

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Κινητά Συστήματα Συνεργασίας / Επιχειρηματολογίας (Argumentation) στον Παγκόσμιο Ιστό / Διαδίκτυο

Διονύσιος Σ. Καρούσος

Μ.Π.Σ: «Μαθηματικά και Σύγχρονες Εφαρμογές»
Μ.Δ.Ε: Υπολογιστικά Μαθηματικά – Πληροφορική στην Εκπαίδευση
Κατεύθυνση: Μαθηματικά των Υπολογιστών και Υπολογιστική Νοημοσύνη

Επιβλέπων: Γ. Μητακίδης, Καθηγητής

Πάτρα, Μάρτιος 2011

ΕΥΧΑΡΙΣΤΙΕΣ

Ο Καθηγητής κ. Γεώργιος Μητακίδης ήταν αυτός που από τη πρώτη στιγμή της συνεργασίας μας μου δίδαξε πολλά και μου έδειξε τον δρόμο για την επιστημονική μου πρόοδο με υπομονή, μεθοδικότητα και στοχευμένες συμβουλές. Αποτελεί ιδιαίτερη τιμή για εμένα το ότι υπήρξα φοιτητής του. Του οφείλω λοιπόν το πρώτο ευχαριστώ μέσα από την καρδιά μου.

Ο Επίκουρος Καθηγητής κ. Δημήτριος Καββαδίας ήταν αυτός που καθ' όλη τη διάρκεια των σπουδών μου, σε εύκολες και δύσκολες στιγμές, στάθηκε δίπλα μου ως πραγματικός «δάσκαλος». Τον ευχαριστώ θερμά για τις γνώσεις που μου προσέφερε αλλά κυρίως για το ήθος, την ευπρέπεια και τη κατανόηση που δείχνει απέναντι στους φοιτητές του.

Ομοίως θέλω να ευχαριστήσω και τον Επίκουρο Καθηγητή κ. Παναγή Καραζέρη ο οποίος μου έκανε τη τιμή να συμμετάσχει στην τριμελή εξεταστική επιτροπή της διπλωματικής μου εργασίας.

Στη διάρκεια των μεταπτυχιακών σπουδών μου, συνεργάσθηκα με αρκετά άτομα τα οποία συνεισέφεραν στην πρόοδο μου, μέσω της συζήτησης, της συνεργασίας και των αμέτρητων ωρών ομαδικής μελέτης. Πάνω απ' όλα όμως, οφείλω ένα μεγάλο ευχαριστώ στους φίλους και συμφοιτητές Νίκο Κόλλια και Αντιγόνη Γκορόγια, όπου η στήριξη που μου προσέφεραν απλόχερα, αποτέλεσε για εμένα βασική αιτία επιτυχίας των σπουδών μου.

Ένα πολύ μεγάλο ευχαριστώ χρωστάω στον αδερφό μου Νίκο, ο οποίος καθόλη τη διάρκεια των σπουδών μου στάθηκε δίπλα μου, όχι μόνο ως αδερφός αλλά και ως «δάσκαλος» που με επιμονή και υπομονή με καθοδηγούσε επιτυχώς σε κάθε τι που με προβλημάτιζε.

Για το τέλος άφησα τους δικούς μου ανθρώπους, αυτούς που τους έχω πολύ βαθιά στη καρδία μου. Οφείλω τουλάχιστον ένα πολύ πολύ μεγάλο ευχαριστώ στους ανθρώπους που με «ανέχτηκαν» όλα αυτά τα χρόνια και «κουράστηκαν» να γίνονται αποδέκτες του άγχους και των προβληματισμών μου, τη γυναίκα μου Μαρία, τη μητέρα μου Βιβή, τον αδερφό μου Σωκράτη και τα «αδέρφια» μου Νίκο και Θανάση.

Κλείνοντας, νιώθω την ανάγκη να αφιερώσω την όλη αυτή προσπάθεια στον πατέρα μου Σωτήρη και το νουνό μου Γιάννη, ως ένα ελάχιστο δείγμα ευγνωμοσύνης.

ΠΕΡΙΛΗΨΗ

Η ραγδαία ανάπτυξη του Παγκόσμιου Ιστού και η ταχύτατη διείσδυσή του σε κάθε έκφανση της καθημερινότητας του σύγχρονου ανθρώπου είναι έκδηλη. Με την ανάπτυξη της τεχνολογίας των κινητών τηλεφώνων νέας γενιάς και των αντίστοιχων δικτύων, η πρόσβαση στον Παγκόσμιο Ιστό μέσω κινητών συσκευών δεν γίνεται μόνο περιστασιακά, αλλά πολλές φορές αποτελεί τον πρωτεύοντα τρόπο πρόσβασης. Επομένως, η ανάγκη αξιοποίησης των δυνατοτήτων των κινητών τηλεφώνων και αυτών που προσφέρει το Web2.0 είναι συνεχώς αυξανόμενη. Σε αυτό το πλαίσιο, η παροχή κάθε προσφερόμενης υπηρεσίας των Συστημάτων Υποστήριξης Επιχειρηματολογίας στο Web και στις κινητές συσκευές, φαίνεται αρκετά υποσχόμενη. Αυτό μπορεί να γίνει είτε δημιουργώντας τις αντίστοιχες Mobile εφαρμογές, είτε δημιουργώντας υπηρεσίες, οι οποίες είναι απόλυτα προσβάσιμες και από τις συσκευές κινητών τηλεφώνων.

Στα πλαίσια της Διπλωματικής εργασίας προσπαθήσαμε να εκμεταλλευτούμε τα πλεονεκτήματα που δίνει ένα κινητό τηλέφωνο, όσον αφορά την ευχρηστία, την ευκολία πρόσβασης και τη χρήση του Web2.0, δημιουργώντας μία Mobile εφαρμογή που παρέχει σε χρήστες κινητού τηλεφώνου, με λειτουργικό σύστημα Windows Mobile, τη δυνατότητα να συμμετέχουν σε ένα συγκεκριμένο Σύστημα Υποστήριξης Επιχειρηματολογίας, το CoPe_it!. Με αυτόν τον τρόπο επιχειρούμε να συνεισφέρουμε στην προσπάθεια διεύρυνσης των πιθανών χρήσεων του συγκεκριμένου συστήματος.

Σε πρώτη φάση μελετήθηκε η εξέλιξη του Παγκόσμιου Ιστού από το Web1.0 στο Web2.0 και οι προσφερόμενες σε αυτόν υπηρεσίες. Έπειτα παρουσιάστηκε η ανάπτυξη του Κινητού Παγκόσμιου Ιστού και το πώς μέσα από τις τεχνολογικές εξελίξεις αλλά και τις καθημερινές συνήθειες του μέσου ανθρώπου, οδηγούμαστε στην ανάγκη ύπαρξης δυνατότητας μεταφοράς κάθε είδους δραστηριότητας και προσφερόμενων υπηρεσιών από το Web στο Mobile Web.

Στη συνέχεια μελετήθηκαν τα Συστήματα Υποστήριξης Επιχειρηματολογίας, ξεκινώντας από την ανάλυση κάποιων βασικών εννοιών, συνεχίζοντας με τα μοντέλα διαχείρισης επιχειρημάτων και καταλήγοντας σε εμπεριστατωμένη έρευνα των αντιστοίχων ηλεκτρονικών εργαλείων υποστήριξης που είναι διαθέσιμα είτε απλά ως ηλεκτρονικές εφαρμογές, είτε ως on-line Web-based εφαρμογές. Έπειτα αποκαλύπτεται το κενό που υπάρχει στο αντίστοιχο πεδίο του Mobile Web από την έλλειψη αντίστοιχης εφαρμογής, με μοναδικό «φωτεινό» σημείο την ύπαρξη εφαρμογών παραπλήσιου ενδιαφέροντος.

Τέλος, παρουσιάστηκε η όλη διαδικασία ανάπτυξης της on-line κινητής εφαρμογής, Mobile CoPe_it!, αφού προηγήθηκε εκτενής παρουσίαση του ήδη υπάρχοντος μητρικού συστήματος.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή 3

1.1 Αντικείμενο της Διπλωματικής Εργασίας 4

1.2 Διάρθρωση της Διπλωματικής Εργασίας 4

ΚΕΦΑΛΑΙΟ 2: Η μετάβαση από το Web στο Mobile Web 6

2.1 Web1.0 7

2.1.1 Τα βασικά μέρη του Παγκόσμιου Ιστού 9

2.2 Web2.0 11

2.2.1 Οι βασικές αρχές του Web2.0 13

2.2.2 Επίπεδα ιεραρχίας του Web2.0 22

2.2.3 Βασικές διαφορές του Web2.0 από το Web1.0 24

2.3 Mobile Web 25

2.3.1 Πρότυπα για το περιεχόμενο του Mobile Web 27

2.3.2 Πρότυπα κινητών συσκευών για χρήση Mobile Web 33

2.3.3 Γλώσσες Σήμανσης Περιεχομένου (Markup Languages) 34

2.3.3.1 WML μία παγκόσμια γλώσσα - πρότυπο για το Mobile Web 34

2.3.3.2 XHTML μία παγκόσμια γλώσσα πρότυπο για το Web 35

2.3.3.3 XHTML Mobile Profile μία παγκόσμια Markup γλώσσα για το Mobile Web . 35

2.3.4 Σύγκριση Web - Mobile Web 36

2.3.5 Ανάπτυξη δικτύων κινητής τηλεφωνίας 39

ΚΕΦΑΛΑΙΟ 3: Συστήματα Υποστήριξης Επιχειρηματολογίας 43

3.1 Εισαγωγή στην επιχειρηματολογία 44

3.1.1 Επιχείρημα 44

3.1.2 Επιχειρηματολογία 45

3.1.3 Επιχειρηματολογική κατασκευή γνώσης ... 45
3.1.4 Αιτιολόγηση (Reasoning) ... 46
3.1.5 Μοντέλα διαχείρισης επιχειρημάτων ... 47
3.2 Εφαρμογές αναπαράστασης σκέψης και απεικόνισης διαλόγων με επιχειρήματα ... 54
3.2.1 Συστήματα Υποστήριξης Επιχειρηματολογίας στο Web ... 62
3.2.2 Συστήματα υποστήριξης επιχειρηματολογίας στο Mobile-Web ... 69
ΚΕΦΑΛΑΙΟ 4: Υλοποιώντας μια Mobile Εφαρμογή Υποστήριξης Επιχειρηματολογίας ... 71
4.1 Το σύστημα CoPe_it! ... 72
4.1.1 Γενικά ... 72
4.1.2 Θεωρητική τεκμηρίωση της λειτουργίας του συστήματος και μοντέλο δεδομένων ... 72
4.1.3 Χαρακτηριστικές λειτουργίες ... 77
4.1.4 Στοιχεία λειτουργίας ... 79
4.2 Βασικές απαιτήσεις της εφαρμογής ... 81
4.3 Σχεδιασμός ... 82
4.4 Ανάπτυξη ... 84
4.4.1 Server Side ... 84
4.4.1.1 Τι είναι τα Web Services; ... 84
4.4.1.2 Mobile CoPe_it! Web Service ... 87
4.4.2 Mobile Application Side ... 89
4.5 Περιγραφή λειτουργίας – Εγχειρίδιο χρήσης ... 91
4.6 Θέματα εξέλιξης ... 100
ΚΕΦΑΛΑΙΟ 5: Σύνοψη ... 102
5.1 Σύνοψη ... 103
5.2 Συμπεράσματα ... 103

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή

Η παρούσα Διπλωματική Εργασία πραγματεύεται την ανάπτυξη της πρώτης κινητής εφαρμογής Συστήματος Υποστήριξης Επιχειρηματολογίας. Το κεφάλαιο αυτό περιγράφει συνοπτικά το πλαίσιο δράσης της εργασίας αλλά και το αντικείμενό της.

1.1 Αντικείμενο της Διπλωματικής Εργασίας

Η ευρεία χρησιμοποίηση των Web2.0 υπηρεσιών του Παγκόσμιου Ιστού και η μεταφορά τους στο όλο και ταχύτερα αναπτυσσόμενο Mobile Web, σε συνδυασμό με τις πάντα χρήσιμες υπηρεσίες των Συστημάτων Υποστήριξης Επιχειρηματολογίας αποτελούν το αντικείμενο μελέτης και έρευνας της συγκεκριμένης Διπλωματικής Εργασίας.

Στο πλαίσιο της έρευνας μελετήθηκαν τα Συστήματα Υποστήριξης Επιχειρηματολογίας, οι υπηρεσίες και οι τεχνολογίες που ακμάζουν στο Παγκόσμιο Ιστό και η όλο και ελκυστικότερη αγορά του Mobile Web, δίνοντας ιδιαίτερη έμφαση στο κατά πόσον έχουν ωριμάσει οι συνθήκες ώστε να ευδοκιμήσει η ανάπτυξη μιας υπηρεσίας Συστημάτων Υποστήριξης Επιχειρηματολογίας στο χώρο του Mobile Web.

Στη συνέχεια και αφού εντοπίστηκε έλλειψη τέτοιων κινητών συστημάτων, προσπαθήσαμε να εκμεταλλευτούμε τα πλεονεκτήματα που δίνει ένα κινητό τηλέφωνο, δημιουργώντας μία Mobile εφαρμογή που παρέχει τη δυνατότητα σε χρήστες κινητού συσκευών, με λειτουργικό σύστημα Windows Mobile, να συμμετέχουν σε ένα συγκεκριμένο Σύστημα Υποστήριξης Επιχειρηματολογίας, το CoPe_it!. Με αυτόν τον τρόπο επιχειρήθηκε να συνεισφέρουμε στην προσπάθεια διεύρυνσης των πιθανών χρήσεων του συγκεκριμένου συστήματος αλλά και να κάνουμε την αρχή για διεύρυνση της χρήσης των Συστημάτων Υποστήριξης Επιχειρηματολογίας στο Mobile Web.

1.2 Διάρθρωση της Διπλωματικής Εργασίας

Στο επόμενο κεφάλαιο παρουσιάζεται η εξέλιξη του Παγκόσμιου Ιστού και των υπηρεσιών του, καταγράφοντας βήμα-βήμα τις αλλαγές που λαμβάνουν χώρα με την πάροδο του χρόνου. Ξεκινώντας από τη δημιουργία του Web1.0, μέχρι την εποχή της νέας γενιάς Παγκόσμιου Ιστού, με το Web2.0. Στο Web2.0 αναλύονται οι βασικές διαφορές ανάμεσα στα Web1.0 και Web2.0 μέσα από παραδείγματα τεχνολογιών, εφαρμογών αλλά και νοοτροπίας. Επίσης στη συγκεκριμένη ενότητα του Web2.0 παρουσιάζεται το τι είναι αυτό που μας οδήγησε στο Web2.0, οι βασικές αρχές της νέας αυτής γενιάς του Παγκόσμιου Ιστού και τα επίπεδα ιεραρχίας της. Στην συνέχεια γίνεται εκτενής μελέτη του Mobile Web και στο πως γίνεται η μετάβαση από τον Παγκόσμιο Ιστό των προσωπικών υπολογιστών στον Παγκόσμιο Ιστό των κινητών

συσκευών μέσω της ανάπτυξης προτύπων, βέλτιστων πρακτικών, των γλωσσών σήμανσης και της ανάπτυξης των δικτύων κινητής τηλεφωνίας.

Στο τρίτο κεφάλαιο μελετώνται τα Συστήματα Υποστήριξης Συνεργασίας μέσω της Επιχειρηματολογίας. Αρχικά αναλύονται κάποιες βασικές έννοιες όπως είναι το επιχείρημα (argument), η επιχειρηματολογία (argumentation), η επιχειρηματολογική κατασκευή γνώσης, η αιτιολόγηση (reasoning) και έπειτα ακολουθεί η ανάλυση σχετικά με τα μοντέλα διαχείρισης επιχειρημάτων. Στη συνέχεια του κεφαλαίου γίνεται αναφορά στην ανάπτυξη πολλών ηλεκτρονικών εφαρμογών-υπηρεσιών που ως στόχο έχουν την υποστήριξη της επιχειρηματολογίας με κυριότερο κοινό χαρακτηριστικό τη χρήση εργαλείων που οπτικοποιούν τη διαδικασία της επιχειρηματολογίας (παρουσιάζονται αρκετά παραδείγματα). Αφού μελετήθηκαν τα συγκεκριμένα συστήματα και τα συγκεκριμένα εργαλεία υποστήριξης τους, παρουσιάζονται αναλυτικά όσα εξ' αυτών βασίζονται στον Παγκόσμιο Ιστό (Web-based on-line tools). Κλείνοντας το κεφάλαιο επισημαίνεται το «βρεφικό» στάδιο ανάπτυξης τέτοιων υπηρεσιών, για χρήστες Mobile Web, μη έχοντας κάποια έτοιμη εφαρμογή, παρά μόνο μία παραπλήσιου ενδιαφέροντος, που σχετίζεται με on-line Mind Mapping.

Στο τέταρτο κεφάλαιο παρουσιάζονται τα στάδια σχεδιασμού και ανάπτυξης του Web Service αλλά και της εφαρμογής για κινητά, αφού πρώτα δίδεται όλο το απαραίτητο θεωρητικό υπόβαθρο για την κατανόηση των πεπραγμένων. Έτσι, αρχικά γίνεται μία εκτενής παρουσίαση του CoPe_it! και έπειτα αναφέρονται οι απαιτήσεις και οι προδιαγραφές για την παρεχόμενη λειτουργικότητα της εφαρμογής μας. Κατόπιν παρουσιάζεται ο σχεδιασμός και η ανάπτυξη που έλαβε χώρα τόσο στη μεριά του συστήματος όσο και στην υλοποίηση της εφαρμογής, ενώ στη συνέχεια παρουσιάζεται η εφαρμογή κινητού που αναπτύχθηκε (λειτουργίες και τρόπο χρήσης) και αναφέρονται ζητήματα που μπορεί να αποτελέσουν μελλοντικές προκλήσεις για περαιτέρω συνέχιση της εργασίας μας.

Τέλος, στο πέμπτο και τελευταίο κεφάλαιο, κλείνοντας το κυρίως μέρος της Διπλωματικής Εργασίας, παρουσιάζεται η σύνοψη της εργασίας και τα συμπεράσματα που εξάγονται από όλη τη διπλωματική για το Web2.0, το Mobile Web, τα Συστήματα Υποστήριξης Επιχειρηματολογίας και την θέση που πρέπει να καταλαμβάνουν οι εφαρμογές των τελευταίων στο Mobile Web.

ΚΕΦΑΛΑΙΟ 2: Η μετάβαση από το Web στο Mobile Web

Ο Παγκόσμιος Ιστός πλέον αλληλεπιδρά με τον σύγχρονο άνθρωπο σχεδόν στα πάντα, καλύπτοντας του ανάγκες ψυχαγωγίας, γνώσης-μάθησης, δημιουργίας, κοινωνικής δικτύωσης, εργασίας και οτιδήποτε άλλο μπορεί να συλλάβει ο ανθρώπινος νους. Η αλληλεπίδραση αυτή κάνει τον χρήστη του Ιστού να επιδιώκει συνεχώς αναβαθμιζόμενες υπηρεσίες αλλά και υπηρεσίες τις οποίες θα απολαμβάνει παντού ανεξαρτήτως χώρου και χρόνου.

Στο παρόν κεφάλαιο παρουσιάζεται η εξέλιξη του Παγκόσμιου Ιστού από τη πρώτη μέρα ύπαρξης του, ξεκινώντας με τα πρώτα βήματα του Web1.0 και συνεχίζοντας με την εξέλιξη του και τη μετέπειτα δημιουργία της νέας γενιάς Web, με το Web2.0. Στο Web2.0 αναλύονται οι βασικές διαφορές ανάμεσα στα Web1.0 και Web2.0 μέσα από παραδείγματα τεχνολογίας, εφαρμογών αλλά και νοοτροπίας. Επίσης στη συγκεκριμένη ενότητα του Web2.0 περιγράφεται το τι είναι αυτό που μας οδήγησε στο Web2.0; Ποιες οι βασικές αρχές του; Και ποια τα επίπεδα ιεραρχίας του. Στην συνέχεια γίνεται αναφορά στο Mobile Web, στο πως γίνεται η μετάβαση από τον Παγκόσμιο Ιστό των προσωπικών υπολογιστών στον Παγκόσμιο Ιστό των κινητών συσκευών (ανάπτυξη προτύπων, οι βέλτιστες πρακτικές, οι γλώσσες σήμανσης, ανάπτυξη δικτύων κινητής τηλεφωνίας) και τι είναι αυτό που μας οδήγησε σε αυτή τη μετάβαση.

2.1 Web1.0

Ο Παγκόσμιος Ιστός (World Wide Web, WWW, W3, Web) όπως ορίζεται από το World Wide Web Consortium (W3C), είναι «ο κόσμος των προσιτών μέσω Διαδικτύου πληροφοριών, που είναι διαθέσιμος μέσω τον ΙστοΣυσκευών, όπως τον υπολογιστή, το τηλέφωνο, τη τηλεόραση ή ακόμη και συσκευές όπως το δικτυωμένο ψυγείο….» [URL: World Wide Web Consortium].

Ο Παγκόσμιος Ιστός είναι ένας διαδικτυακός χώρος πληροφοριών στον οποίο τα στοιχεία κάθε είδους πληροφορίας (πόροι του ιστού), προσδιορίζονται από μοναδικά προσδιοριστικά αποκαλούμενα URIs και διασυνδέονται με συνδέσεις που ορίζονται μέσα σε αυτό τον χώρο.

Δεν είναι λίγες οι φορές που αρκετοί συγχέουν την έννοια του Παγκόσμιου Ιστού με αυτή του Διαδικτύου (Internet), θεωρώντας πως έχουμε να κάνουμε με την ίδια έννοια. Στην πραγματικότητα η αντίληψη αυτή είναι λανθασμένη, καθώς το Web δεν είναι το Διαδίκτυο αλλά μία από τις εφαρμογές του Διαδικτύου. Παγκόσμιος Ιστός χωρίς Διαδίκτυο δεν υφίσταται, ενώ Διαδίκτυο χωρίς Ιστό μπορεί να υπάρξει. Ωστόσο ο Παγκόσμιος Ιστός είναι αυτός που φανέρωσε την τεράστια χρησιμότητα του Διαδικτύου και εκτόξευσε στα ύψη την συχνότητα χρησιμοποίησης του. Η πιο ευδιάκριτη διαφορά ανάμεσα στις δύο έννοιες εντοπίζεται στην ύπαρξη και υλικής υπόστασης του Διαδικτύου, σε αντίθεση με τον Παγκόσμιο Ιστό που αποτελείται αποκλειστικά από πακέτα πληροφορίας.

Η τεχνολογία του Ιστού καθιστά δυνατή τη δημιουργία υπερκειμένου (Hypertext), μία διασύνδεση δηλαδή πάρα πολλών μη ιεραρχημένων στοιχείων που πρωτύτερα ήταν απομονωμένα. Τα στοιχεία αυτά δεν είναι απαραίτητα σε μορφή κειμένου καθώς μπορούν να πάρουν και άλλες μορφές, όπως εικόνας και ήχου.

Είναι πολλοί αυτοί που οραματίστηκαν την ανάπτυξη ενός διασυνδεδεμένου συστήματος από πληροφορίες και δεδομένα. Ο Vannevar Bush, άλλοτε κοσμήτορας του τμήματος Μηχανολόγων του MIT, διετέλεσε διευθυντής του Αμερικανικού Γραφείου Επιστημονικών Ερευνών και Ανάπτυξης στη διάρκεια του Β’ Παγκόσμιου Πολέμου και ήταν άνθρωπος που επέβλεψε το σχεδιασμό της πρώτης ατομικής βόμβας. Το 1945 σ’ ένα άρθρο του στην Atlantic Monthly με τίτλο: «As We May Think» [Bush, 1945] περιέγραψε μία φώτο-ηλεκτρομηχανική μηχανή που την ονόμασε Memex, η οποία θα μπορούσε με χρήση ψηφιοποίησης φωτοκύτταρων και ενός συστήματος στιγμιαίας φωτογράφισης, να δημιουργεί και να ακολουθεί παραπομπές από και προς διάφορα τμήματα εγγράφων, αποθηκευμένων σε μικροφίλμ.

Ο Ted Nelson, ένας επαγγελματίας οραματιστής, έγραψε το 1965 για «Λογοτεχνικές Μηχανές» [Nelson, 1965], υπολογιστές που θα έδιναν τη δυνατότητα στον άνθρωπο να γράφει και να δημοσιεύει κείμενα σε μία νέα, μη γραμμική μορφή, την οποία ονόμασε υπερκείμενο (hypertext). Το υπερκείμενο ήταν «μη σειριακό» κείμενο το οποίο ο αναγνώστης δεν ήταν αναγκασμένος να διαβάσει με μία ορισμένη σειρά, αλλά μπορούσε να ακολουθήσει συνδέσμους που παρέπεμπαν σε άλλα έγγραφα και από ένα σύντομο παράθεμα να ανατρέξει στο πρότυπο έγγραφο. Ο Nelson είχε συλλάβει ένα πρωτοποριακό σχέδιο, το Xanadu [Nelson, 1981], συμφωνά με το οποίο όλες οι πληροφορίες του κόσμου θα μπορούσαν να εκδοθούν σε υπερκείμενο. Οραματιζόταν ότι η κάθε αναφορά έπρεπε από ένα σύνδεσμο να παραπέμπει τον αναγνώστη στη πηγή του και ταυτόχρονα να επιτρέπει στον συγγραφέα να αποζημιώνεται με ένα μικροποσό, κάθε φορά που κάποιος διαβάζει το κείμενο του. Οραματίστηκε μια ουτοπική κοινωνία, όπου όλοι οι άνθρωποι θα μπορούσαν να επικοινωνούν σαν ίσοι προς ίσους και να μοιράζονται όλες τις πληροφορίες τους. Αγωνίστηκε χρόνια ολόκληρα για να πείσει κάποιους να χρηματοδοτήσουν το σχέδιο του, χωρίς ποτέ να το κατορθώσει.

Ο Douglas Engelbart, ερευνητής στο Πανεπιστήμιο του Stanford, παρουσίασε στη δεκαετία του '60 έναν τρόπο ηλεκτρονικής συνεργασίας με το όνομα NLS (On Line System) [Engelbart, 1962]. Ο Engelbart πίστεψε ότι το υπερκείμενο ήταν ένα εργαλείο που θα μπορούσε να βοηθήσει τους ανθρώπους να συνεργαστούν σαν μια ομάδα. Για να μπορεί να κινεί ευκολότερα τον κέρσορα του υπολογιστή του πάνω στο κείμενο και να επιλέγει πιο άκοπα τους υπερκειμενικούς συνδέσμους, ο Engelbart επινόησε μια νέα συσκευή: ένα κομμάτι ξύλο που στο κάτω μέρος του είχε αισθητήρες και μία μπίλια. Η συσκευή αυτή ήταν το πρώτο ποντίκι. Η βασική ιδέα του Engelbart ήταν ότι ο άνθρωπος και ο υπολογιστής μπορούσαν να αλληλεπιδράσουν ο ένας στον άλλον φυσικά και αβίαστα. Δυστυχώς όπως ο Bush και ο Nelson, έτσι και ο Engelbart προπορευόταν πολύ της εποχής του. Έπρεπε να περάσουν ακόμη δεκαπέντε χρόνια για να γίνει η επανάσταση των προσωπικών υπολογιστών, που θα έκανε το «ποντίκι» του Engelbart τόσο γνωστό και συνηθισμένο όσο ένα μολύβι.

