



Πανεπιστήμιο Πειραιώς

Τμήμα Ψηφιακών Συστημάτων
Πρόγραμμα Μεταπτυχιακών Σπουδών Διδακτική της Τεχνολογίας
και Ψηφιακά Συστήματα

Διπλωματική Εργασία

Σχεδιασμός και υλοποίηση πλατφόρμας εκμάθησης και πρόβλεψης
συμπεριφοράς χρήστη στο πλαίσιο ανάπτυξης γνωστικών
λειτουργιών σε ασύρματες κινητές συσκευές

Δημηροπούλου Πολύμνια ME/07087
Επιβλέπων Καθηγήτρια: Βέρα Αλεξάνδρα Σταυρουλάκη

Σεπτέμβριος 2011

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας είναι να παρουσιαστούν και να μελετηθούν στοιχεία σχετικά με τις γνωστικές λειτουργίες σε ασύρματες κινητές συσκευές. Η παρούσα εργασία περιγράφει κάποια βιβλιογραφικά δεδομένα σχετικά με το αντικείμενο αυτό , καθώς επίσης δίνει και κάποιες πληροφορίες για την λειτουργία των Web Services. Τέλος, αναπτύχθηκε μια custom εφαρμογή για την επιλογή και χρήση υπηρεσιών σε ασύρματες κινητές συσκευές.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Ευχαριστίες

Θερμά ευχαριστώ την υπεύθυνη καθηγήτρια Βέρα – Αλεξάνδρα Σταυρουλάκη για την επίβλεψη, την βοήθεια και την κατανόηση που μου παρείχε για την ολοκλήρωση της διπλωματικής μου εργασίας.

Επίσης θερμές ευχαριστίες εκφράζω στον κύριο Παναγιώτη Δεμέστιχα και στον κύριο Γεώργιο Ευθύμογλου , συνεξεταστές της παρούσας διπλωματικής εργασίας.

Τέλος ευχαριστώ την οικογένεια μου για την υποστήριξη της κατά την διάρκεια όλων των σπουδών μου.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Συντομογραφίες

Λατινικά

C_RMES	Cognitive Equipment Reconfigurable System
GPRS	General Packet Radio Services
UMTS	Universal Mobile Telecommunication System
QoS	Quality of Service
RAT	Radio Access Technology
CPT	Conditional Probability Table
RDF	Resource Description Framework

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1	Εισαγωγή	1
1.1	Εισαγωγή	1
1.2	Στόχος Εργασίας	1
1.3	Δομή Εργασίας	1
2	Βιβλιογραφική Επισκόπηση	2
2.1	Εισαγωγή	2
2.2	Διαχείριση γνωστικών ασύρματων τερματικών	3
2.2.1	Cognitive Reconfigurable Equipment Management Systems	4
2.2.2	Σχεδίαση Συστήματος	5
2.2.3	Λειτουργία Συστήματος	8
2.3	Γνωστικά Συστήματα	9
2.3.1	Γνωστικά Δίκτυα	10
2.4	Προδιαγραφές και σχεδίαση Συστήματος	12
2.5	Τεχνικές Πρόβλεψης Συμπεριφοράς χρήστη	15
2.6	Διαδικασίες εκμάθησης συμπεριφοράς χρήστη	18
2.7	Machine Learning	22
2.7.1	Βασικά συστατικά	23
2.8	Προφίλ χρηστών	27
2.9	Web Services	29
3	Μεθοδολογία	37
3.1	Εισαγωγή	37
3.2	Εργαλεία	37
3.3	Σκεπτικό Επίλυσης	37
4	Αποτελέσματα	40
4.1	Εισαγωγή	40
4.2	Αποτελέσματα Εφαρμογής	41
5	Βιβλιογραφία	46
	ΠΑΡΑΡΤΗΜΑ Ι	47
	ΠΑΡΑΡΤΗΜΑ Ι Ι	51

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ ΚΑΙ ΠΙΝΑΚΩΝ

Εικόνα 1 : Σχεδίαση C_RMES[4]	5
Εικόνα 2 : Οντότητα Επιλογής και επαναχρησιμοποίησης[4]	7
Εικόνα 3 : Λειτουργία C_REMS συστήματος[4]	8
Εικόνα 4 : Γνωστικός Κύκλος[2]	10
Εικόνα 5 : Γνωστικά Device Management[3]	13
Πίνακας 6 : Παράμετροι παρατήρησης[3]	16
Πίνακας 7 : Συνδυασμός παραμέτρων[3]	16
Εικόνα 8 : Αλληλεξάρτηση παραμέτρων παρατήρησης και εξόδου [3]	17
Εικόνα 9 : Formula	18
Εικόνα 10 : CPT πίνακας[3]	20
Εικόνα 11 : Στοιχεία παραμέτρων[3]	20
Εικόνα 12 : Πιθανές τιμές[3]	21
Εικόνα 13 : Data Quality[6]	24
Εικόνα 14 : Επιβλεπόμενη μάθηση[6]	25
Εικόνα 15 : Μη επιβλεπόμενη μάθηση[6]	25
Εικόνα 16 : Ενισχυτική μάθηση[6]	26
Εικόνα 17 : Εμπειρικό λάθος	26
Εικόνα 18 : Δομή προφίλ χρηστών[6]	28
Εικόνα 19 : Τύπου I	32
Εικόνα 20 : Τύπου II	33
Εικόνα 21 : Ρόλοι και λειτουργίες Web Services	34
Εικόνα 22 : WSDL	35
Εικόνα 23 : Υλοποίηση WSDL	36
Εικόνα 24 : Διάγραμμα ροής της Εφαρμογής	38
Εικόνα 25 : Login Page εφαρμογής	41
Εικόνα 26 : Login	42
Εικόνα 27 : Λάθος User name και password	42
Εικόνα 28 : Επιλογή υπηρεσίας	43
Εικόνα 29 : Επιλογή παρόχου	43
Εικόνα 30 : Προτιμήσεις χρήστη	44
Εικόνα 31 : Top rated προτιμήσεις χρήστη	45
Εικόνα 32 : Καταγραφή προτίμησης χρήστη	45

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

1 Εισαγωγή

1.1 Εισαγωγή

Η παρούσα εργασία παρουσιάζει πληροφορίες για τα γνωστικά networks και την πρόβλεψη συμπεριφοράς χρήστη σε ασύρματες κινητές συσκευές. Η εφαρμογή που αναπτύχθηκε , προσπαθεί να πλαισιώσει όλες τις πρακτικές που έχουν αναπτυχθεί μέχρι τώρα με τον καλύτερο δυνατό τρόπο.

1.2 Στόχος Εργασίας

Η παρούσα εφαρμογή προσπαθεί να καταγράψει τις προτιμήσεις του χρήστη χρησιμοποιώντας μια βάση δεδομένων και εισάγοντας και εξάγοντας όλες τις απαραίτητες πληροφορίες από αυτήν

1.3 Δομή Εργασίας

Η παρούσα εργασία αποτελείται από τη βιβλιογραφική επισκόπηση και την ανάπτυξη μιας custom εφαρμογής.

2 Βιβλιογραφική Επισκόπηση

2.1 Εισαγωγή

Στο παρόν κεφάλαιο γίνεται βιβλιογραφική αναφορά στα γνωστικά συστήματα και στις μεθόδους πρόβλεψης συμπεριφοράς χρήστη. Η ανάγκη για επαναχρησιμοποίηση πόρων σε ασύρματα κινητά δίκτυα έχει οδηγήσει την τεχνολογία να βρει τρόπους για την κάλυψη των αναγκών αυτών. Τα γνωστικά ασύρματα δίκτυα έχουν δημιουργηθεί για την εξυπηρέτηση του σκοπού αυτού. Πιο συγκεκριμένα, τα γνωστικά ασύρματα δίκτυα χρησιμοποιούν μέσα για να χειριστούν αποτελεσματικά την πολυπλοκότητα την πρόσβασης στοχεύουν στην ενοποίηση των διαφορετικών μορφών πρόσβασης σε ασύρματες συσκευές, δίνοντας έμφαση σε ένα radio oriented περιβάλλον. Ο απώτερος σκοπός αυτών των συστημάτων είναι να παρέχουν υπηρεσίες στον τελικό χρήστη λαμβάνοντας υπόψη τις προτιμήσεις του και τα τεχνικά χαρακτηριστικά του περιβάλλοντος του χρήστη. Για να επιτευχθεί κάτι τέτοιο πρέπει να υποστηρίζεται τόσο από την μεριά του δικτύου όσο και από την μεριά της τερματικής συσκευής του χρήστη. Έτσι είναι απαραίτητο να οι τερματικές συσκευές να έχουν κατάλληλους μηχανισμούς προκειμένου να εξατομικεύουν τις υπηρεσίες με τον καλύτερο δυνατό τρόπο χωρίς να γίνονται περισσότερες ενέργειες από τον χρήστη. Με γνώμονα αυτό οι τα γνωστικά και επαναχρησιμοποιήσιμα δίκτυα εμφανίζονται ως συμπληρωματικές έννοιες στα B3G δίκτυα.

2.2 Διαχείριση γνωστικών ασύρματων τερματικών

Η ευρεία χρήση των ασύρματων δικτύων τα έχει οδηγήσει σε μια πολλαπλότητα νέων και ισχυρών λύσεων πρόσβασης. Το σύνολο των τεχνολογιών που χρησιμοποιούνται μέχρι σήμερα σχετικά με τα ασύρματα δίκτυα βασίζονται σε GPRS, UMTS, WLAN IEEE 802.11, WiMAX IEEE 802.16 και DVB.

Έρευνες σχετικά με τα B3G/4G δίκτυα προσπαθούν να βρουν τρόπους για την αποτελεσματική χρήση των προτύπων. Ο τελικός σκοπός είναι να αξιοποιηθεί όλο το φάσμα των τεχνολογιών πρόσβαση σε ασύρματες συσκευές. Σύμφωνα με την B3G τεχνολογία τα διαφορετικά RATs (Radio Access Technologies) μπορούν να συγκλίνουν σε ένα radio περιβάλλον στο οποίο ο χρήστης είναι πάντα συνδεδεμένος με τον βέλτιστο τρόπο και μπορεί να αλλάζει συχνότητες. Τα ασύρματα γνωστικά δίκτυα έχουν προκύψει ως συμπληρωματική έννοια στο B3G δίκτυα. Γενικότερα, τα γνωστικά δίκτυα καθορίζουν την συμπεριφορά τους με τρόπους προληπτικούς και μη, με βάση το εξωτερικό περιβάλλον, τους αντικειμενικούς σκοπούς, τις πολιτικές δικτύου και την γνώση. Στις περιπτώσεις των γνωστικά τερματικών, ο ορισμός αυτός μπορεί να μεταφραστεί ως η δυνατότητα να επιλεγεί η πιο κατάλληλη τεχνολογία με τρόπο ασφαλή και έμπιστο.

Τα ασύρματα γνωστικά δίκτυα αποτελούνται από επαναχρησιμοποιήσιμες οντότητες και διαχείριση των λειτουργιών τους.

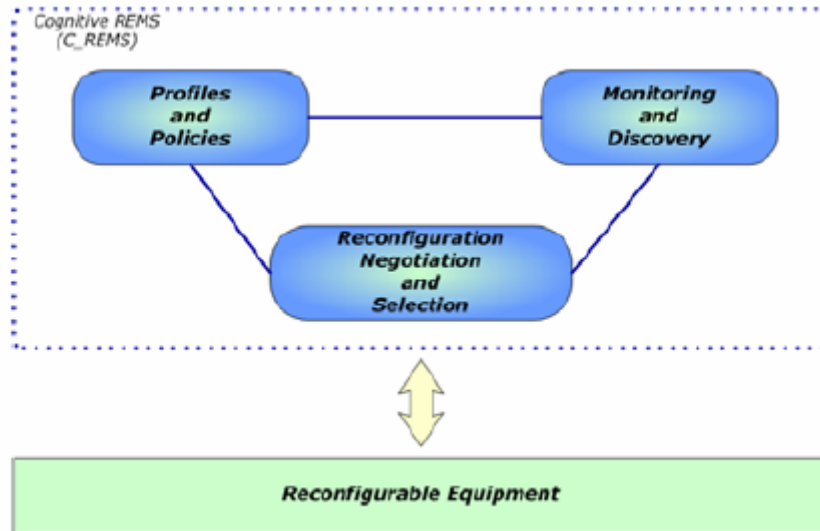
2.2.1 Cognitive Reconfigurable Equipment Management Systems

Ένα C_RMES σύστημα πρέπει ενισχύει τα χαρακτηριστικά του εξοπλισμού με τις δυνατότητες που περιγράφονται παρακάτω:

- Διαχείριση προτιμήσεων χρήστη, δυνατοτήτων εξοπλισμού και πολιτικών δικτύου. Ακριβής περιγραφή και αναπαράσταση των πληροφοριών καθώς και ρύθμιση και ανανέωση των αντίστοιχων προφίλ χρήστη και πολιτικών δικτύου.
- Απόκτηση πληροφοριών περιεχομένου. Αναλυτικότερα, ανεύρεση πρόσβασης δικτύου και παρακολούθηση του αντίστοιχου περιβάλλοντος. Κατά την ανεύρεση γίνονται περιοδικοί έλεγχοι ώστε να δούμε κατά πόσο μια καινούρια λύση είναι καταλληλότερη από αυτήν που χρησιμοποιείται ήδη. Συγκεκριμένα, με τον τρόπο αυτό μπορούμε να δούμε κατά πόσο ένα καινούριο RAT στην περιοχή της υπηρεσίας μπορεί να προσφέρει αποτελεσματικότερα την υπηρεσία. Η παρακολούθηση του περιβάλλοντος αναφέρεται στην συσσώρευση στατιστικών από διαφορετικά RATs, προκειμένου να δούμε την κατάσταση τους και κατά πόσο χρειάζεται αναδιαμόρφωση.
- Διαπραγματεύσεις με άλλα δίκτυα για την καταλληλότερη επιλογή. Κατά την διαδικασία της επιλογής μπορεί να οδηγηθούμε σε αλλαγή RAT. Αυτή η επιλογή αποτελείται από τις προτιμήσεις του χρήστη, τις ικανότητες του εξοπλισμού, την υπηρεσία, την τοποθεσία και την ώρα. Επιπρόσθετα, η επιλογή αυτή δεν πρέπει να περιοριστεί σε τεχνολογίες που είχαν εγκατασταθεί παρελθοντικά στην οντότητα.

2.2.2 Σχεδίαση Συστήματος

Στο παρακάτω σχήμα φαίνονται οι βασικές οντότητες που απαρτίζουν ένα C_RMES σύστημα. (Προφίλ και πολιτικές, Ανεύρεση και παρακολούθηση, επιλογή και επαναχρησιμοποίηση).



