήκες αφού είναι πολλοί πιο δύσκολο να καταστραφούν από τις απλές ετικέτες Barcode. Έτσι δίνουν την δυνατότητα καταγραφής ύστερα από κάποια καταστροφή. Π.χ υγρασία, πλημύρα, καύσιμα και άλλες δύσκολες συνθήκες και είναι εύκολα αναγνώσιμες ύστερα από καιρό.
- Εύκολη ανάκληση παρτίδων σκευάσματος που για κάποιο λόγο έχει καταστεί επικίνδυνο.
- Εύκολη αλλαγή τιμής στην περίπτωση που η ετικέτα είναι επανεγγράψιμη (Read - Write). Οπότε και ασφάλεια ότι θα πουληθεί με την σωστή τιμή.
- Κάποιες ενέργειες γίνονται πιο γρήγορα με αποτέλεσμα την αύξηση της παραγωγικότητας των υπαλλήλων και την καλύτερη εξυπηρέτηση των πελατών.



9.2 Φάρμακο με RFID αυτοκόλλητη ετικέτα

### Συμπεράσματα - Ανακεφαλαίωση

Η πρότυπη αυτή εφαρμογή θα μπορέσει να βοηθήσει σε μεγάλο βαθμό την οργάνωση του υπάρχοντος συστήματος προσφέροντας μεγαλύτερη αυτοματοποίηση και όλα τα πλεονεκτήματα – ωφέλει των RFID που προαναφέρθηκαν.

Σίγουρα για την ανάπτυξη ενός τέτοιου πιλοτικού συστήματος απαιτεί μια λεπτομερέστερη μελέτη κόστους τόσο για τις ετικέτες που θα πρέπει να τοποθετήσουν οι φαρμακευτικές στα φάρμακα όσο και για τους readers και το ενδιάμεσο λειτουργικό που θα πρέπει να διαθέτει ο φαρμακοποιός, προκειμένου να διασφαλιστεί η ποιότητα του συστήματος με τα λιγότερα δυνατά χρήματα.

## 10. Παραδοτέα

- Backup βάσης SQL Server (Pharmacy\_20120317.bak)
- Πηγαίος κώδικας (Source Code)

Για την εφαρμογή διαχείρισης της εφαρμογής και των φαρμακείων.

- PharmacyWebService1
- PharmacyServiceClient

Για το Web site αναζήτησης φαρμάκου και φαρμακείου.

- AvailablePharmacyWS
- PharmacyClientWebSite

- Documentation

Περιέχει πληροφορίες όπως :

- Βιβλιογραφία
- Σχεδιασμό της εφαρμογής (Use Cases, Sequence Diagrams, σχεδιασμό βάσης (ER- diagrams, RM-1 Σχεσιακό σχήμα)
- Την αρχιτεκτονική της εφαρμογής
- Τρόπος υλοποίησης
- User Manual

## 11. Συντομογραφίες

**RFID** Radio Frequency Identification

**AJAX** Asynchronous JavaScript and XML

**SOAP** Simple Object Access Protocol

**WSDL** Web Services Description Language

**JQuery** a new JavaScript Library

**.NET Framework** software framework that runs primarily on Microsoft Windows

**XML** eXtensible Markup Language

**DTD** Document Type Definition

**UDDI** Universal Description, Discovery and Integration

**SOAP** Simple Object Access Protocol

**Apothecary** – Αποτελεί ένα ιστορικό όνομα για τον γιατρό ο οποίος σχεδίαζε και διένειμε φαρμακευτικά είδη για γιατρούς, χειρουργούς και ασθενείς, όρος ο οποίος έχει πλέον αντικατασταθεί – εξυπηρετείται από τον φαρμακοποιό (ή χημικό ή πτυχιούχο φαρμακοποιό) και μερικά άτομα που τους φροντίζουν.

## 12. Αναφορές

- [1] Μιχαήλ Μαβής, 'Τεχνολογία RFID Προκλήσεις και Απειλές', Ηλεκτρονικές Απειλές, [http://library.tee.gr/digital/m2142/m2142\\_mavis.pdf](http://library.tee.gr/digital/m2142/m2142_mavis.pdf), 2006
- [2] Trinity Systems, Τεχνολογία και Λύσεις, [http://www.trinitysystems.gr/Trinity\\_RFID\\_Brochure.pdf](http://www.trinitysystems.gr/Trinity_RFID_Brochure.pdf)
- [3] Μωυσιάδου Ευα, R.F.I.D. και επιχειρηματικές εφαρμογές, 2008
- [4] Marianne Stanczak ,The Many Uses of RFID, <http://www.csa.com/discoveryguides/rfid/review3.php>
- [5] Θεοχάρης Δημητρίου, *Web Services και SOAP*, Πανεπιστήμιο Μακεδονίας, Τμήμα Εφαρμοσμένης Πληροφορικής Θεσσαλονίκη, <http://pdplab.it.uom.gr/project/soap/Downloads/WebservicesTheory.pdf>, 2007
- [6] Roseindia.net (2007), "Web Services"  
<http://www.roseindia.net/webservices/webservices.shtml> [6 Μαρτίου 2007]
- [7] Βασιλακόπουλος Γ., Σχεδιασμός βάσεων δεδομένων, Πειραιάς 2009
- [8] Πρέντζα Αδριάννα, Ανάλυση και σχεδιασμός συστημάτων, Πανεπιστήμιο Πειραιά - Τμήμα ψηφιακών συστημάτων, 2010 - 2011
- [9] docs.jquery.com, <http://docs.jquery.com/Tutorials>
- [10] Tony Tomov, jQuery Plugin - jqGrid, <http://www.trirand.com/blog/jqgrid/downloads/jqgriddocs.pdf>, 2009
- [11] Steven Holzner, JQuery, Learn jQuery the Quick and Easy Way, Peachpit Press, July 2009
- [12] Nick Randolph and David Gardner, Professional Visual Studio 2008 (Wrox Programmer to Programmer), <http://msdn.microsoft.com>, C# Programmer's Reference
- [13] Αναγνώστου Κώστας, Εισαγωγή στην C#/.NET, <http://videogameslab.wordpress.com/2009/01/08/intro-csharp-net/>, Ιανουάριος 2008
- [14] code.google.com, Google Chart Tools, <http://code.google.com/intl/el-GR/apis/chart/>,  
developers.google.com, Google Map Java Script API v3,  
[https://developers.google.com/maps/documentation/javascript/Developer's\\_Guide](https://developers.google.com/maps/documentation/javascript/Developer's_Guide),  
<https://developers.google.com/maps/documentation/javascript/tutorial>,  
API Reference,  
<https://developers.google.com/maps/documentation/javascript/reference>,  
Code Samples,  
<https://developers.google.com/maps/documentation/javascript/examples/>