Το επόμενο σημαντικό επίτευγμα στην πορεία για τη σύνδεση των πληροφοριών όλου του κόσμου ήταν ασφαλώς το Internet, μια γενική επικοινωνιακή υποδομή που ενώνει μεταξύ τους υπολογιστές. Σε αυτήν βασίζεται και ο Παγκόσμιος Ιστός. Οι καινοτομίες του Donald Davis, του Paul Barrab, του Vint Cerf, του Bob Kahn και των συνεργατών τους είχαν ήδη λάβει χώρα τη δεκαετία του 70, αλλά «τώρα» μόλις άρχισαν να αξιοποιούνται σε ευρεία κλίμακα.

Το Web με την σημερινή του μορφή πήρε σάρκα και οστά πολύ αργότερα, στα τέλη της δεκαετίας του '80. Συγκεκριμένα το 1989 στα εργαστήρια του CERN (Κέντρο Έρευνας Σωματιδιακής Φυσικής) στην Γενεύη, ο Tim Berners Lee ανέπτυξε ένα σύστημα διασυνδεδεμένων πληροφοριών το οποίο ήταν προσβάσιμο από διαφορετικά συστήματα υπολογιστών, οι οποίοι βρίσκονταν μακριά ο ένας με τον άλλον [Berners-Lee, 1992]. Τότε οι χρήστες χρησιμοποιούσαν τις τεχνολογίες Tex και PostScript για την δημιουργία περιεχομένου, ενώ πιο λίγοι ήταν αυτοί που χρησιμοποιούσαν την πολύπλοκή γλώσσα SGML (η HTML είναι υποσύνολο της SGML), τεχνολογίες οι οποίες δεν μπορούσαν να τρέξουν σε όλα τα συστήματα υπολογιστών λόγω της πολυπλοκότητας και του αποθηκευτικού όγκου τους.

Ο Tim Berners Lee τότε συνειδητοποίησε πως χρειαζόταν κάτι πιο απλό, το οποίο θα μπορούσε να τρέξει με την ίδια αποτελεσματικότητα από έναν πολύ μικρό υπολογιστή μέχρι και ένα σύνθετο γραφικό υπολογιστικό σύστημα. Έτσι αναπτύχθηκε η HTML (Hypertext Markup Language), η οποία πραγματοποιούσε με πολύ εύκολο τρόπο αυτό που ο Tim Berners είχε σκεφτεί, και το οποίο συνεργάστηκε επίσης πολύ εύκολα με το πρωτόκολλο TCP/IP που χρησιμοποιούσαν από τότε οι υπολογιστές για να επικοινωνούν μεταξύ τους.

Το 1991 το CERN ξεκίνησε να λειτουργεί στο Web μαζί με μια λίστα ηλεκτρονικού ταχυδρομείου (mailing list) με το όνομα www-talk. Πολλοί άνθρωποι, είτε για επαγγελματικούς σκοπούς, είτε για την ερασιτεχνική ενασχόλησή τους με το αντικείμενο, βοήθησαν να εξελιχθεί το Web φτιάχνοντας ιστοσελίδες και σχεδιάζοντας Browsers όπως το Cello, Viola και MidasWWW. Ο διασημότερος όμως Browser της εποχής εκείνης ήταν ο X Window Mosaic Browser ο οποίος σχεδιάστηκε από την εταιρία National Center for Supercomputer Application (NCSA).

Το World Wide Web, από το 1991 άρχισε να μεγαλώνει με ταχύτατους ρυθμούς εξαφανίζοντας άλλα συστήματα διακίνησης πληροφοριών βασισμένα στην τεχνολογία του Internet, όπως το WAIS, το Hytelnet, το Gopher και το UseNet.

2.1.1 Τα βασικά μέρη του Παγκόσμιου Ιστού

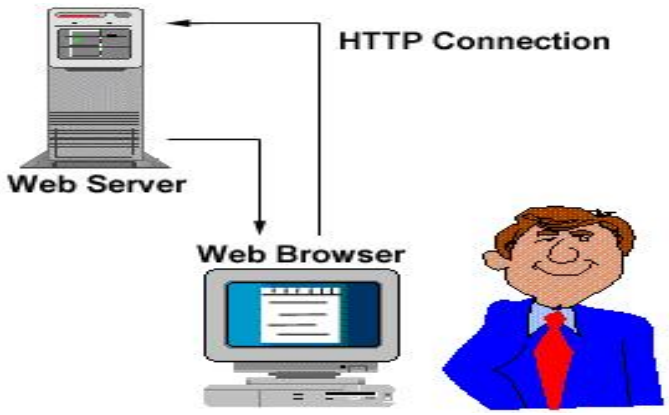
Οι βασικές συνιστώσες του Παγκόσμιου Ιστού, όπως φαίνονται και στην Εικόνα 2.1, είναι οι διακομιστές (web servers), οι χρήστες του, το πρωτόκολλο HTTP και τα προγράμματα ανάγνωσης ιστοσελίδων (web browsers). Αναλυτικότερα:

- **Οι Web Servers** είναι υπολογιστές που αποθηκεύουν και διακινούν πληροφορίες, δεδομένα, και έγγραφα χρησιμοποιώντας το Internet. Τα έγγραφα αυτά είναι στη πλειοψηφία τους ιστοσελίδες, αλλά μπορεί να είναι και άλλα αρχεία όπως π.χ. λογιστικά φύλλα, παρουσιάσεις, αρχεία κειμένου, συμπιεσμένα αρχεία γενικώς.

- **Οι χρήστες του Web** είναι υπολογιστές συνδεδεμένοι στο Internet που μπορούν να κατεβάσουν δεδομένα από τους Web Servers.

- **Το πρωτόκολλο HTTP** (Hypertext Transfer Protocol) που αποκαλείται και ως πρωτόκολλο του Web, είναι ένα σύνολο από κανόνες που ελέγχουν και καθορίζουν την διακίνηση των ιστοσελίδων από τους Web Servers στους υπολογιστές των χρηστών. Όταν πληκτρολογούμε την ηλεκτρονική διεύθυνση μίας ιστοσελίδας, ξεκινάμε πάντα με http, έτσι καθορίζουμε τον τρόπο με τον οποίο θα πραγματοποιηθεί η συνδιαλλαγή μεταξύ του υπολογιστή μας και του Server στον οποίο φυλάσσεται η ιστοσελίδα που σκοπεύουμε να δούμε.

- **Τα προγράμματα ανάγνωσης ιστοσελίδων** (Browsers), είναι λογισμικό που χρησιμοποιούν οι χρήστες για να παρουσιάσουν κείμενο, γραφικά, ήχο, multimedia, που έχουν κατεβάσει από το Internet. Οι πιο δημοφιλείς browsers είναι ο Microsoft Internet Explorer, ο Mozilla Firefox, ο Safari, ο Opera και ο Google Chrome.



Εικόνα 2.1: *Τα βασικά μέρη του Παγκόσμιου Ιστού. [Πηγή: http://www.extropia.com]*

Το World Wide Web ως πληροφοριακό σύστημα παρέχει συγκεκριμένο μοντέλο δεδομένων το οποίο βασίζεται σε κόμβους και υπερσυνδέσμους. Το μοντέλο δεδομένων του παγκόσμιου ιστού βασίζεται σε κόμβους (nodes), άγκυρες (anchors) και συνδέσμους ή υπερσυνδέσμους (links ή hyperlinks). Οι κόμβοι είναι φορείς του περιεχομένου, οι άγκυρες είναι περιοχές του

περιεχομένου των κόμβων από όπου ξεκινούν ή καταλήγουν σύνδεσμοι και σύνδεσμοι συνδέουν δύο κόμβους.

2.2 Web2.0

Ο όρος Web2.0 (Ιστός 2.0), χρησιμοποιείται για την περιγραφή της υποτιθέμενης ή προτεινόμενης νέας γενιάς του Παγκόσμιου Ιστού η οποία βασίζεται στην όλο και μεγαλύτερη δυνατότητα των χρηστών του Διαδικτύου να μοιράζονται πληροφορίες και να συνεργάζονται σε απευθείας σύνδεση (on-line). Αυτή η νέα γενιά αποτελεί μια δυναμική διαδικτυακή πλατφόρμα στην οποία μπορούν να αλληλεπιδρούν χρήστες χωρίς να χρειάζεται να έχουν εξειδικευμένες γνώσεις σε θέματα υπολογιστών και δικτύων.

Η φράση Web2.0 ειπώθηκε για πρώτη φορά το 2004 κατά τη διάρκεια ενός συνεδρίου μεταξύ της O'Reilly Media και της MediaLive International όπου προτείνονταν ιδέες για την αναβάθμιση του Παγκόσμιου Ιστού [O'Reilly, 2005]. Ο Dale Dougherty (αντιπρόεδρος της O'Reilly Media), παρατήρησε ότι το Διαδίκτυο είχε αρχίσει να γίνεται πολύ δημοφιλές και σημαντικό μέρος της καθημερινότητας όλο και μεγαλύτερου ποσοστού ανθρώπων. Συνεχώς έβγαιναν νέες υπηρεσίες, εφαρμογές και ιστοσελίδες οι οποίες αναγνωρίζονταν από το ευρύ κοινό σε σύντομο χρονικό διάστημα. Επιπλέον όλο και μεγαλύτερος αριθμός εταιριών άρχισαν να στρέφουν την επιχειρηματική τους δράση στο Διαδίκτυο και να προσπαθούν να φέρουν τους καταναλωτές τους προς αυτή τη κατεύθυνση. Παραδείγματα τέτοιων υπηρεσιών αποτελούν οι δικτυακοί τόποι κοινωνικής δικτύωσης (social networking sites), τα wikis, τα εργαλεία επικοινωνίας και οι Folksonomies (μέθοδος που χρησιμοποιείται για τη δημιουργία αλλά και την διαχείριση ετικετών έτσι ώστε να σχολιαστεί και να ταξινομηθεί ένα περιεχόμενο). Αν και ο παραπάνω όρος προτείνει μια νέα έκδοση του Παγκόσμιου Ιστού (World Wide Web), δεν αναφέρεται στον εκσυγχρονισμό τεχνικών προδιαγραφών, αλλά σε αλλαγές στον τρόπο που οι προγραμματιστές λογισμικού και οι τελικοί χρήστες χρησιμοποιούν το Διαδίκτυο.



Εικόνα 2.2*: Ο χάρτης του Web2.0. [Πηγή:* http://www.stonesoft.com*]*

Δύο σημαντικοί συγγραφείς-ερευνητές δίνουν τους δικούς τους ορισμούς για το Web2.0 [Ahmed, 2007]: «Το Web2.0 είναι περισσότερο μία ιδέα στο μυαλό των ανθρώπων παρά μία πραγματικότητα. Είναι μία ιδέα στην οποία δίνεται έμφαση η αμοιβαιότητα μεταξύ του χρήστη και αυτού που προσφέρει υπηρεσίες. Με άλλα λόγια, εννοούμε τη γνήσια αλληλεπίδραση, απλά επειδή δίνεται η δυνατότητα στους χρήστες τόσο να ανεβάζουν πληροφορίες στο Διαδίκτυο όσο και να κατεβάζουν.» [Stephen Fry]. Σύμφωνα με τον παραπάνω ορισμό η ιδέα του Web2.0 περιλαμβάνει την παραγωγή και τη διάδοση περιεχομένου από τους χρήστες δίνοντας τους τη δυνατότητα και την ελευθερία να το μοιραστούν και να το ξαναχρησιμοποιήσουν. Αυτό έχει σαν αποτέλεσμα μία αύξηση στην οικονομική αξία του Διαδικτύου καθώς οι χρήστες μπορούν να κάνουν περισσότερα πράγματα on-line.

«Το Web2.0 είναι μία επιχειρησιακή επανάσταση στη βιομηχανία των ηλεκτρονικών υπολογιστών που προκλήθηκε από την ιδέα της θεώρησης του Διαδικτύου σαν πλατφόρμα και την προσπάθεια για κατανόηση των κανόνων για επιτυχία σε αυτή τη νέα πλατφόρμα.» [Tim O'Reilly]. Κατά τον Tim O'Reilly, δηλαδή, το Web2.0 θεωρείται σαν μία επιχείρηση που δέχεται το Διαδίκτυο σαν πλατφόρμα και χρησιμοποιεί τις δυνατότητές του (όπως για παράδειγμα το παγκόσμιο κοινό, δηλαδή την υπάρχουσα ομάδα των χρηστών του Web). Ο παραπάνω ορισμός αποτελεί και τον τελευταίο συγκεντρωτικό ορισμό του Web2.0 καθώς το ακριβές νόημά του παραμένει ανοιχτό προς αντιπαράθεση. Μερικοί ειδικοί, συμπεριλαμβανομένου και του Tim Berners Lee, έχουν αμφισβητήσει το γεγονός ότι ο όρος έχει κάποιο πραγματικό νόημα.

2.2.1 Οι βασικές αρχές του Web2.0

Τον Οκτώβριο του 2004 κατά τη διάρκεια του περιβόητου συνεδρίου που πρωτοαναφέρθηκε ο όρος Web2.0, μεταξύ της O'Reilly Media και της MediaLive International, ο Tim O'Reilly και ο John Battelle κατέληξαν σε κάποια βασικά συμπεράσματα. [O'Reilly, 2005] Αυτά τα συμπεράσματα για πολλούς αποτελούν τις βασικές αρχές του Web2.0. Τα συμπεράσματα τους είναι τα παρακάτω:

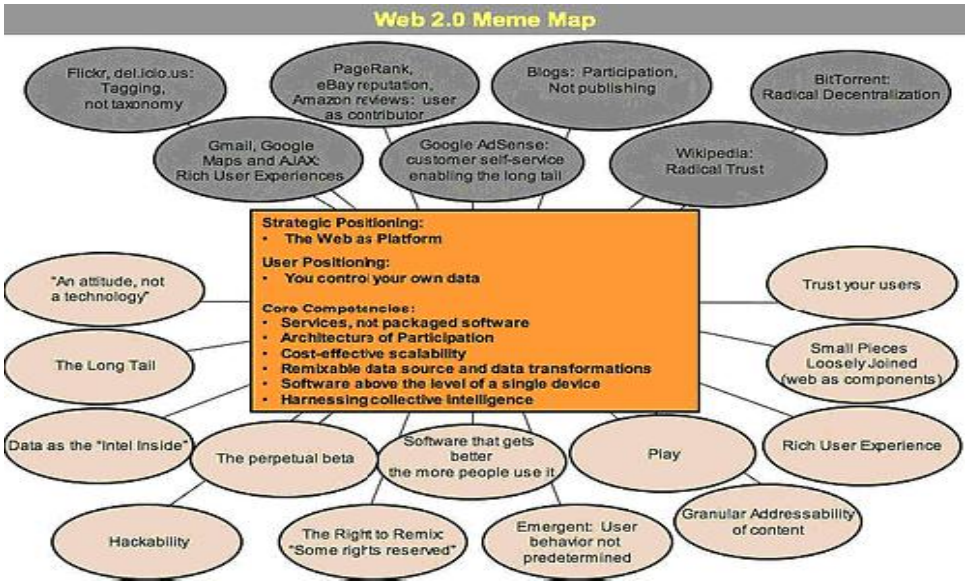
- Χρήση του Web ως πλατφόρμα
- Αξιοποίηση της συλλογική νοημοσύνης (συμμετοχή και αλληλεπίδραση των χρηστών)
- Διαχείριση των βάσεων δεδομένων
- Τέλος στον κύκλο εκδόσεων λογισμικού (τέλος στο «Perpetual Beta»)
- «Ελαφριά» προγραμματιστικά μοντέλα
- Λογισμικό πάνω από το επίπεδο της μιας και μόνο συσκευής
- Πλούσιες εμπειρίες χρήστη

Αναλυτικότερα τα συμπεράσματα αυτά έχουν ως εξής:

Το Web ως διαδικτυακή πλατφόρμα

Το Διαδίκτυο στην νέα του μορφή δεν αποτελείται πλέον από μεμονωμένες εφαρμογές και υπηρεσίες, αλλά αποτελεί μία ενιαία πλατφόρμα, ένα σύστημα χωρίς ιδιοκτήτη, που παραμένει συνδεδεμένο χάρη σε ένα σύνολο πρωτοκόλλων και προτύπων και στηρίζεται σε μία συνολική διάθεση για συνένωση και συνεργασία.

Η Εικόνα 2.3 παρουσιάζει ένα μνημονικό χάρτη του Web2.0 που είχε αναπτυχθεί σε μία σύσκεψη ανταλλαγής ιδεών κατά την διάρκεια του FOO Camp, ενός συνεδρίου στο O'Reilly Media.



Εικόνα 2.3: *Μνημονικός χάρτης του Web2.0. [Πηγή: http://oreilly.com]*

Δημιουργός του σλόγκαν «Το Διαδίκτυο σαν πλατφόρμα» είναι ο Dale Dougherty. Ο Dougherty ανέφερε συγκεκριμένα, στα πλαίσια ενός συνεδρίου για το μέλλον του Web: «Το Web2.0 καθιστά το Διαδίκτυο ως μια πλατφόρμα, που εκτείνεται σε όλες τις συσκευές. Οι εφαρμογές Web2.0 είναι εκείνες οι οποίες δημιουργούν τα ουσιώδη πλεονεκτήματα αυτής της πλατφόρμας, αποτελώντας ένα συνεχώς αναβαθμιζόμενο λογισμικό, το οποίο γίνεται τόσο καλύτερο όσο περισσότεροι άνθρωποι το χρησιμοποιούν. Η χρήση και η αναδιάταξη δεδομένων, από μεμονωμένους χρήστες, από πολλαπλές πηγές και η παροχή των δικών τους δεδομένων και υπηρεσιών σε μια μορφή που επιτρέπει την αναδιάταξη αυτών από τρίτους, δημιουργεί μια «αρχιτεκτονική της συμμετοχής» η οποία επιφέρει δικτυακές επιρροές, αναπτύσσοντας περαιτέρω την πρώτη γενιά του Παγκόσμιου Ιστού (Web1.0) προσδίδοντας πλούσιες εμπειρίες στον χρήστη».

Συμπερασματικά, λοιπόν, θα λέγαμε πως ο όρος Web2.0 εκφράζει την αλλαγή στον τρόπο με τον οποίο χρησιμοποιείται πλέον το Διαδίκτυο. Δεν αποτελεί κάποιο νέο τεχνολογικό πρότυπο, αλλά αναφέρεται στη χρήση του Διαδικτύου ως μια πλατφόρμα εφαρμογών, όπου η συμμετοχή των χρηστών στην ανάπτυξη και διαχείριση του περιεχομένου, αλλάζει τη φύση και την αξία της πληροφορίας. Πιο συγκεκριμένα, οι λέξεις Blogspot/Wordpress (Ιστολόγια - Blogs) [URL: Blogspot], Facebook [URL: Facebook], Twitter [URL: Twitter], MySpace [URL: MySpace], Wikipedia [URL: Wikipedia], Del.icio.us [URL: Del.icio.us] και Flickr [URL: Flickr] υπάρχουν στην καθημερινότητά μας και αποτελούν εφαρμογές στις οποίες κάθε χρήστης έχει και ένα

προφίλ, αυτό αντιστοιχεί σε μια προσωπική σελίδα γι' αυτόν, το περιεχόμενο της οποίας το καθορίζει ο ίδιος σαν να ήταν ο δημιουργός της. Με αυτόν το τρόπο (συνδέοντας την θεωρία με την πράξη) το Διαδίκτυο είναι μια πλατφόρμα πάνω στην οποία στηρίζονται κάποιες εφαρμογές σαν τις προαναφερθείσες. Ο Παγκόσμιος Ιστός αποκτά έναν συλλογικό - κοινωνικό (Social) χαρακτήρα και ο χρήστης μετατρέπεται από αναγνώστης σε δημιουργό του δικού του προσωπικού περιεχομένου.

Αξιοποίηση της συλλογική νοημοσύνης

Η βασική αρχή που βρίσκεται πίσω από την επιτυχία των επικρατέστερων ιστότοπων του Διαδικτύου είναι ότι χρησιμοποίησαν και εκμεταλλεύτηκαν τη συλλογική νοημοσύνη των χρηστών του Διαδικτύου. Καινοτόμες εφαρμογές/υπηρεσίες που λαμβάνουν υπόψη αυτή την αρχή είναι αυτές που κυριαρχούν στο Διαδίκτυο:

- Η Wikipedia είναι μια on-line εγκυκλοπαίδεια που βασίζεται στο γεγονός ότι κάθε χρήστης του Διαδικτύου μπορεί να προσθέσει οποιαδήποτε εγγραφή, η οποία μπορεί να τροποποιηθεί από κάποιον άλλον. Δηλαδή, το κείμενο και το περιεχόμενο της εγκυκλοπαίδειας δημιουργούνται δυναμικά. Η Wikipedia είναι μεταξύ των 100 κορυφαίων ιστοσελίδων και πολλοί πιστεύουν ότι θα φτάσει μεταξύ των 10 πρώτων.

- Ιστότοποι όπως το Flickr πρωτοπορούν χρησιμοποιώντας την ιδέα που κάποιοι άλλοι αποκαλούν «Folskonomy» (σε αντίθεση με την «Taxonomy»). Δηλαδή, μια μέθοδο αθροιστικής κατηγοριοποίησης των ιστότοπων που χρησιμοποιεί ελεύθερα επιλεγμένες λέξεις-κλειδιά που συχνά ονομάζονται tags (ετικέτες). Η χρήση των ετικετών επιτρέπει την ύπαρξη πολλαπλών και επικαλυπτόμενων συνδέσεων που το ίδιο το μυαλό χρησιμοποιεί, παρά την ύπαρξη αυστηρών κατηγοριών. Π.χ. μια φωτογραφία στο Flickr από ένα λίμνη μπορεί να έχει ως ετικέτα είτε «λίμνη» είτε «γαλάζια», επιτρέποντας την ανάκτηση της από διαφορετικά ερωτήματα που γεννιούνται από τη δραστηριότητα του χρήστη.

Η χρήση υπερσυνδέσμων είναι το βασικό θεμέλιο του Διαδικτύου. Όπως η νευρική σύναψη στον εγκέφαλο, όπου οι συνειρμοί γίνονται πιο έντονοι μέσω της επανάληψης, έτσι και οι συνδέσεις του Διαδικτύου αυξάνονται συστηματικά ως αποτέλεσμα της αθροιστικής δραστηριότητας από όλους τους χρήστες του Διαδικτύου.

Διαχείριση των βάσεων δεδομένων

Κάθε σημαντική διαδικτυακή εφαρμογή έχει υποστηριχθεί από μια εξειδικευμένη βάση δεδομένων (π.χ. οι βάσεις δεδομένων των Google, eBay, Yahoo, MapQuest, Amazon). Η διαχείριση μιας βάσης δεδομένων είναι μια βασική ικανότητα των Web2.0 εφαρμογών, με αποτέλεσμα πολλές φορές τέτοιου είδους εφαρμογές να ονομάζονται infoware αντί για software. Στο χώρο του Διαδικτύου μπορούμε να συναντήσουμε πολλές περιπτώσεις στις οποίες ο έλεγχος της βάσης δεδομένων έχει οδηγήσει σε ασυνήθιστα μεγάλα κέρδη και στο γενικότερο έλεγχο της αγοράς. Με την πάροδο του χρόνου το να αποκομίσεις μια επιχειρηματική υπεροχή στο Διαδίκτυο γίνεται πιο δύσκολα μέσω του ελέγχου των εφαρμογών λογισμικού παρά με τον έλεγχο των πηγών δεδομένων.

Μια από τις πιο πρόσφατες πετυχημένες εφαρμογές είναι η δικτυακή χαρτογράφηση. Ο έντονος ανταγωνισμός που υπάρχει σε αυτό τον τομέα δηλώνει πως η αδυναμία μιας εταιρείας να καταλάβει τη σημασία της ιδιοκτησίας των δεδομένων της εφαρμογής σίγουρα θα υπονομεύσει την ανταγωνιστική της θέση.

Το MapQuest [URL: MapQuest] πρωτοπόρησε στην κατηγορία της χαρτογράφησης το 1995. Ωστόσο όταν το Yahoo και η Microsoft, και πιο πρόσφατα το Google, αποφάσισαν να εισέλθουν στην αγορά, ήταν εύκολο για αυτούς να προσφέρουν μια ανταγωνιστική εφαρμογή δίνοντας άδεια χρήσης των ίδιων δεδομένων.

Σε αντίθεση με το MapQuest, το Amazon [URL: Amazon] εμπλούτισε «αμείλικτα» τα δεδομένα, προσθέτοντας πίνακες περιεχομένων, ευρετήρια, εικόνες και δειγματικό υλικό. Ακόμη πιο σημαντική ήταν η εξουσιοδότηση των χρηστών να σχολιάζουν τα δεδομένα, με αποτέλεσμα μετά από 10 χρόνια ο Amazon να αποτελεί την πιο αξιόλογη πηγή αναφοράς για τα βιβλιογραφικά δεδομένα για όλους τους ενδιαφερόμενους καταναλωτές.

Αν το MapQuest είχε επίσης επιτρέψει στους χρήστες τον σχολιασμό των χαρτών και των κατευθύνσεων τότε θα έδινε μεγαλύτερη αξία στην εφαρμογή, δυσκολεύοντας περισσότερο την είσοδο των ανταγωνιστών στην αγορά απλά και μόνο ελευθερώνοντας τη βάση δεδομένων.
Το πρόσφατο λανσάρισμα του Google Maps [URL: Google Maps] έχει προκαλέσει ανταγωνισμό μεταξύ των πωλητών εφαρμογών και των προμηθευτών δεδομένων. Ο «ελαφρύς» προγραμματισμός του Google έχει οδηγήσει στη δημιουργία ενός μεγάλου αριθμού αξιόλογων υπηρεσιών με τη μορφή των mashups που συνδέουν το Google Maps με άλλες διαδικτυακά-

προσπελάσιμες πηγές δεδομένων. Εξέχον παράδειγμα τέτοιου mashup αποτελεί αυτό που συνδυάζει το Google Maps με δεδομένα αγοράς σπιτιών και ενοικίασης διαμερισμάτων, με αποτέλεσμα τη δημιουργία ενός πολύτιμου και αλληλεπιδραστικού εργαλείου αναζήτησης κατοικίας.

Είναι φανερό ότι η νέα γενιά εφαρμογών θα ενεργοποιηθεί από τα πρότυπα και τις λύσεις σε αυτούς τους τομείς, που μετατρέπουν αποδοτικά προκαθορισμένες κατηγορίες δεδομένων σε αξιόπιστα υποσυστήματα.