Εικόνα 1 : Σχεδίαση C_RMES[4]

- Προφίλ και πολιτικές : Αυτή η οντότητα συντηρεί και παρέχει πληροφορίες σχετικά με την συμπεριφορά του χρήστη σε σύγκριση με τις δυνατότητες του εξοπλισμού του δικτύου, τις προτιμήσεις του χρήστη, τις απαιτήσεις και τους περιορισμούς καθώς και τις πολιτικές του δικτύου. Το προφίλ του εξοπλισμού, π.χ. πληροφορίες σχετικά με τις δυνατότητες του τερματικού, προσδιορίζει ένα σύνολο από ενδεχόμενες διαμορφώσεις εξοπλισμού. Πιο συγκεκριμένα, οι διαμορφώσεις αυτές περιλαμβάνουν τα RATs με τα οποία το τερματικό είναι ικανό να λειτουργήσει και το αντίστοιχο φάσμα που μπορεί να χρησιμοποιηθεί. Πληροφορίες που σχετίζονται με το προφίλ του χρήστη προσδιορίζουν ένα σύνολο εφαρμογών που μπορούν να χρησιμοποιηθούν από τον συγκεκριμένο χρήστη σε συγκεκριμένο επίπεδο QoS. Υψηλή χρησιμότητα μιας

εφαρμογής και QoS υποδεικνύουν υψηλή δείκτη ενδιαφέροντος από τον χρήστη για τον συγκεκριμένο επίπεδο της προσφερόμενης υπηρεσίας. Οι πολιτικές που χρησιμοποιούνται στο σύστημα αποτελούν τους περιορισμούς του παρόχου του δικτύου που πρέπει να ληφθούν υπόψη. Σχετικά με τον χρήστη, οι πολιτικές μπορούν να περιορίσουν τις εφαρμογές και το QoS που επιτρέπεται να χρησιμοποιηθούν. Οι πολιτικές που ακολουθούνται από την μεριά του τερματικού, επιτρέπουν για χρήση ένα υποσύνολο από τις συνολικές δυνατότητες. Οι πολιτικές του δικτύου μπορούν επίσης να καθορίσουν αλγορίθμους και παραμέτρους που πρέπει να χρησιμοποιηθούν για την διαχείριση του περιεχομένου. Η οντότητα αυτή πρέπει να αναπτύξει γνώση σχετικά με την συμπεριφορά, τις προτιμήσεις και τις απαιτήσεις του χρήστη. Πιο συγκεκριμένα, πρέπει να έχει πληροφορίες για την κινητικότητα του χρήστη σε συγκεκριμένη εφαρμογή, τοποθεσία και ώρα. Πληροφορίες σχετικά με τις προτιμήσεις του χρήστη για QoS κρίνονται ως αναγκαίες.

- **Ανεύρεση και παρακολούθηση:** Ο σκοπός αυτής της ενότητας είναι να παρέχει πληροφορίες σχετικές με το περιεχόμενο. Σε περίπτωση που ανιχνευτεί κάποια απόκλιση θα πρέπει να είναι σε θέση να αρχικοποιήσει μια σχετική διάρθρωση δικτύου. Η παρακολούθηση γίνεται σε περιοδικά χρονικά διαστήματα και συλλέγει πληροφορίες σχετικά με την τρέχουσα σύνδεση και τις εφαρμογές που χρησιμοποιούνται. Μόλις συλλεχθούν αυτές οι πληροφορίες, συγκρίνονται με τα προκαθορισμένα κατώτατα όρια. Στην περίπτωση που υπάρχουν παραβιάσεις, η διαδικασία ανεύρεσης ανιχνεύει πιθανά εναλλακτικές διαμορφώσεις. Οι γνωστικές λειτουργικότητες αυτής της οντότητας αποτελούνται από συμπληρωματικές πληροφορίες για το τρέχων περιεχόμενο και τις προβλέψεις για μελλοντικό περιεχόμενο. Συμπληρωματικές πληροφορίες για το παρόν περιεχόμενο δίνουν περισσότερη δύναμη στις διαδικασίες ανεύρεσης και παρακολούθησης. Οι προβλέψεις του περιεχομένου βασίζονται

στην παρακολούθηση καταστάσεων σε διαφορετική τοποθεσία και ώρα.

- Επιλογή και Επαναχρησιμοποίηση: Η οντότητα αυτή αποφασίζει την καταλληλότερη διαμόρφωση για την διαχείριση του τερματικού με κριτήριο τα QoS, το τρέχον περιεχόμενο, το προφίλ του χρήστη και τις πολιτικές που χρησιμοποιεί το δίκτυο.

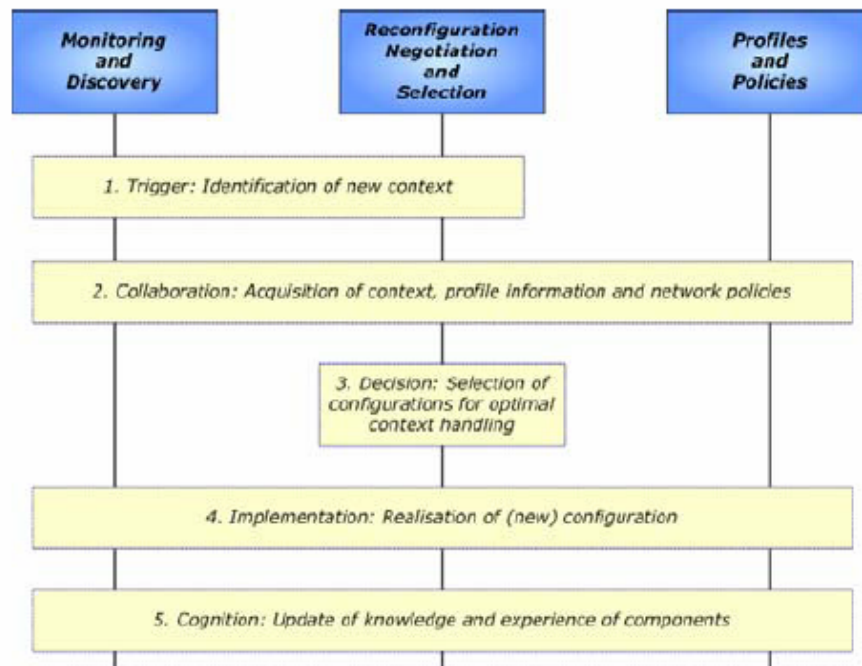


Εικόνα 2 : Οντότητα Επιλογής και επαναχρησιμοποίησης[4]

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑΣ

2.2.3 Λειτουργία Συστήματος

Στην παρούσα ενότητα θα δούμε τα στάδια λειτουργίας ενός C_RMES συστήματος. (Εικόνα 3)



Εικόνα 3 : Λειτουργία C_REMS συστήματος[4]

Το σενάριο χρήσης ξεκινάει από την πυροδότηση στην φάση 1. Αυτό μπορεί να γίνει από την οντότητα της ανεύρεσης και παρακολούθησης η' και από την οντότητα των προφίλ και πολιτικών. Στην δεύτερη φάση η οντότητα της επιλογής και επαναχρησιμοποίησης διαχειρίζεται τις απαραίτητες πληροφορίες από το προφίλ του χρήστη και τις πολιτικές του δικτύου. Στη συνέχεια επιλέγεται η καταλληλότερη διαμόρφωση για το περιεχόμενο. Στη φάση 4 εφαρμόζονται όλες οι επιλογές που έγιναν στα προηγούμενα στάδια. Και τέλος, γίνεται ενημέρωση όλων των οντοτήτων που συμμετέχουν στο σύστημα για τις επιλεγμένες τιμές.

2.3 Γνωστικά Συστήματα

Ως γνωστικό system θεωρείται ένα σύστημα το οποίο έχει την ικανότητα να αντιλαμβάνεται τα χαρακτηριστικά και τις παραμέτρους του περιβάλλοντος και να προσαρμόζει κατάλληλα τους πόρους και τις λειτουργίες του με βάση τις αρχές του (κανόνες, πολιτικές συστήματος). [3] Τα συστήματα αυτά εκτελούν τις έξι (6) βασικές διαδικασίες που περιγράφονται παρακάτω:

1. Καταγραφή και παρατήρηση του περιβάλλοντος που λειτουργούν.
2. Προσανατολισμός του συστήματος.
3. Σχεδιασμός για μελλοντική ενέργεια του συστήματος.
4. Λήψη αποφάσεων για την πορεία της παραπάνω ενέργειας.
5. Ενέργεια η οποία επηρεάζει τη λειτουργία του συστήματος.
6. Συλλογή αποτελεσμάτων από την παραπάνω ενέργεια.

Στο πρώτο βήμα απαιτείται το σύστημα παρατηρεί και συλλέγει πληροφορίες για το περιβάλλον στο οποίο λειτουργεί. Στο δεύτερο βήμα το σύστημα, λαμβάνοντας υπ' όψιν τα αποτελέσματα από το πρώτο βήμα, προσανατολίζει τις λειτουργίες του. Στο επόμενο βήμα σχεδιάζει μια μελλοντική ενέργεια του. Στο τέταρτο βήμα εκτελείται η ενέργεια του τρίτου βήματος ώστε στο τελευταίο βήμα να συλλεχθούν τα αποτελέσματα για περαιτέρω μελέτη και επαναπροσανατολισμό του συστήματος. [1]

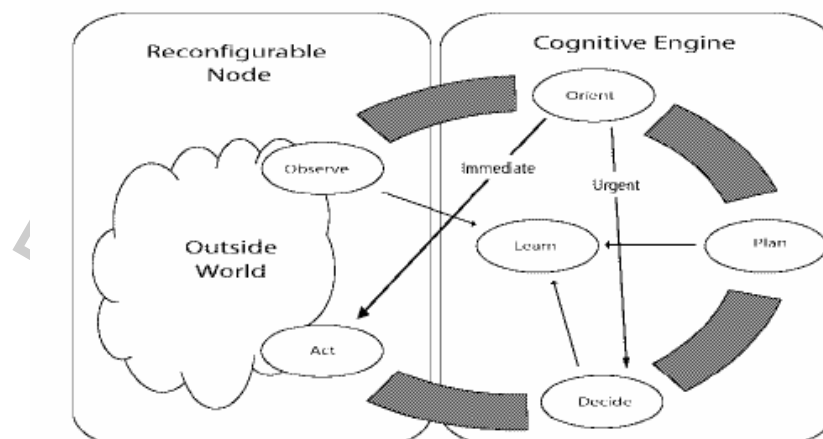
Στα επόμενα κεφάλαια αυτής της ενότητας περιγράφεται αναλυτικά μια προσέγγιση για την δημιουργία ενός τέτοιου δικτύου.

2.3.1 Γνωστικά Δίκτυα

Ως γνωστικά networks ορίζεται ένα δίκτυο το οποίο έχει σκοπό να προσφέρει αξιόπιστη επικοινωνία υψηλού εύρους ζώνης λόγω της δυνατότητας του να παρατηρεί την τρέχουσα κατάσταση του συστήματος και να το αναπροσαρμόζει με βάση το περιβάλλον που λειτουργεί. Επιπρόσθετα, τα δίκτυα αυτά δεν εξαρτώνται άμεσα από την υλοποίησή τους, είναι ανεξάρτητα από αυτήν και έτσι παρέχουν υπηρεσίες γρήγορα και αξιόπιστα ικανοποιώντας τους χρήστες. Τα κοινά δίκτυα συνήθως εγκλωβίζονται στην δομή υλοποίησής τους και δεν έχουν την δυνατότητα να ανακατασκευαστούν με βάση κάθε φορά τις τρέχουσες ανάγκες. Στο παρακάτω σχήμα απεικονίζεται ένας αναπροσαρμόσιμος κόμβος που λειτουργεί στα πλαίσια ενός γνωστικά network.

Στο παρακάτω σχήμα απεικονίζεται η λειτουργία ενός γνωστικού κύκλου (Cognition Cycle). Ένας γνωστικός κύκλος μπορεί να χωριστεί σε δύο ενότητες

- Cognitive Engine: Περιέχει όλες τις απαραίτητες διαδικασίες ενός γνωστικού συστήματος.
- Reconfigurable Node: Παρατηρεί το εξωτερικό περιβάλλον και ενεργεί ανάλογα.



Εικόνα 4 : Γνωστικός Κύκλος[2]

2.3.1.1 Reconfigurable Node

Στα γνωστικά δίκτυα που περιγράφηκαν παραπάνω, είναι σημαντικό να υπάρχει ένας κόμβος δικτύου που να μπορεί να προσαρμόζεται κάθε φορά στις νέες απαιτήσεις του. Ένας τέτοιος κόμβος πρέπει να αποτελείται από ετερογενή στοιχεία τόσο hardware όσο και software. Κάθε μέρος του κόμβου θα πρέπει να αναπαριστάται ως μεμονωμένο συστατικό του. Για παράδειγμα ένα στοιχείο του μπορεί να ενσωματώνει ένα ολόκληρο στρώμα (όπως το στρώμα δικτύου ή το φυσικό στρώμα στο μοντέλο OSI).

Κάθε στοιχείο θα αποτελείται από μία ομάδα αλληλοεξαρτούμενων παραμέτρων προκειμένου να ελέγχουν κάθε λειτουργία του δικτύου. Χρησιμοποιείται μια component-based σχεδίαση γιατί προσφέρει την ικανότητα να διαχειρίζεται τον υψηλό βαθμό πολυπλοκότητας που απαιτείται από ένα τέτοιο σύστημα. Επιπρόσθετα η προσέγγιση δεν απαιτεί βαθιά γνώση για κάθε στοιχείο του κόμβου και έτσι μπορεί να κατασκευαστεί πιο εύκολα ένας κόμβος που θα εξυπηρετεί ένα αρκετά μεγάλο εύρος λειτουργιών.

Η παρατήρηση και η μετέπειτα ενέργεια είναι δύο σημαντικοί παράγοντες για τον παραπάνω περιγραφόμενο κόμβο. Με την έννοια παρατήρηση εννοούμε οποιαδήποτε διαδικασία κάνει το δίκτυο προκειμένου να συλλέξει πληροφορίες από το εξωτερικό περιβάλλον στο οποίο λειτουργεί. Η παρατήρηση γίνεται με δύο αλληλοσυμπληρωμένους μηχανισμούς από και προς τόσο του reconfigurable node όσο και της γνωστικά engine.

Ο πρώτος μηχανισμός επιτρέπει σε οποιοδήποτε στοιχείο του δικτύου να ενημερώνει τον διαχειριστή πόρων για οτιδήποτε σημαντική κίνηση παρατηρεί στο εξωτερικό περιβάλλον.

Ο δεύτερος μηχανισμός αποτελείται από μια διαμοιρασμένη βάση δεδομένων. Κάθε στοιχείο μπορεί να εισάγει οποιαδήποτε πληροφορία στην βάση δεδομένων προκειμένου να χρησιμοποιηθεί από την Cognition Engine.

Παρατηρούμε λοιπόν ότι έχουμε δύο μηχανισμούς push και pull αντίστοιχα προκειμένου να κυκλοφορήσει πληροφορία μεταξύ του

reconfigurable node και της cognition engine για την καλύτερη και αποδοτικότερη λειτουργία του δικτύου.

Στη συνέχεια και αφού έχουν συλλεχθεί και επεξεργαστεί οι πληροφορίες, το δίκτυο καλείται να πράξει ανάλογα προκειμένου να παρέχει γρήγορα, σε καλή ποιότητα και μικρό κόστος υπηρεσία στον χρήστη.

2.4 Προδιαγραφές και σχεδίαση Συστήματος

Η αποδοχή και η ικανοποίηση του χρήστη είναι δύο βασικά κριτήρια για την επιτυχία της προσφερόμενης υπηρεσίας στον χρήστη. Πολύ σημαντικός επιχειρηματικός σκοπός είναι η παροχή υπηρεσιών σε διαφορετικούς τύπους χρηστών και σε διαφορετικές περιοχές και ώρες. Τέτοιες υπηρεσίες μπορούν να έχουν περιεχόμενο voice, audio, data και video που παρέχονται με διαφορετικούς τρόπους.

Οι παραπάνω υπηρεσίες πρέπει να παρέχονται με τον καλύτερο δυνατό συνδυασμό ποιότητας υπηρεσιών (QoS) και κόστους.

Τα επίπεδα της ποιότητας υπηρεσιών είναι συνδεδεμένα με τα παρακάτω μετρίσιμα μεγέθη:

- Απόδοση (ταχύτητα, καθυστερήσεις, θόρυβος κ.α.).
- Διαθεσιμότητα (χωρητικότητα κ.α.).
- Αξιοπιστία.
- Ασφάλεια.

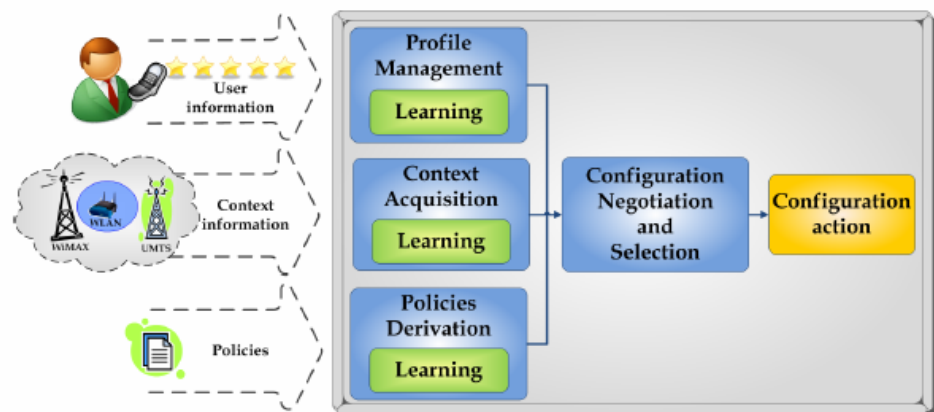
Τα επίπεδα κόστους πρέπει να κάνουν τον χρήστη να αισθάνεται ότι η προσφερόμενη υπηρεσία είναι στην καλύτερη κοστολογίσιμη τιμή για το προσωπικό του όφελος του.

Προκειμένου να επιτευχθεί αυτός ο επιχειρηματικός σκοπός, η χρήση των ασύρματων B3G δικτύων απαιτεί έναν συνδυασμό πολλών προδιαγραφών όπως η εξατομίκευση της υπηρεσίας, το πλαίσιο επίγνωσης, βέλτιστη συνδεσιμότητα, πρόβλεψη και δυνατότητα απομακρυσμένης χρήσης της υπηρεσίας.

Για τις παραπάνω προδιαγραφές απαιτείται ένα Γνωστικά Device Management σύστημα που θα ενσωματώνει τα παρακάτω χαρακτηριστικά:

- Διαχείριση προτιμήσεων χρήστη σε σχέση με τις ικανότητες της τερματικής συσκευής και τον τρόπο χρήσης του δικτύου. Αυτό το χαρακτηριστικό αναφέρεται στην ακριβή περιγραφή των πληροφοριών αυτών καθώς και την ρύθμιση των αντίστοιχων παραμέτρων του προφίλ του χρήστη και του δικτύου.
- Συλλογή πληροφοριών για την τρέχουσα κατάσταση και τοποθεσία του χρήστη και του τερματικού που χρησιμοποιεί. Σκοπός εδώ είναι να αξιολογηθεί η αναγκαιότητα επαναχρησιμοποίησης , και να προσφερθεί η καταλληλότερη διαμόρφωση δικτύου.

Στο παρακάτω σχήμα βλέπουμε όλες τις απαραίτητες οντότητες που συμμετέχουν σε ένα Γνωστικά Device Management.



Εικόνα 5 : Γνωστικά Device Management[3]

Το σύστημα αυτό περιέχει οντότητες για Διαχείριση προφίλ χρηστών, ενημέρωση περιεχομένου, δημιουργία προφίλ χρηστών, διαμόρφωση και διαπραγμάτευση προτιμήσεων. Η αναγκαιότητα των οντοτήτων αυτών προήλθε από τις προδιαγραφές που πρέπει να έχει ένα τέτοιο σύστημα προκειμένου να υιοθετήσει την πρόβλεψη συμπεριφοράς του χρήστη. Όλες οι οντότητες περιλαμβάνουν γνωστικές λειτουργικότητες για την ανάπτυξη γνώσεων σχετικά με τις προτιμήσεις του χρήστη και τον εντοπισμό των αλλαγών στην συμπεριφορά του για περαιτέρω χρήση του συστήματος. Επιπρόσθετα, ένα τέτοιο σύστημα μπορεί να γνωρίζει και να θέτει άμεσα σε ισχύ παρελθοντικές αποτελεσματικές λύσεις.

Αναλυτικότερα, κάθε μια από τις οντότητες έχει τα παρακάτω χαρακτηριστικά:

- Διαχείριση προφίλ χρηστών: Ο ρόλος αυτής της οντότητας είναι να διαχειρίζεται πληροφορίες σχετικές με τις προτιμήσεις χρήστη σύμφωνα με δυνατότητες του τερματικού του χρήστη. Η οντότητα αυτή χρησιμοποιεί μηχανισμούς για την απόκτηση γνώσεων στην συμπεριφορά χρηστών.
- Ενημέρωση περιεχομένου: Η λειτουργικότητα αυτής της οντότητας μπορεί να διαχωριστεί σε δύο διεργασίες, ανεύρεση και παρακολούθηση πληροφοριών. Κύριος σκοπός τις ανεύρεσης πληροφοριών είναι ο περιοδικός έλεγχος για νέες παραμετροποιήσεις. Η παρακολούθηση έχει ως σκοπό τη συλλογή στατιστικών για την τρέχουσα σύνδεση δικτύου.
- Δημιουργία προφίλ χρηστών: Αυτή η οντότητα προβλέπει και διαχειρίζεται πληροφορίες σχετικά με τις πολιτικές που ακολουθούνται. Οι πολιτικές χρήσης χρησιμοποιούνται ως παράμετροι κατά τη διαδικασία επιλογής καλύτερης παραμετροποίησης.

- Διαμόρφωση και διαπραγμάτευση προτιμήσεων: Αυτή η οντότητα αποφασίζει ποια είναι η καταλληλότερη παραμετροποίηση του τερματικού με βάση την ποιότητα υπηρεσίας.

2.5 Τεχνικές Πρόβλεψης Συμπεριφοράς χρήστη

Τα δίκτυα που περιγράφηκαν παραπάνω έχουν άμεση σχέση με τεχνικές πρόβλεψης συμπεριφοράς χρήστη. Σε συνδυασμό μπορούν να επιτύχουν ασφαλέστερες και γρηγορότερες υπηρεσίες. Στο παρόν κεφάλαιο θα αναφερθούμε στο user profiling και σε διάφορες τεχνικές που χρησιμοποιούνται.

Η συλλογή πληροφοριών σχετικά με τις προτιμήσεις χρήστη είναι σημαντική για μελλοντική χρήση της υπηρεσίας, για τον λόγο αυτό έχουν αναπτυχθεί διάφορες τεχνικές συλλογής πληροφοριών.

Κύριο μέλημα των τεχνικών αυτών είναι να αντιστοιχηθούν οι προτιμήσεις του χρήστη με παραμέτρους από το προφίλ του.

Αυτές οι παράμετροι μπορούν να χρησιμοποιηθούν μετά κατά την διαδικασία επιλογής βέλτιστης παραμετροποίησης.

Οι παράμετροι αυτοί είναι κατηγοριοποιημένοι σε δύο βασικές κατηγορίες : Παρατήρησης και εξόδου.

Οι παράγοντες παρατήρησης είναι μετρήσιμα ποσά το οποία προκύπτουν από το σύστημα σε ένα συγκεκριμένο χρονικό διάστημα ή παρέχονται από τον χρήστη μέσω ελάχιστης αλληλεπίδρασης του με το σύστημα.

Ο χρήστης επιλέγει μια υπηρεσία, στην αρχική κατάσταση το σύστημα θεωρεί ότι ο χρήστης δεν έχει καταχωρημένες προτιμήσεις. Κάθε φορά που ο χρήστης επιλέγει μια υπηρεσία εμφανίζεται ένα ερωτηματολόγιο το οποίο τον προτρέπει να συμπληρώσει βαθμολογίες, με βάση την κλίμακα Likert, για την υπηρεσία που επέλεξε και για τον τρόπο που του προσφέρθηκε. [5]

Στον παρακάτω πίνακα καταγράφονται οι παράμετροι παρατήρησης

Παράμετροι Παρατήρησης	Πιθανές Τιμές
Υπηρεσία	Audio, Video, Data, Voice
Ποιότητα Υπηρεσιών	Υψηλή , Μεσαία, Χαμηλή
Τοποθεσία	Η τιμή παρέχεται από το Context Acquisition component
Ωρολογιακή Ζώνη	Η τιμή παρέχεται από το Context Acquisition component

Πίνακας 6 : Παράμετροι παρατήρησης[3]

Οι παράμετροι εξόδου εξαρτώνται άμεσα από τις τιμές των παραμέτρων παρατήρησης. Οι τιμές τους ανανεώνονται δυναμικά με την πάροδο του χρόνου. Οι παράμετροι αυτοί χρησιμοποιούνται για την επιλογή της καλύτερης υπηρεσίας με βάση ένα συγκεκριμένο πλαίσιο που αφορά τον χρήστη (τοποθεσία, ωρολογιακή ζώνη και ρόλος χρήστη).

Οι πιθανές τιμές των παραμέτρων ανήκουν στην κλίμακα [1,5], όπου (1)Καθόλου ικανοποιητικό, (2)Λίγο ικανοποιητικό, (3)Ικανοποιητικό, (4)Αρκετά ικανοποιητικό, (5) Πολύ ικανοποιητικό.

Οι αποδεκτές τιμές κόστους μπορούν να προέλθουν από ένα σύνολο διατιμήσεων συμβολαίου του χρήστη. Στο παρακάτω πίνακα βλέπουμε πως μπορούν να συνδυαστούν οι παράμετροι παρατήρησης και εξόδου.

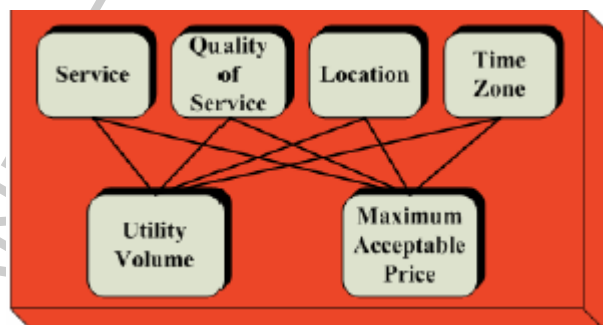
Output Parameter	Potential Value
Utility Volume	$u_{cxt}(s_1, q_1), u_{cxt}(s_2, q_2), u_{cxt}(s_1, q_2), \dots$
Maximum Acceptable Price	$mp_{cxt}(s_1, q_1), mp_{cxt}(s_1, q_2), mp_{cxt}(s_2, q_1), \dots$
<i>where s_i denotes service i, q_j denotes QoS level j, and cxt denotes context, i.e. a particular combination of location and time zone</i>	

Πίνακας 7 : Συνδυασμός παραμέτρων[3]

Οι πιθανές τιμές της παραμέτρου utility volume για κάποιο συγκεκριμένο περιεχόμενο ενημερώνονται με βάση την κρίση του χρήστη για την συγκεκριμένη υπηρεσία που κλήθηκε να επιλέξει από τους πιθανούς παρόχους. Για τον λόγο αυτό στα αρχικά στάδια της διαδικασίας, οι επιλογές γίνονται με βάση το περιεχόμενο της υπηρεσίας και τις πολιτικές που εφαρμόζονται στο δίκτυο, οι πληροφορίες αυτές προέρχονται από τις αντίστοιχες οντότητες που περιγράφηκαν παραπάνω.

Κάθε φορά που ο χρήστης επιλέγει μια υπηρεσία, μια διαδικασία αξιολόγησης του δίνει την δυνατότητα να παρέχει τον βαθμό ικανοποίησης του. Όπως αναφέρθηκε παραπάνω η κλίμακα του Likert, όπου το (1) αντιπροσωπεύει την πλήρη δυσαρέσκεια και το (5) την πλήρη ικανοποίηση, χρησιμοποιείται για να καταγράψει τον βαθμό ικανοποίησης του χρήστη.

Με τον τρόπο αυτό ακόμα κι ένας χρήστης που δεν έχει εξοικείωση με την τεχνολογία μπορεί να παρέχει τον βαθμό ικανοποίησης του μέσα από ένα ερωτηματολόγιο. Υπάρχει πάντα η δυνατότητα ο χρήστης να παρακάμψει το στάδιο της αξιολόγησης, κάτι το οποίο σημαίνει ότι δεν θα είναι το ίδιο επαναχρησιμοποιήσιμο το δίκτυο την επόμενη φορά.



Εικόνα 8 : Αλληλεξάρτηση παραμέτρων παρατήρησης και εξόδου [3]

Η εκτίμηση των προτιμήσεων βασίζονται στην αξιολόγηση των υπηρεσιών από την μεριά του χρήστη με βάση το QoS και το κόστος. Σκοπός είναι να υπολογιστεί η πιθανότητα μιας τιμής με βάση την ποιότητα (QoS) για μια συγκεκριμένη υπηρεσία δοθέντων τοποθεσίας της ώρας.

Προκειμένου να μοντελοποιηθεί η αλληλεξάρτηση μεταξύ των παραμέτρων ενός προφίλ χρήστη (Εικόνα 5) και να παραχθεί μια πιθανολογική σχέση, χρησιμοποιείται ένα Bayesian δίκτυο.

2.6 Διαδικασίες εκμάθησης συμπεριφοράς χρήστη

Η διαδικασία ανάπτυξης γνώσης για την πρόβλεψη συμπεριφοράς χρήστη αποτελείται από δύο φάσεις. Η αρχική φάση είναι αποτελείται από όλες τις απαραίτητες ενέργειες που γίνονται για την συλλογή πληροφοριών. Η επόμενη φάση αποτελείται από την προσέγγιση μελλοντικών πράξεων του χρήστη με βάση τις πληροφορίες που έχουν συλλεχθεί. Παρακάτω περιγράφεται ο υπολογισμός της παραμέτρου εξόδου Utility Volume.

Για την εκτίμηση μιας πιθανής τιμής του επιπέδου ικανοποίησης του χρήστη για μια συγκεκριμένη υπηρεσία και QoS σε συγκεκριμένη τοποθεσία και ώρα χρησιμοποιούνται Bayesian έννοιες.

Αναπτύσσεται μια μέθοδος προκειμένου να υπολογίζεται στιγμιαία η συμπεριφορά χρήστη ακολουθώντας την παρακάτω formula :

$$P_{adapted,n} = w_{hist} * P_{adapted,n-1} + w_{instant} * (1 - |P_{adapted,n-1} - P_{instant,n}|) * P_{instant,n}$$

Εικόνα 9 : Formula

Όπου :

- $|X|$: Η απόλυτη τιμή της μεταβλητής X .
- n : Τρέχουσα περίπτωση
- $P_{adapted,n}$: Προσαρμοσμένη πιθανότητα την στιγμή n
- $P_{adapted,n-1}$: Προσαρμοσμένη πιθανότητα την στιγμή $n-1$
- $P_{instant,n}$: Τρέχουσα στιγμιαία εκτίμηση.
- $W_{instant}$ και W_{hist} : Είναι το βάρος που αποδίδεται σε παρελθοντική και παρούσα εκτίμηση.

Το επόμενο βήμα είναι να παραχθεί Conditional Probability Table (CPT) με βάση το Utility Volume. Για να γίνει πιο κατανοητό, οι όροι στιγμιαία και προσαρμοσμένη εκτίμηση πρέπει να αναλυθούν.

Η στιγμιαία εκτίμηση για ένα συγκεκριμένο χρονικό περιθώριο αντιπροσωπεύει τα μετρίσιμα μεγέθη του συστήματος και τον τρόπο με τον οποίον μοντελοποιήθηκαν με βάση τον CPT πίνακα.

Αναλυτικότερα, αν το σύστημα κληθεί να προσφέρει μια συγκεκριμένη υπηρεσία σε συγκεκριμένο QoS και κόστος δοθέντων τοποθεσίας και ώρας , τότε ανατρέχει στον CPT πίνακα προκειμένου να προσφέρει την υπηρεσία ικανοποιητικά.