Ένα στοιχείο που πρέπει να σημειωθεί είναι η ανησυχία και το ενδιαφέρον των χρηστών σχετικά με τη μυστικότητα και τα δικαιώματα που κατέχουν σε σχέση με τα προσωπικά τους δεδομένα. Στις πρώτες δικτυακές εφαρμογές τα δικαιώματα των δημιουργών (copyright) είχαν μόνο αόριστα επιβληθεί. Ωστόσο, όσο οι εταιρείες αρχίζουν να συνειδητοποιούν ότι ο έλεγχος των δεδομένων μπορεί να αποτελέσει την κύρια πηγή ανταγωνιστικής υπεροχής, πιθανό να δούμε να πραγματοποιούνται αυξημένες προσπάθειες σχετικά με τον έλεγχο.

Ωστόσο, αν και η άνοδος του ιδιόκτητου λογισμικού οδήγησε στην τάση του «Ανοιχτού Λογισμικού», είναι αναμενόμενο ότι η άνθηση των ιδιόκτητων βάσεων δεδομένων θα καταλήξει σε μια τάση «Ελεύθερων Δεδομένων» μέσα στην επόμενη δεκαετία. Είναι ήδη ορατά τα σημάδια μιας εξισορροπητικής τάσης στα έργα «ανοιχτών δεδομένων» (open data projects) όπως η Wikipedia, τα Creative Commons [URL: Creative Commons] και στα έργα λογισμικού (software projects) όπως το Greasemonkey, που επιτρέπουν στους χρήστες να έχουν τον έλεγχο σχετικά με την απεικόνιση των δεδομένων στον προσωπικό τους υπολογιστή.

Τέλος στον κύκλο εκδόσεων λογισμικού

Όταν O'Reilly ρωτήθηκε για το «Perpetual Beta», προσπάθησε να πει ότι ο κλασικός κύκλος έκδοσης λογισμικού γίνεται πάρα πολύ αργά και είναι στραμμένος στις σύγχρονες επιχειρήσεις, και ότι οι επιχειρήσεις δεν πρέπει ποτέ να σταματήσουν να ενισχύουν τις εφαρμογές και τις υπηρεσίες τους. Αυτό συνδέεται με την ιδέα ότι το λογισμικό δεν είναι ένα προϊόν, αλλά υπηρεσία και κανείς δε θέλει μια υπηρεσία που γίνεται ξεπερασμένη σε λίγους μήνες. Οι επιχειρήσεις πρέπει να βελτιώνουν τις υπηρεσίες τους συνεχώς και αυτό σημαίνει δημιουργία

νέων λειτουργιών κάθε λίγες εβδομάδες. Αυτό είναι που δίνει τα μεγάλα ανταγωνιστικά πλεονεκτήματα στην εφαρμογή ιστού:

- Οι χρήστες δεν είναι απαραίτητο να κατεβάσουν τη νέα έκδοση για να πάρουν τη νέα λειτουργία.
- Οι επιχειρήσεις δεν είναι απαραίτητο να διατηρήσουν την παλιά συμβατότητα (κανένας δεν θα τρέξει ποτέ μια ξεπερασμένη έκδοση).
- Οι χρήστες παίρνουν μέρος στην ανάπτυξη, και μπορούν να προτείνουν το νέο χαρακτηριστικό γνώρισμα που προστίθεται.
- Οι επιχειρήσεις μπορούν να δοκιμάσουν τα νέα χαρακτηριστικά γνωρίσματα καθημερινά και να επιλέξουν πόσο να επενδύσουν σε κάθε ένα από αυτά.
- Οι χρήστες μπορούν να έχουν πρόσβαση στην υπηρεσία από οπουδήποτε.
- Οι επιχειρήσεις μπορούν να παραδώσουν μια διαθέσιμη υπηρεσία οπουδήποτε.

Όταν οι συσκευές και τα προγράμματα συνδέονται με το Διαδίκτυο, οι εφαρμογές δεν είναι πλέον χειροποίητα αντικείμενα λογισμικού, αλλά τρέχουσες υπηρεσίες. Επομένως, είναι καλό να προσθέσουμε χαρακτηριστικά σε μία κανονική βάση γνωρίζοντας την εμπειρία του χρήστη και έπειτα να δεσμεύουμε τους χρήστες σε πραγματικό χρόνο σαν ελεγκτές, και να οργανώσουμε την υπηρεσία έτσι ώστε να ξέρουμε πώς αυτοί οι άνθρωποι χρησιμοποιούν τα νέα χαρακτηριστικά γνωρίσματα.

«Ελαφριά» προγραμματιστικά μοντέλα

Από τη στιγμή που οι υπηρεσίες Διαδικτύου έγιναν η δεσπόζουσα τάση, οι μεγάλες εταιρείες έδωσαν μάχη με μια σωρό από πολύπλοκες διαδικτυακές υπηρεσίες με σκοπό να κατασκευάσουν υψηλά αξιόπιστα προγραμματιστικά περιβάλλοντα για κατανεμημένες εφαρμογές. Το Διαδίκτυο, λοιπόν, άρχισε να πετυχαίνει ανατρέποντας τη θεωρία του υπερκειμένου και αντικαθιστώντας την με μια απλή πρακτικότητα για ένα ιδανικό σχεδιασμό. Όμοια, η τεχνική RSS (Really Simple Syndication) αποτελεί την πιο ευρέως διαδεδομένη και πλήρως ανεπτυγμένη υπηρεσία Διαδικτύου εξαιτίας της απλότητάς της.

Η ίδια αναζήτηση για απλότητα είναι εμφανής και σε άλλες συστηματοποιημένες διαδικτυακές υπηρεσίες. Χαρακτηριστικό παράδειγμα η αποδέσμευση του Google Maps που πραγματοποιήθηκε πρόσφατα από το Google. Η AJAX (JavaScript και XML) [Garrett, 2005] διεπαφή του γρήγορα αποκρυπτογραφήθηκε από τους Hackers οι οποίοι στη συνέχεια χρησιμοποίησαν τα δεδομένα σε νέες υπηρεσίες.

Οι πιο πολλές υπηρεσίες Διαδικτύου που χρηματοδοτούνται από εταιρείες παρουσιάζουν μια πολυπλοκότητα με σκοπό τη δημιουργία μιας ισχυρής και σφιχτής ένωσης. Αυτό μπορεί να είναι αναγκαίο σε κάποιες περιπτώσεις αλλά από την άλλη μπορούν να διατηρηθούν κάποιες εφαρμογές περισσότερο ελεύθερες, ακόμα και πιο ευπαθείς, όπως το Web2.0. υποστηρίζει.

Οι απλές διαδικτυακές υπηρεσίες, όπως η RSS και όσες στηρίζονται στο REST πρωτόκολλο [Fielding, 2000], χρησιμοποιούν τη syndication στα δεδομένα (ένα μερίδιο του ιστότοπου διατίθεται σε άλλους ιστότοπους ή συνδρομητές) παρά το να ελέγχουν τι ακριβώς συμβαίνει στο άλλο άκρο της σύνδεσης. Αυτή η ιδέα είναι θεμελιώδης στο Διαδίκτυο και είναι γνωστή ως end-to-end αρχή.

Συστήματα όπως το αρχικό δίκτυο καθώς και οι τεχνολογίες RSS και AJAX έχουν ένα κοινό: η φραγή σε σχέση με την επαναχρησιμοποίηση είναι πολύ χαμηλή. Το μεγαλύτερο μέρους του χρήσιμου λογισμικού είναι ανοιχτού κώδικα. Η επιλογή «Προβολή Κώδικα» που υπάρχει σε ένα φυλλομετρητή του Διαδικτύου έχει κάνει δυνατό σε οποιοδήποτε χρήστη να αντιγράψει τον κώδικα της ιστοσελίδας οπουδήποτε άλλου χρήστη. Η RSS σχεδιάστηκε για να εξουσιοδοτήσει το χρήστη να βλέπει το περιεχόμενο που αυτός θέλει, όποτε αυτό είναι επιθυμητό, χωρίς να χρειάζεται η άδεια από τον παροχέα των αντίστοιχων πληροφοριών. Οι πιο πετυχημένες υπηρεσίες Διαδικτύου είναι εκείνες που οδηγούν σε νέες κατευθύνσεις που ούτε και οι δημιουργοί τους είχαν φανταστεί.

Στο μεγαλύτερο δικτυακό βιβλιοπωλείο (κατά πολλούς), το Amazon.com η βάση δεδομένων του βρίσκεται, από το 2002 και ύστερα, στα χέρια χιλιάδων χρηστών, χωρίς να ελέγχεται τι ακριβώς αυτοί βλέπουν και πώς τη χειρίζονται. Και αυτό δεν έγινε με κακόβουλο τρόπο αφού η ίδια η εταιρεία το επέτρεψε. Το αποτέλεσμα είναι εκατομμύρια χρήστες να βρουν την πληροφορία που χρειάζονται από τον συγκεκριμένο ιστότοπο και να προβαίνουν σε αγορές μέσω του καταστήματος αυτής της εταιρείας.

Οι μεγάλες εταιρείες εγκαταλείπουν κάτι ακριβό αλλά κερδίζουν σε κάποιο άλλο τομέα. Για παράδειγμα, η Wikipedia έχει χάσει τον κεντρικό έλεγχο της αρθογραφίας με αντάλλαγμα την ταχύτητα και την ευρύτητα. Η Napster εγκατέλειψε την ιδέα του «καταλόγου» (όλα τα τραγούδια που ένας πωλητής προσφέρει) και κέρδισε σε ευρύτητα. Η Amazon εγκατέλειψε την ιδέα ενός φυσικού κτιρίου στο κέντρο μιας πόλης με αποτέλεσμα να εξυπηρετεί ολόκληρο τον κόσμο. Το Google αμέλησε αρχικά τους μεγάλους πελάτες και πέτυχε να κερδίσει το 80% αυτών των οποίων οι ανάγκες δεν είχαν ακόμη καλυφθεί.

Λογισμικό πάνω από το επίπεδο της μιας και μόνο συσκευής

Ένα από τα χαρακτηριστικά του Web2.0 που αξίζει να σημειωθεί είναι το γεγονός ότι δεν περιορίζεται πια στην πλατφόρμα ενός προσωπικού υπολογιστή. Βέβαια, οποιαδήποτε διαδικτυακή εφαρμογή μπορεί να θεωρηθεί ως λογισμικό που ξεπερνά το επίπεδο μιας απλής συσκευής. Άλλωστε ακόμα και η πιο απλή εφαρμογή περιλαμβάνει τουλάχιστον δύο υπολογιστές, αυτόν που φιλοξενεί τον εξυπηρετητή δικτύου (web server) και αυτόν που φιλοξενεί τον περιηγητή (browser). Η ανάπτυξη του Διαδικτύου ως πλατφόρμα επεκτείνει αυτή την ιδέα σε σύνθετες εφαρμογές που απαρτίζονται από υπηρεσίες που παρέχονται από διάφορους υπολογιστές.

Το iTunes [URL: iTunes] που προσφέρει αναπαραγωγή και οργάνωση ψηφιακών αρχείων ήχου και εικόνας, είναι το καλύτερο παράδειγμα που τηρεί την παραπάνω αρχή. Αποτελεί ένα δικτυακό κατάστημα που βασίζεται στην τεχνολογία RSS για την προσφορά των υπηρεσιών του. Αυτή η εφαρμογή εκτείνεται από μια συσκευή σε ένα μαζικό διαδικτυακό back-end, με τον προσωπικό υπολογιστή να παίζει το ρόλο της τοπικής κρυφής μνήμης και του σταθμού ελέγχου. Έχουν πραγματοποιηθεί πολλές απόπειρες στο παρελθόν ώστε να μεταφερθεί το δικτυακό περιεχόμενο σε φορητές συσκευές, αλλά ο iPod/iTunes συνδυασμός είναι μια από τις πρώτες τέτοιου είδους εφαρμογές που σχεδιάστηκαν ώστε να συνδέσουν διάφορες συσκευές. Το TiVo [URL: TiVo] είναι ένα ακόμη αντιπροσωπευτικό παράδειγμα.

Το iTunes και το TiVo επιδεικνύουν και άλλες επιπλέον αρχές του Web2.0:

- Ουσιαστικά δεν αποτελούν καθεαυτού δικτυακές εφαρμογές, αλλά αυξάνουν την ισχύ της δικτυακής πλατφόρμας, χρησιμοποιώντας την ως ένα αόρατο τμήμα της υποδομής τους.

- Η ουσιαστική προσφορά τους έχει να κάνει κυρίως με τη διαχείριση δεδομένων
- Δεν είναι πακεταρισμένες εφαρμογές αλλά υπηρεσίες.
- Τέλος, δείχνουν μια αναπτυσσόμενη χρήση της συλλογικής νοημοσύνης

Πλούσιες εμπειρίες χρήστη

Το Διαδίκτυο χρησιμοποιήθηκε για να παρέχει «applets» (εφαρμογές Διαδικτύου) και άλλου είδους ενεργό περιεχόμενο μέσα από έναν φυλλομετρητή. Η εισαγωγή της Java [URL: Java] το 1995 δομήθηκε γύρω από την παροχή τέτοιων applets. Η Javascript και έπειτα η DHTML σχεδιάστηκαν ως εργαλεία «ελαφρού» προγραμματισμού με σκοπό την υποστήριξη ικανότητας προγραμματισμού από την πλευρά του πελάτη και τις πιο πλούσιες εμπειρίες του χρήστη. Πριν κάποια χρόνια, η Macromedia [URL: Macromedia] επινόησε τον όρο «Rich Internet Applications» ώστε να δώσει έμφαση στις ικανότητες του Flash ως προς την παροχή του όχι απλά πολυμεσικού περιεχομένου αλλά και εμπειρίες εφαρμογών με GUI-στυλ (Graphical User Interface - Γραφική Διεπαφή Χρήστη).

Οι RIA τεχνικές (Rich Internet Application - Εμπλουτισμένες Εφαρμογές Διαδικτύου), όπως η AJAX, έχουν εξελιχθεί ώστε να μπορούν βελτιώνουν την εμπειρία του χρήστη σε εφαρμογές Διαδικτύου που βασίζονται στο φυλλομετρητή. Η AJAX περιλαμβάνει μια ιστοσελίδα που απαιτεί ενημέρωση ενός τμήματος του περιεχομένου της, και μεταβάλλοντας αυτό το τμήμα στην έξοδο του φυλλομετρητή, χωρίς να χρειάζεται ανανέωση από το χρήστη ολόκληρης της σελίδας. Επιπλέον, υπάρχουν ιδιόκτητες εφαρμογές (όπως το Google Maps) και ανοιχτές φόρμες που μπορούν να χρησιμοποιούν τις APIs των υπηρεσιών Διαδικτύου, τα syndication feeds και τα screen scraping.

Παρόλα αυτά, η δυνατότητα του δικτύου να προσφέρει ολόπλευρες εφαρμογές δεν κατέρριψε την δεσπόζουσα τάση μέχρι που έκανε την εμφάνιση του το Gmail της Google, που γρήγορα ακολουθήθηκε από το Google Maps, δικτυακές εφαρμογές με πλούσια περιβάλλοντα χρήστη και αλληλεπίδραση ισοδύναμη με αυτή που υποστηρίζουν οι εφαρμογές των προσωπικών υπολογιστών. Η συλλογή των τεχνολογιών που το Google χρησιμοποίησε ήταν η christened AJAX.

Πολλές από τις ικανότητες που τώρα εξερευνούν έχουν κάνει την εμφάνισή τους εδώ και πολλά χρόνια. Στα τέλη της δεκαετίας του ’90, και η Microsoft [URL: Microsoft] και η Netscape [URL: Netscape] οραματίζονταν τις ικανότητες που τελικά τώρα συνειδητοποιήθηκαν αλλά η μάχη μεταξύ τους έκανε δύσκολη τη δημιουργία εφαρμογών που υποστηρίζονται από τους φυλλομετρητές. Όταν η Microsoft τελικά «κέρδισε» τότε έγινε πιθανή μια τέτοιου είδους εφαρμογή. Και ενώ ο Firefox ξαναέφερε στο φως τον ανταγωνισμό στην αγορά των φυλλομετρητών, τουλάχιστον δε ξαναείδαμε τον καταστροφικό ανταγωνισμό σχετικά με τα δικτυακά πρότυπα, που κρατούσε πίσω την πρόοδο στη δεκαετία του ’90.

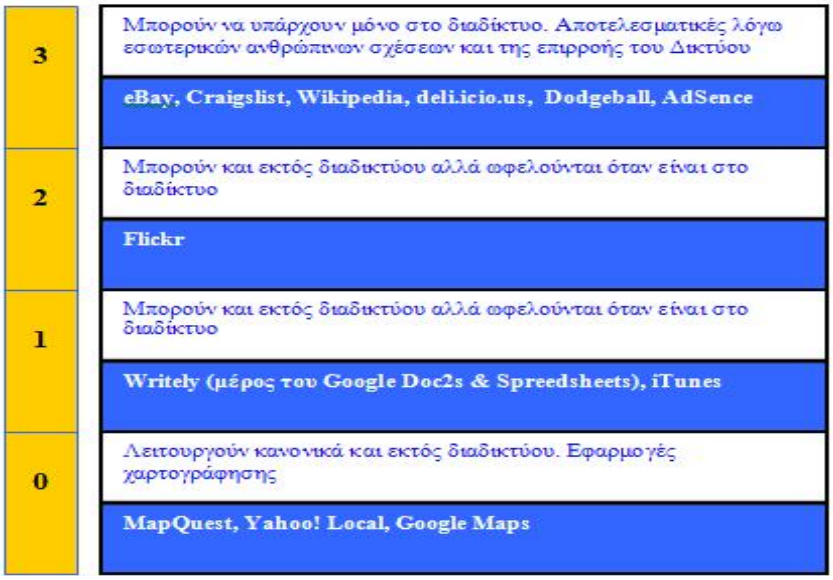
Είναι εύκολο να δούμε πως το Web2.0 θα ανακατασκευάσει το βιβλίο διευθύνσεων (address book). Ένα τέτοιο βιβλίο του Web2.0 – στυλ θα χειρίζεται το address book του προσωπικού υπολογιστή ή του τηλεφώνου ως μια μνήμη επαφών, που ο χρήστης θα ζητά από το σύστημα να θυμάται. Στο μεταξύ, μια δικτυακή ατζέντα, με το στυλ του Gmail, θα θυμάται κάθε σταλμένο ή λαμβανόμενο μήνυμα, κάθε ηλεκτρονική διεύθυνση, κάθε τηλεφωνικό αριθμό που χρησιμοποιείται, και θα δημιουργεί ένα ευρετικό κοινωνικό δίκτυο για να αποφασίζει ποιον θα επιλέξει ως εναλλακτικό στην περίπτωση που μια επαφή δε βρέθηκε στην τοπική μνήμη. Αν δε βρεθεί η επαφή εκεί, το σύστημα θα ρωτά το ευρύτερο κοινωνικό δίκτυο.

Ένας Web2.0 επεξεργαστής κειμένου θα ακολουθεί το Wiki-στυλ, δηλαδή την αθροιστική συμμετοχή στη δημιουργία του κειμένου, με παράλληλη υποστήριξη των πλούσιων δυνατοτήτων μορφοποίησης που υποστηρίζουν οι επεξεργαστές κειμένου των προσωπικών υπολογιστών. Ο Writely είναι ένα καλό αντιπροσωπευτικό παράδειγμα μιας τέτοιας εφαρμογής. Η επανάσταση του Web 2.0 δε θα περιοριστεί στις εφαρμογές των προσωπικών υπολογιστών.

Στο μέλλον, οι νέες εφαρμογές του Web2.0 θα περιλαμβάνουν δυνατότητες μάθησης από τους χρήστες, χρησιμοποιώντας την αρχιτεκτονική της συμμετοχής, ώστε να υπερτερούν στο χώρο του λογισμικού διεπαφής αλλά και στην αφθονία των διαμοιραζόμενων δεδομένων.

2.2.2 Επίπεδα ιεραρχίας του Web2.0

Ο Tim O’Reilly χώρισε το Web2.0 σε τέσσερα επίπεδα ιεραρχίας με βάση τις επιχειρήσεις και τα προϊόντα που ενσωματώνουν τις παραπάνω αρχές (βλέπε Εικόνα 2.4).



Εικόνα 2.4: *Επίπεδα ιεραρχίας του Web2.0.*

Παρακάτω αναφέρονται περιγραφικά αυτά τα επίπεδα, καθώς και παραδείγματα από εταιρείες και προϊόντα που συμπεριλαμβάνονται στο αντίστοιχο επίπεδο:

- Εφαρμογές 3ου επιπέδου: Γνωστές σαν «Web2.0 - oriented», που μπορούν να υπάρχουν μόνο στο Διαδίκτυο και είναι αποτελεσματικές λόγω των εσωτερικών ανθρώπινων σχέσεων και της επιρροής του δικτύου. Οι εφαρμογές αυτού του επιπέδου μπορούν να αυξήσουν την αποτελεσματικότητα τους, τόσο περισσότερο όσο οι άνθρωποι τις χρησιμοποιούν. Παραδείγματα τέτοιων τεχνολογιών είναι το eBay, Craigslist, Wikipedia, del.icio.us, Skype, dodgeball και το AdSense.

- Εφαρμογές 2ου επιπέδου: Μπορούν να διενεργηθούν εκτός Διαδικτύου, παρόλα αυτά όμως ωφελούνται όταν είναι στο Διαδίκτυο. Χαρακτηριστικό παράδειγμα είναι το Flickr, το οποίο ωφελείται από το τη κοινή βάση δεδομένων των φωτογραφιών.

- Εφαρμογές 1ου επιπέδου: Είναι επίσης διαθέσιμες εκτός Διαδικτύου παρόλα αυτά όμως ωφελούνται όταν είναι στο Διαδίκτυο. Χαρακτηριστικά παραδείγματα είναι το Writely (τώρα πλέον μέρος του Google Doc2s & Spreadsheets) και το i-Tunes (λόγω του τμήματος της μουσικής αποθήκης).

- Εφαρμογές 0ου επιπέδου: Επίσης λειτουργούν κανονικά εκτός Διαδικτύου. Παραδείγματα αποτελούν το MapQuest, το Yahoo! Local και το Google Maps (εφαρμογές χαρτογράφησης που χρησιμοποιούν συνεισφορές από τους χρήστες για να το αναπτύξουν).

2.2.3 Βασικές διαφορές του Web2.0 από το Web1.0

Η κύρια ερώτηση που οδήγησε στην μετακίνηση από το Web1.0 στο Web2.0, είναι «πώς μπορώ να πάρω αυτό το στοιχείο και να το μοιραστώ με άλλους ανθρώπους;». Δεδομένης αυτής της ερώτησης, όλοι οι ιστότοποι που υπήρχαν εκείνη τη στιγμή στο Διαδίκτυο προσπάθησαν να απαντήσουν με διαφορετικές προσεγγίσεις κάθε φορά.

Παρακάτω παρατίθενται κάποιες διαφορές μεταξύ Web1.0 και Web2.0:

- Το Web1.0 άρχισε ως ένας ηλεκτρονικός τόπος για τις επιχειρήσεις να μεταδίδουν τις πληροφορίες στους χρήστες/πελάτες. Τώρα, αρκετά χρόνια αργότερα, το μέρος αυτό έχει μετατραπεί σε μια θέση όπου και οι άνθρωποι μπορούν να προσφέρουν πληροφορίες.

- Στην αρχή, οι χρήστες διάβαζαν ότι οι άλλοι έγραφαν και ήταν συμβιβασμένοι με αυτή τη γνώση που αποκτούσαν. Αλλά, με την πάροδο του χρόνου, αυτοί οι ίδιοι χρήστες αποφάσισαν ότι θέλησαν να δημιουργήσουν μέρος των πληροφοριών που διαμοιράζονται.

- Η κίνηση από το Web1.0 προς το Web2.0 έχει σχέση με τους ανθρώπους που ανταλλάσσουν πληροφορίες για να γίνουν μέρος μιας μεγάλης κοινότητας.

- Πλέον, οι χρήστες δεν είναι παθητικοί δέκτες απλά συλλέγοντας την πληροφορία που τους διατίθενται, αλλά παίρνουν μέρος σε Blog, σε Forum και σχολιάζουν κάθε τι που γνωρίζουν.

- Στις εφαρμογές Web2.0, οι διεπαφές είναι πιο φιλικές, οι πληροφορίες είναι περισσότερες, οι άνθρωποι είναι περισσότερο συνδεδεμένοι και τα διάφορα δεδομένα αναζητούνται ευκολότερα σε σχέση με τις υπηρεσίες Web1.0.

Τέλος, για να γίνει πιο κατανοητός ο «διαχωρισμός» ανάμεσα στο Web1.0 και στο Web2.0 παρατίθεται ο Πίνακας 2.1 που φέρνει σε αντιπαράθεση μία σειρά από τεχνολογίες και εφαρμογές Web2.0 με τις «αντίστοιχες» που υπήρχαν στο Web1.0. [O'Reilly, 2005]

Τομέας	Web 1.0	Web 2.0
Διαφήμιση μέσω Διαδικτύου	Double-Click	Google AdSence
On-line διαχείριση φωτογραφιών	Ofoto	Flickr
Εφαρμογές κατανομής αρχείων	Akamai	BitTorrent
Πηγές μουσικής	Mp3.com	Napster
Εγκυκλοπαίδειες	Britannica Online	Wikipedia
Προσκλήσεις	Evite	Upcoming.org και EVDB
Κατάταξη ιστότοπου σε λίστες αποτελεσμάτων	Domain Name Speculation	Search Engine Optimization
Τεχνικές διαφήμισης	Page Views	Cost Per Click
Απόσπαση δεδομένων από έξοδο προγράμματος	Screen Scraping	Web Services
Δημιουργία περιεχομένου	Publishing	Participation
Διαχείριση περιεχομένου	Content Management Systems	Wikis
Ταξινόμηση περιεχομένου	Directories (taxonomy)	Tagging (floskonomy)
Λήψη περιεχομένου	Stickiness	Syndication
Ιστοσελίδες	WebPages	Web Blogs
Εφαρμογές	Standalone apps	Mashups
Λογισμικό	Λογισμικό ως προϊόν	Λογισμικό ως υπηρεσία
Πύλες	Portals	Composite applications

***Πίνακας 2.1**: Παραδείγματα τεχνολογιών και εφαρμογών Web1.0 και Web2.0. [Πηγή: http://oreilly.com]*

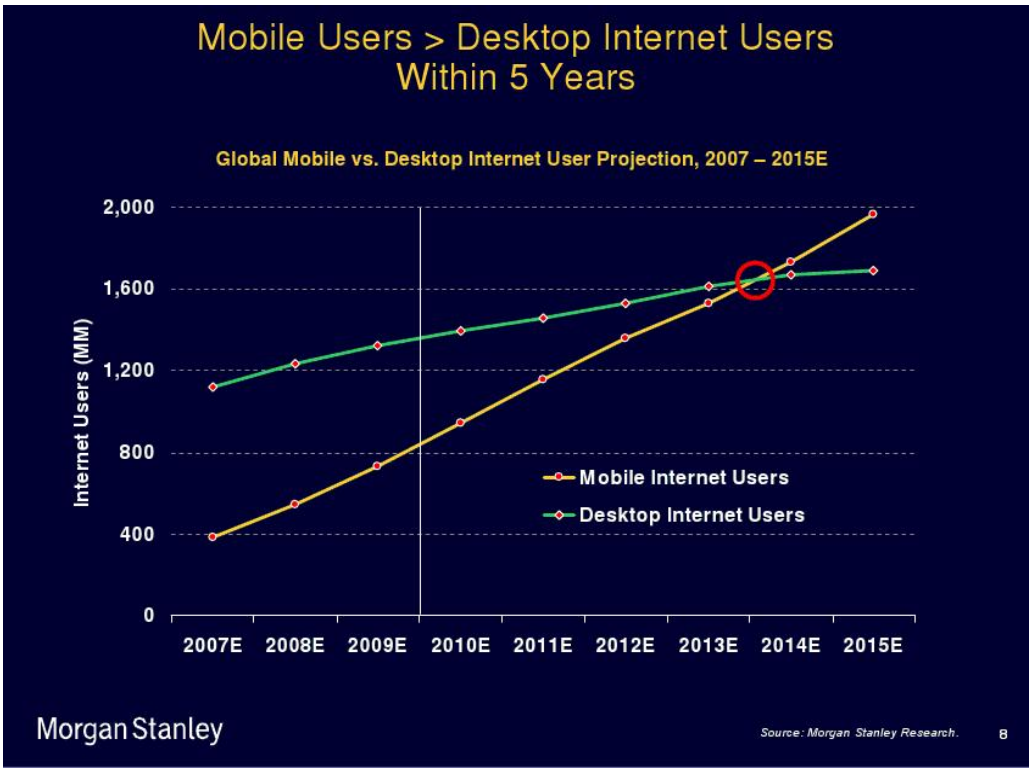
2.3 Mobile Web

Ο κινητός Ιστός (Mobile Web) αναφέρεται στη χρήση των Διαδικτυακών εφαρμογών, ή της μηχανή αναζήτησης, που βασίζονται στην πρόσβαση στο Διαδίκτυο από μια κινητή συσκευή, όπως ένα απλό κινητό τηλέφωνο, ένα Smartphone ή ενός Tablet PC, μέσω ενός ασύρματου δικτύου.

Παραδοσιακά, η πρόσβαση στον Ιστό γινόταν μέσω των υπηρεσιών της σταθερής τηλεφωνίας. Εντούτοις ο Ιστός τείνει να γίνει κινητός. Το 2008 ήταν ένα έτος ορόσημο για τη μετάβαση από σταθερή σε κινητή χρήση του Ιστού, καθώς επιτεύχθηκε η κινητή πρόσβαση στο Διαδίκτυο ενός προσωπικού υπολογιστή μέσω δικτύου κινητής τηλεφωνίας [URL: International Telecommunication Union 2009]. Στην πραγματικότητα, η μετατόπιση της χρήσης από τη σταθερή πρόσβαση στη κινητή πρόσβαση Ιστού άρχισε να πραγματοποιείται με ιδιαιτέρα γοργά βήματα από το 2007 με την εμφάνιση στην αγορά των πρώτων κινητών συσκευών τύπου Multitouch Smartphone (iPhone 2G [URL: iPhone 2G], Samsung SPH-M4650 [Samsung SPH-

M4650 - Review]) και πιο πρόσφατα από το 2010 με την εμφάνιση των υπολογιστών Multitouch Tablet (iPad [URL: iPad], Samsung Galaxy Tab [URL: Samsung Galaxy Tab], κλπ). Οι δύο παραπάνω πλατφόρμες είναι αυτές που συνέβαλαν και συνεχίζουν να συμβάλλουν ιδιαιτέρα στην πρόσβαση στο Διαδίκτυο και πιο συγκεκριμένα στο Mobile Web, χάρη στο ότι πλέον διαθέτουν προηγμένες εφαρμογές βασισμένες στο χρήση του Διαδικτύου-Web και ειδικότερα οι Web-Browsers τους έχουν εξαιρετική απόδοση, προσφέροντας μια πολύ δυνατή εμπειρία σε κάθε χρήστη του Διαδικτύου, έχοντας αφήσει πίσω τη μετριότητα των Browsers κινητών τηλεφώνων προηγούμενων γενεών.

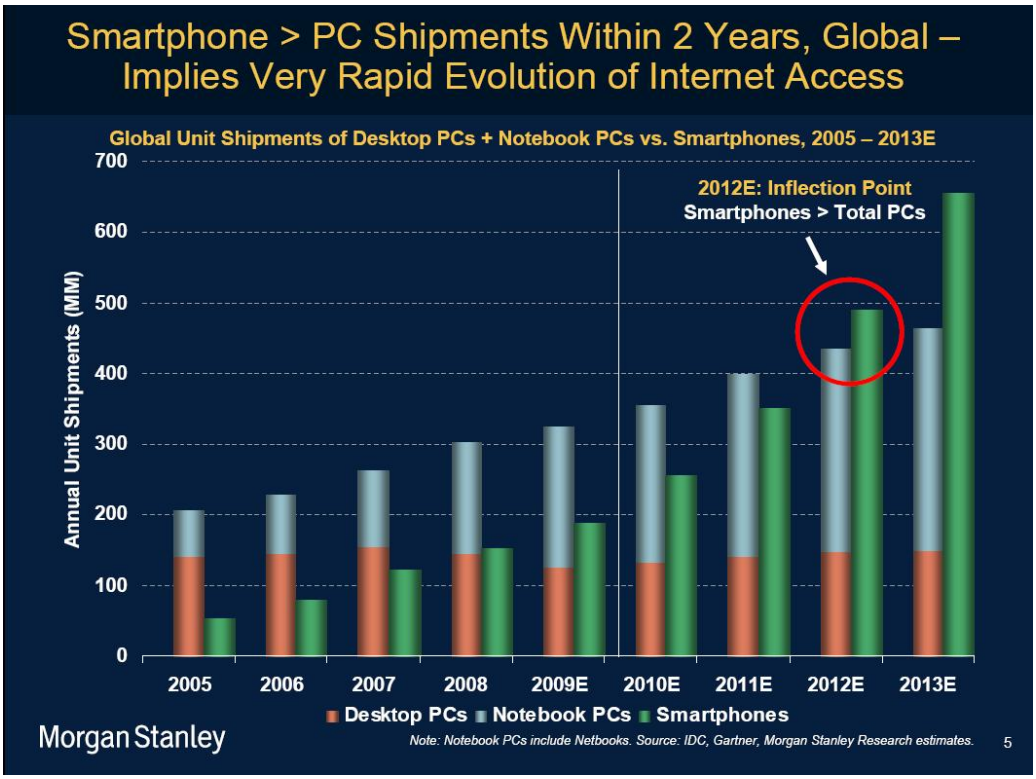
Είναι πλέον γεγονός πως η βελτίωση των δικτύων κινητής τηλεφωνίας, η βελτιωμένη ποιότητα κατασκευή κινητών συσκευών και η όλο πιο αυξανόμενη επεξεργαστική ισχύς των συσκευών αυτών, οδηγούν στη ραγδαία ανάπτυξη του Mobile Web και πιο γενικά του Mobile Internet. Σύμφωνα με στοιχεία που δημοσιεύτηκαν τον Απρίλιο του 2010 από την Mary Meeker [Meeker et al., 2010a] (Διευθύντρια στη Morgan Stanley [URL: Morgan Stanley] και επικεφαλής της ερευνητικής ομάδας τεχνολογίας) στην καθιερωμένη ετήσια αναφορά με τίτλο «κράτος του Διαδικτύου», οι χρήστες του Mobile Internet στα μέσα του 2014, θα ξεπεράσουν σε αριθμό αυτούς του Desktop Internet [Meeker, 2010a]. (βλέπε Εικόνα 2.5)



Εικόνα 2.5*: Διάγραμμα χρήστες Mobile Internet Vs Desktop Internet. [Πηγή: http://www.morganstanley.com]*

Αυτή η τάση, δηλαδή η ραγδαία ανάπτυξη του Mobile Internet σε σχέση με το Desktop Internet, μπορεί εύκολα να ενισχυθεί, αν αναλογιστεί κανείς πως σύμφωνα με μελέτη που πραγματοποιήθηκε από την informa Telecoms & Media [URL: informa Telecoms & Media - One billion subscribers to own smartphone devices in 2013], ο αριθμός των χρηστών smartphone αναμένεται να ξεπεράσει το 1 δισεκατομμύριο έως το 2013.

Αξιοσημείωτο είναι πως σύμφωνα με μία ακόμα παρουσίαση της Morgan Stanley με τίτλο «Internet Trends», που πραγματοποιήθηκε από την Mary Meeker τον Ιούνιο του 2010 [Meeker et al., 2010b] στη Νέα Υόρκη καταγράφεται πως στα μέσα του 2012 ο αριθμός των χρηστών Smartphones θα ξεπερνά τον αριθμό των χρηστών προσωπικών υπολογιστών (σταθερών και Netbook) (βλέπε Εικόνα 2.6)



***Εικόνα 2.5**: Διάγραμμα χρήστες Smartphone Vs PC. [Πηγή: http://www.morganstanley.com]*

2.3.1 Πρότυπα για το περιεχόμενο του Mobile Web

Οι υπηρεσίες στο Mobile Web μπορεί να έχουν το μεγάλο πλεονέκτημα της ελευθερίας κινήσεων λόγω της φορητότητας τους, εντούτοις η πρόσβαση στο Mobile Web σήμερα ακόμα υποφέρει από προβλήματα διαλειτουργικότητας, ευχρηστίας και δυνατότητας χρησιμοποίησης.

Τα προβλήματα διαλειτουργικότητας προέρχονται από τον τεμαχισμό των πλατφορμών των κινητών συσκευών, των λειτουργικών συστημάτων και των μηχανών αναζήτησης. Τα προβλήματα δυνατότητας χρησιμοποίησης είναι επικεντρωμένα γύρω από το μικρό φυσικό μέγεθος των οθονών των κινητών τηλεφώνων και άλλων μικροσυσκευών, σε σχέση με τις οθόνες των προσωπικών υπολογιστών.

Μία προσέγγιση για την αντιμετώπιση των παραπάνω προβλημάτων είναι η ανάπτυξη προτύπων (standards). Η πρωτοβουλία κινητού Ιστού (Mobile Web Initiative) [URL: W3C - Mobile Web Initiative] είναι μια νέα πρωτοβουλία από την W3C (World Wide Web Consortium) όπου έχει σαν σκοπό την ανάπτυξη καλύτερων πρακτικών και τεχνολογιών σχετικών με το Mobile Web. Ο σκοπός της πρωτοβουλίας είναι να γίνει το «ξεφύλλισμα» του Ιστού από τις κινητές συσκευές πιο αξιόπιστο και πιο προσιτό για τον χρήστη. Ο κύριος στόχος είναι να εξελιχθούν τα πρότυπα των μορφών δεδομένων (Data Formats) από τους παρόχους υπηρεσιών Ιντερνέτ ώστε να προσαρμόζονται στις προδιαγραφές και τις ιδιαιτερότητες των κινητών συσκευών. Το W3C έχει δημοσιεύσει τις οδηγίες για το κινητό περιεχόμενο (Mobile Content) και εξετάζει ενεργά το πρόβλημα της ποικιλομορφίας των συσκευών. Για το σκοπό αυτό προσπαθεί να δημιουργήσει μια βάση δεδομένων η οποία θα περιέχει περιγραφή για τις διάφορες κινητές συσκευές και τα χαρακτηριστικά τους.

Το W3C αναπτύσσει επίσης ένα σχέδιο για την αξιολόγηση της εγκυρότητας του περιεχομένου του Mobile Web, μέσω του mobileOK scheme [W3C, mobileOK Scheme 1.0, 2009], το οποίο θα βοηθήσει τους προγραμματιστές να ελέγξουν κατά πόσον το περιεχόμενο της σελίδας τους είναι συμβατό με το Mobile Web. Δίδεται ιδιαίτερη έμφαση στην προσαρμογή του περιεχομένου, πράγμα που αποτελεί το κλειδί για την επίτευξη του πανταχού παρόντος Ιστού, σε συνδυασμό με την «αποθήκη» περιγραφής των κινητών συσκευών.

Επίσης η mTLD (η εταιρία που έχει κατοχυρώσει το «.mobi») έχει δημιουργήσει, και διαθέτει δωρεάν, ένα εργαλείο, επονομαζόμενο MobiReady Report [URL: MobiReady Report], δοκιμής της ετοιμότητας ως προς την ικανότητα της ιστοσελίδας να ανταποκριθεί σε μία κινητή συσκευή. Το συγκεκριμένο εργαλείο κάνει μια ανάλυση της σελίδας και δίνει έτοιμο το αποτέλεσμα σε μια «.mobi» σελίδα. Αυτά τα τεστ που χρησιμοποιεί MobiReady Report χρησιμοποιούν τις καλύτερες πρακτικές και τα καλύτερα πρότυπα.

Παρακάτω παρατίθεται η λίστα των εξήντα βέλτιστων πρακτικών (πρότυπα) που πρέπει να ακολουθήσουν οι δημιουργοί, συντηρητές των ιστοσελίδων ώστε να πετύχουν την ιδανική παροχή περιεχομένου στον Mobile Web, με απώτερο στόχο τη βελτίωση της εμπειρίας των χρηστών του Mobile Web, όπως αυτές περιγράφονται στα επίσημα έγγραφα του W3C [W3C - Mobile Web Best Practices 1.0, 2008]:

1. Θεματική συνέπεια (Thematic consistency): Να εξασφαλιστεί ότι το περιεχόμενο που παρέχεται από την πρόσβαση σε ένα URI δίνει μία θεματικά συνεκτική εμπειρία όταν είναι προσβάσιμο από διάφορες συσκευές.

2. Δυνατότητες (Capabilities): Να εκμεταλλεύεται τις δυνατότητες των συσκευών ώστε να παράσχει μια βελτιωμένη εμπειρία στο χρήστη.

3. Ελλείψεις (Deficiencies): Να λειφθούν τα κατάλληλα μέτρα για να επιλύσουν ελλιπείς υλοποιήσεις.

4. Δοκιμή (Testing): Διενέργεια δοκιμών σε πραγματικές συσκευές, καθώς και εξομοιωτές.

5. URIs: Κρατήστε το μήκος του URI μικρό.

6. NavBar: Παρέχει μόνο την βασική πλοήγηση, στην κορυφή της σελίδας.

7. Ισορροπία (Balance): Πρέπει να λαμβάνεται υπόψη η εξισορρόπηση μεταξύ του να υπάρχουν πάρα πολλές συνδέσεις (Links) σε μια σελίδα και του να ζητάνε από το χρήστη να ακολουθήσει πάρα πολλές συνδέσεις για να φτάσει σε αυτό που ψάχνει.

8. Πλοήγηση (Navigation): παρέχει συνεπείς μηχανισμούς πλοήγησης.

9. Πλήκτρα πρόσβασης (Access Keys): Να αντιστοιχούν τα Access Keys σε Link μενού πλοήγησης με συχνά επισκεπτόμενη λειτουργικότητα.

10. Link Target ID: Να αναγνωρίζεις καθαρά (με σαφήνεια) το στόχο κάθε Link.

11. Link Target Format: Σημείωσε τη μορφή του Target File εκτός αν γνωρίζεις ότι το υποστηρίζει η συσκευή.

12. Image Maps: Μην χρησιμοποιείτε Image Maps αν δεν ξέρετε ότι η συσκευή τους υποστηρίζει αποτελεσματικά.

13. Αναδυόμενα παράθυρα (Pop ups Windows): Μην προκαλείτε να εμφανιστούν Pop-Ups Windows ή άλλα παράθυρα και να μην αλλάζετε το τρέχον παράθυρο χωρίς να ενημερώνεται ο χρήστης.

14. Αυτόματη ανανέωση (Auto refresh): Μην δημιουργείτε περιοδικά αυτόματα ανανεώσιμες σελίδες, αν δεν έχει ενημερωθεί ο χρήστης και αν δεν του παρέχετε ένα μέσο για να το σταματήσει αυτό.

15. Ανακατεύθυνση (Redirection): Μην χρησιμοποιείτε σήμανση για να ανακατευθύνετε σελίδες αυτόματα. Αντ' αυτού, διαμορφώστε το Server έτσι ώστε να πραγματοποιεί ανακατευθύνσεις μέσω 3xx κωδικών HTTP.

16. Εξωτερικές πηγές (External resources): Κρατήστε τον αριθμό των εξωτερικά συνδεόμενων πηγών στο ελάχιστο.

17. Κατάλληλο (Suitable): Βεβαιωθείτε ότι το περιεχόμενο είναι κατάλληλο για χρήση σε ένα πλαίσιο κινητού τηλεφώνου.

18. Σαφήνεια (Clarity): Χρησιμοποιήστε σαφή και απλή γλώσσα.

19. Οριοθέτηση (Limited): Περιορίστε το περιεχόμενο σε αυτό που ο χρήστης έχει ζητήσει.

20. Χρησιμοποιήσιμο Μέγεθος Σελίδας (Page Size Usable): Χωρίστε τις σελίδες σε χρησιμοποιήσιμα αλλά περιορισμένου μεγέθους τμήματα.

21. Όριο Μεγέθους Σελίδας (Page Size Limit): Βεβαιωθείτε ότι το συνολικό μέγεθος της σελίδας συμφωνεί με τους περιορισμούς της μνήμης της συσκευής.

22. Κύλιση (Scrolling): Περιορίστε το scrolling σε μία κατεύθυνση αν δεν μπορείτε να αποφύγετε και 2ο scrolling

23. Κεντρική Έννοια (Central Meeting): Βεβαιωθείτε ότι το υλικό που είναι κρίσιμο για το μήνυμα της σελίδας προηγείται αυτού που δεν είναι τόσο κρίσιμο.

24. Γραφικά για το Χώρο (Graphics for Spacing): Μην χρησιμοποιείτε γραφικά για το χώρο (κενό).

25. Μεγάλα Γραφικά (Large Graphics): Μην χρησιμοποιείτε εικόνες που δεν μπορούν να αναπαραχθούν από τη συσκευή. Αποφύγετε μεγάλες εικόνες ή εικόνες υψηλής ανάλυσης, εκτός εάν αυτές οι εικόνες περιέχουν κρίσιμες πληροφορίες.

26. Χρήση του Χρώματος (Use of Color): Διασφάλισε ότι οι πληροφορία που μεταφέρεται με χρώματα είναι επίσης διαθέσιμη και χωρίς το χρώμα.

27. Αντίθεση Χρώματος (Color Contrast): Βεβαιωθείτε ότι στο προσκήνιο και στο παρασκήνιο οι χρωματικοί συνδυασμοί παρέχουν επαρκή αντίθεση.

28. Αναγνωσιμότητα Εικόνας Παρασκηνίου (Background Image Readability): Όταν χρησιμοποιείτε εικόνες φόντου βεβαιωθείτε ότι το περιεχόμενο παραμένει αναγνώσιμο στη συσκευή.

29. Τίτλος Σελίδας (Page Title): Δώστε ένα σύντομο αλλά περιγραφικό τίτλο σελίδας.

30. Όχι Πλαίσια (No Frames): Μην χρησιμοποιείτε πλαίσια.

31. Δομή (Structure): Χρησιμοποιείστε χαρακτηριστικά της γλώσσας σήμανσης (Markup Language) για να επισημάνει τη λογική δομή του εγγράφου.

32. Υποστήριξη Πινάκων (Table Support): Μην χρησιμοποιείτε πίνακες εκτός αν είναι γνωστό πως η συσκευή τους υποστηρίζει.

33. Ένθετοι Πίνακες (Tables Nested): Μην χρησιμοποιείτε ένθετους πίνακες.

34. Πίνακες Διάταξης (Tables Layout): Μην χρησιμοποιείτε πίνακες για τη διάταξη.

35. Εναλλακτικοί Πίνακες (Tables Alternative): Όπου είναι δυνατό, χρησιμοποιήστε μια εναλλακτική λύση για την παρουσίαση σε μορφή πίνακα.

36. Εναλλακτικό Κείμενο για κάθε στοιχείο (Non-Text Alternatives): Να παρέχεται ένα ισοδύναμο κειμένου για κάθε στοιχείο που δεν είναι κείμενο (πχ εικόνα).

37. Αντικείμενα ή Scripts (Objects or Scripts): Μη βασίζεστε σε ενσωματωμένα αντικείμενα (embedded objects) ή σε scripts.

38. Καθορισμός Μεγέθους Εικόνας (Images Specify Size) Να καθορίζετε το μέγεθος των εικόνων στη σήμανση (Markup), αν έχουν ένα εγγενές μέγεθος.

39. Αλλαγή Μεγέθους Εικόνας (Images Resizing): Αλλάξτε το μέγεθος της εικόνας στον Server, εάν έχουν εγγενές μέγεθος.

40. Κατάλληλη σήμανση (Valid Markup): Δημιουργήστε έγγραφα που πιστοποιούνται με δημοσιευμένες επίσημες γραμματικές.

41. Μέτρα (Measures): Μην χρησιμοποιείτε μετρήσεις Pixel και απόλυτες μονάδες στις τιμές χαρακτηριστικών της γλώσσας σήμανσης και των ιδιοτήτων του CSS (Cascade Style Sheet).

42. Χρήση CSS (Style Sheets Use): Χρησιμοποιείτε CSS για να ελέγχετε τη διάταξη και την παρουσίαση, εάν η συσκευή είναι γνωστό ότι δεν τα υποστηρίζει.

43. Υποστήριξη CSS (Style Sheets Support): Οργανώστε τα έγγραφα έτσι ώστε εάν χρειαστεί να μπορούν να διαβαστούν και χωρίς το CSS.

44. Μέγεθος CSS (Style Sheets Size): Κρατήστε το μέγεθος του CSS μικρό.

45. Ελαχιστοποίηση (Minimize): Χρησιμοποιήστε λακωνική και αποτελεσματική σήμανση.

46. Υποστήριξη Μορφοποίησης Περιεχομένου (Content Format Support): Αποστείλατε το περιεχόμενο σε μια μορφή που είναι γνωστό ότι υποστηρίζεται από τη συσκευή.

47. Προτιμώμενη Μορφοποίηση Περιεχομένου (Content Format Preferred): Όπου είναι δυνατόν, να στείλετε το περιεχόμενο σε μια προτιμώμενη μορφή.

48. Υποστήριξη Κωδικοποίησης Χαρακτήρων (Character Encoding Support): Βεβαιωθείτε ότι το περιεχόμενο είναι κωδικοποιημένο με τη χρήση κωδικοποίησης χαρακτήρων που υποστηρίζεται από τη συσκευή.

49. Χρήση Κωδικοποίησης Χαρακτήρων (Character Encoding Use) Αναφέρατε στην απάντηση την κωδικοποίηση χαρακτήρων που χρησιμοποιείται.

50. Μηνύματα Σφάλματος (Error Messages): Να παρέχετε ενημερωτικά μηνύματα λάθους και ένα μέσο για την πλοήγηση από το μήνυμα σφάλματος πίσω στις χρήσιμες πληροφορίες.

51. Προσωρινά Αρχεία (Cookies): Να μην βασίζεστε στο αν είναι διαθέσιμα τα Cookies.

52. Διάθεση Προσωρινής Μνήμης (Cashing): Να παρέχετε τις προσωρινά αποθηκευμένες πληροφορίες στη μνήμη Cash σε αποκρίσεις HTTP.

53. Γραμματοσειρές (Fonts): Μη βασίζεστε στην υποστήριξη των γραμματοσειρών που σχετίζονται με το Styling.

54. Ελαχιστοποίηση Πληκτρολόγησης (Minimize Keystrokes): Κρατήστε τον αριθμό των πλήκτρων στο ελάχιστο.

55. Αποφυγή Ελεύθερου Κείμενου (Avoid Free Text): Αποφύγετε την εισαγωγή ελεύθερου κειμένου, όπου είναι δυνατόν.

56. Παροχή εξ ορισμού τιμών (Provide Defaults): Παρέχετε προεπιλεγμένες εξ ορισμού τιμές, όπου είναι δυνατόν.

57. Εξ Ορισμού Τρόπος Εισαγωγής (Default Input Mode) Διευκρινίστε μία κατάσταση ή γλώσσα ή μορφή εισαγωγής εξ ορισμού κειμένου, εάν η συσκευή το υποστηρίζει.

58. Σειρά ετικετών (Tab Order): Δημιουργήστε μια λογική σειρά, μέσω συνδέσμων, στοιχείων ελέγχου φόρμας και αντικειμένων.

59. Τοποθέτηση Ετικετών στα στοιχεία έλεγχου (Control Labeling): Βάλτε ετικέτες σε όλα τα στοιχεία ελέγχου κατάλληλα και αντιστοιχίστε τις ετικέτες μονοσήμαντα με τα στοιχεία ελέγχου.

60. Έλεγχος Θέσης (Control Position): Τοποθετήστε τις ετικέτες ώστε έτσι ώστε να διατάσσονται κατάλληλα σε σχέση με τα στοιχεία ελέγχου στα οποία αναφέρονται.

2.3.2 Πρότυπα κινητών συσκευών για χρήση Mobile Web

Οι κατασκευαστές των κινητών τηλεφώνων για να μπορέσουν να ανταπεξέλθουν σε μία ανταγωνιστική αγορά δημιουργούν προϊόντα-συσκευές που έχουν πολλά και διαφορετικά χαρακτηριστικά. Αποτέλεσμα αυτού είναι τόσο η ανάπτυξη όσο και ο σχεδιασμός των εφαρμογών του Mobile Web να μην είναι μια απλή διαδικασία για τους προγραμματιστές. Αυτή η συμπεριφορά ονομάζεται πολυμορφία (Device Market Fragmentation ή Device Diversity) των συσκευών και μερικά παραδείγματα τέτοιων διαφορών όσον αφορά τα κινητά τηλέφωνα μπορεί να είναι σε επίπεδο:

- Υλικού (μέγεθος οθόνης, πληκτρολόγιο)
- Δικτύου (ταχύτητα σύνδεσης, bandwidth)
- Συμπεριφοράς browser (Opera, NetFront, Safari, IE, Firefox Mobile κλπ)

Στον Πίνακα 2.2 εμφανίζονται τα πρότυπα χαρακτηριστικά που πρέπει να πληρούν τα κινητά τηλέφωνα για τις εφαρμογές του Mobile Web.

XHTML MP 1.0	Τα κινητά πρέπει να υποστηρίζουν ως Markup Language την XHTML MP 1.0
256 χρώματα	Οι οθόνες των κινητών πρέπει να υποστηρίζουν 256 χρώματα τουλάχιστον
120x120 Pixels	οι οθόνες των κινητών πρέπει να είναι τουλάχιστον 120x120 Pixels
10KB περιεχομένου	το μέγεθος του περιεχομένου (Content) δεν πρέπει να ξεπερνά τα 10KB
9.6Kbps Bandwidth	Το εύρος ζώνης των δεδομένων δεν πρέπει να ξεπερνά τα 9.6KB το δευτερόλεπτο
GIF και JPEG	Οι συσκευές των κινητών τηλεφώνων πρέπει να υποστηρίζουν μηχανισμούς συμπίεσης GIF και JPEG
UTF-8	χρησιμοποιείται ως κωδικοποίηση χαρακτήρων η UTF-8

***Πίνακας 2.2**: Πρότυπα χαρακτηριστικά κινητών τηλέφωνων για Mobile Web. [Πηγή: http://www.passani.it/gap/]*

2.3.3 Γλώσσες Σήμανσης Περιεχομένου (Markup Languages)

Για τις εφαρμογές του Mobile Web χρησιμοποιούνται γλώσσες σήμανσης περιεχομένου όπως η WML (Wireless Markup Language), η XHTML (Extensible Hypertext Markup Language) και η XHTML MP (Extensible Hypertext Markup Language Mobile Profile). Κάθε μία από αυτές έχει τα δικά της χαρακτηριστικά που σε αρκετές περιπτώσεις είναι κοινά αφού όλες έχουν τη ρίζα τους στην XML. Ωστόσο κάθε μία διαφοροποιείται από την άλλη σε κάποια σημεία της σύνταξής τους.