Parent Node(s)			Utility Volume				
Service	QoS	TimeZone	1	2	3	4	5
Audio	Low	Personal	0,45	0,2	0,2	0,05	0,1
		Professional	0,2	0,35	0,2	0,1	0,15
	Medium	Personal	0,15	0,55	0,1	0,05	0,15
		Professional	0,25	0,1	0,5	0,1	0,05
	High	Personal	0,05	0,1	0,05	0,1	0,7
		Professional	0,1	0,05	0,05	0,25	0,55
Video	Low	Personal	0,55	0,05	0,2	0,05	0,15
		Professional	0,6	0,1	0,1	0,1	0,1
	Medium	Personal	0,15	0,2	0,35	0,25	0,05
		Professional	0,15	0,2	0,1	0,45	0,1
	High	Personal	0,1	0,15	0,1	0,4	0,25
		Professional	0,15	0,1	0,15	0,2	0,4
Data	Low	Personal	0,35	0,25	0,15	0,2	0,05
		Professional	0,5	0,15	0,05	0,25	0,05
	Medium	Personal	0,1	0,15	0,5	0,15	0,1
		Professional	0,3	0,4	0,15	0,05	0,1
	High	Personal	0,15	0,3	0,05	0,45	0,05
		Professional	0,2	0,15	0,05	0,2	0,4

Εικόνα 10 : CPT πίνακας[3]

Η προσαρμοσμένη εκτίμηση, αντιπροσωπεύει το αποτέλεσμα που παράχθηκε από την formula που περιγράφηκε παραπάνω.

Step	Service	Quality of Service	Location	Time Zone	User Feedback
0	-----	-----	-----	-----	-----
1	Audio	Medium	Home	Personal	4
2	Data	Low	Work	Professional	1
3	Video	High	Home	Personal	5
4	Data	High	Work	Professional	5
5	Audio	Low	Home	Personal	2
6	Voice	Low	Home	Personal	1

Εικόνα 11 : Στοιχεία παραμέτρων[3]

Step	1	2	3	4	5
0	--	--	--	--	--
1	0,15	0,55	0,1	0,05	0,15
2	0,5	0,15	0,05	0,25	0,05
3	0,1	0,15	0,1	0,4	0,25
4	0,2	0,15	0,05	0,2	0,4
5	0,45	0,2	0,2	0,05	0,1
6	0,35	0,25	0,15	0,2	0,05

Εικόνα 12 : Πιθανές τιμές[3]

Κάθε φορά που χρησιμοποιείται μια υπηρεσία, δημιουργείται ένας πίνακας παραμέτρων (Υπηρεσία, QoS, Κόστος, Τοποθεσία και ώρα). Για κάθε μια γραμμή του CPT πίνακα δημιουργείται ένας πίνακας πιθανών τιμών.

Αυτός ο πίνακας γεμίζει με τιμές κάθε φορά που ο χρήστης επιλέγει να χρησιμοποιήσει μια υπηρεσία. Στη συνέχεια υπολογίζεται η στιγμιαία εκτίμηση με βάση τον CPT πίνακα και ακολουθεί ο υπολογισμός της προσαρμοσμένης εκτίμησης με βάση την formula (Εικόνα 6).

Σαν συμπέρασμα μπορούμε να πούμε ότι οι στιγμιαίες εκτιμήσεις χαρακτηρίζονται από σημαντικές διακυμάνσεις σε αντίθεση με τις προσαρμοσμένες εκτιμήσεις στις οποίες οι διακυμάνσεις εμφανίζονται πιο ήπιες.

2.7 Machine Learning

Ο όρος machine learning εννοεί μια επιστήμη που ασχολείται με τον σχεδιασμό και την ανάπτυξη αλγορίθμων προκειμένου να παραχθούν αποτελέσματα σχετικά με την συμπεριφορά του χρήστη μέσω εμπειρικών δεδομένων. Η επιστήμη αυτή επιτρέπει σε μια υπολογιστική μηχανή μέσω επαγωγικών αλγορίθμων να παράγει στατιστικά στοιχεία.

Το machine learning συνήθως αναφέρεται σε αλλαγές που γίνονται στα συστήματα και είναι συσχετισμένες με τεχνητή νοημοσύνη. Τέτοιες αλλαγές μπορεί να αφορούν αναγνώριση, διάγνωση, προγραμματισμό, πρόληψη κ.α. Το machine learning συγκλίνει από διάφορες πηγές, μερικές αναφέρονται παρακάτω:

- Στατιστικές : Ένα μακροχρόνιο πρόβλημα στις στατιστικές είναι το πώς αυτές χρησιμοποιούν δείγματα από άγνωστες κατανομές προκειμένου να έρθουν σε ένα αποτέλεσμα.
- Brain Models: Μη γραμμικές οντότητες με δεδομένα εισόδου που έχουν συγκεκριμένο βάρος αντιπροσωπεύουν τους βιολογικούς νευρώνες του ανθρώπινου εγκεφάλου. Οι περισσότερες μελέτες στον τομέα αυτόν διαπραγματεύονται κατά πόσο ένα δίκτυο με τέτοιες οντότητες μπορεί να προσομοιώσει την λειτουργία του ανθρώπινου εγκεφάλου.
- Θεωρία προσαρμοστικού ελέγχου: Οι θεωρίες ελέγχου διαπραγματεύονται το πρόβλημα του ελέγχου μιας διαδικασίας με άγνωστες παραμέτρους που πρέπει να εκτιμηθούν κατά την διάρκεια της διαδικασίας και όχι από πριν. Συχνά οι παράμετροι αυτοί αλλάζουν κατά τη διάρκεια της διαδικασίας, και η θεωρίες ελέγχου πρέπει να είναι σε θέση να καταγράψουν τις αλλαγές αυτές.
- Ψυχολογικά Μοντέλα: Η επιστήμη της ψυχολογίας έχει ασχοληθεί με το πώς λειτουργεί ο άνθρωπος σε ένα μεγάλο φάσμα δραστηριοτήτων. Ένα πρόσφατο παράδειγμα είναι το δίκτυο EPAM για αποθήκευση και ανάκτηση πληροφοριών για συνώνυμες έννοιες.

- Τεχνητή νοημοσύνη: Η τεχνητή νοημοσύνη είναι άρρηκτα συνδεδεμένη με το machine learning. Με την χρήση διάφορων αισθητήρων καταγράφονται οι ενέργειες του χρήστη σε μια υπολογιστική μηχανή και αναλύονται με κατάλληλους αλγορίθμους και εργαλεία προκειμένου να σχηματιστεί το αντίστοιχο προφίλ του.
- Εξελικτικά μοντέλα: Η διάκριση των εννοιών της εξέλιξης και της μάθησης στην επιστήμη των υπολογιστών μπορεί να μην είναι ξεκάθαρη και για τον λόγο αυτό αναπτύχθηκαν τεχνικές που ακολουθούν βιολογικά πρότυπα και μοντέλα.

2.7.1 Βασικά συστατικά

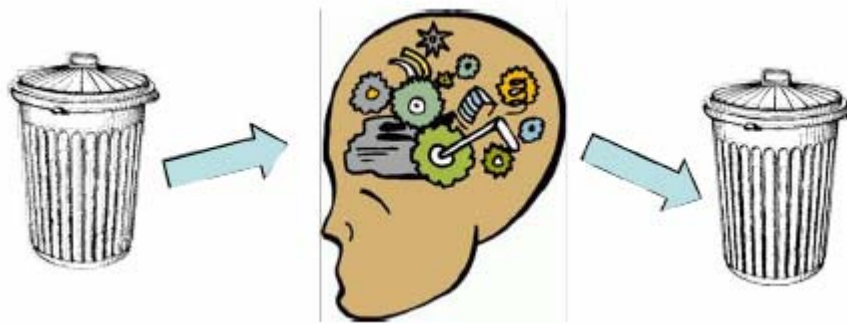
Στην ενότητα αυτή θα παρουσιαστούν τα βασικά συστατικά του machine learning. Περιληπτικά είναι τα ακόλουθα:

- Δεδομένα.
- Εργασίες.
- Μηχανήματα εκμάθησης
 - Υπολογιστικά μοντέλα (τρόπος αναπαράστασης γνώσης):
 - Δέντρα απόφασης.
 - Νευρωνικά δίκτυα.
 - Bayesian μοντέλα.
- Αλγόριθμος μάθησης (πως η γνώση προσαρμόζεται στις παρατηρήσεις):
 - Διάδοση επιστροφής.
 - Προσδοκία – Μεγιστοποίηση.
- Επικύρωση: Μετράει την εκμάθηση της ποιότητας και της απόδοσης.

Έστω ότι έχουμε ένα σύνολο x από δεδομένα εισόδου και ένα σύνολο y από δεδομένα εξόδου και το διάστημα (x,y) σαν ένα δείγμα προς παρατήρηση. Πρέπει να ορίσουμε ένα σύνολο προς εξέταση $D = f(X_1; Y_1).. (X_n; Y_n)$ και το υποθετικό διάστημα, $h: X \rightarrow Y$. Με βάση αυτά ορίζεται και ο αλγόριθμος μάθησης ως μία αντιστοιχία $L: X \times Y \rightarrow H$, η

οποία χρησιμοποιώντας το D επιλέγει μια συνάρτηση από το H με προφητικό τρόπο.

Ένα από τα προβλήματα που συναντάται συχνά στο machine learning έχει να κάνει με την ποιότητα των δεδομένων εισόδου, τα οποία θα επηρεάσουν την όλη διαδικασία και κατά συνέπεια και τα δεδομένα εξόδου.



Εικόνα 13 : Data Quality[6]

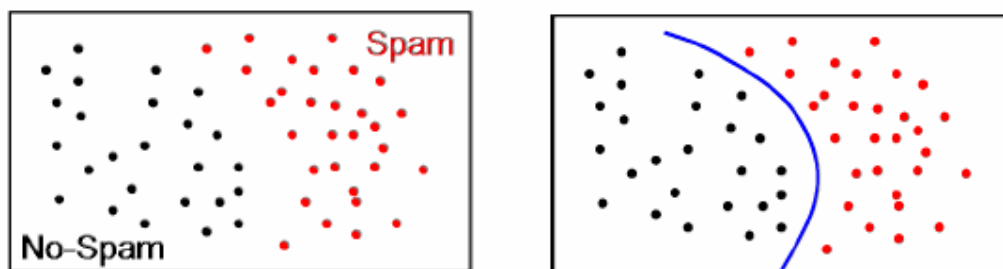
Ένα machine learning μοντέλο μπορεί να είναι τόσο καλό όσο και τα δεδομένα εισόδου που λαμβάνει. Η ποιότητα του αλγορίθμου αυξάνεται ανάλογα με το μέγεθος και την ποιότητα των δεδομένων εισόδου. Προκειμένου να ελαχιστοποιηθούν τα προβλήματα που μπορεί να προκληθούν από την κακή ποιότητα δεδομένων, πρέπει να υπάρχει μια προ-επεξεργασία τους.

Έτσι πριν ξεκινήσει η διαδικασία γίνεται ένα φιλτράρισμα δεδομένων και μια προετοιμασία τους προκειμένου να εισέλθουν στο μοντέλο. Μερικές από αυτές τις δραστηριότητες μπορεί να είναι διόρθωση λαθών, ελλείψεις δεδομένων και δημιουργία “θορύβου”. Με την διαδικασία εύρεσης χαρακτηριστικών τιμών δεδομένων μπορούμε να κλιμακώσουμε και να εξομαλύνουμε τα δεδομένα με δυνατότητα επιλογής και εξόρυξης τους.

Ανάλογα με την εργασία που πρέπει να γίνει, οι αλγόριθμοι διαφοροποιούνται μεταξύ τους. Οι διαφορετικές εργασίες συχνά απαιτούν και διαφορετικούς βαθμούς ανάδρασης:

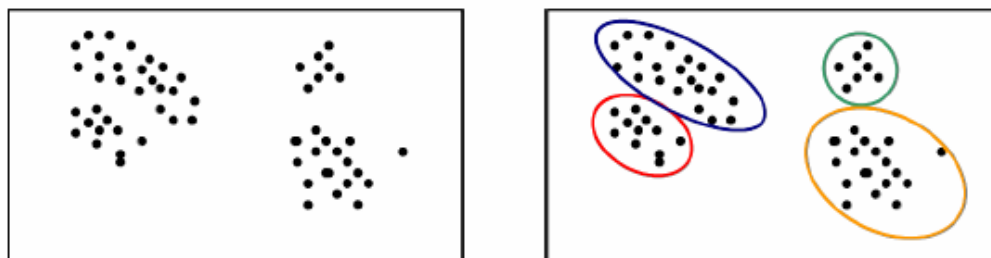
- Επιβλεπόμενη μάθηση.
- Μάθηση χωρίς επίβλεψη.
- Ενισχυτική μάθηση.

Με τον όρο επιβλεπόμενη μάθηση εννοούμε μια machine learning διαδικασία από επιβλεπόμενα δεδομένα εκπαίδευσης.



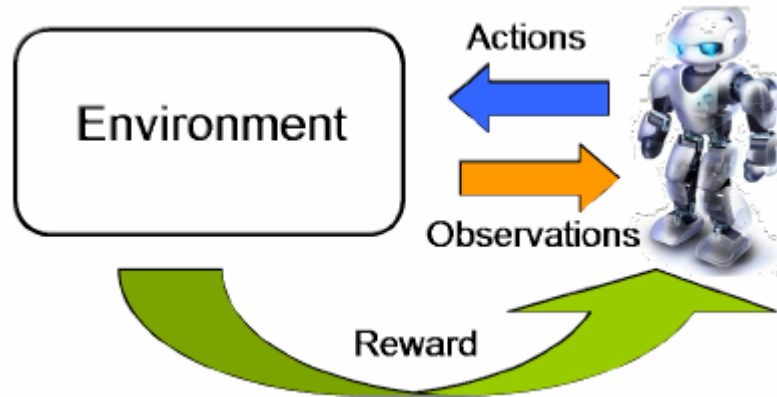
Εικόνα 14 : Επιβλεπόμενη μάθηση[6]

Η μη επιβλεπόμενη μάθηση προσπαθεί να ομαδοποιήσει τα δεδομένα εισόδου βρίσκοντας κοινά χαρακτηριστικά μεταξύ τους.



Εικόνα 15 : Μη επιβλεπόμενη μάθηση[6]

Η ενισχυτική μάθηση επιλέγει την καλύτερη δράση με βάση τα οφέλη και τις ζημίες που προέρχονται από την αλληλεπίδραση με το εξωτερικό περιβάλλον.



Εικόνα 16 : Ενισχυτική μάθηση[6]

Στόχος είναι να βρεθεί ένας αλγόριθμος L που επιλέγει μια υπόθεση h η οποία γενικοποιεί τα αφανή δεδομένα. Οποιαδήποτε υπόθεση προσεγγίζει σωστά τη συνάρτηση στόχου μέσα σε ένα αρκετά μεγάλο υπο-σύνολο δειγμάτων, θα προσεγγίζει σωστά και τη συνάρτηση στόχου σε όλο το δείγμα δεδομένων.

Όπως σε κάθε αλγόριθμο, έτσι κι εδώ υπάρχει πάντα το εμπειρικό λάθος το οποίο ορίζεται ως εξής, όπου h είναι η υπόθεση και D το δείγμα :

$$Err_{\mathcal{D}}(h) = \frac{1}{N} \sum_{(x_i, y_i) \in \mathcal{D}} L(h(x_i), y_i)$$

Εικόνα 17 : Εμπειρικό λάθος

Συνοψίζοντας, τα σημεία κλειδιά στο machine learning είναι τα ποιοτικά δεδομένα, η διαφοροποίηση ανάλογα με τον τύπο ανάδρασης, τα μηχανήματα εκμάθησης και οι μετρήσεις της απόδοσης της μάθησης.

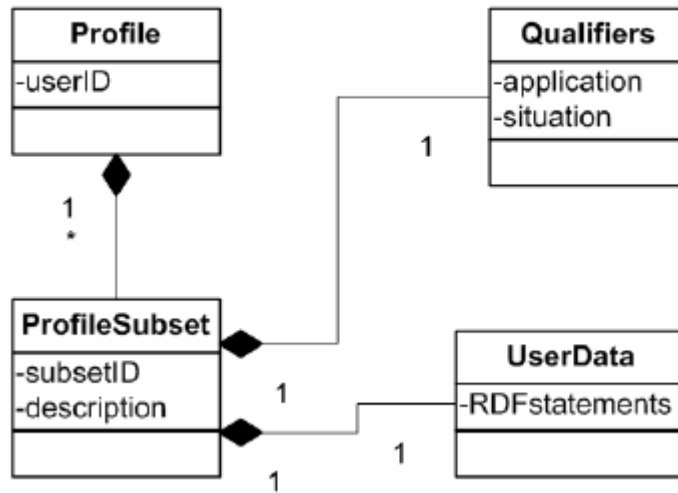
2.8 Προφίλ χρηστών

Στην ενότητα αυτή περιγράφονται τα βασικά χαρακτηριστικά και οι λειτουργίες των user profiles. Οι χρήστες μιας εφαρμογής μπορούν να ομαδοποιηθούν ανάλογα με τις ενέργειες που κάνουν σε μοντέλα χρήσης. Κάθε μοντέλο χρήσης αποτελείται από υπο-μοντέλα με βάση το περιβάλλον στο οποίο αναπτύσσεται η εφαρμογή[6].