2.3.3.1 WML μία παγκόσμια γλώσσα - πρότυπο για το Mobile Web

Η WML είναι βασισμένη στην XML (Extensible Markup Language) και προορίζεται για ανάπτυξη εφαρμογών σε συσκευές που υποστηρίζουν το WAP (wireless application protocol) όπως τα κινητά τηλέφωνα. Έχει σχεδιαστεί λοιπόν για συσκευές μικρού εύρους και μικρής προσπέλασης. Η WML μοιάζει αρκετά με την HTML στο ότι και οι δύο γλώσσες παρέχουν πλοηγική υποστήριξη, δεδομένα εισόδου, Hyperlinks, κείμενο, αναπαράσταση εικόνας και φόρμες. Στην WML κάθε έγγραφο είναι γνωστό με τον όρο «Deck». Τα δεδομένα μέσα σε ένα deck δομούνται σε μία ή περισσότερες σελίδες που ονομάζονται «Cards» και κάθε μία από αυτές αναπαριστά μία αλληλεπίδραση με το χρήστη. Πολλές φορές που η μνήμη της συσκευής είναι περιορισμένη πρέπει τα Cards να χωρίζονται σε πολλαπλά decks ώστε να μην υπερφορτώνεται

ένα Deck με όλα τα Cards. Στη συνέχεια τα WML decks αποθηκεύονται σε ένα συνηθισμένο Web Server, ενώ όταν ζητηθούν οι WML cards από κάποια συσκευή αυτές μεταφέρονται από μια WAP gateway στη συσκευή του κινητού. Οι Gateways στέλνουν τις σελίδες σε μορφή κατάλληλη για υποδοχή από τα κινητά.

Πρέπει να επισημανθεί ότι η πραγματική δύναμη της WML είναι ότι παρέχει μία διεπαφή για το υλικό του κινητού τηλεφώνου προκειμένου να ξεκινήσει μία κλήση βασισμένη σε αυτό που ζητήθηκε από το χρήστη.

2.3.3.2 XHTML μία παγκόσμια γλώσσα πρότυπο για το Web

Η XHTML ανήκει και αυτή στην οικογένεια των XML γλωσσών και είναι σχεδόν ταυτόσημη με την HTML, ωστόσο αποτελεί μία καλύτερα δομημένη εκδοχή της. Είναι δηλαδή ένας συνδυασμός της XML και της HTML, υιοθετώντας όλα τα στοιχεία της HTML και την αυστηρή σύνταξη της XML. Υποστηρίζεται από σχεδόν όλους τους Browsers και συστήνεται ως γλώσσα ανάπτυξης εφαρμογών του Mobile Web από την W3C (World Wide Web Consortium, το οποίο αναπτύσσει τεχνολογίες ώστε οι λειτουργίες του Web να βελτιώνονται συνεχώς).

2.3.3.3 XHTML Mobile Profile μία παγκόσμια Markup γλώσσα για το Mobile Web

Η XHTML MP γλώσσα αποτελεί υποσύνολο της XHTML η οποία είναι μια αυστηρότερη εκδοχή της HTML. Είναι ουσιαστικά η XHTML Basic με κάποια επιπρόσθετα στοιχεία και ιδιότητες της XHTML. Ο στόχος της XHTML MP είναι να συνενώσει τις τεχνολογίες που χρησιμοποιούνται για την πρόσβαση στο Διαδίκτυο από τα κινητά με αυτές για το WWW (World Wide Web). Πριν την έλευση αυτής της γλώσσας σήμανσης περιεχομένου οι σχεδιαστές των WAP ιστοσελίδων έκαναν χρήση της WML ενώ οι σχεδιαστές των Web ιστοσελίδων χρησιμοποιούσαν την HTML. Με την παρουσία όμως αυτής της Markup γλώσσας, δεν είναι μόνο η παρουσίασή τους καλύτερη αλλά δίνεται παράλληλα και η δυνατότητα να χρησιμοποιούνται οι ίδιες τεχνολογίες για τον σχεδιασμό των WAP και Web ιστοσελίδων.

Τα πλεονεκτήματα της XHTML MP είναι:

- Μπορούν να χρησιμοποιηθούν τα ίδια εργαλεία για την ανάπτυξη τόσο ιστοσελίδων όσο και WAP σελίδων με αποτέλεσμα να μειώνεται το κόστος σχεδιασμού αλλά και ο χρόνος εκμάθησης νέων εργαλείων.

- Εφόσον ο χρήστης είναι εξοικειωμένος με τεχνολογίες Διαδικτύου όπως η HTML, XHTML και CSS μπορεί να σχεδιάσει μια WAP σελίδα.

- Συνηθισμένοι Web Browsers μπορούν να χρησιμοποιηθούν σαν εξομοιωτές για δοκιμή των WAP σελίδων.

- Η XHTML MP υποστηρίζει WAP CSS, κάτι το οποίο επιτρέπει το διαχωρισμό του περιεχομένου και της παρουσίασης σε διαφορετικά αρχεία. Αυτό σημαίνει ότι το περιεχόμενο χρειάζεται μία φορά να γραφεί ενώ η παρουσίαση μπορεί να αλλάζει ανάλογα με τη συσκευή που χρησιμοποιείται κάθε φορά. Επιπλέον, με τη βοήθεια του WAP CSS ο χρήστης αποκτά μεγαλύτερο έλεγχο στην παρουσίαση (π.χ. μπορεί να αλλάζει το μέγεθος της γραμματοσειράς και τα χρώματα).

2.3.4 Σύγκριση Web - Mobile Web

Αρκετές είναι οι διαφορές που παρουσιάζουν οι εφαρμογές του Mobile Web έναντι του κλασικού Web. Αρχικά, όσον αφορά τα εξωτερικά και πιο εμφανή κατά συνέπεια χαρακτηριστικά των δύο συσκευών (κινητό τηλέφωνο – υπολογιστής γραφείου) το ενδιαφέρον μονοπωλεί η οθόνη και το πληκτρολόγιο εξαιτίας της διαφοράς μεγέθους σε σχέση πάντα με έναν υπολογιστή γραφείου.

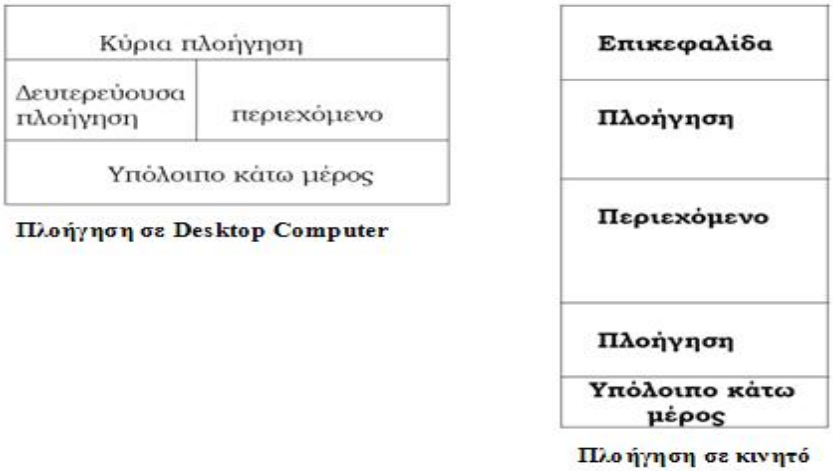
Αυτά με τη σειρά τους δημιουργούν νέες διαφορές που σχετίζονται με την πλοήγηση, το μέγεθος και τη μορφή των σελίδων αλλά και με το μέγεθος των μηνυμάτων. Εισάγονται λοιπόν κάποιοι περιορισμοί που δυσκολεύουν σε μεγάλο βαθμό το χρήστη. Οι περιορισμοί-χαρακτηριστικά αυτά είναι:

- **Μέγεθος οθόνης** - Το μέγεθος της οθόνης στην οποία εμφανίζονται τα δεδομένα αποτελεί έναν σημαντικό παράγοντα που βάζει το Web σε πλεονεκτική θέση έναντι του Mobile Web. Το σχετικά μικρό μέγεθος της οθόνης των κινητών συσκευών καθιστά δύσκολη ή και αδύνατη πολλές φορές την εμφάνιση κειμένου και γραφικών όπως αυτά θα εμφανίζονταν σε έναν υπολογιστή γραφείου. Το μέγεθος σε συνδυασμό με την έλλειψη δεικτικής συσκευής (Pointing

Device) και τα μικρά πλήκτρα ή το περιορισμένο πληκτρολόγιο μετατρέπουν σε εξαιρετικά δύσκολη εργασία την εισαγωγή δεδομένων. Αυτό με τη σειρά του συνεπάγεται ότι μεγάλα URLs είναι δύσκολο να πληκτρολογηθούν σωστά ενώ είναι ακόμη αρκετά δύσκολο να συμπληρωθούν φόρμες. Η δυσκολία που εισάγει το μικρό μέγεθος της οθόνης γίνεται περισσότερο αντιληπτή λαμβάνοντας υπόψη ότι μία συσκευή κινητού τηλεφώνου επιτρέπει 20-25 χαρακτήρες ανά γραμμή και 5-7 γραμμές κειμένου 22 στην οθόνη. Παρόλα αυτά τα νεότερα Smartphone όπως το iPhone [URL: iPhone] και αρκετές Android [URL: Android OS] και Windows Mobile [URL: Windows Mobile OS] συσκευές ξεπερνούν κάποιες από αυτές τις δυσκολίες.

- **Πλοήγηση** - Τα κινητά τηλέφωνα σε αντίθεση με τους υπολογιστές γραφείου δεν διαθέτουν ποντίκι (Pointing Device) με αποτέλεσμα η πλοήγηση στο Διαδίκτυο να γίνεται αποκλειστικά από το πληκτρολόγιο, κατ' επέκταση να αυξάνεται το Scrolling (κύλιση) και όπως είναι λογικό να μειώνεται η ευελιξία της πλοήγησης. Αν ο χρήστης επιθυμεί να προσπελάσει μία ιστοσελίδα προκειμένου να ενημερωθεί και να αναζητήσει άμεσα μία πληροφορία και αντί για αυτό καταλήξει να αναλώνει το χρόνο του με Scrolling τότε είναι σχεδόν σίγουρο ότι θα εγκαταλείψει την προσπάθεια αυτή. Αντί αυτού είναι σημαντικό, η γραμμή πλοήγησης να βρίσκεται σε κάποιο εμφανές σημείο της σελίδας (συνήθως στο πάνω μέρος) ώστε ο χρήστης να είναι σε θέση να επιλέξει γρήγορα αυτό που αναζητά.

Στην Εικόνα 2.7 φαίνεται ο διαφορετικός τρόπος με τον οποίο δομείται μία σελίδα όσον αφορά το μηχανισμό πλοήγησης σε έναν υπολογιστή γραφείου και σε ένα κινητό τηλέφωνο Παρότι δεν είναι αδύνατο να χρησιμοποιηθεί σε ένα κινητό τηλέφωνο το στυλ της πλοήγησης που χρησιμοποιείται σε έναν υπολογιστή γραφείου, γενικότερα δε λειτουργεί εξίσου καλά. Επιπλέον, είναι δυνατόν να ανατεθούν σε κάποια πλήκτρα (access keys) λειτουργίες όπως το Scrolling up & down.



Εικόνα 2.7*: Μηχανισμός πλοήγησης Desktop Vs Κινητό.*

- **Ταχύτητα** - Η ταχύτητα με την οποία προσπελαύνονται οι σελίδες στο Mobile Web είναι αισθητά μικρότερη από ότι στο Web για έναν υπολογιστή γραφείου και μάλιστα σε αρκετές περιπτώσεις είναι μικρότερη μιας Dial-Up σύνδεσης. Όπως είναι φυσικό η μικρή ταχύτητα (Low Speed) οδηγεί σε μεγάλους χρόνους αναμονής ιδιαίτερα εάν το περιεχόμενο είναι μεγάλο ή χρειάζεται πλοήγηση μεταξύ σελίδων.

Επιπλέον, το κόστος αναμονής ενισχύεται από το γεγονός ότι οι σελίδες Web είναι πιθανόν να περιέχουν και πληροφορίες που ο χρήστης δεν έχει ζητήσει και για τις οποίες δεν ενδιαφέρεται άμεσα, όπως για παράδειγμα διαφημιστικά παράθυρα ή μεγάλες εικόνες. Με τη σειρά του λοιπόν αυτό το «πρόσθετο υλικό» συμβάλλει στη μείωση της χρηστικότητας και στην αύξηση της καθυστέρησης (Latency).

- **Έλλειψη παραθύρων** - Ένας υπολογιστής γραφείου προσφέρει στο χρήστη τη δυνατότητα να ανοίγει περισσότερα από ένα παράθυρα κάθε φορά, πράγμα ιδιαίτερα χρήσιμο αφού μπορεί να εκτελεί παράλληλα διάφορες λειτουργίες. Αυτό πρακτικά σημαίνει ότι στην οθόνη του υπολογιστή γραφείου είναι ενεργά περισσότερα του ενός παράθυρα και ο χρήστης είναι αυτός που θα επιλέξει τη σειρά με την οποία θα περιηγηθεί σε αυτά. Μπορεί δηλαδή να υπάρξει εναλλαγή των παραθύρων, γεγονός που μειώνει το χρόνο αναμονής του χρήστη. Αντίθετα, στη μεγαλύτερη γκάμα των κινητών τηλεφώνων δε συμβαίνει ακριβώς το ίδιο. Μόνο μια σελίδα μπορεί να προσπελαστεί κάθε φορά ενώ επιπλέον ο χρήστης μπορεί να έχει πρόσβαση στις Web σελίδες σειριακά, με αποτέλεσμα να χάνει σε λειτουργικότητα και χρόνο. Το γεγονός αυτό οδηγεί σε μείωση της χρηστικής ικανότητας του Mobile Web.

- **Είδη προσπελάσιμων σελίδων** - Μεγάλος αριθμός κινητών τηλεφώνων δεν υποστηρίζουν Flash ή κάποιο άλλο τρίτο λογισμικό ώστε να μπορέσει να εξασφαλιστεί ασφαλής σύνδεση με κάποιες ιστοσελίδες. Κατά συνέπεια μειώνεται η πιθανότητα να βρεθεί ο χρήστης αντιμέτωπος με ανεπιθύμητες δυσλειτουργίες (του κινητού τηλεφώνου) που θα προκαλούσε η προσπέλαση αυτών των ιστοσελίδων.

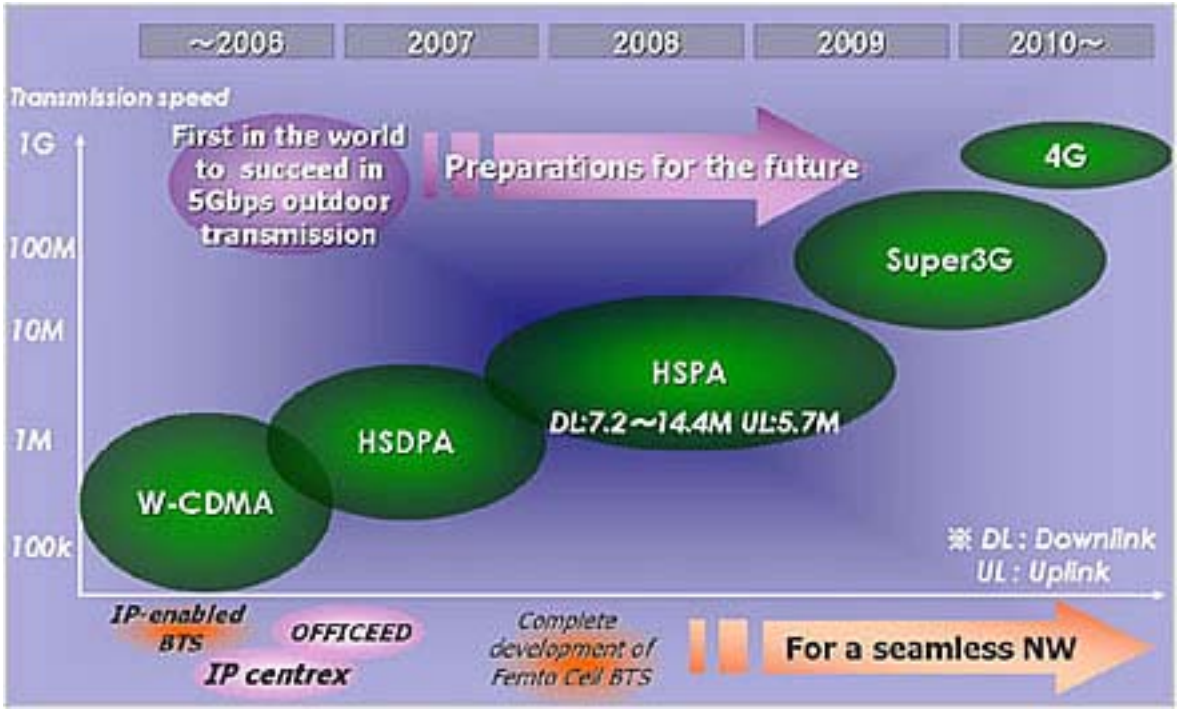
- **Μέγεθος μηνυμάτων** - Μία από τις σημαντικότερες εφαρμογές του Διαδικτύου είναι η υπηρεσίες του ηλεκτρονικού ταχυδρομείου (e-mail). Αποτελεί έναν εύχρηστο και γρήγορο τρόπο επικοινωνίας. Βασικό πλεονέκτημα των εφαρμογών του Web είναι ότι δεν θέτει περιορισμούς για τον αριθμό των χαρακτήρων που μπορούν να σταλούν. Αντίθετα, το Mobile Web περιορίζει το χρήστη θέτοντας ένα άνω όριο στον αριθμό των χαρακτήρων ενός e-mail μηνύματος, με αποτέλεσμα να απαιτείται η αποστολή πολλαπλών μηνυμάτων. Αυτό καθυστερεί τον χρήστη και παράλληλα τον κουράζει.

- **Ισχύς επεξεργαστή** - Συγκρίνοντας την ισχύ του επεξεργαστή ενός υπολογιστή γραφείου με αυτή ενός κινητού τηλεφώνου ο χρήστης διαπιστώνει ότι ο πρώτος υπερτερεί αρκετά έναντι του δεύτερου. Ο επεξεργαστής ενός υπολογιστή γραφείου προσφέρει ικανοποιητική ισχύ για να μπορεί ο χρήστης να εκτελεί διάφορες λειτουργίες σε μικρό χρόνο. Αντίθετα, ο επεξεργαστής των κινητών τηλεφώνων προσφέρει μικρότερη επεξεργαστική ισχύ και είναι συνεπώς είναι πιο χρονοβόρος.

2.3.5 Ανάπτυξη δικτύων κινητής τηλεφωνίας

Μέσω των δικτύων κινητής τηλεφωνίας μπορούν να αποκτήσουν πρόσβαση στο Internet κινητά τηλέφωνα, φορητοί υπολογιστές και υπολογιστές παλάμης. Ωστόσο, ο τρόπος πρόσβασης, όπως επίσης και η ποιότητά της, εξαρτώνται από το είδος κινητής τηλεφωνίας ή ακριβέστερα, από τη «γενιά» της. Ως γνωστόν, η κινητή τηλεφωνία διακρίνεται σε γενιές. Πρόκειται για διαφορετικές τεχνολογικές πλατφόρμες, που απεικονίζουν την εξέλιξη στην κινητή τηλεφωνία. Μολονότι συνήθως γίνεται λόγος για τέσσερεις γενιές, την πρώτη, τη δεύτερη και την τρίτη, τις οποίες για συντομία ονομάζουμε 1G, 2G, 3G και 4G, στην πραγματικότητα και οι γενιές και οι τεχνολογικές υποδομές μέσω των οποίων εκφράζονται είναι περισσότερες των τεσσάρων. Είναι χαρακτηριστικό ότι μεταξύ δεύτερης και τρίτης γενιάς παρεμβάλλεται η γενιά 2.5 (2.5G), ενώ περισσότερες της μιας τεχνολογικές υποδομές στεγάζονται κάτω από την «ομπρέλα» μιας γενιάς

(βλέπε Εικόνα 2.8), όπως θα δούμε και παρακάτω, ενώ κάλλιστα κάθε διαφορετική υποδομή θα μπορούσε να ταυτίζεται και με διαφορετική γενιά.



Εικόνα 2.8: *Διάγραμμα γενεών κινητής τηλεφωνίας (ταχύτητα, έτος). [Πηγή:* http://www.3g.co.uk*]*

Οι γενεές κινητής τηλεφωνίας έχουν ως εξής:

- **1G γενιά κινητής τηλεφωνίας -** Η αρχή έγινε με την 1G, που έκανε την εμφάνισή της πριν από το 1990 σε ορισμένες χώρες της Ευρώπης και στην Αμερική. Ογκώδεις συσκευές, χαμηλότατες ταχύτητες, χαμηλή ποιότητα και μετάδοση φωνής με αναλογικό τρόπο, αλλά και πολλά προβλήματα σύνδεσης ήταν τα χαρακτηριστικά της 1G. Στη συγκεκριμένη γενιά δεν υπήρχε η δυνατότητα χρήσης Internet διότι ούτε οι κινητές συσκευές κατάλληλες υπήρχαν αλλά ούτε και τα δίκτυα είχαν αυτή τη δυνατότητα.

- **2G γενιά κινητής τηλεφωνίας -** Η δεύτερη γενιά κινητής τηλεφωνίας εισήχθη στο παγκόσμιο στερέωμα στις αρχές της δεκαετίας του '90, κυρίως μέσω των τεχνολογικών υποδομών GSM (Global System for Mobile Communications) [URL: ETSI - European Telecommunications Standards Institute, Mobile technologies GSM] με χρήση σε Ευρώπη, Αφρική, Ασία, TDMA (Time Division Multiple Access) με χρήση στις ΗΠΑ, CDMA (Code Division Multiple Access) [URL: CDMA Development Group] σε ΗΠΑ και Ασία και PDC (Personal Digital Cellular) στην Ιαπωνία. Όλες οι παραπάνω τεχνολογικές πλατφόρμες

εντάσσονται στη 2G. Κύρια χαρακτηριστικά του GSM είναι η ψηφιακή μετάδοση φωνής με καλή πιστότητα και η ψηφιακή μετάδοση δεδομένων με χαμηλή ταχύτητα, μόλις 9,6 Kbps. Η πρώτη προσπάθεια σύνδεσης στο Internet μέσω των δικτύων της 2G έγινε πριν από μερικά χρόνια με την εμφάνιση του WAP (Wireless Application Protocol). Οι χαμηλές όμως ταχύτητες μετάδοσης δεδομένων και η ανυπαρξία συσκευών που να υποστηρίζουν ικανοποιητική απεικόνιση των ιστοσελίδων οδήγησε το WAP σε αποτυχία.

Στη συνέχεια μία άλλη τεχνολογία, η HSCSD (High Speed Circuit Switched Data) [URL: ETSI, HSCSD], ανέλαβε να υποστηρίξει τη γρήγορη μετάδοση δεδομένων μέσω του δικτύου GSM και πέτυχε να διπλασιάσει (και να τριπλασιάσει, ενίοτε) την ταχύτητα διαμεταγωγής δεδομένων, έναντι της ταχύτητας του GSM. Κάπου εδώ ολοκληρώνεται η πρώτη περίοδος των κινητών τεχνολογιών σύνδεσης με το Internet, που δεν στέφθηκε με επιτυχία, με εξαίρεση ορισμένες μόνο χώρες.

- **2.5G γενιά κινητής τηλεφωνίας** - Από εκείνο το σημείο και εφεξής τα δίκτυα κινητής τηλεφωνίας αρχίζουν να γίνονται πιο ελκυστικά για τους χρήστες που επιθυμούν να έχουν πρόσβαση στο Διαδίκτυο μέσω του κινητού τους ή άλλης φορητής συσκευής. Αυτό οφείλεται στο ότι οι γενιές που ακολούθησαν ήταν (και είναι) ασύγκριτα πιο γρήγορες, καθώς και στο ότι κυκλοφόρησαν στην αγορά οι "κατάλληλες" συσκευές (μεγάλες έγχρωμες οθόνες, με υποστήριξη πολυμέσων κ.λπ.). Έτσι, την αργή και χαμηλών δυνατοτήτων 2G ακολούθησε η γενιά 2.5. Τα τεχνολογικά πρότυπα της 2.5G είναι το GPRS (General Packet Radio Service) [URL: ETSI, GPRS] και το EDGE (Enhanced Data Rates For Global Evaluation) [URL: ETSI, EDGE], με πιο διαδεδομένο το GPRS, το οποίο χρησιμοποιείται και στην Ελλάδα. Η 2.5G, εκτός από άριστη ποιότητα μετάδοσης φωνής, προσφέρει μετάδοση δεδομένων σε πολύ υψηλότερες ταχύτητες έναντι της 2G, και, θεωρητικά, μπορεί να φθάσει και τα 115Kbps για το GPRS και τα 384Kbps για το EDGE.

- **3G γενιά κινητής τηλεφωνίας** - Η τρίτη γενιά κινητής τηλεφωνίας ενσωματώνει, αρκετά πρότυπα με πιο διαδεδομένα τα πρότυπα UMTS (Universal Mobile Telecommunications System) [URL: ETSI, UMTS] και WCDMA (Wideband Code Division Multiple Access) [URL: ETSI, W-CDMA]. Η 3G προσφέρει πολλές δυνατότητες υπηρεσιών πολυμέσων (λ.χ. παρακολούθηση τηλεοπτικών προγραμμάτων από το κινητό), συνεχή σύνδεση με το Διαδίκτυο

και μετάδοση κάθε είδους δεδομένων σε πολύ υψηλές ταχύτητες, από 64Kbps έως 384Kbps σε πρώτο στάδιο, και μέχρι τα 2Mbps αργότερα.

- **3.5G γενιά κινητής τηλεφωνίας -** Στη συνέχεια έκανε την εμφάνιση της μια άλλη τεχνολογία, η HSDPA (High-Speed Downlink Packet Access), η οποία αποτελεί μια ενισχυμένη μορφή της 3ης γενιάς κινητής τηλεφωνίας, που ανήκει στην οικογένεια της HSPA (High-Speed Packet Access). Αυτή η τεχνολογία αποκαλείται και ως 3.5G ή G+ ή Turbo 3G και επιτρέπει στα δίκτυα που είναι βασισμένα στο πρότυπο UMTS να έχουν υψηλότερες ταχύτητας μεταφορά δεδομένων. Οι τρέχουσες επεκτάσεις της HSDPA υποστηρίζουν ταχύτητες συνδέσεων 1.8, 3.6, 7.2 και 14.0 Mbps.