Η δομή ενός προφίλ χρήστη περιλαμβάνει τις παρακάτω οντότητες :

- Προφίλ : Περιέχει όλες τις απαραίτητες πληροφορίες για έναν χρήστη. Οι πληροφορίες αυτές , ανάλογα με το περιεχόμενό τους, μπορούν να ομαδοποιηθούν σε διαφορετικά υποσύνολα (π.χ. προσωπικά δεδομένα, δεδομένα λογαριασμού κ.α.)
- Υποσύνολα προφίλ: Περιέχει μια εφαρμογή και όλα τα απαραίτητα στοιχεία του χρήστη για μέσα σε ένα συγκεκριμένο περιεχόμενο. Ένας χρήστης μπορεί να έχει πολλά υποσύνολα προφίλ. Είναι απαραίτητο να υπάρχει ένα κύριο προφίλ χρήστη για κάθε περιεχόμενο μιας εφαρμογής που χρησιμοποιεί.
- Qualifier: Περιλαμβάνει metadata ενός προφίλ του χρήστη. Το περιεχόμενο αποτελείται από ζευγάρια μεταβλητής-τιμής. Για παράδειγμα η μεταβλητή location έχει την τιμή home.
- Δεδομένα χρήστη: Περιέχουν τριπλέτες RDF αποτελούμενες από υποκείμενο, κατηγορημα και αντικείμενο.

Οι οντότητες αυτές αλληλοσχετίζονται μεταξύ τους σύμφωνα με το παρακάτω σχήμα.



Εικόνα 18 : Δομή προφίλ χρηστών[6]

2.9 Web Services

Ως web services ορίζονται αυτόνομες, ανεξάρτητες, εφαρμογές που μπορούν να περιγραφούν, να διατεθούν, να ανατεθούν και να εκτελεστούν σε ενιαίο δικτυακό περιβάλλον.

Τα Web Services αποτελούν μια αναδυόμενη τεχνολογία η οποία επιτρέπει σε νόμιμες εφαρμογές που τρέχουν σε διαφορετικές μηχανές να ανταλλάσσουν δεδομένα και να συνεργάζονται μεταξύ τους χωρίς τη χρήση επιπρόσθετου λογισμικού ή υλικού.

Οι εφαρμογές που βασίζονται σε Web Services μπορούν να ανταλλάσσουν δεδομένα αδιαφορώντας για την γλώσσα προγραμματισμού που χρησιμοποιεί η κάθε μια, για την πλατφόρμα στην οποία βασίζεται και στο πρωτόκολλο διαδικτύου που ακολουθείται. Τα Web Services είναι αυτό-περιγραφικές και αυτόνομες δίκτυοκεντρικές μονάδες οι οποίες εκτελούν επιχειρησιακές λειτουργίες. Αναπτύσσονται με ευκολία επειδή βασίζονται σε κοινά πρότυπα και υπάρχουσες τεχνολογίες όπως XML και HTTP.

Από τα βασικά χαρακτηριστικά των web services είναι η ετερογένεια, η ανεξαρτησία τους από το σημείο υλοποίησης, η αμεσότητα και τέλος ότι είναι αναλλοίωτα στον χρόνο.

Η χρήση των web services έχει τα παρακάτω πλεονεκτήματα:

- Γρήγορη και εύκολη ανάπτυξη συστημάτων.
- Αποδοτική ανάπτυξη εφαρμογών.
- Διασυνδεδιμότητα.
- Άμεση ολοκλήρωση.
- Περιορισμός πολυπλοκότητας.
- Μειώνουν το κόστος της διασύνδεσης της εφαρμογής (interface cost).
- Παρέχουν γενικής χρήσης μηχανισμό για την υιοθέτηση επιχειρηματικής διαδικασίας.
- Αναπτύσσονται τόσο στο εσωτερικό της εφαρμογής όσο και στο ευρύτερο περιβάλλον που αναπτύσσεται.

Τα Web Services στοχεύουν τόσο στους ανθρώπους ως χρήστες αλλά και σε αυτοματοποιημένες εφαρμογές. Δρουν ως πόροι σε άλλες εφαρμογές, οι

οποίες αιτούνται και ξεκινούν τα web services, με ή χωρίς ανθρώπινη παρέμβαση. Παρέχουν ευελιξία και είναι αρθρωτές, αυτό-γνωσες και αυτό-περιγραφικές εφαρμογές που γνωρίζουν τι λειτουργίες κάνει η εφαρμογή, τι δεδομένα εισόδου απαιτεί και τι αποτελέσματα έχει και ενημερώνει σχετικά άλλες εφαρμογές, χρήστες ή web services.

Τα web services επιτρέπουν στους σχεδιαστές ενός συστήματος να δομήσουν τις εφαρμογές τους με βάση υπάρχων κώδικα χωρίς να χρειαστεί να παράγουν καινούριο. Θα μπορούσαν να θεωρηθούν ως μια λογική συνέχεια των object oriented και component based εφαρμογών. Επιπρόσθετα, αποτελούν αυτόνομες μονάδες οι οποίες αναπτύσσονται σε πρότυπα πλατφόρμας διαμεσολαβητή που μπορεί να περιγράψει, εκδίδει, εντοπίζει, ενορχηστρώσει, προγραμματίσει και να εκτελέσει.

Τα Web Services επεκτείνουν την ιδέα του λογισμικού ως υπηρεσία και ενσωματώνουν την παροχή σύνθετων επιχειρηματικών διαδικασιών και συναλλαγών ως υπηρεσία. Είναι χαλαρά συνδεδεμένες μονάδες, λόγω της ευχέρειας των πρωτοκόλλων, των διασυνδέσεων και των υπηρεσιών εγγραφής τους επιτρέπεται στις εφαρμογές που σχετίζονται με αυτά, με διαφορές από τις προϋπάρχουσες αρχιτεκτονικές. Κάθε ένα web service είναι μια αυτόνομη λογισμική μονάδα η οποία εκτελεί μια μοναδική εργασία.

Από την μεριά του προγραμματισμού, παρέχουν μια είσοδο η οποία επιτρέπει την χρήση Web Service σε απομονωμένες εφαρμογές.

Μπορούν να εντοπιστούν δυναμικά και να συμπεριληφθούν σε υπάρχουσες εφαρμογές και βρίσκονται κατανεμημένα στο διαδίκτυο κάνοντας χρήση ενός γνωστού πρωτοκόλλου μεταφοράς δεδομένων στο διαδίκτυο πάνω από HTTP συνδέσεις.

Οι υπηρεσίες αλληλεπιδρούν μεταξύ τους δυναμικά και χρησιμοποιούν πρότυπα τεχνολογιών διαδικτύου καθιστώντας έτσι εφικτή την διασύνδεση συστημάτων που διαφορετικά θα χρειαζόταν εκτεταμένη προσπάθειες επέκτασης. Μια σύνδεση χαρακτηρίζεται ως σφιχτή όταν οι εφαρμογές χρειάζεται να γνωρίζουν πως λειτουργούν οι υπόλοιπες εφαρμογές που αλληλεπιδρούν. Ως χαλαρή χαρακτηρίζεται μια σύνδεση που δεν απαιτεί τέτοιου είδους πληροφορίες, παρέχοντας παράλληλα ευελιξία και διαλειτουργικότητα.

Οι κλήσεις των web services μπορούν να γίνονται είτε σύγχρονα (απομακρυσμένα) είτε ασύγχρονα.

Στην σύγχρονη κλήση ο χρήστης στέλνει ένα αίτημα και περιμένει ένα μήνυμα απόκρισης πριν συνεχίσει σε επόμενα βήματα. Στην ασύγχρονη κλήση ο χρήστης στέλνει ένα αίτημα και η εφαρμογή μπορεί να στείλει αίτημα απόκρισης σε μελλοντικό χρονικό διάστημα χωρίς αυτό να αποκλείει τον χρήστη να συνεχίσει σε επόμενα βήματα.

Τα web services μπορούμε επίσης να τα διαχωρίσουμε σε :

- Αντικαταστάσιμα Web Services: Οι υπηρεσίες που παρέχονται από πολλούς πάροχους και μπορούν να αντικατασταθούν χωρίς να επηρεασθεί η λειτουργικότητα της εφαρμογής, εφόσον η διεπαφή της υπηρεσίας είναι όμοια.
- Web Services κρίσιμης αποστολής: Είναι οι υπηρεσίες που παρέχονται από ένα πάροχο, και αν αντικαθιστούν επηρεάζουν την λειτουργικότητα της εφαρμογής.

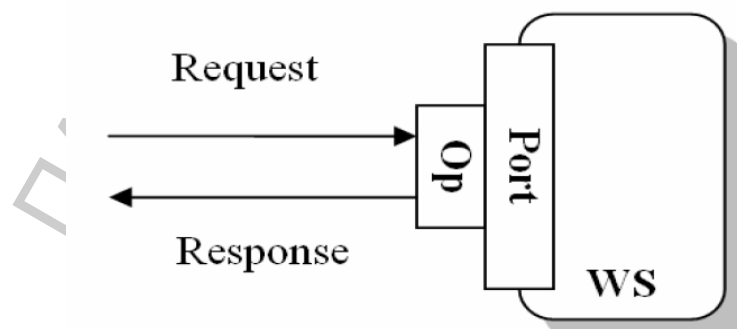
Τα web services ορίζουν ένα σαφή διαχωρισμό ανάμεσα στη διασύνδεση και την υλοποίηση. Η διασύνδεση της υπηρεσίας καθορίζει την λειτουργικότητα που είναι εμφανής στον τελικό χρήστη και παρέχει τα μέσα για την είσοδο σε αυτή. Η υλοποίηση της υπηρεσίας, πραγματοποιεί ένα ξεχωριστό μέρος της διεπαφής, είναι οι λεπτομέρειες της υλοποίησης δεν φαίνονται στον τελικό χρήστη.

Τα web services μπορούν να έχουν μια από τις παρακάτω τοπολογίες :

- Απλές ή Πληροφοριακές (τύπου I): Είναι web services που υποστηρίζουν απλές δράσεις (αίτημα/απόκριση). Περιμένουν για το αίτημα του χρήστη, το επεξεργάζονται και αποκρίνονται. Κάνουν μια απλή διεργασία χωρίς να κρατάνε κάτι στην μνήμη τους.

Διαχωρίζονται σε τρεις υποκατηγορίες

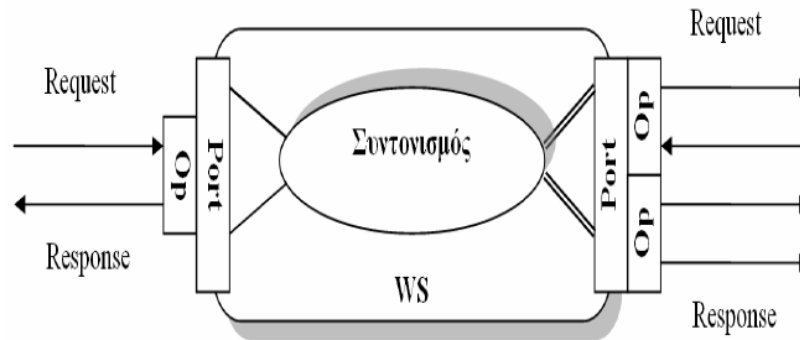
- Υπηρεσίες καθαρού περιεχομένου: Παρέχουν προγραμματιστική πρόσβαση σε περιεχόμενο (π.χ. Ειδήσεις, Καιρός κ.α.)
- Υπηρεσίες εμπορίου: Πιο σύνθετες υπηρεσίες που απαιτούν συνάθροιση πληροφοριών (π.χ Logistics).
- Υπηρεσίες σύνδεσης πληροφοριών: Υπηρεσίες προστιθέμενης αξίας που επιτρέπουν την πρόσβαση σε εμπορικά sites (π.χ. Σύστημα κράτησης θέσεων)



Εικόνα 19 : Τύπου I

- Περίπλοκες (τύπου II): Είναι web services που υλοποιούν κάποια μορφή συντονισμού μεταξύ εισερχομένων και εξερχομένων

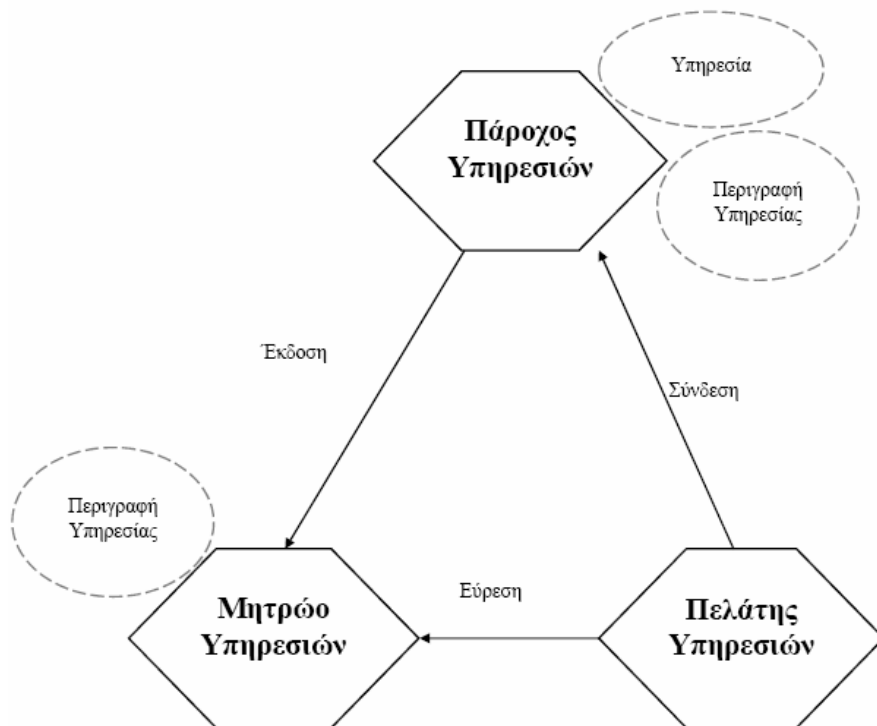
εργασιών. Μπορούν να συνθέτουν τόσο προγραμματιστικές όσο και διαδραστικές υπηρεσίες.



Εικόνα 20 : Τύπου II

Τα web services έχουν τους παρακάτω ρόλους και λειτουργίες :

- Ο πάροχος Web Service εκδίδει τις υπηρεσίες στον πράκτορα εξεύρεσης (Discovery agency).
- Ο πελάτης των υπηρεσιών διερευνά για επιθυμητές υπηρεσίες χρησιμοποιώντας το μητρώο του πράκτορα.
- Με βάση τα αποτελέσματα αναζήτησης, ο πελάτης καλεί το Web Service που του παρέχει ο πάροχος.
- Η λειτουργία έκδοσης αποτελείται από:
 - Περιγραφή υπηρεσίας: Πληροφορίες επιχειρηματικής φύσης και τεχνικές.
 - Εγγραφή υπηρεσίας: Εγγραφή στο μητρώο των τριών χαρακτηριστικών.
- Η λειτουργία εύρεσης αποτελείται από:
 - Εύρεση των υπηρεσιών από τον πράκτορα εξεύρεσης.
 - Εντοπισμός της επιθυμητής από τα αποτελέσματα της έρευνας.
- Η λειτουργία της σύνδεσης αποτελείται από:
 - Απ'ευθείας σύνδεση.
 - Σύνδεση με διαμεσολαβητή.



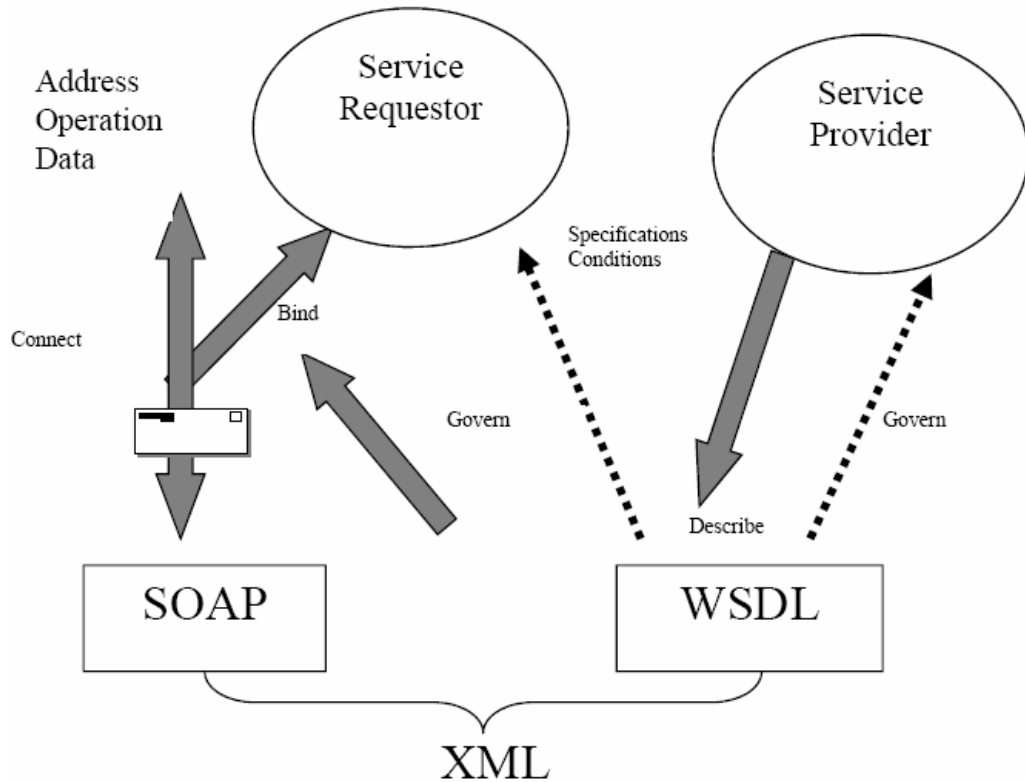
Εικόνα 21 : Ρόλοι και λειτουργίες Web Services

Για όλα τα παραπάνω πλεονεκτήματα που παρουσιάζουν τα Web Services, υπάρχει μια τάση τα τελευταία χρόνια να μιλάμε για service-oriented προγραμματισμό. Ο service-oriented προγραμματισμός περιέχει ένα σύνολο υπηρεσιών όπως ο σχεδιασμός εφαρμογών. Ένας τέτοιος τύπος προγραμματισμού υποστηρίζει την δημιουργία ταχέων και χαμηλού κόστους συνθέσεων που αποτελούνται από κατακευματισμένες εφαρμογές.

Ταυτόχρονα αντιπροσωπεύει και τη συγχώνευση πολλαπλών τεχνολογιών όπως, κατακευματισμένα συστήματα, κατασκευή λογισμικού, πληροφοριακά συστήματα, γλώσσες προγραμματισμού, δίκτυο-κεντρικού προγραμματισμού και XML μηνυμάτων.

Για την επικοινωνία με Web Services χρησιμοποιούνται αρχεία WSDL, τα οποία προσφέρουν έναν ομοιόμορφο μηχανισμό για την περιγραφή αφηρημένης διεπαφής υπηρεσιών και συγκεκριμένου πρωτοκόλλου σύνδεσης που υποστηρίζει την υπηρεσία. Ένα αρχείο WSDL περιγράφει τι κάνει το συγκεκριμένο web service, που ανήκει και πως καλείται. Η γλώσσα

WSDL βασίζεται στην XML και χρησιμοποιεί τα αντίστοιχα πρωτόκολλα για την ανταλλαγή μηνυμάτων.

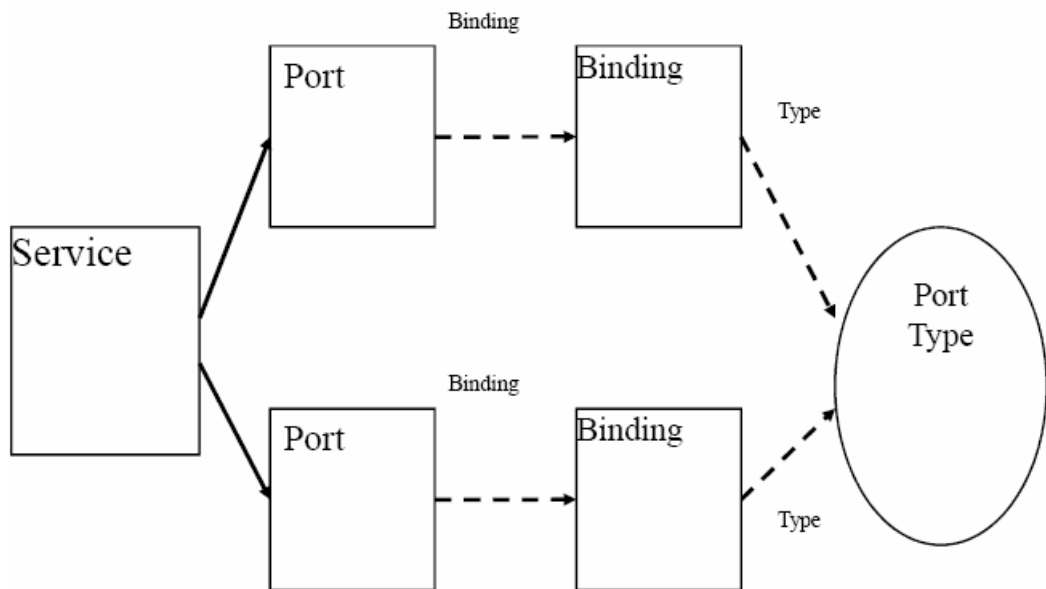


Εικόνα 22 : WSDL

Οι προδιαγραφές της WSDL μπορούν να διαχωριστούν σε δύο μέρη:

- Τον καθορισμό της διεπαφής της υπηρεσίας που περιγράφει την γενική δομή του Web Service.
- Την υλοποίηση της υπηρεσίας που συνδέει το αφηρημένο μέρος (abstract) σε μια συγκεκριμένη διεύθυνση δικτύου, σε ένα συγκεκριμένο πρωτόκολλο, και σε πραγματικές δομές δεδομένων.

Η WSDL καθορίζει γραμματική και σύνταξη που περιγράφει ένα Web Service σαν μια συλλογή από επικοινωνιακά άκρα. Χρησιμοποιεί XML και τοποθετείται πάνω από ένα XML schema. Παρέχει όλα τα μέσα για την ομαδοποίηση των μηνυμάτων σε λειτουργίες και τις λειτουργίες σε επαφές. Τα δεδομένα που ανταλλάσσονται μεταξύ των άκρων έχουν καθοριστεί ως μέρος των μηνυμάτων και κάθε δραστηριότητα επεξεργασίας που επιτρέπεται σε ένα άκρο θεωρείται ως μια λειτουργία. Λειτουργίες που έχουν τις σχετικές άδειες χρήσης ομαδοποιούνται σε ένα άκρο σε τύπους θύρας.



Εικόνα 23 : Υλοποίηση WSDL

3 Μεθοδολογία

3.1 Εισαγωγή

Στο παρόν κεφάλαιο παρουσιάζεται ο τρόπος υλοποίησης της εφαρμογής σε γλώσσα προγραμματισμού C#. Η εφαρμογή μέσω Web Services παρέχει στον χρήστη υπηρεσίες βασισμένες στις προτιμήσεις του. Ο χρήστης αλληλεπιδρά με την εφαρμογή μέσω ενός friendly interface σε mobile συσκευή.

3.2 Εργαλεία

Τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της συγκεκριμένης εφαρμογής είναι τα παρακάτω :

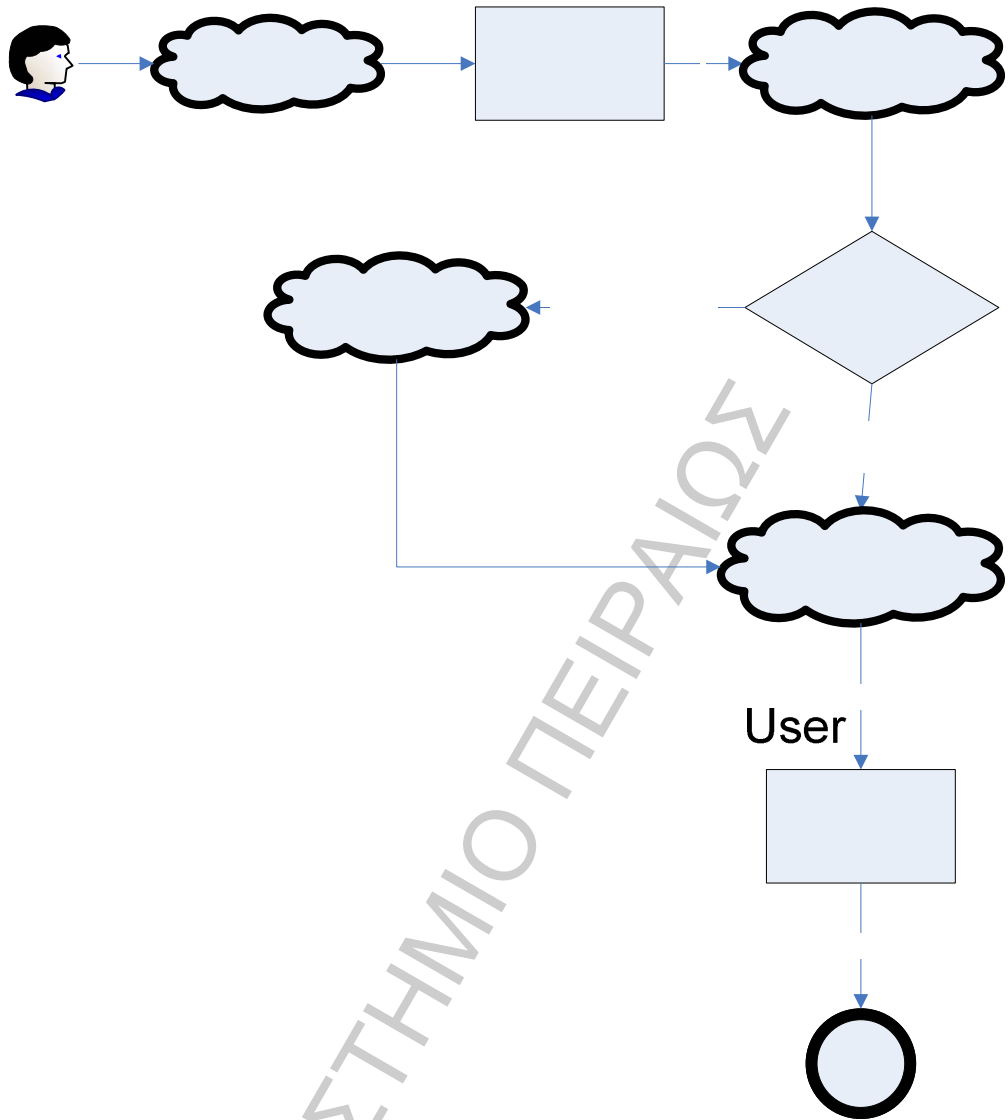
- Microsoft Visual Studio 2008
- SQL Server 2008.

3.3 Σκεπτικό Επίλυσης

Σκοπός της συγκεκριμένης εφαρμογής είναι να παρέχει υπηρεσίες και να καταγράφει τις προτιμήσεις του χρήστη σε σύντομο χρονικό διάστημα. Τα χαρακτηριστικά και οι προτιμήσεις του χρήστη αποθηκεύονται σε μία βάση δεδομένων (Sql Database).

Κάθε φορά που ο χρήστης επιλέγει μια υπηρεσία η εφαρμογή εμφανίζει με τη χρήση web service προηγούμενες επιλογές του ως προς το δίκτυο, την ποιότητα υπηρεσίας και το κόστος της.

Μετά την επιλογή της υπηρεσίας η εφαρμογή ενημερώνει την βάση δεδομένων μέσω web service με πληροφορίες όπως το δίκτυο, η ποιότητα υπηρεσίας και το κόστος.



Εικόνα 24 : Διάγραμμα ροής της Εφαρμογής

Στο παραπάνω διάγραμμα ροής εμφανίζονται τέσσερα web services τα οποία χρησιμοποιούνται στην εφαρμογή κατά την αλληλεπίδραση της με τον χρήστη. Αναλυτικότερα :

- Web Service (1): Είναι το login Web Service, το οποίο παίρνει ως παραμέτρους username / password και επιστρέφει μια Boolean μεταβλητή που πιστοποιεί τον χρήστη.
- Web Service (2): Είναι το AVL_Services Web Service, το οποίο παίρνει ως παραμέτρους location και service. Οι πιθανές τιμές για το location είναι : Office, Home, Other και για το Service : Voice, Video, Data. Η εφαρμογή εντοπίζει την τοποθεσία του χρήστη και με βάση την επιλογή του service του δείχνει όλα τα διαθέσιμα δίκτυα με πληροφορίες για την ποιότητα (QoS) και το κόστος (Cost).
- Web Service (3): Είναι το Insert preferences Web Service. Αυτό χρησιμοποιείται σε δύο σημεία της εφαρμογής. Εισάγει στην βάση δεδομένων όλες τις απαραίτητες πληροφορίες από την επιλογή του χρήστη. Σαν παραμέτρους παίρνει , service, location χρήστη, QoS της επιλεγμένης υπηρεσίας, user id, κόστος της επιλεγμένης υπηρεσίας, πάροχος της επιλεγμένης υπηρεσίας.
Το Web Service αυτό στην ουσία ανοίγει ένα session στην βάση της εφαρμογής και εισάγει δεδομένα στον πίνακα που κρατάει πληροφορίες για τις προτιμήσεις του χρήστη.
- Web Service (4): Είναι το User Services WS. Παίρνει ως παραμέτρους το user id ,το location και το service και επιστρέφει τις προτιμήσεις του συγκεκριμένου χρήστη για την τοποθεσία που βρίσκεται και την υπηρεσία που έχει επιλέξει. Κάθε φορά που επιλέγει μια εγγραφή αυξάνεται ο counter και εμφανίζονται οι επιλογές με σειρά προτιμήσεως.

4 Αποτελέσματα

Στο παρόν κεφάλαιο παρουσιάζεται αναλυτικά η εφαρμογή που υλοποιήθηκε μέσω ενός friendly interface το οποίο μπορεί να χρησιμοποιηθεί σε mobile συσκευή.