- **4G γενιά κινητής τηλεφωνίας -** Τέλος, η συνέχεια σε αυτή τη διαδοχή των γενιών δόθηκε με το λανσάρισμα της τέταρτης γενιάς κινητής τηλεφωνίας (4G), στα μέσα του 2010. Η τεχνολογία 4G είναι ένας νέος τρόπος χρήσης των δικτύων, με έμφαση κυρίως στη μετάδοση δεδομένων. Δανείζεται δηλαδή στοιχεία της τελευταίας γενιάς Wi-Fi και της ασύρματης τεχνολογίας μικρής εμβέλειας. Το 4G σημαίνει ταχύτερη μετάδοση δεδομένων. Η αναβάθμιση σε 4G είναι περισσότερο για να ενισχύσει όσα μπορούν να κάνουν οι χρήστες σε ένα δίκτυο 3G, όπως επεσήμανε ο Mat Carter (πρόεδρος του τμήματος 4G της Sprint). «Θεωρήστε το ως τη διαφορά ανάμεσα στην παρακολούθηση τηλεόρασης σε χαμηλή ανάλυση, σε σχέση με την υψηλή», εξήγησε ο ίδιος. «Από τη στιγμή που θα δεις την υψηλή ανάλυση, είναι δύσκολο να γυρίσεις στην προηγούμενη. Το ίδιο συμβαίνει και με το 4G», συμπλήρωσε. Σε ότι αφορά την ταχύτητα, αλήθεια είναι ότι το 4G είναι σίγουρα ταχύτερο από τα 3G δίκτυα, περίπου κατά τέσσερις φορές. Υπάρχει πάντως ένα εμφανές πλεονέκτημα του 4G σε σχέση με το 3G. Αν και δεν είναι πάντα ταχύτερο όταν πρόκειται για το «κατέβασμα» ενός αρχείου, είναι σίγουρα ταχύτερο σε ότι αφορά την έναρξη μετάδοσης των δεδομένων. Αυτό σημαίνει ότι ενδείκνυται στις περιπτώσεις ανταλλαγής δεδομένων, ενώ και οι τηλεφωνικές κλήσεις μπορούν να επωφεληθούν από τις μικρότερες καθυστερήσεις.

ΚΕΦΑΛΑΙΟ 3: Συστήματα Υποστήριξης Επιχειρηματολογίας

Για την ορθή μελέτη των Συστημάτων Επιχειρηματολογίας και ειδικότερα των Συστημάτων Υποστήριξης Συνεργασίας μέσω της Επιχειρηματολογίας, αρχικά αναλύονται κάποιες βασικές έννοιες όπως είναι το επιχείρημα (Argument), η επιχειρηματολογία (Argumentation), η επιχειρηματολογική κατασκευή γνώσης, η Αιτιολόγηση (Reasoning) και έπειτα ακολουθεί η ανάλυση σχετικά με τα μοντέλα διαχείρισης επιχειρημάτων. Στη συνέχεια του κεφαλαίου γίνεται αναφορά στην ανάπτυξη πολλών ηλεκτρονικών περιβαλλόντων που ως στόχο έχουν την υποστήριξη της επιχειρηματολογίας με κυριότερο κοινό χαρακτηριστικό τη χρήση εργαλείων που οπτικοποιούν τη διαδικασία της επιχειρηματολογίας. Τέλος γίνεται αναφορά στον περιορισμένο αριθμό τέτοιων εργαλείων βασισμένων στον Παγκόσμιο Ιστό (Web-Based) και αναπτύσσονται ορισμένα χαρακτηριστικά παραδείγματα. Κλείνοντας το κεφάλαιο επισημαίνεται το «βρεφικό» στάδιο ανάπτυξης τέτοιων εργαλείων, για χρήστες Mobile Web, μη έχοντας κάποια έτοιμη εφαρμογή, παρά μόνο μία «συγγενικού» ενδιαφέροντος εφαρμογή, που σχετίζεται με On-line Mind Mapping.

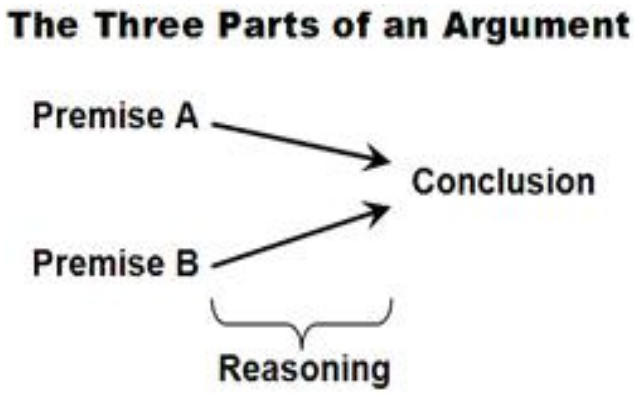
3.1 Εισαγωγή στην επιχειρηματολογία

3.1.1 Επιχείρημα

Μελετώντας την επιχειρηματολογία σίγουρα δεν θα ήταν σωστό να παραληφθεί ο προσδιορισμός του δομικού στοιχείου αυτής, του επιχειρήματος. Έχουν διατυπωθεί κατά καιρούς πολλοί ορισμοί που προσπαθούν να εξηγήσουν τι είναι επιχείρημα και να προσδιορίσουν τα συστατικά στοιχεία του.

Έτσι λοιπόν ένα επιχείρημα μπορεί να θεωρηθεί μια σειρά λογικών προτάσεων που εκφράζονται για να υποστηρίξουν μια θέση ή μια ένθερμη ανταλλαγή συγκλινουσών ή αντίθετων απόψεων [oxford dictionary].

Οι κυριότεροι εμπνευστές της δομής ενός επιχειρήματος είναι οι Stephen E. Toulmin και Charles Arthur Willard που έθεσαν τις βάσεις για την επεξήγηση αυτού και προσέγγισαν τα επιχειρήματα ως βάσεις για μια πειστική επικοινωνία. Όλα τα επιχειρήματα έχουν δομή η οποία είτε σχεδιάζεται και ακολουθείται εσκεμμένα είτε είναι προϊόν ατομικής ανακάλυψης. [URL: changingminds]. Σε γενικές γραμμές όμως θα μπορούσαμε να πούμε πως ένα επιχείρημα αποτελείται από έναν ή περισσότερους ισχυρισμούς (Premise), που πρέπει να αποδειχθούν πως είναι αληθείς, μια μέθοδος αιτιολόγησης-υποστήριξης (Reasoning) του ισχυρισμού και ένα συμπέρασμα (Conclusion) (βλέπε Εικόνα 3.1).



Εικόνα 3.1*: Τα τρία μέρη ενός επιχειρήματος. [Πηγή: http://www.thwink.org]*

3.1.2 Επιχειρηματολογία

Η επιχειρηματολογία (Argumentation) μπορεί να ορισθεί ως μία αλληλεπίδραση μεταξύ διαφόρων, ενδεχομένως συγκρουόμενων επιχειρημάτων, που έχει σαν στόχο την κατάληξη σε ένα συμπέρασμα καθολικά αποδεκτό.

Επίσης η επιχειρηματολογία μπορεί να ορισθεί ως «η δημιουργία απόψεων συνοδευόμενων από λόγους-επιχειρήματα υπέρ ή κατά ενός ισχυρισμού σε συνδυασμό με ερωτήσεις, επεξηγήσεις και διευκρινήσεις» και δεν είναι η στεγνή αντιπαράθεση ιδεών και απόψεων [Baker, 2003].

Είναι σαφές πως είναι δύσκολο να οριστεί η επιχειρηματολογία σε ένα πλαίσιο όπου λείπει η συνεργασία. Άλλωστε για να υπάρξει εποικοδομητική επιχειρηματολογία θα πρέπει οι συμμετέχοντες να συγκρουστούν, να εξωτερικεύσουν τις απόψεις τους, να εξετάσουν τις απόψεις των άλλων, να τις αξιολογήσουν, ίσως να αναθεωρήσουν τις δικές τους και από κοινού να καταλήξουν σε μια κοινά αποδεκτή λύση.

3.1.3 Επιχειρηματολογική κατασκευή γνώσης

Έρευνες του παρελθόντος έδειξαν πως η επιχειρηματολογική γνώση είναι ένα αναπόσπαστο κομμάτι όχι μόνο στην αξιολόγηση-κρίση καθημερινών καταστάσεων [Kuhn, 1991] αλλά και στη συμμετοχή σε επιστημονικές συζητήσεις [Driver et al., 2000; Sandowal & Millwood, 2005]. Ο όρος επιχειρηματολογική γνώση αναφέρεται στην κατανόηση του πώς να εξετάζει και να εκτιμά κανείς δεδομένα ώστε να είναι σε θέση να δημιουργήσει επιχειρήματα που θα τον τοποθετούν υπέρ ή κατά μιας θέσης.

Στην επιχειρηματολογική κατασκευή γνώσης οι εκπαιδευόμενοι εμπλέκονται στην επιχειρηματολογία με σκοπό να μάθουν πώς να επιχειρηματολογούν μέσα σε κάποιο πλαίσιο [Kuhn et al., 1997], κατασκευάζοντας επιχειρήματα σε ανάδραση με τους συνομιλητές τους, αλλά και να αποκτήσουν γνώσεις σχετικά με το αντικείμενο – θέμα προς συζήτηση [Weinberger & Fischer, 2006; Andriessen et al., 2003]. Ο καλύτερος τρόπος να μάθεις να επιχειρηματολογείς είναι να πάρεις ενεργά μέρος σε συζητήσεις στις οποίες πρέπει να ακολουθήσεις ορισμένες οδηγίες για να εκφράσεις τα επιχειρήματά σου καθώς επιλύεις κάποιο πρόβλημα. [Kuhn, 1991].

Η επιχειρηματολογική κατασκευή γνώσης λοιπόν στοχεύει σε δύο αποτελέσματα: στην απόκτηση γνώσης σχετικά με το αντικείμενο (Domain Specific Knowledge) και σχετικά με την

επιχειρηματολογία (Argumentative Knowledge). Αυτό σημαίνει ότι στην πρώτη περίπτωση να καταστήσει ικανούς τους εκπαιδευόμενους να εφαρμόσουν την νέο-αποκτηθείσα γνώση σε κάποια περίσταση, αξιολογώντας το νέο υλικό, ενεργοποιώντας την προηγούμενη γνώση και συνδέοντάς της με τη νέα. Στη δεύτερη περίπτωση να κατανοήσουν πως συντάσσεται ένα επιχείρημα (ισχυρισμός, απόδειξη, προσδιορισμός κατά Toulmin) και μια σειρά επιχειρημάτων (επιχείρημα, αντεπιχείρημα, ολοκλήρωση) [Andriessen et al., 2003].

3.1.4 Αιτιολόγηση (Reasoning)

Η αιτιολόγηση (Reasoning) μπορεί να θεωρηθεί ένα αναπόσπαστο κομμάτι της επιχειρηματολογίας διότι αναφέρεται στη παράθεση επιχειρημάτων, που καταλήγουν σε συμπέρασμα, ως προϊόν λογικής ανάλυσης ή απόφασης. Στην ουσία αφορά την κατάθεση επιχειρημάτων και αποδείξεων που χρησιμοποιούνται στην διάρκεια ενός συλλογισμού και στην αιτιολόγηση της απόφασης-άποψης. «Για ποιο λόγο πιστεύω αυτό που πιστεύω; Πως το υποστηρίζω;»

Κατά τη διάρκεια της επιχειρηματολογίας τα άτομα βγάζουν συμπεράσματα από δεδομένη γνώση για να καταλήξουν σε μία λύση η οποία δεν τους δόθηκε από την αρχή [Voss & Means, 1991]. Τα συμπεράσματα που εξάγονται κατά τη διάρκεια αυτής της διεργασίας έχουν παρόμοια δομή με ένα επιχείρημα και τα ίδια κριτήρια αξιολόγησης εφαρμόζονται σε αυτά.

Οι Means & Voss συμπέραναν πως η διαφωνία-επιχειρηματολογία βοηθάει στην αποθήκευση και ανάκτηση της γνώσης στη μνήμη και στη δημιουργία μοντέλων που ενθαρρύνουν την εξαγωγή συμπερασμάτων, λύση προβλημάτων και τη μάθηση [Means & Voss, 1996]. Συχνά όμως τα άτομα δεν είναι σε θέση να αιτιολογήσουν τις απόψεις τους και να τις βασίσουν πάνω σε λογικά συμπεράσματα και δεδομένα, ούτε να δεχτούν τους συλλογισμούς και τις δικαιολογίες των υπολοίπων. Ο Kuhn [Kuhn, 1991] συσχέτισε αυτή τη παρατήρηση με τις απόψεις των ατόμων σχετικά με τη γνώση και τα «πιστεύω» που θεωρούν πως έχουν και έτσι ισχυρίστηκε πως υπάρχουν δύο είδη γνώσεων:

- Απόλυτη επιστημολογία (Absolute Epistemology): η γνώση μου επικρατεί σε σχέση με όλες τις εναλλακτικές λύσεις – δεν υπάρχει κάτι άλλο εκτός από αυτό που πιστεύω.

- Αξιολογική επιστημολογία (Evaluative Epistemology): η γνώση δεν είναι στατική και αλλάζει συνεχώς.

Τα άτομα που τείνουν προς την απόλυτη επιστημολογία δεν αιτιολογούν ούτε διαφοροποιούν τις απόψεις τους ως εκ τούτου χρειάζονται περισσότερη υποστήριξη στην διαδικασία της αιτιολόγησης που θα πρέπει να ξεκινήσει από την αναθεώρηση των δικών τους σκέψεων.

Βέβαια η αιτιολόγηση μπορεί να γίνει και συνεργατικά λαμβάνοντας υπόψη πως η συνεργασία μπορεί συχνά να οδηγήσει στην ατομική μάθηση [Billig, 1987; Kuhn & Udell, 2003]. Έτσι οι Reznitskaya, Anderson, McNurlen, Nguyen-Jahiel, Arcodidou και Kim το 2001 [Reznitskaya et al., 2001] εφάρμοσαν τη συνεργατική αιτιολόγηση (Collaborative Reasoning) σε παιδιά δημοτικού σκοπεύοντας να τα καταστήσουν ικανά να επιχειρηματολογούν. Με τη βοήθεια του δασκάλου τα παιδιά κατέθεταν τα επιχειρήματά τους, αμφισβητούσαν τους συμμαθητές τους, ζητούσαν διευκρινίσεις, έκαναν ερωτήσεις. Απεδείχθη πως η μέθοδος αυτή βοηθάει στη δημιουργία «σχήματος» (Schema) και αφηρημένων δομών που βοηθάνε στην οργάνωση και ανάκτηση γνώσης σχετική με το επιχείρημα, ενθαρρύνουν την κατασκευή νέων επιχειρημάτων και θέτουν τις βάσεις για την εκτίμηση και τον σχολιασμό των αντιπάλων επιχειρημάτων [Anderson & Pearson, 1984; Reznitskaya et al., 2001].

3.1.5 Μοντέλα διαχείρισης επιχειρημάτων

Υπάρχουν αρκετά θεωρητικά μοντέλα διαχείρισης επιχειρημάτων όπως το μοντέλο Toulmin [Toulmin, 1958], η μέθοδος IBIS (Issue-Based Information System) [Kunz & Ritter, 1970], το μοντέλο PHI (Procedural Hierarchy of Issues) [McCall, 1991], το QOC (Questions, Options and Criteria) [MacLean et al., 1991], το DRL (Decision Representation Language) [Lee, 1991], το DRIM (Design Recommendation and Intent Model) [Pena-Mora et al., 1993], το WinWin Spiral Model [Boehm et al., 1998], η προσέγγιση του Compendium [Conklin et al., 2003] και το RATSpeak [Burge, 2005].

Στη συνέχεια θα αναλύσουμε τη μεθοδολογία το τεσσάρων «γνωστότερων» μοντέλων από τα παραπάνω:

Το μοντέλο Toulmin

Τον τελευταίο αιώνα η μελέτη επικεντρώθηκε στη σειριακή δομή ενός επιχειρήματος και έτσι οι μελετητές προσπάθησαν να προσδιορίσουν τη «γραμματική» ενός επιχειρήματος, όπως αυτό συμβαίνει σε μια πρόταση. Ποια δηλαδή είναι τα απαραίτητα συστατικά ενός καλά ορισμένου επιχειρήματος. Έτσι λοιπόν πρώτος ο Toulmin [Toulmin, 1958] προσδιόρισε πως ένα επιχείρημα έχει τα εξής δομικά στοιχεία:

- **Claim** (Ισχυρισμός – πρόταση προς συζήτηση - συμπέρασμα)

Το Claim είναι μια πρόταση την οποία ζητάμε από το άλλο άτομο να την αποδεχθεί. Αυτό περιλαμβάνει τις πληροφορίες που του ζητάμε να δεχθεί ως αληθείς ή τις ενέργειες που θέλουμε να δεχθεί και να θεσπίσει. Πχ: Θα πρέπει να χρησιμοποιήσετε ένα ακουστικό βαρηκοΐας.

Πολλοί άνθρωποι αρχίζουν με μια αξίωση, αλλά στη συνέχεια διαπιστώνουν ότι αυτή αμφισβητείται. Αν ένα άτομο απλά ζητήσει να γίνει κάτι, το άλλο άτομο δεν θα το πράξει έτσι απλά. Θα ρωτήσει γιατί θα πρέπει να συμφωνήσει και θα θελήσει αποδείξεις για τον συγκεκριμένο ισχυρισμό-αξίωση. Αυτό είναι που κάνει τα Grounds σημαντικά.

- **Grounds ή Data** (Σκεπτικό - οι λόγοι ή τα αποδεικτικά στοιχεία που ενισχύουν τον ισχυρισμό)

Τα Grounds είναι η βάση της πραγματικής πειθούς και αποτελείται από τα δεδομένα και τις «σκληρές πραγματικότητες», καθώς και το σκεπτικό πίσω από τον ισχυρισμό. Είναι η αλήθεια στην οποία βασίζεται ό ισχυρισμός. Το σκεπτικό (Ground) μπορεί να περιλαμβάνει την απόδειξη της τεχνογνωσίας και των βασικών καταστάσεων στις οποίες το υπόλοιπο του επιχειρήματος είναι χτισμένο.

Η πραγματική αλήθεια των στοιχείων μπορεί να είναι μικρότερη από 100%, όπως και όλα τα δεδομένα που βασίζονται στην αντίληψη και ως εκ τούτου, υπάρχουν κάποια στοιχεία υποθετικά. Πχ: Πάνω από το 70% όλων των ατόμων άνω των 65 ετών έχουν δυσκολία ακοής.

Οι πληροφορίες είναι συνήθως ένα πολύ ισχυρό στοιχείο της πειθούς, αν και επηρεάζουν τους ανθρώπους με διαφορετικό τρόπο. Εκείνοι που είναι δογματικοί, λογικοί ή ορθολογιστές έχουν περισσότερες πιθανότητες να πειστούν από πραγματικά δεδομένα. Όσοι λειτουργούν συναισθηματικά θα εμείνουν στις δικές τους θέσεις και θα αμφισβητήσουν τις πληροφορίες ή θα προσπαθήσουν να τις αγνοήσουν. Δηλαδή, κάποιοι θα δεχτούν τα δεδομένα χωρίς ερώτηση.

Κάποιοι θα τα απορρίψουν αμέσως και άλλοι θα ψάξουν βαθύτερα και θα απαιτήσουν περισσότερες εξηγήσεις. Εδώ είναι που εισάγεται το «Warrant».

- **Warrant** (Δικαιολογία που αποδεικνύει γιατί τα δεδομένα υποστηρίζουν τον ισχυρισμό)

Ένα Warrant συνδέει τα δεδομένα και τους άλλα σκεπτικά σε έναν ισχυρισμό (Claim) «νομιμοποιώντας» τον ισχυρισμό. Η δικαιολογία (Warrant) μπορεί να είναι ρητή ή σιωπηρή και αφανής. Πχ: Ένα ακουστικό βαρηκοΐας βοηθά τους περισσότερους ανθρώπους να ακούσουν καλύτερα.

Η δικαιολογία μπορεί να είναι απλή και μπορεί επίσης να είναι μεγαλύτερο επιχείρημα, με επιπλέον επιμέρους στοιχεία. Σε πολλά επιχειρήματα, συχνά οι δικαιολογίες είναι αυτονόητες και ως εκ τούτου ασταθείς. Αυτό δίνει τη δυνατότητα στο άλλο άτομο να εξετάσει και να εκθέσει τα Warrants, ίσως για να δείξει ότι είναι αδύναμα ή αβάσιμα.

- **Backing** (Υποστήριξη - προαιρετικές πληροφορίες που στηρίζουν το Warrant)

Το Backing για ένα επιχείρημα παρέχει πρόσθετη υποστήριξη για να το Warrant απαντώντας σε διάφορες ερωτήσεις. Πχ: Ακουστικά βαρηκοΐας είναι διαθέσιμα τοπικά.

- **Qualifier** (προσθέτει βαθμό βεβαιότητας στο συμπέρασμα)

Το «Qualifier» δείχνει την δύναμη το στοιχείων του Warrant, περιλαμβάνοντας λέξεις όπως «πιο», «συνήθως», «πάντα» ή «μερικές φορές». Τα επιχειρήματα μπορούν έτσι να κυμαίνονται από ισχυρούς ισχυρισμούς ως γενικά αρκετά «πλαδαρά» ή κατά ένα μεγάλο μέρος και συχνά μάλλον αβέβαια είδη δήλωσης. Πχ: Τα ακουστικά βαρηκοΐας βοηθούν τους περισσότερους ανθρώπους.

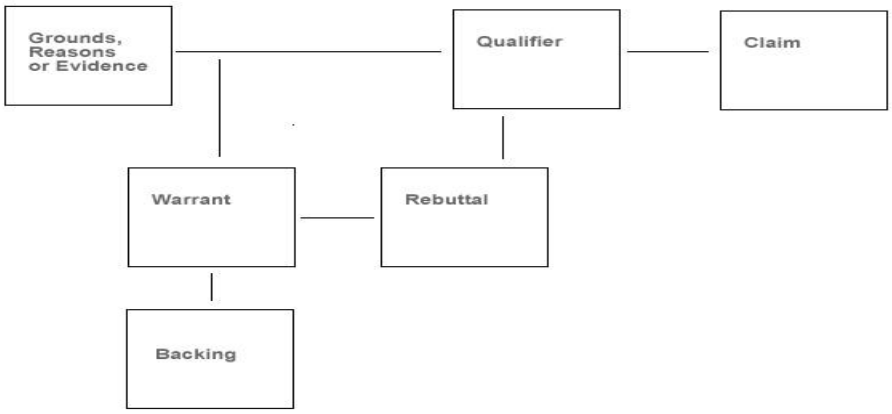
Μια άλλη παραλλαγή είναι η επιφύλαξη, η οποία μπορεί να δώσει τη δυνατότητα της εσφαλμένης αξίωσης. Πχ: Αν δεν υπάρχουν αποδείξεις για το αντίθετο, τα ακουστικά βαρηκοΐας δεν κάνουν καμία ζημιά στα αυτιά.

Το «Qualifiers» και οι επιφυλάξεις χρησιμοποιούνται ευρέως από τους διαφημιστές που περιορίζονται στο να μη λένε ψέματα. Έτσι χρησιμοποιούν λέξεις όπως «συνήθως», «ουσιαστικά», «εκτός» και ούτω καθεξής προκειμένου να ενισχύσουν τον ισχυρισμό τους.

- **Rebuttal** (Διάψευση - αντεπιχείρημα)

Παρά την προσεκτική κατασκευή του επιχειρήματος, μπορεί να υπάρχουν ακόμα αντεπιχειρήματα που μπορούν να χρησιμοποιηθούν. Αυτά μπορεί να ανατραπούν, είτε μέσω συνεχούς διαλόγου, ή με τη προεξόφληση του αντεπιχειρήματος δίνοντας την ανατροπή κατά την αρχική παρουσίαση του επιχειρήματος. Πχ: Υπάρχει ένα γραφείο υποστήριξης που ασχολείται με τεχνικά προβλήματα.

Οποιαδήποτε διάψευση είναι ένα επιχείρημα από μόνο του, και έτσι μπορεί να περιλαμβάνει έναν ισχυρισμό (Claim) μια δικαιολογία (Warrant), μία υποστήριξη (Backing) και ούτω καθεξής. Ασφαλώς, μπορεί να περιλαμβάνει και μια διάψευση (Rebuttal).



Εικόνα 3.2: *Τα έξι δομικά στοιχεία ενός επιχειρήματος κατά τον Toulmin. [Πηγή: http://www-rohan.sdsu.edu]*

Εκτός από τα έξι προαναφερθέντα δομικά στοιχεία στο επιχείρημα μπορούμε να προσδιορίσουμε άλλα τρία σημαντικά στοιχεία:

- Assumptions (οι Παραδοχές που γίνονται)
- Counter examples - counter Arguments (Αντιπαραδείγματα - Αντεπιχειρήματα)
- Implications (Συνέπειες)

Πολύ αργότερα από το 1958, οι Van Eemeren και Grootendorst [Van Eemeren & Grootendorst, 1999] ισχυρίστηκαν πως το παραπάνω γραμματικό μοντέλο δεν μπορεί να εφαρμοστεί σε πραγματικές συνθήκες και μάλιστα πως καλύπτει μόνο τον υπερασπιστή της αρχικής άποψης και όχι τον «αντίπαλο». Ισχυρίστηκαν πως το κυριότερο πρόβλημα αυτής της θεωρίας είναι πως δεν λαμβάνει υπόψη ότι η συνομιλία πρέπει να τοποθετείται μέσα σε κάποιο εννοιολογικό και κοινωνικό πλαίσιο.

Οι εκπαιδευτικές επιστήμες θεώρησαν πως ένα άλλο πρόβλημα είναι ότι το γραμματικό μοντέλο του Toulmin αγνοεί την εξέλιξη [Leitao, 2001] καθώς και την φύση της επιχειρηματολογίας που είναι υψηλού επιπέδου [Voss et al., 1983]. Έτσι στράφηκαν στην ανάλυση της επιχειρηματολογίας ως τύπο διαλόγου και ανέπτυξαν και άλλες προσεγγίσεις όπως:

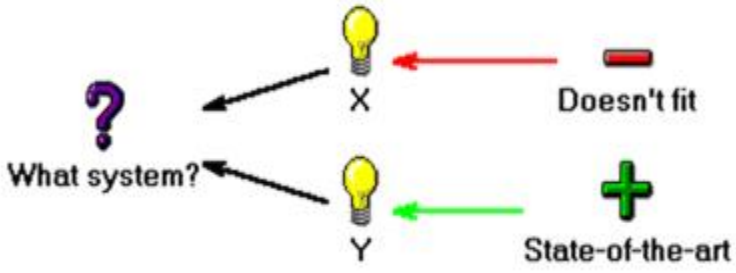
- Formal Dialectics [Barth & Krabbe, 1982]: Περιγράφει την επιχειρηματολογία ως έναν διάλογο ανάμεσα σε έναν υποστηρικτή και έναν «αντίπαλο».
- Pragma Dialectics [Van Eemeren & Grootendorst, 1992, 1999]: περιγράφει τη διάδραση ανάμεσα στις δυο πλευρές με όρους αναγκαίων συνθηκών για μια κριτική συζήτηση παρά με κανόνες λογικής για την κατασκευή επιχειρημάτων.

Η κριτική όμως που δέχτηκε το μοντέλο του Toulmin δεν αναιρεί πως η θεωρία του αποτέλεσε βασικό στοιχείο για τις μετέπειτα θεωρίες.

Το μοντέλο IBIS

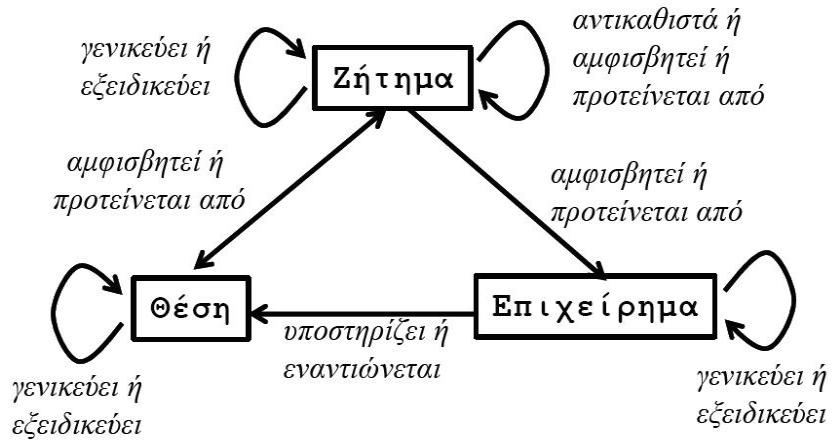
Το μοντέλο IBIS αναπτύχθηκε από τους Kunz και Rittel [Kunz et al., 1970] και είναι μια προσπάθεια να τοποθετηθεί η επιχειρηματολογία σε ένα μοντέλο που να αναπαριστά τις θέσεις και τις εναλλακτικές που προτάθηκαν από τους συμμετέχοντες. Η αναπαράσταση δεν είναι απαραίτητο να γίνεται γραφικά, αλλά βασίζεται στην αρχή ότι η διαδικασία σχεδιασμού ενός πολύπλοκου προβλήματος είναι μια συνομιλία μεταξύ συμμετεχόντων που όλοι έχουν τον δικό τους επιστημονικό τομέα και στην οποία κάνουν κοινοκτημοσύνη την εμπειρία τους, την γνώση τους (ρητή/άρρητη) προς επίλυση των σχεδιαστικών προβλημάτων.

Γύρω από ένα θέμα προς επίλυση, που αποκαλείται και topic, τοποθετούνται οι συνομιλητές και καταθέτουν αρκετές προτάσεις/θέσεις. Κάθε φορά που τοποθετούνται αναφέρουν για ποιο λόγο υποστηρίζουν την άποψη τους (Reasoning) και έτσι δημιουργούνται επιχειρήματα μέχρι να αποδειχθεί η θέση που υποστηρίχθηκε. Μια πρόταση/θέση με την σειρά της μπορεί να έχει ένα ή περισσότερα επιχειρήματα τα οποία την ενισχύουν ή την αναιρούν. Σε αυτή τη διαδικασία δημιουργείται μια πιο δομημένη άποψη του αρχικού «topic» αλλά και των πιθανών λύσεων του. [Conklin & Begeman, 1988]



Εικόνα 3.2: *Το σετ των επιτρεπόμενων ρητορικών κινήσεων στο IBIS. [Πηγή: Conklin & Begeman, 1988]*

Η διαβούλευση στο IBIS στοιχειοθετείται γύρω από τρεις οντότητες: Τα ζητήματα (issues) που αντιπροσωπεύουν καθετί που χρειάζεται συζήτηση και που τίθεται σε διαβούλευση κατά τη διάρκεια του σχεδιασμού, τις θέσεις (positions) που συνιστούν εναλλακτικές στάσεις επίλυσης ή θεώρησης των ζητημάτων, και τα επιχειρήματα (arguments) που είναι δηλώσεις υπέρ ή κατά μιας ή περισσότερων θέσεων. Οι συσχετίσεις που συνδέουν ζητήματα, θέσεις και επιχειρήματα είναι διαφόρων τύπων και δηλώνουν ποιοτικά πως ένα ζήτημα υποστηρίζεται από μια θέση (π.χ. σχεδιαστική άποψη) στη βάση συγκεκριμένων επιχειρημάτων.

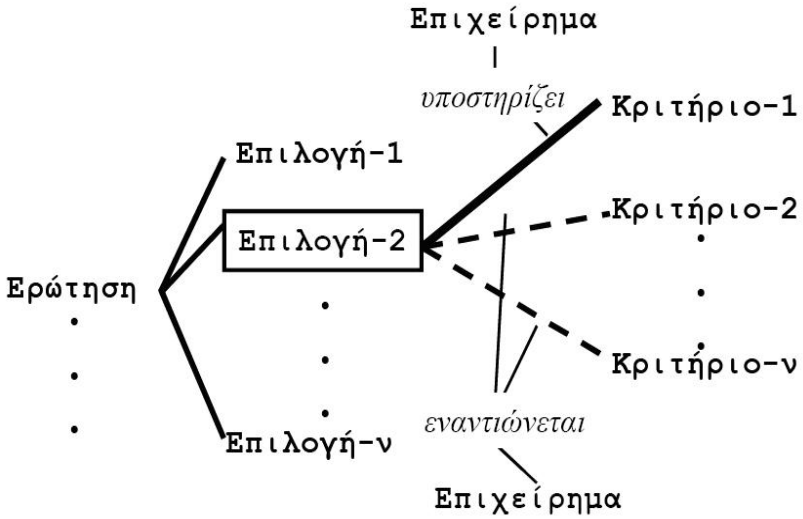


Εικόνα 3.3*: Το μετα-μοντέλο IBIS. [Πηγή: Conklin & Begeman, 1988]*

Στο IBIS η διαβούλευση έχει δενδρική δομή και αρχίζει όταν ένας από τους εμπλεκόμενους εταίρους υποβάλλει το αρχικό ζήτημα που συνιστά την κορυφή σε ένα IBIS δένδρο. Το δένδρο εμπλουτίζεται ή αυξάνεται με την πάροδο της διαβούλευσης και την υποβολή ζητημάτων, θέσεων και επιχειρημάτων από τους εταίρους. Η διαδικασία αυτή συνεχίζεται μέχρι να υπάρξει απόφαση σχετικά με το ζήτημα που μελετάται.

Το μοντέλο QOC

Μια πρόταση που ακολούθησε χρονικά την ανάπτυξη του IBIS, και η οποία είναι ιδιαίτερα δημοφιλής, είναι η προσέγγιση που προτείνουν οι MacLean, Bellotti και Moran με τον τίτλο Questions, Options and Criteria [MacLean et al., 1991]. Η μέθοδος QOC εξειδικεύει μια γραφική τεχνοτροπία για τη διατύπωση και αιτιολόγηση εναλλακτικών προτάσεων σχετικά με χαρακτηριστικά σχεδιαστικών κατασκευασμάτων. Ουσιαστικά, η μέθοδος QOC αφορά τη διαχείριση του σχεδιαστικού χώρου στον οποίο εμπίπτει ένα σχεδιαστικό τέχνημα. Για το σκοπό αυτό, η μέθοδος ορίζει συγκεκριμένες οντότητες και συσχετίσεις μεταξύ αυτών.



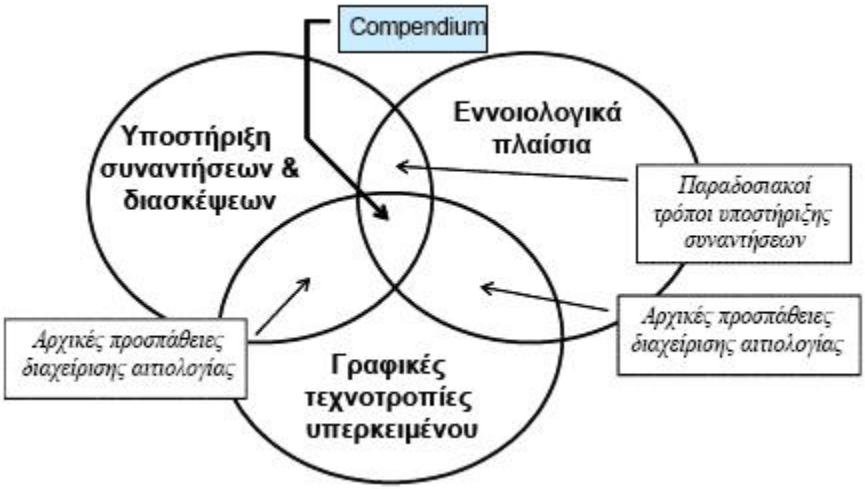
***Εικόνα 3.4**: Το μοντέλο QOC. [Πηγή: MacLean et al., 1991]*

Οι οντότητες σε ένα QOC γράφημα είναι τύπου: επιλογές (Options) που αντιπροσωπεύουν τα χαρακτηριστικά ενός τεχνήματος, ερωτήσεις (Questions) που είναι το μέσο οργάνωσης των διαφόρων επιλογών και αντιστοιχούν στα ζητήματα του IBIS, κριτήρια (Criteria) που καθορίζουν την επιλογή μεταξύ των επιλογών, και επιχειρήματα (Arguments) που, όπως στο IBIS, υποστηρίζουν ή εναντιώνονται σε στοιχεία ενός QOC μοντέλου. Ένα τυπικό QOC μοντέλο έχει τη γραφική μορφή που απεικονίζεται στην Εικόνα 3.4.

Η προσέγγιση του Compendium

Το Compendium [Conklin et al., 2003] αποτελείται από αρκετά εργαλεία, που στο σύνολο τους συνιστούν μια σύγχρονη προσπάθεια στην κατεύθυνση συλλογικού συμπερασμού (group sense making). Τα εργαλεία του Compendium χτίζουν πάνω σε αρχικές προσπάθειες διαχείρισης

αιτιολογίας και μνήμης ομάδων, προσδιορίζοντας επιπλέον μια συνολική προσέγγιση στη σύγχρονη διαχείριση γνώσεων (knowledge management). Η προσέγγιση αυτή βασίζεται σε τρεις πυλώνες (ανάπτυξη εννοιολογικών πλαισίων, υποστήριξη συναντήσεων και συσκέψεων, και γραφικές τεχνοτροπίες υπερκειμένου) που κατά καιρούς έχουν διερευνηθεί ως μεμονωμένα γνωστικά πεδία (βλέπε Εικόνα 3.4)



***Εικόνα 3.4**: Η προσέγγιση του Compendium. [Πηγή: Conklin et al., 2003]*

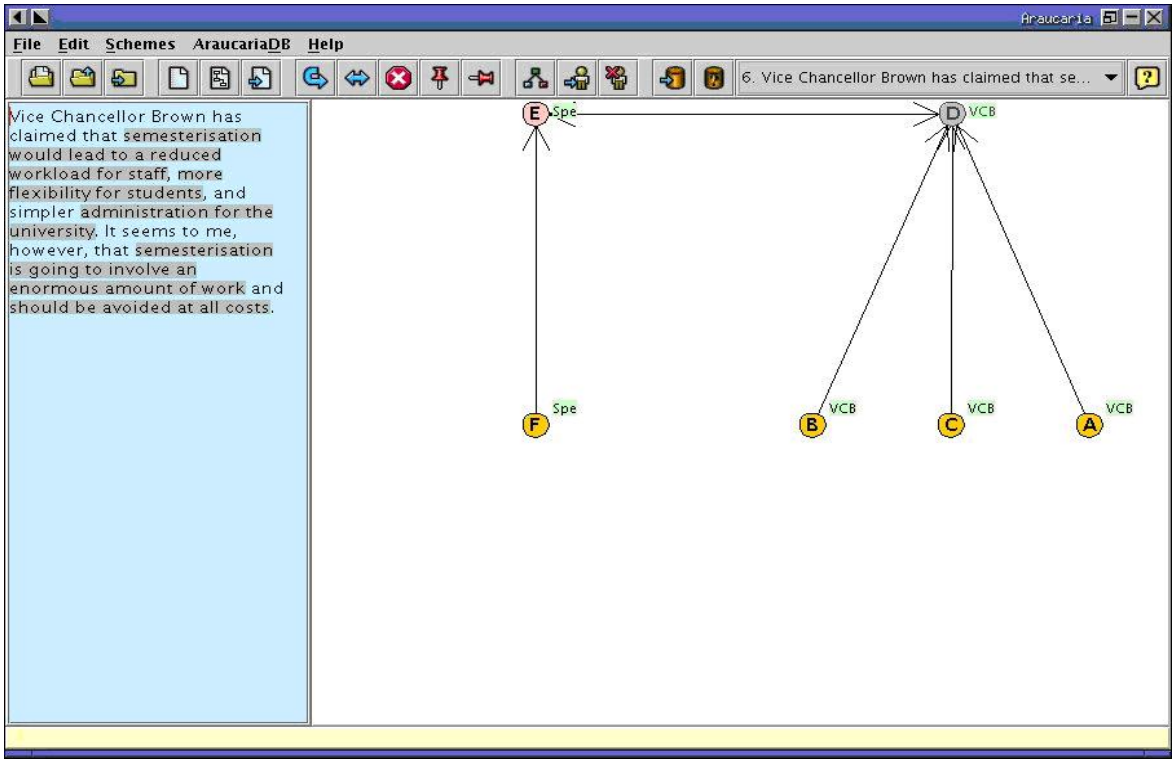
Τα εργαλεία υποστήριξης συνεργασίας του Compendium είναι τρία, και περιλαμβάνουν: (α) ένα γραφικό σύστημα υπερκειμένου (Graphical Hypertext) σχεδιασμένο για μοντελοποίηση ημιδομημένων υπερκειμένων, (β) ένα διαχειριστή (Compendium Facilitator) που εργάζεται με την ομάδα των συνεργαζόμενων, δρώντας ως γέφυρα μεταξύ της διαβούλευσης που πραγματοποιείται κατά τη διάρκεια μιας συνάντησης, και της καταγραφής και παρουσίασης της διαβούλευσης μέσω υπολογιστή, και (γ) μοντέλα (διαγραμματικές τεχνοτροπίες) όπως του IBIS, που δομούν τη συζήτηση και καταγράφουν την πρόοδο της διαδικασίας.

3.2 Εφαρμογές αναπαράστασης σκέψης και απεικόνισης διαλόγων με επιχειρήματα

Συστήματα υποστήριξης επιχειρηματολογίας που βασίζονται στα μοντέλα διαχείρισης του προηγούμενου κεφαλαίου μπορούμε να συναντήσουμε σε ηλεκτρονικές εφαρμογές αναπαράστασης σκέψης και απεικόνισης διαλόγων με επιχειρήματα όπως τα Araucaria [Reed & Rowe, 2004], το Hermes [Karacapilidis & Papadias, 1998], το Athena [Rolf et al., 2002], το Compendium [Buckingham Shum et al., 2006], το Belvedere [Suthers et al., 1995], το

Reason!Able [Van Gelder, 2002, 2003] και πολλά άλλα. Παρακάτω περιγράφονται οι προαναφερθείσες εφαρμογές:

Το **Araucaria** [URL: Araucaria Software] είναι ένα εργαλείο λογισμικού για τα επιχειρήματα. Βοηθά τον χρήστη στην αναδημιουργία και στη σχεδίαση ενός επιχειρήματος, με τη χρήση απλού Point and Click Interface. Υποστηρίζει επίσης τα σχέδια επιχειρηματολογίας και παρέχει ένα εξατομικευμένο από το χρήστη σύνολο σχεδίων με το οποίο μπορεί να αναλύσει τα επιχειρήματα. Μόλις αναλυθούν τα επιχειρήματα, μπορούν να σωθούν με ένα Portable σχήμα που ονομάζεται AML (γλώσσα σήμανσης επιχειρήματος), βασισμένο σε XML.

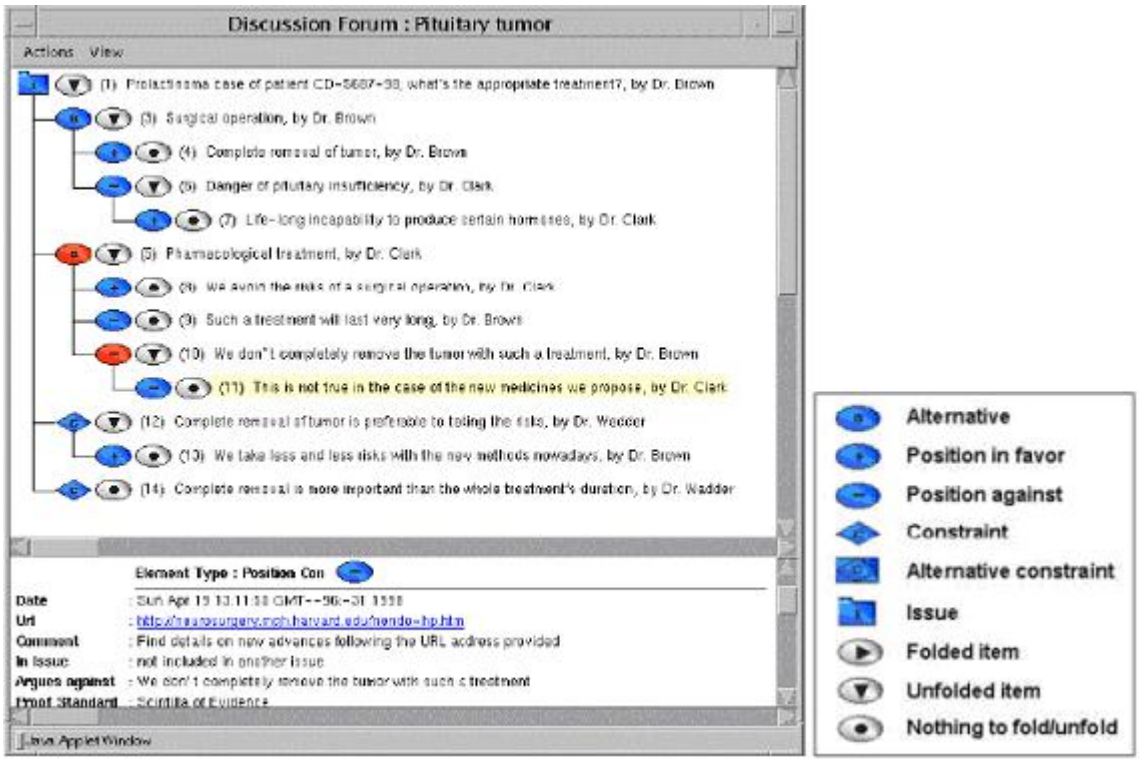


***Εικόνα 3.5**: Περιβάλλον εργασίας Araucaria. [Πηγή: http://araucaria.computing.dundee.ac.uk]*

Το **Hermes** [URL: Hermes System] παρέχει μια πλατφόρμα ασύγχρονης επικοινωνίας (Forum) που υποστηρίζει την επιχειρηματολογία χρησιμοποιώντας σύμβολα στην αρχή κάθε μηνύματος που οπτικοποιούν τη δομή-πορεία του διαλόγου. Η επεξήγηση μερικών συμβόλων που χρησιμοποιούνται είναι:

- Issue: Θέση προς συζήτηση
- Position in favor/against: Επιχείρημα υπέρ ή κατά μια άποψης αντίστοιχα

- Alternatives: Εναλλακτική άποψη
- Constraints: υποδηλώνει τη σχέση μεταξύ επιχειρημάτων και μαρτυρά την ισχύς μεταξύ αυτών.



***Εικόνα 3.6**: Παράδειγμα στο Hermes. [Πηγή: http://www-sop.inria.fr/aid/hermes/]*

Επίσης το κάθε σχόλιο αριθμείται για να γίνεται πιο εύκολη αναφορά σε αυτό και περιέχει πληροφορίες σχετικά με το ποιος το υπέβαλλε, πότε υπεβλήθη και αν υπάρχει κάποια σχετική URL που να το υποστηρίζει.

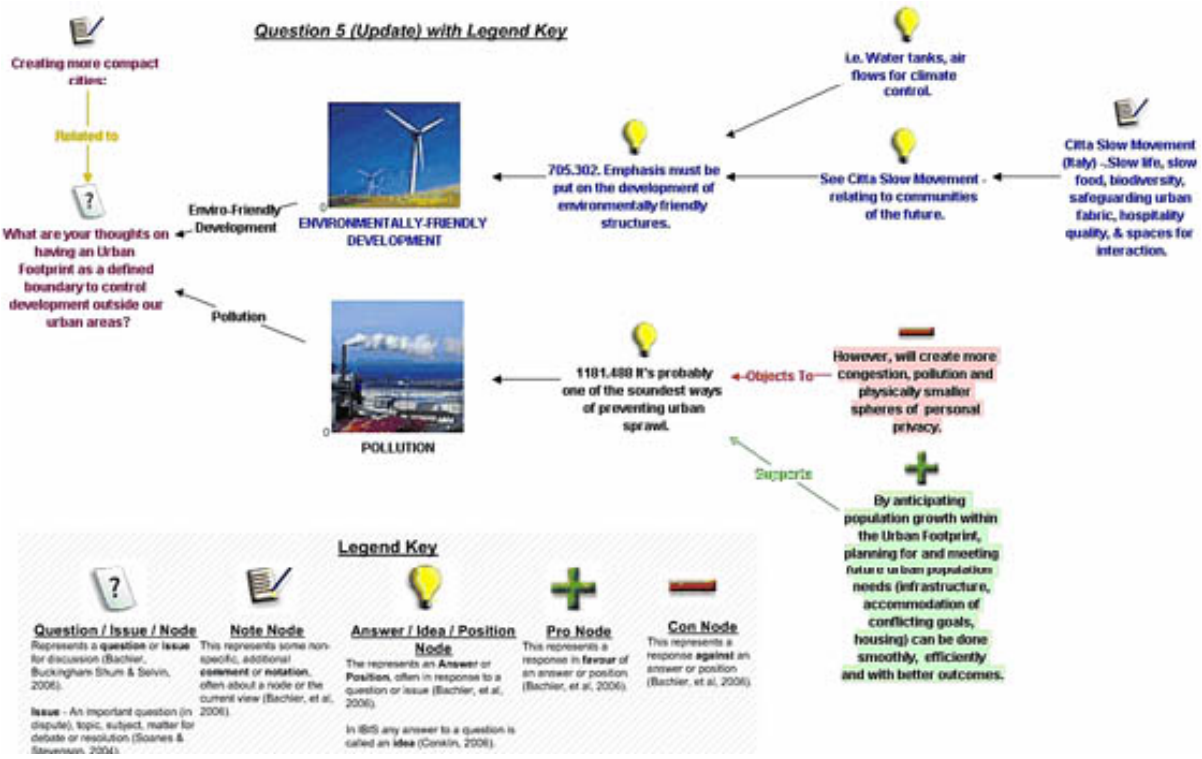
Το **Compendium** [URL: Compendium] σχεδιάστηκε για να παρέχει σε πραγματικό χρόνο καταγραφή (Mapping) των συζητήσεων σε συσκέψεις, συνεργατική μοντελοποίηση και διαχείριση της οργανωτικής μνήμης (Organizational Memory). Οι αποφάσεις υποδηλώνονται, καθώς και οι παραθέσεις σε αυτές (εξωτερικά δεδομένα) όπως σημειώσεις, λογιστικά φύλλα κτλ. Ανάλογα με το θέμα που θέλει κανείς να διαπραγματευτεί υπάρχει δυνατότητα για χρήση διαφόρων εργαλείων.

Όταν χρησιμοποιείται για επιχειρηματολογία χρησιμοποιεί πλήθος από διαφορετικούς κόμβους που υποδηλώνονται με διαφορετικά εικονίδια, δηλαδή το θέμα προς συζήτηση αναπαριστάται με

ένα «?», τα σχόλια με έναν λαμπτήρα, επιχειρήματα υπέρ μιας άποψης με το «+» και κατά με το «-», κάποιο επιχείρημα που λειτουργεί υποστηρικτικά σε κάποιο άλλο, αλλά αντίθετα σε άλλο με «+/-». Επίσης υπάρχει δυνατότητα σύνδεσης- αναφοράς σε εξωτερική πηγή δεδομένων, σε διαφορετικό χάρτη αλλά και σε παράθεση ιδεών με κουκκίδες. Η συμφωνία απεικονίζεται με έναν «κόμβο απόφασης».

Οι σύνδεσμοι που ενώνουν τους κόμβους χαρακτηρίζονται από μεγάλη ποικιλία. Μπορούν να απεικονιστούν είτε με μονόδρομα βέλη, είτε με αμφίδρομα, είτε με εντελώς απλά. Υποδηλώνουν σχέσεις όπως: ανταπόκριση, υποστήριξη, αντίθεση, πρόκληση, διευκρίνηση, συμπλήρωση, σχέση αλλά και οποιοδήποτε άλλη σχέση θέλει να δημιουργήσει ο χρήστης.

Τέλος, παρέχει και διαγράμματα, που όμως δεν εξειδικεύονται στην επιχειρηματολογία αλλά στην επίλυση συγκρούσεων.

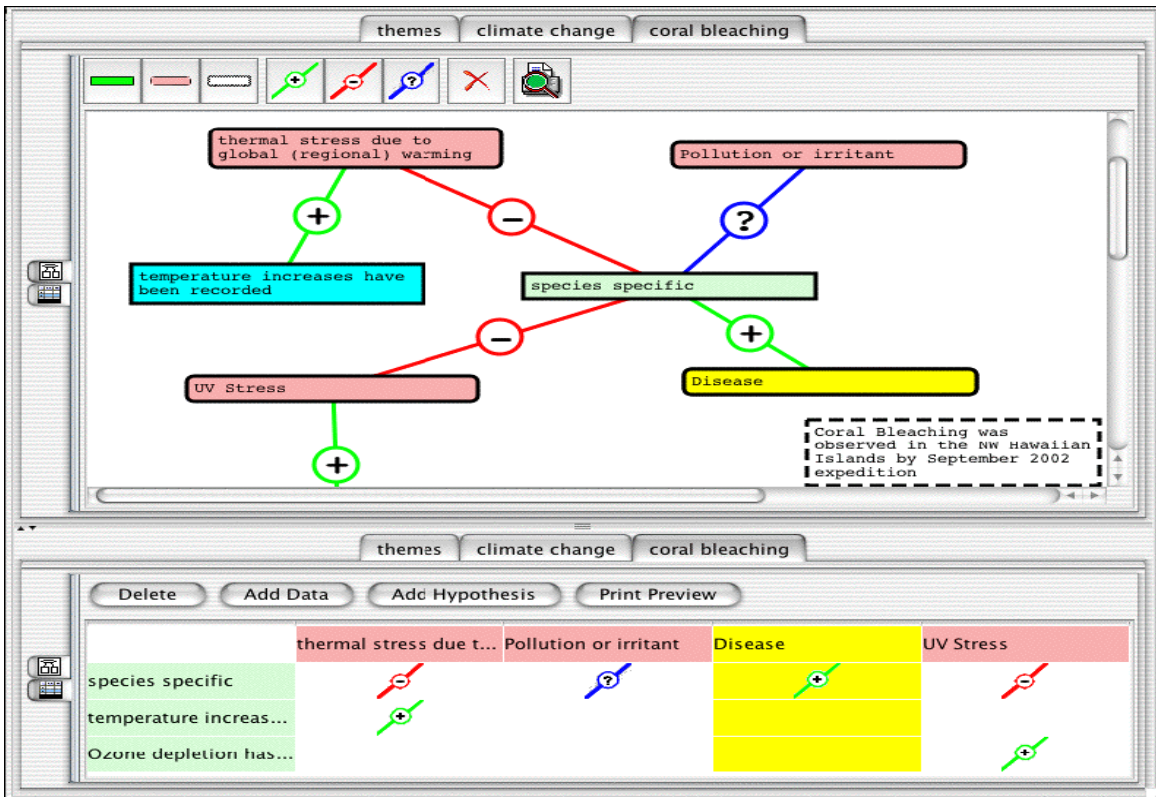


Εικόνα 3.7: *Παράδειγμα Compendium. [Πηγή: http://compendium.open.ac.uk]*

Το **Belvedere** αναπτύχθηκε στο πανεπιστήμιο του Pittsburg από την ερευνητική ομάδα ανάπτυξης γνωστικών εργαλείων και επιτρέπει τόσο τη σύγχρονη όσο και την ασύγχρονη επικοινωνία. Το Belvedere αποτελείται από έναν κοινό χώρο για τη δόμηση «διαγραμμάτων διερεύνησης», τα οποία σχετίζουν δεδομένα και υποθέσεις με αποδεικτικές σχέσεις. Η οντολογία

των αναπαραστάσεων είναι δεδομένη και αμετάβλητη και κάθε είδους έννοια που χρησιμοποιείται στην στοιχειοθέτηση επιχειρημάτων αντιστοιχεί σε μία γραφική αναπαράσταση. («Δεδομένα», «Υπόθεση», «Αξίωμα», «Σχόλιο»).