4.1 Εισαγωγή

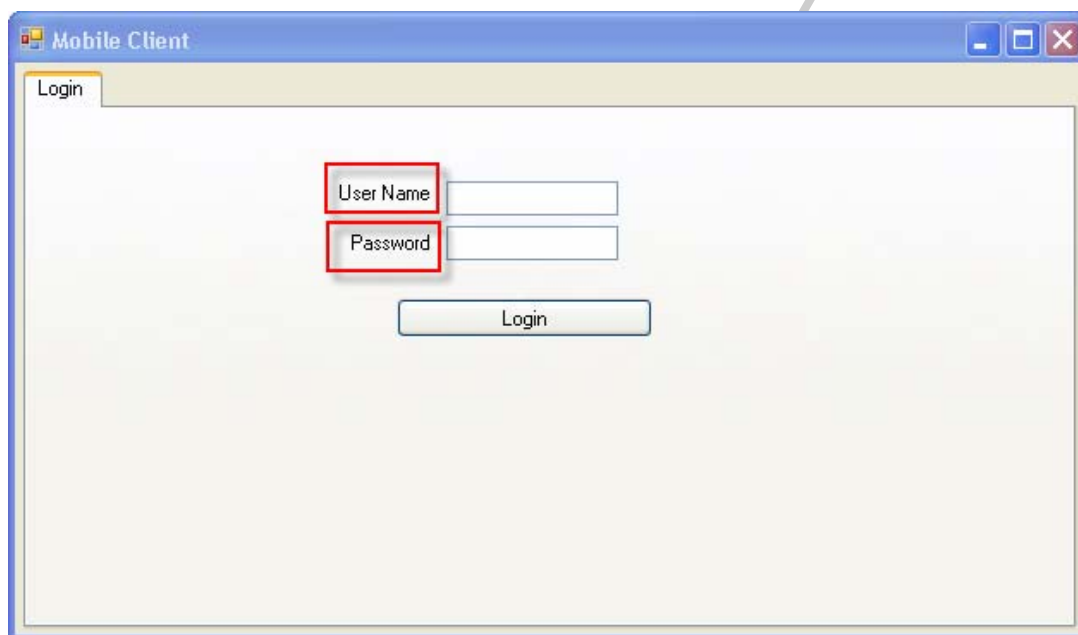
Όλοι οι χρήστες της εφαρμογής για να χρησιμοποιήσουν την παρούσα εφαρμογή πρέπει να είναι αποθηκευμένοι σε μια βάση δεδομένων, με όλα τα απαραίτητα στοιχεία τους. Περιληπτικά τα βήματα και οι ενέργειες που γίνονται είναι οι εξής:

- Πιστοποιημένος χρήστης κάνει login στην εφαρμογή χρησιμοποιώντας ένα κωδικό όνομα (User name) και τον κωδικό του (password). Αν ο συνδυασμός αυτός δεν είναι έγκυρος, η εφαρμογή ειδοποιεί τον χρήστη.
- Ο χρήστης επιλέγει ποια υπηρεσία επιθυμεί να χρησιμοποιήσει (Voice ,Video, Data). Η εφαρμογή έχει καταγράψει την τοποθεσία που βρίσκεται ο χρήστης μέσω της κινητής του συσκευής (Home, Office, Other) και του εμφανίζει μια λίστα με όλα τα διαθέσιμα δίκτυα με βάση την επιλεγμένη υπηρεσία και την τοποθεσία του χρήστη. Η λίστα περιέχει πληροφορίες για τον πάροχο (Vendor), το κόστος (Cost) και την ποιότητα της υπηρεσίας (QoS).
- Ο χρήστης έχει δύο επιλογές στο σημείο αυτό:
 1. Να επιλέξει παροχή υπηρεσιών από την λίστα.
 2. Να δει τις προηγούμενες επιλογές του για την συγκεκριμένη υπηρεσία και τοποθεσία που βρίσκεται.
- Όταν ο χρήστης επιλέξει τον πάροχο που θα τον εξυπηρετήσει, η εφαρμογή αποθηκεύει την προτίμηση αυτή στην βάση δεδομένων, προκειμένου να υπάρχει μια top rated λίστα για τον κάθε χρήστη με βάση την υπηρεσία και την τοποθεσία.

4.2 Αποτελέσματα Εφαρμογής

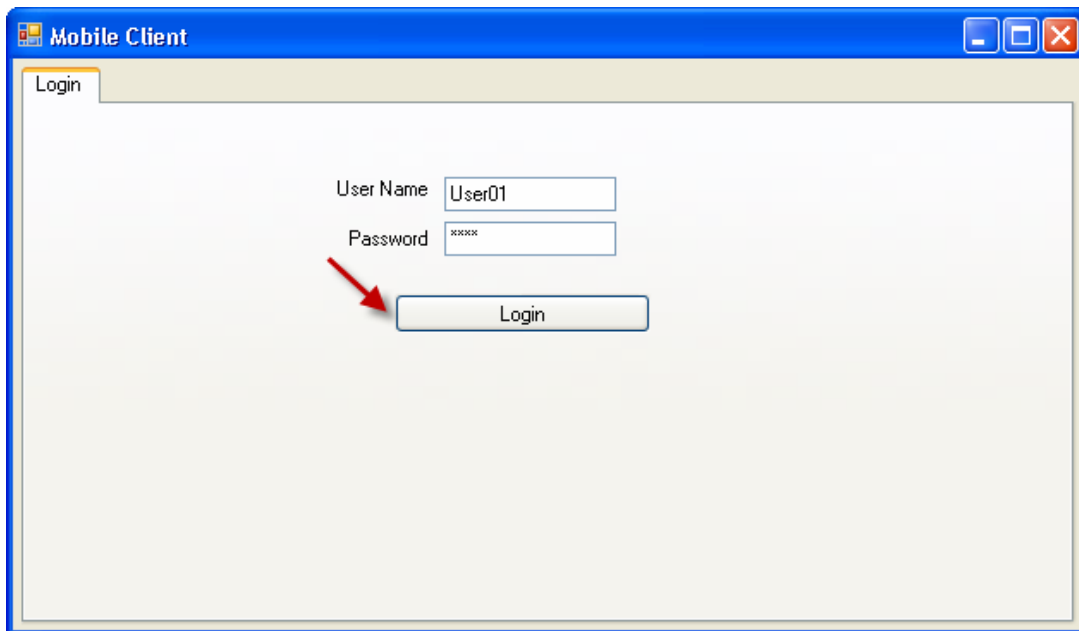
Παρακάτω παρουσιάζονται screenshots της παρούσας εφαρμογής.

Κατά την εκκίνηση της εφαρμογής ζητείται από τον χρήστη να παρέχει User Name και password.



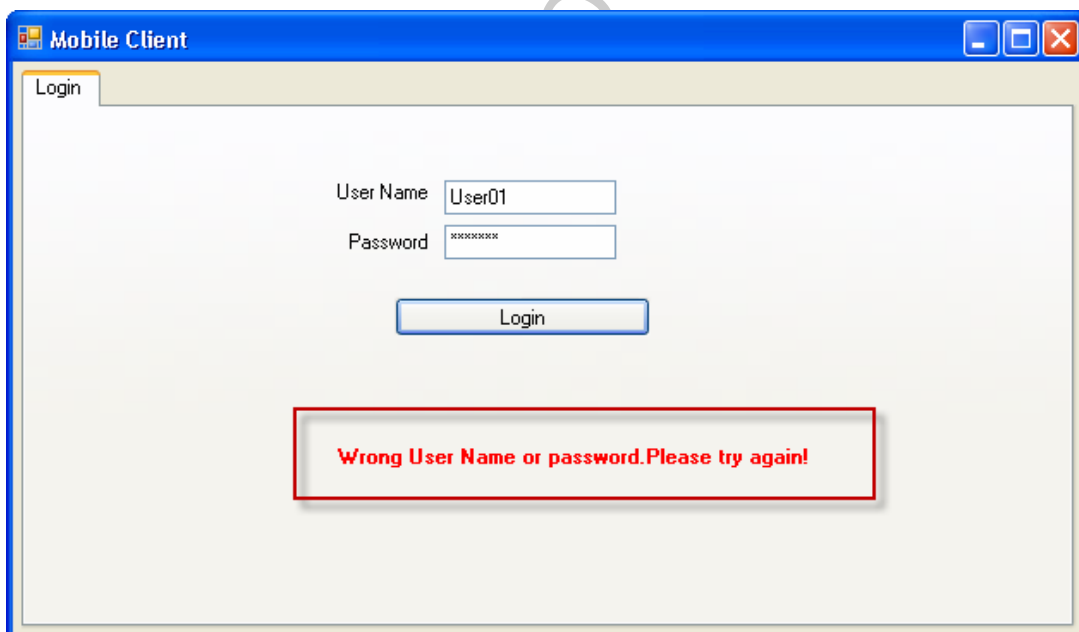
Εικόνα 25 : Login Page εφαρμογής

Ο χρήστης εισάγει τα credentials του και πατάει το κουμπί Login προκειμένου να εισέλθει στην εφαρμογή



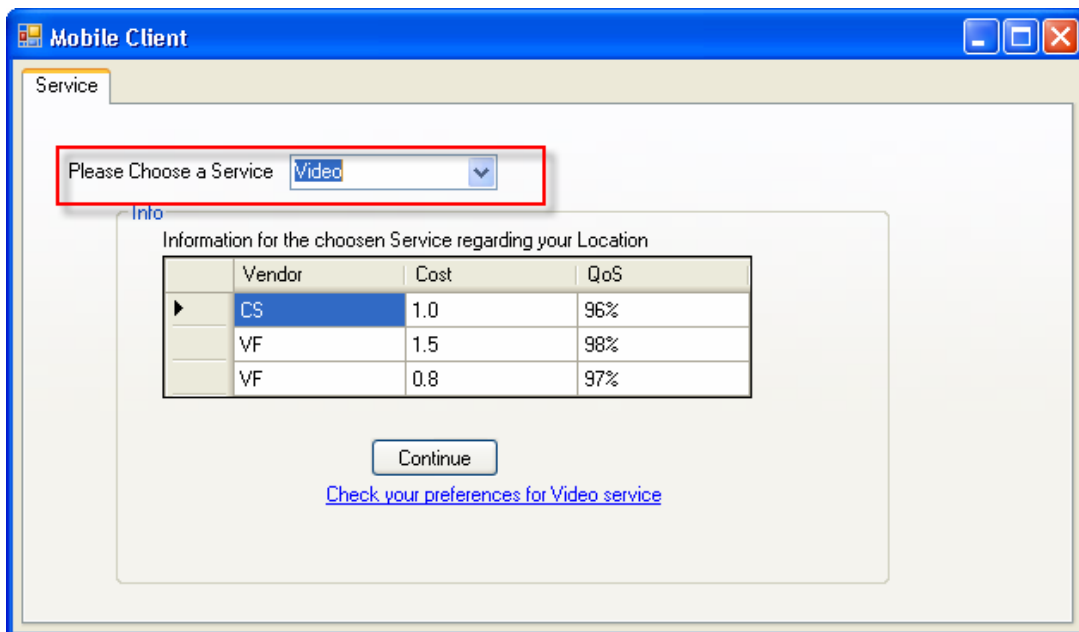
Εικόνα 26 : Login

Αν ο συνδυασμός User Name και password δεν είναι σωστό, η εφαρμογή ειδοποιεί τον χρήστη και τον προτρέπει να ξαναδοκιμάσει



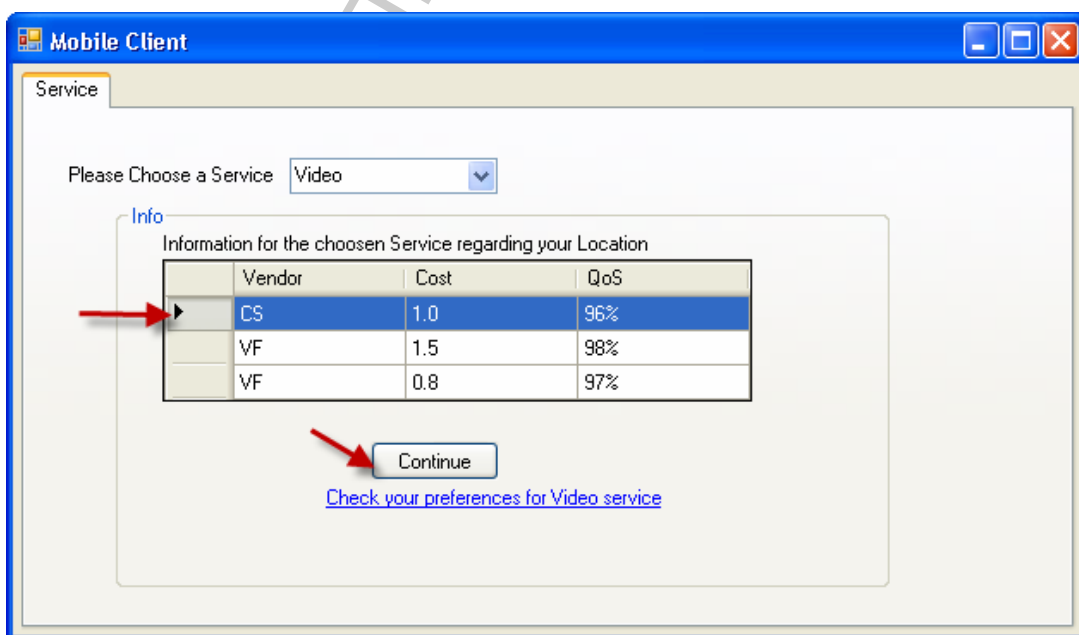
Εικόνα 27 : Λάθος User name και password

Ο χρήστης επιλέγει την υπηρεσία που θέλει και η εφαρμογή του δείχνει όλους τους διαθέσιμους παρόχους με βάση την τοποθεσία που βρίσκεται



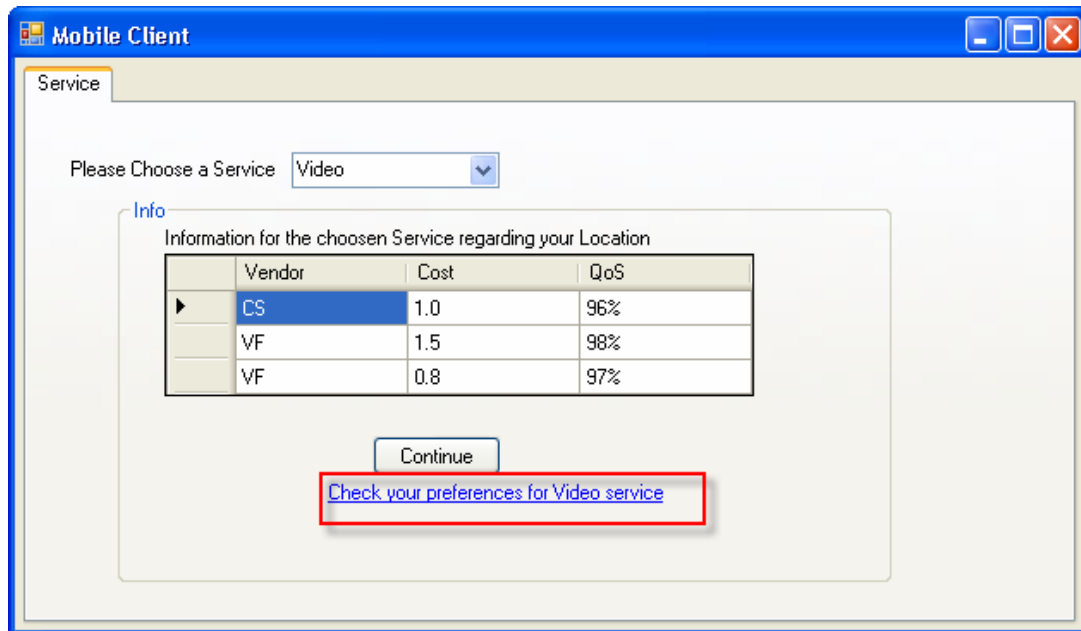
Εικόνα 28 : Επιλογή υπηρεσίας

Στην περίπτωση που ο χρήστης δεν θέλει να δει προηγούμενες επιλογές του, διαλέγει έναν πάροχο και πατάει Continue προκειμένου να χρησιμοποιήσει την υπηρεσία που επέλεξε.



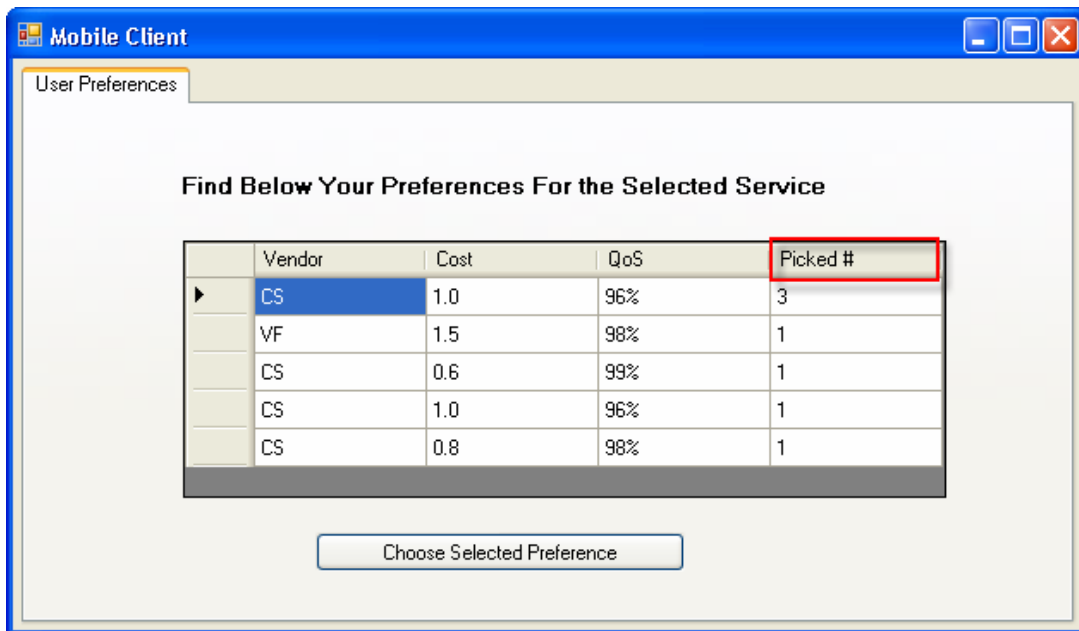
Εικόνα 29 : Επιλογή παρόχου

Στην περίπτωση που ο χρήστης θέλει να δει τις προτιμήσεις του για την υπηρεσία που επέλεξε , πατάει το link



Εικόνα 30 : Προτιμήσεις χρήστη

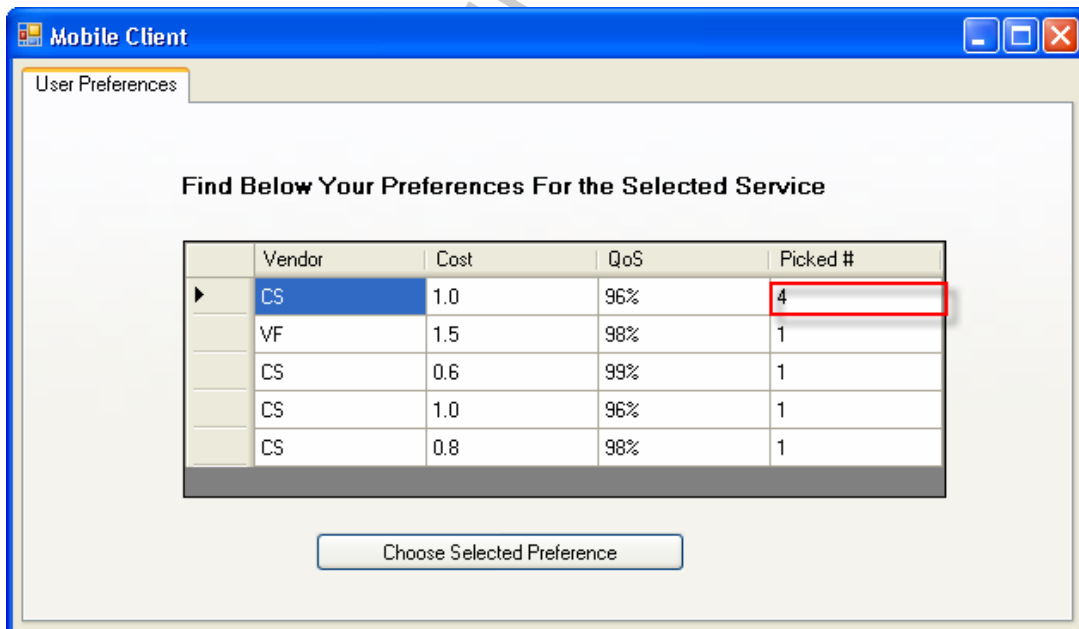
Η εφαρμογή παρουσιάζει όλες τις επιλογές του χρήστη με βάση την υπηρεσία και την τοποθεσία που βρίσκεται. Οι πληροφορίες αυτές είναι ταξινομημένες με αύξουσα σειρά με βάση την top επιλογή του χρήστη.



Εικόνα 31 : Top rated προτιμήσεις χρήστη

Στο παρόν σενάριο ο χρήστης επιλέγει την μαρκαρισμένη επιλογή η οποία είναι ήδη η top rated.

Την επόμενη φορά που θα κάνει login και θα θέλει να χρησιμοποιήσει την ίδια υπηρεσία , το Picked# θα πρέπει να έχει αυξηθεί κατά 1.



Εικόνα 32 : Καταγραφή προτίμησης χρήστη

5 Βιβλιογραφία

- [1] J.Mitola III, Cognitive radio for flexible mobile multimedia communications. In Mobile Multimedia Communications, 1999
- [2] Paul Sutton, Linda E.Doyle και Keith E.Nolan, A Reconfigurable Platformfor Γνωστικά Networks, 2006
- [3] V.Stavroulaki, A.Katidiotis, D.Petromanolakis, Management Functionality for Cognitive Wireless Terminal, 2008
- [4] V.Stavroulaki , Y.Kritikou, P.Demestichas, Introducing cognition in the management of equipment in the future wireless world, 2008
- [5] Likert, R., A Technique for the Measurement of Attitudes, Archives of Psychology 140: pp. 1-55, 1932
- [6]J. Orwant. Heterogeneous Learning in the Doppelgänger User Modeling System. User Modeling and User-Adapted Interaction, 4:107–130, 1995.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑΣ

ΠΑΡΑΡΤΗΜΑ Ι

Στο παρόν παράρτημα ακολουθεί ο κώδικας υλοποίησης για τα Web Services που χρησιμοποιούνται στην παρούσα εφαρμογή.

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace WebServices
{
    /// <summary>
    /// Summary description for AVL_Services
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    public class AVL_Services : System.Web.Services.WebService
    {

        [WebMethod]
        public Avail_serv.S_AVL_SERVDataTable avl_srv(string
        srv, string loc)
        {
            Avail_serv.S_AVL_SERVDataTable avl = new
            Avail_serv.S_AVL_SERVDataTable();
            Avail_servTableAdapters.S_AVL_SERVTableAdapter avltbl =
            new WebServices.Avail_servTableAdapters.S_AVL_SERVTableAdapter();
            avltbl.Fill(avl, srv, loc);
            return avl;
        }
    }
}
```

```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Data.SqlClient;
using System.Collections.Generic;

namespace WebServices
{
    /// <summary>
    /// Summary description for Service1
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    public class Insert_Pref : System.Web.Services.WebService
    {

        [WebMethod]
        public bool InsertData(string code, string loc, string
usr, string cost, string qos, string ven)
        {
            SqlConnection mySqlConnection = new
SqlConnection(@"Server=SOUSHIBA\SQLEXPRESS;Database=MSC_db;Trusted_
Connection=Yes;");
            mySqlConnection.Open();

            SqlCommand mySqlCommand =
mySqlConnection.CreateCommand();
            mySqlCommand.CommandText = "INSERT INTO S_SERVICES
(" +
                " CODE, LOCATION, USER_NAME,COST, QOS,VENDOR " +
                ") VALUES (" +
                "@code, @loc, @usr, @cost, @qos,@ven" +
                ")";

            mySqlCommand.Parameters.Add("@code",
SqlDbType.NVarChar, 50);
            mySqlCommand.Parameters.Add("@loc",
SqlDbType.NVarChar, 50);
            mySqlCommand.Parameters.Add("@usr",
SqlDbType.NVarChar, 50);
            mySqlCommand.Parameters.Add("@cost",
SqlDbType.NVarChar, 50);
            mySqlCommand.Parameters.Add("@qos",
SqlDbType.NVarChar, 50);
            mySqlCommand.Parameters.Add("@ven",
SqlDbType.NVarChar, 50);

            mySqlCommand.Parameters["@code"].Value = code;
            mySqlCommand.Parameters["@loc"].Value = loc;
            mySqlCommand.Parameters["@usr"].Value = usr;
        }
    }
}

```

```

mySqlCommand.Parameters["@cost"].Value = cost;
mySqlCommand.Parameters["@qos"].Value = qos;
mySqlCommand.Parameters["@ven"].Value = ven;

int i = 0;
i = mySqlCommand.ExecuteNonQuery();
mySqlConnection.Close();
    if(i>0)
    {
        return true;
    }
    else
    {
        return false;
    }
} //Web Method
} //Class
} //Namespace

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace WebServices
{
    /// <summary>
    /// Summary description for Service1
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    public class Login : System.Web.Services.WebService
    {
        [WebMethod]
        public int GetUsr(string username, string passw)
        {
            Users.S_USERSDataTable users = new
Users.S_USERSDataTable();
            UsersTableAdapters.S_USERSTableAdapter ustbl = new
WebServices.UsersTableAdapters.S_USERSTableAdapter();
            ustbl.Fill(users, username, passw);
            int countrows = users.Rows.Count;
            return countrows;
        }
    }
}
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

```



```

namespace WebServices
{
    /// <summary>
    /// Summary description for Service1
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    public class Preferences : System.Web.Services.WebService
    {
        [WebMethod]
        public prefer.S_SERVICESDataTable pref (string USR, string
SRV, string VND)
        {
            prefer.S_SERVICESDataTable usrpref = new
prefer.S_SERVICESDataTable();
            preferTableAdapters.S_SERVICESTableAdapter preftbl =
new WebServices.preferTableAdapters.S_SERVICESTableAdapter();
            preftbl.Fill(usrpref, USR,SRV,VND);
            return usrpref;
        }
    }
}

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace WebServices
{
    /// <summary>
    /// Summary description for Services_Usr
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [ToolboxItem(false)]
    public class Services_Usr : System.Web.Services.WebService
    {
        [WebMethod]
        public Services_users_ds.S_SERVICESDataTable GetExisting
(string loc, string usr)
        {
            Services_users_ds.S_SERVICESDataTable prf = new
Services_users_ds.S_SERVICESDataTable();
            Services_users_dsTableAdapters.S_SERVICESTableAdapter
test = new
WebServices.Services_users_dsTableAdapters.S_SERVICESTableAdapter()
;
            test.Fill(prf, loc, usr);
            return prf;
        }
    }
}

```

ΠΑΡΑΡΤΗΜΑ Ι Ι

Στο παρόν παράρτημα ακολουθεί ο κώδικας υλοποίησης για το interface που χρησιμοποιούνται στην παρούσα εφαρμογή.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace CustomClient
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            this.Client.TabPages.Remove(tabService);
            this.Client.TabPages.Remove(tabAvail);
            this.Client.TabPages.Remove(Pref);
            this.AutoSizeMode =
System.Windows.Forms.AutoSizeMode.GrowAndShrink;

            this.AutoSize = true;
        }
        private int RandomNumber(int min, int max)
        {
            Random random = new Random();
            return random.Next(min, max);
        }
        private void Loginbtn_Click(object sender, EventArgs e)
        {
            localhost_Login.Login usr = new
CustomClient.localhost_Login.Login();
            int rusr = usr.GetUsr(UserNametxt.Text.ToString(),
passtxt.Text.ToString());
            int loc = RandomNumber(1, 4);
            string location = "";

            if (loc == 1)
            {
                Resulttxt.Visible = true;
                location = "Office";
            }
            if (loc == 2)
            {
```

```

        Resulttxt.Visible = true;
        //location = "Home";
        location = "Home";
    }
    if (loc == 3)
    {
        Resulttxt.Visible = true;
        // location = "Other";
        location = "Other";
    }

    if (rusr == 0)
    {
        Resulttxt.Visible = true;
        Resulttxt.Text = "Wrong User Name or
password.Please try again!";
    }
    else
    {
        this.Client.TabPages.Add(tabService);
        this.Client.TabPages.Remove(tabLogin);

        locationlbl.Text = location.ToString();
    }
}

private void linkLabel1_LinkClicked(object sender,
LinkLabelLinkClickedEventArgs e)
{
    this.Client.TabPages.Add(Pref);
    this.Client.TabPages.Remove(tabService);

    string loc3 = locationlbl.Text.ToString();
    string srv2 = Servicecb.Text.ToString();

    services_users_ds1.S_SERVICES.Rows.Clear();
    CustomClient.localhost_Existing.Services_Usr prf = new
CustomClient.localhost_Existing.Services_Usr();
    services_users_ds1.Merge(prf.GetExisting(loc3,
UserNametxt.Text.ToString()));
}

private void Servicecb_SelectedIndexChanged(object sender,
EventArgs e)
{
    string loc2 = locationlbl.Text.ToString();
    string srv = Servicecb.Text.ToString();

    string lnk = "Check your preferences for " +
Servicecb.Text.ToString() + " service";
    linkLabel1.Visible = true;
    linkLabel1.Text = lnk.ToString();
    avail_serv1.S_AVL_SERV.Rows.Clear();

    CustomClient.localhost_Avl.AVL_Services avl = new
CustomClient.localhost_Avl.AVL_Services();
}

```

```

        avail_serv1.Merge(avl.avl_srv(srv, loc2));
    }

    void dataGridView1_SelectionChanged(object sender,
EventArgs e)
    {
        if (avail_serv1.S_AVL_SERV.Count!= 0)
        {
            int i = dataGridView1.CurrentRow.Index;

            costlbl.Text =
(string)dataGridView1.Rows[i].Cells[1].Value;
            qoslbl.Text =
(string)dataGridView1.Rows[i].Cells[2].Value;
            vendorlbl.Text =
(string)dataGridView1.Rows[i].Cells[0].Value;
        }
    }

    private void button1_Click(object sender, EventArgs e)
    {

        this.Client.TabPages.Add(tabAvail);
        this.Client.TabPages.Remove(tabService);
        srvlbl.Text = Servicecb.Text.ToString();

        string dloc = locationlbl.Text.ToString();
        string dcode = Servicecb.Text.ToString();

        localhost_Insert.Insert_Pref ins = new
CustomClient.localhost_Insert.Insert_Pref();
        bool dins = ins.InsertData(dcode, dloc,
UserNametxt.Text.ToString(), costlbl.Text.ToString(),
qoslbl.Text.ToString(), vendorlbl.Text.ToString());
    }

    private void label17_Click(object sender, EventArgs e)
    {
    }

    private void Form1_Load(object sender, EventArgs e)
    {
        // TODO: This line of code loads data into the
'user_pref.S_SERVICES' table. You can move, or remove it, as
needed.
        //
this.s_SERVICESTableAdapter.Fill(this.user_pref.S_SERVICES);
    }

    private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {

```

```

    }

    private void button4_Click(object sender, EventArgs e)
    {
        this.Client.TabPages.Add(tabAvail);
        this.Client.TabPages.Remove(Pref);

        string dloc = locationlbl.Text.ToString();
        string dcode = Servicecb.Text.ToString();

        int i = dataGridView2.CurrentRow.Index;

        costlbl2.Text =
        (string)dataGridView2.Rows[i].Cells[1].Value;
        qoslbl2.Text =
        (string)dataGridView2.Rows[i].Cells[2].Value;
        vendorlbl2.Text =
        (string)dataGridView2.Rows[i].Cells[0].Value;

        localhost_Insert.Insert_Pref ins = new
        CustomClient.localhost_Insert.Insert_Pref();
        bool dins = ins.InsertData(dcode, dloc,
        UserNametxt.Text.ToString(), costlbl2.Text.ToString(),
        qoslbl2.Text.ToString(), vendorlbl2.Text.ToString());

    }

    private void dataGridView2_CellContentClick(object sender,
    DataGridViewCellEventArgs e)
    {

    }

    private void toolStripMenuItem1_Click(object sender,
    EventArgs e)
    {

    }
}

```