Σε κάθε κόμβο μπορεί να εμφανίζεται το όνομα του χρήστη και να σηματοδοτείται με έντονο χρώμα ο ενεργός κόμβος και σύνδεσμος. Εκτός από αυτό ο χρήστης έχει τη δυνατότητα να καθορίσει την ισχύ-βαρύτητα του επιχειρήματος, που εκφράζεται με το πάχος της εξωτερικής γραμμής του κόμβου αλλά και να το συνδέσει με κάποια πηγή στο Διαδίκτυο. Οι σύνδεσμοι μεταξύ των κόμβων αλλά και ανάμεσα σε συνδέσμους, που υποδηλώνουν τη σχέση μεταξύ αυτών, αναπαριστώνται με διαφορετικά χρώματα για να είναι πιο ευδιάκριτα, π.χ. ο πράσινος σύνδεσμος ενώνει επιχειρήματα που λειτουργούν υποστηρικτικά ενώ ο κόκκινος εκφράζει την αντίθεση – διαφωνία, ο ουδέτερος σύνδεσμος «ΚΑΙ» συνδέει όμοια επιχειρήματα.



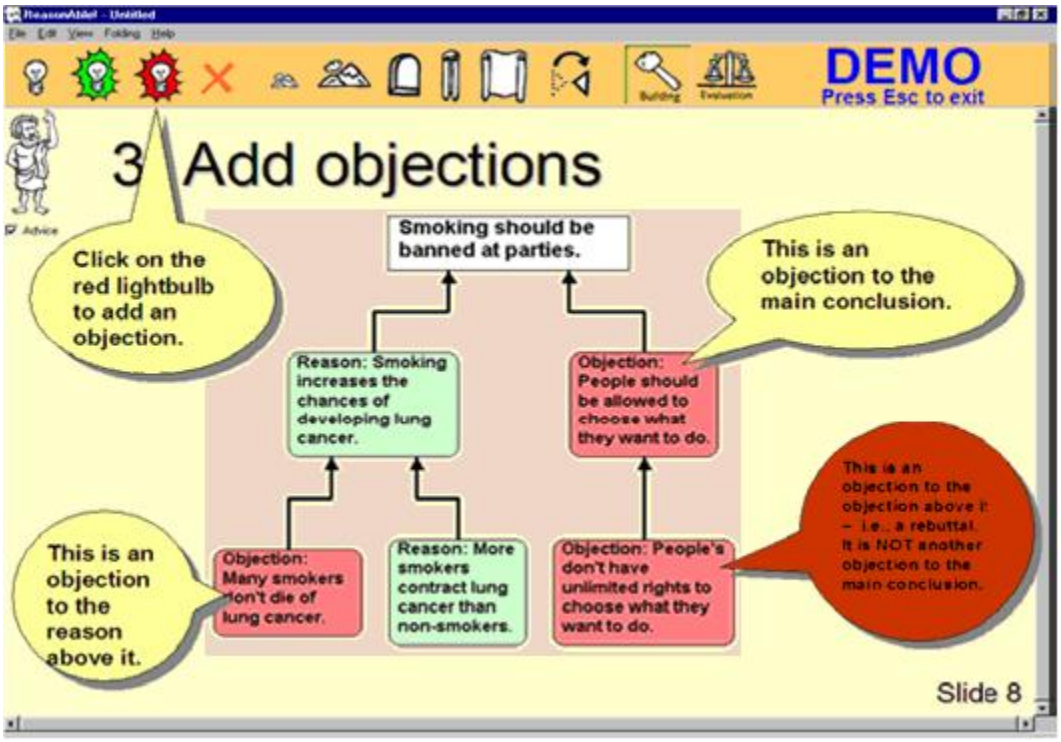
***Εικόνα 3.8**: Περιβάλλον εργασίας Belvedere. [Πηγή: http://lilt.ics.hawaii.edu/lilt/software/belvedere]*

Το **Reason!Able** [URL: Reason!Able] είναι ένα ηλεκτρονικό περιβάλλον που αναπτύχθηκε για να υποστηρίξει την επιχειρηματολογία σε παιδιά πρωτοβάθμιας εκπαίδευσης, για τον λόγο αυτό έχει και ιδιαίτερα ευχάριστο και απλό Interface.

Χρησιμοποιεί κόμβους για να αναπαραστήσει τη πορεία της επιχειρηματολογίας, χωρίς όμως αυτοί να διαφέρουν στο σχήμα παρά μόνο στο χρώμα. Οι ονομασίες των κόμβων παρουσιάζονται παρακάτω:

- Conclusion: ο αρχικός ισχυρισμός που απεικονίζεται με ένα λευκό ορθογώνιο.
- For/against: επιχειρήματα υπέρ/κατά μιας άποψης. Ένας πράσινος κόμβος συμβολίζει ένα επιχείρημα υπέρ ενός άλλου ή υπέρ της αρχικής άποψης και ένα κόκκινο ακριβός το αντίθετο.
- Premises: Επεξήγηση επιχειρήματος (πως το επιχείρημα βοηθάει στη θέση που εκφράστηκε).

Οι σύνδεσμοι που χρησιμοποιούνται υποδηλώνουν τις σχέσεις όπως αυτοί υποδεικνύονται από τους κόμβους. Ένα εύχρηστο χαρακτηριστικό του περιβάλλοντος είναι η δυνατότητα «Evaluation» που παρέχει, δηλαδή ένα επιχείρημα μπορεί να χρωματιστεί έντονα κόκκινο/πράσινο ανάλογα με το πόσο ισχυρό είναι.



Εικόνα 3.8: *Περιβάλλον εργασίας Reason!Able. [Πηγή: http://rationale.austhink.com/reasonable]*

Παρακάτω παρατίθεται ένας πίνακας όπου περιέχει αν όχι όλα, το μεγαλύτερο αριθμό των εργαλείων που χρησιμοποιούνται για την απεικόνιση επιχειρημάτων, σκέψης και διαλόγων όπως τα εντοπίσαμε ύστερα από έρευνα στο Web. Τα περισσότερα από αυτά τα εργαλεία του πίνακα

ήταν καταγεγραμμένα στο άρθρο «Computer-Supported Argumentation: A Review of the State of the Art» [Scheuer et al., 2010] του διεθνή περιοδικού «Computer Supported Collaborative Learning».

Εφαρμογή	Περιγραφή	Αναφορά
AcademicTalk	synchronous collaborative argumentation	McAlister et al., 2004 http://www.londonmet.ac.uk/ltri/research/projects/at.htm
Aquanet	collaborative, configurable ontology	Marshall et al., 1991
Araucaria	transcript, argument schemes, central database for argument exchange	Reed & Rowe, 2004 http://Araucaria.computing.dundee.ac.uk
Argdf	semantic ***Web-Based*** argumentation system	Zablith, 2007 http://argdf.org/
Argue / ArguMed	Argument assistance, legal domain	Verheij, 2003 http://www.ai.rug.nl/~verheij/aaa/argumed3.htm
Argumentum	***Web-Based*** argumentation system, Forum	http://arg.umentum.com/
ARGUNAUT	educational, support system for human moderators, used with Digalo	De Groot et al., 2007, McLaren et al., in press http://www.argunaut.org
ArguNet	collaborative, ***Web-Based***	Schneider et al., 2007 http://www.argunet.org
Athena	Educational, report generator, support reasoning and argumentation.	Rolf & Magnusson, 2002 http://www.athenasoft.org
AVER	Argument visualization for evidential reasoning, criminal investigations	van den Braak & Vreeswijk, 2006
AVERs	Criminal investigation	Bex et al., 2007
bCisive 2	Thinking, planning & presenting	http://bcisive.austhink.com/
bCisive online	decision-making and planning, ***Web-Based***	http://www.bcisiveonline.com
Belvedere	educational, collaborative, multiple views, scientific/evidential reasoning	Suthers et al., 1995; Suthers et al., 2001; Suthers, 2003 http://lilt.ics.hawaii.edu/lilt/software/belvedere
Carneades	support of multiple proof standards, IBIS	Gordon et al., 2007 http://carneades.berlios.de
ClaiMaker / ClaimFinder	Provides a forms-based user interface for modeling claims, and discovery services for searching/filtering/visualizing the database. ***Web-Based***	Buckingham Shum et al., 2003 http://kmi.open.ac.uk/projects/scholonto
CoChemEx	educational, collaborative, inquiry learning, chemistry, scripted	Tsovaltzi et al., in press
Cohere	***Web-Based*** visual tool to create, connect and share Ideas.	http://cohere.open.ac.uk/index.php
Collaboratorium	collaborative, IBIS	Klein & Iandoli, 2008 [1]; Malone & Klein, 2007 http://cci.mit.edu/research/climate.html
Compendium	successor of Questmap, collaborative, IBIS	Buckingham Shum et al., 2006, Okada & Buckingham Shum, 2008

		http://compendium.open.ac.uk
Convince Me	educational, model of coherent reasoning	Ranney and Schank, 1998 http://www.soe.berkeley.edu/~schank/convinceme
CoPe_it!	Successor of Hermes, (also) educational, collaborative, multiple views, support of multiple proof-standards, decision support, IBIS. ***Web-based***	Karacapilidis et al., 2009 http://copeit.cti.gr
DebateGraph	collaborative, local views, ***Web-Based***	http://www.debategraph.org
Debatepedia	Collaborative, Wiki-Based	http://wiki.idebate.org
Digalo	educational,collaborative, configurable ontology	Schwarz and Glassner, 2007 http://www.dunes.gr
Hermes	collaborative, support of multiple proof-standards, decision support, IBIS, ***Web-Based***	Karacapilidis & Papadias, 2001 http://www-sop.inria.fr/aid/hermes
IBIS/gIBIS	collaborative, notational support to solve wicked problems	Conklin & Begeman, 1988
iLogos	educational, argument mapping, causal diagrams	Easterday et al., 2007 http://www.phil.cmu.edu/projects/argument_mapping
Interloc	successor of AcademicTalk, educational, collaborative, sentence openers, configurable dialogue games	Ravenscroft et al., 2008 http://www.interloc.org
KIE/SenseMaker, WISE	Educational, collaborative, container visualization, inquiry learning, science learning	Bell, 1997; Bell & Linn, 2000, Linn et al., 2003 http://tels.sourceforge.net/sensemaker
LARGO	Educational, legal argumentation, ITS	Pinkwart et al., 2006
Legalese	legal argumentation	Hair, 1991
OVA	Online visualization of argument, ***Web-Based***	http://ova.computing.dundee.ac.uk/
Questmap	collaborative, IBIS	Carr, 2003
Rationale	educational, multiple argument modes	Van Gelder, 2007 http://rationale.austhink.com
Reason!Able	educational	Van Gelder, 2002, Van Gelder, 2003 http://rationale.austhink.com/reasonable
Room 5	collaborative, legal argumentation, implements dialogue game	Loui et al., 1997
SEAS	decision support, argument templates; table, starburst and constellation depictions of multidimensional arguments	Lowrance, 2007; Lowrance et al., 2008 http://www.ai.sri.com/~seas
TC3	educational, collaborative, tool suite to support collaborative writing of argumentative texts	Munneke et al., 2003
Zeno	Predecessor of Carneades and Hermes, support of multiple proof-standards, decision support, IBIS	Gordon & Karacapilidis, 1997

Πίνακας 3.1: *Εργαλεία - εφαρμογές υποστήριξης επιχειρηματολογίας (computer-supported)*

Σημείωση: ***---*** *Web-Based* ***---*** *εκτός λειτουργίας*

3.2.1 Συστήματα Υποστήριξης Επιχειρηματολογίας στο Web

Τα εργαλεία-εφαρμογές Υποστήριξης Επιχειρηματολογίας που μετά από εκτενή ερεύνα εντοπίσαμε ανέρχονται στον αριθμό τριάντα εννέα. Τα δέκα (με χρυσό χρωματισμό στον Πίνακα 3.1) από αυτά είναι κατασκευασμένα για on-line χρήση στον Παγκόσμιο Ιστό, από τα οποία αυτή τη στιγμή λειτουργούν τα οχτώ. Μερικά από αυτά είναι:

Το ArgDF [URL: ArgDF] είναι ένα σύστημα βασισμένο στον σημασιολογικό ιστό (Semantic Web) και δομημένο στην κορυφή της οντολογίας AIF-RDF. Το σύστημα ArgDF επιτρέπει στους χρήστες να δημιουργούν και να ρωτούν τα σημασιολογικά σχολιασμένα επιχειρήματα στον Παγκόσμιο Ιστό χρησιμοποιώντας διαφορετικά σχέδια επιχειρηματολογίας. Το σύστημα επιτρέπει επίσης στους χρήστες να χειριστούν τα επιχειρήματα από την επίθεση (Attacking) ή να υποστηρίζουν μέρη των υφιστάμενων επιχειρημάτων, και χρησιμοποιεί τα υπάρχοντα μέρη ενός επιχειρήματος για τη δημιουργία νέων επιχειρημάτων.

Επίσης το ArgDF προσφέρει ευέλικτες λειτουργίες, όπως την ικανότητα να δημιουργείς νέα σχέδια επιχειρηματολογίας από τη διεπαφή του χρήστη. Έτσι, ArgDF είναι μια ανοιχτή πλατφόρμα όχι μόνο για την αναπαράσταση των επιχειρημάτων, αλλά και για την οικοδόμηση αλληλένδετων και συνεχώς εξελισσόμενων δικτύων επιχειρημάτων.

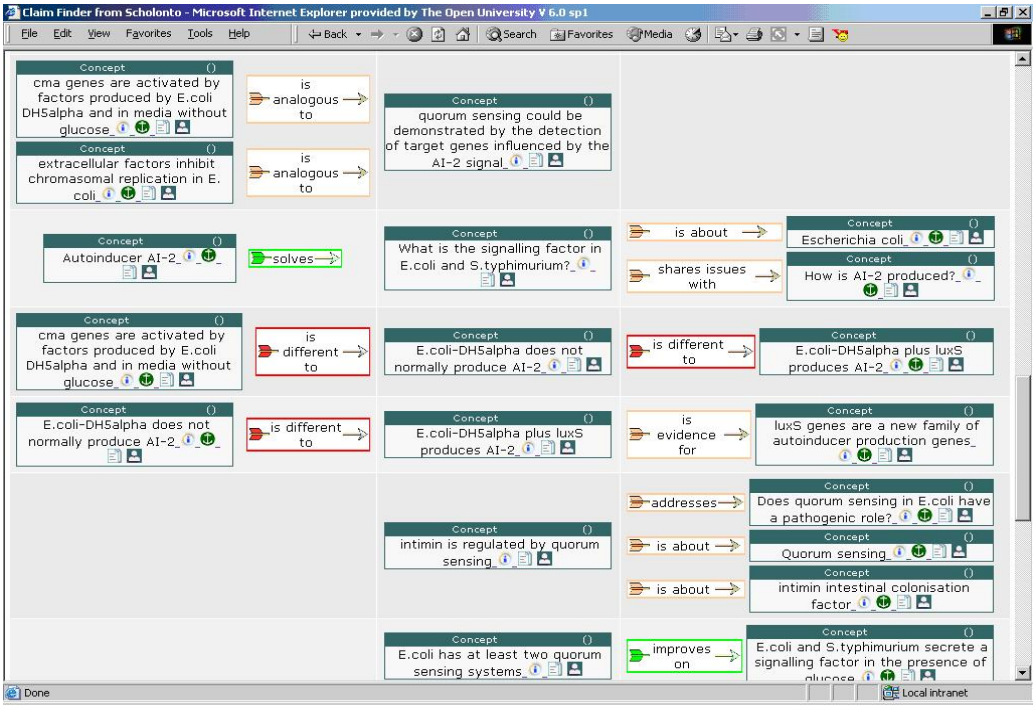


***Εικόνα 3.9**: Παράδειγμα στο ArgDF. [Πηγή: Fouad Zablith, Master thesis, 2007]*

Το **ClaiMaker / ClaimFinder** [URL: ClaiMaker / ClaimFinder] είναι ένα εργαλείο που βοηθάει στην έκφραση επιχειρημάτων και στην οπτικοποίηση ενός ισχυρισμού που υποστηρίζεται ή απορρίπτεται από δεδομένα, υποθέσεις κλπ. Για να γίνει αυτό χρησιμοποίει τρία είδη κόμβων:

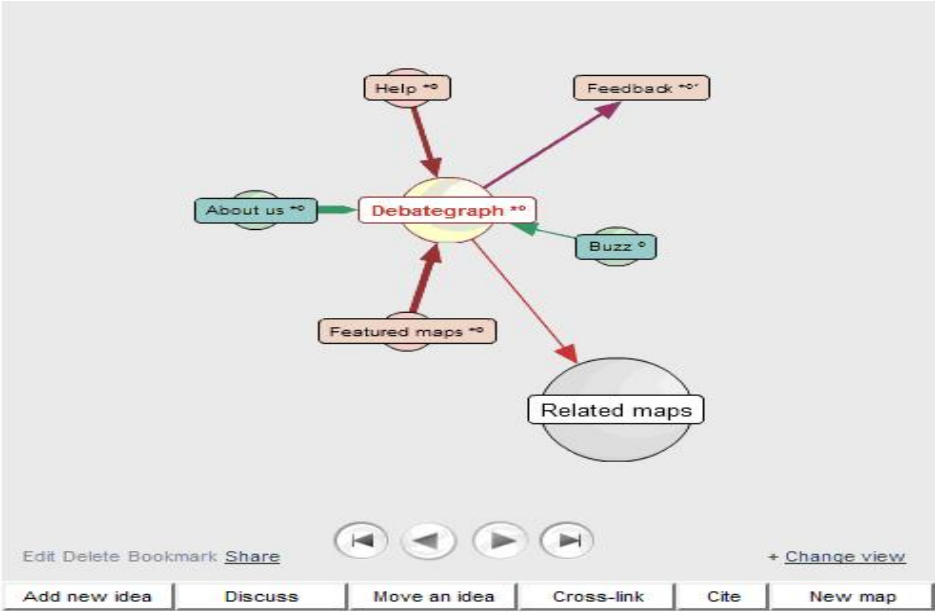
- Ισχυρισμός (Claim): είναι η αρχική θέση που τίθεται προς συζήτηση, αναφέρεται συνήθως σε κάποιο επιστημονικό έγγραφο και περιέχει στοιχεία που χαρακτηρίζουν αυτό αλλά και συνδέεται με αυτό μέσω URL. Σηματοδοτείται με μοβ χρώμα.
- Σχόλιο (Concept): περιέχει σχόλιο του χρήστη, με τίτλο, περίληψη, τύπο (μπορεί ένα σχόλιο να είναι απόδειξη, ανάλυση, δεδομένα, μεθοδολογία, άποψη, πρόβλημα κ.α.) αλλά και προαιρετική σύνδεση σε πηγή που το υποστηρίζει. Σηματοδοτείται με μπλε χρώμα.
- Sets: Σύνολο από Concepts που έχουν κάτι κοινό. Σηματοδοτείται με κόκκινο χρώμα.

Κάθε κόμβος παρέχει μια σειρά από μεταδεδομένα που εμφανίζονται με τη μορφή εικονιδίων. Στις ακμές των κόμβων μπορούν να χρησιμοποιηθούν ήδη προκαθορισμένες λέξεις που εκφράζουν αντίθεση, υποστήριξη, σχολιασμό κλπ. αλλά δίνεται η δυνατότητα να δημιουργηθεί και σύνδεσμος με οποιοδήποτε χαρακτηρισμό που να εξυπηρετεί τις ανάγκες του χρήστη και της εκάστοτε εργασίας.



***Εικόνα 3.10**: Παράδειγμα στο ClaimFinder. [Πηγή: Buckingham Shum, Stanford HCI Seminar, 2004]*

Το **Debategraph** [URL: Debategraph] είναι ένα Web-Based εργαλείο, το οποίο προσφέρει την δυνατότητα στους χρήστες να προβάλουν τα χαρακτηριστικά ενός διαλόγου είτε σε μορφή κειμένου, είτε σε αλληλεπιδραστική, δυναμική μορφή, μέσω δηλαδή ενός «χάρτη διαβούλευσης» (Debate Map). Ανάμεσα στις δυνατότητες που προσφέρει στους χρήστες είναι να δημιουργούν διαλόγους και να προσθέτουν νέα επιχειρήματα, τα οποία μπορούν και να φιλτράρουν ή να βαθμολογήσουν, να δημοσιοποιούν τους διαλόγους, να τους διαμοιράζονται και να τους επαναχρησιμοποιούν, αλλά και να παρουσιάζουν την εξέλιξη των διαλόγων με τροφοδοσία RSS. Επιπλέον, ιδιαίτερο χαρακτηριστικό είναι ότι μέσω του «χάρτη διαβούλευσης» παρουσιάζονται στον χρήστη όχι μόνο όσες ιδέες έχουν παρατεθεί, αλλά και η σχέση που υπάρχει ανάμεσά τους.



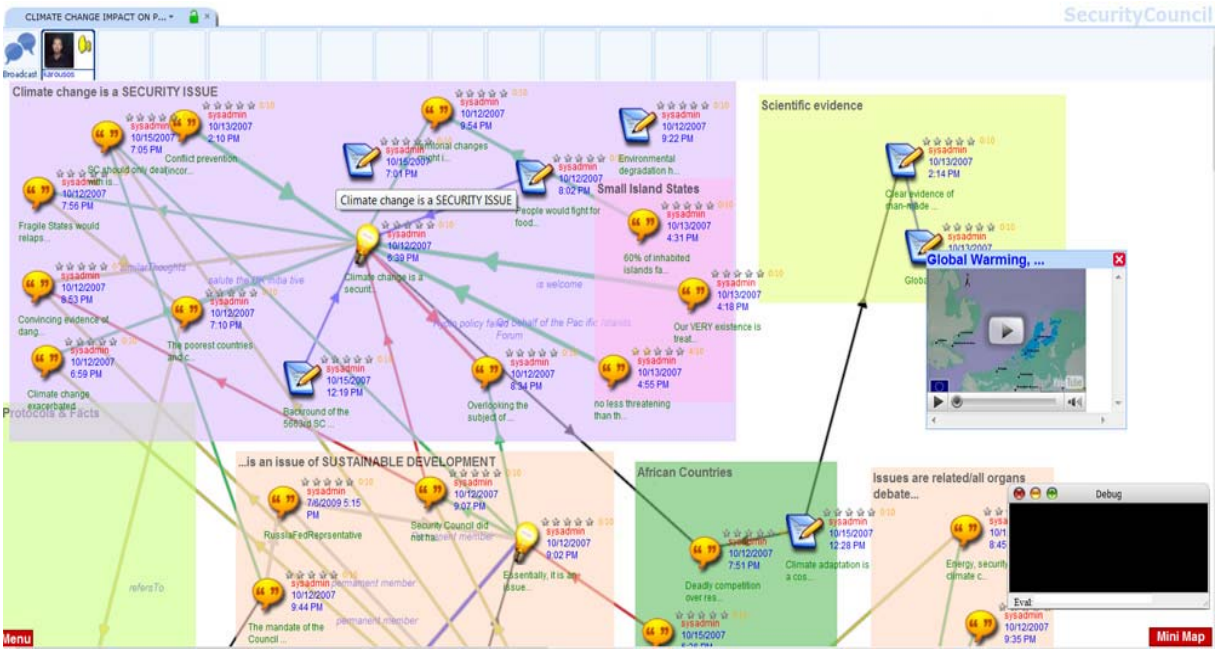
***Εικόνα 3.11**: Βασική προβολή χώρου εργασίας στο Debategraph. [Πηγή: http://debategraph.org]*

Το **CoPe_it!** [URL: CoPe_it!] είναι ένα Web-based εργαλείο το οποίο υποστηρίζει την συνεργασία με επιχειρήματα. Αποτελεί ένα πρωτότυπο λογισμικό φτιαγμένο από το EA-ITY στα πλαίσια του ευρωπαϊκού ερευνητικού έργου «Palette» (IST).

Κυριότερα χαρακτηριστικά του CoPe_it! αποτελούν, η συλλογή και η επεξεργασία πληροφορίας, η υποστήριξη σύγχρονου και ασύγχρονου διαλόγου, η βαθμιαία τυποποίηση πληροφορίας και η υποστήριξη λήψης απόφασης.

Βασικές δομικές μονάδες του συστήματος αποτελούν τα Μέλη (Members), οι Κοινότητες (Communities), οι Χώροι Εργασίας (Workspaces) και τα Γνωστικά Αντικείμενα (Items). Πιο συγκεκριμένα το «Community» είναι μία κοινότητα μελών, που συνήθως έχουν κοινά ενδιαφέροντα (π.χ. κοινότητα για Mobile Internet), μέσα στην οποία δημιουργούνται χώροι εργασίας "Workspaces", με συγκεκριμένη θεματολογία (π.χ. «ο καλύτερος Mobile Browser»), ώστε να μπορεί κάθε εγγεγραμμένο μέλος στον συγκεκριμένο χώρο να ανταλλάσει απόψεις με άλλα μέλη και να βοηθηθεί μέσω επιχειρημάτων να λάβει μια απόφαση. Η διαδικασία ανταλλαγής απόψεων, επιχειρημάτων γίνεται με τη δημιουργία διάφορων γνωστικών αντικειμένων (items), όπως είναι μια Ιδέα (Idea), μια Σημείωση (Note), ένα Σχόλιο (Comment), τα οποία μπορούν να είναι σε μορφή κειμένου (Text), URL και ενός αρχείου (File). Οποιοδήποτε μέλος μπορεί να δημιουργήσει ένα ή παραπάνω «Communities» και αντίστοιχα ένα ή παραπάνω «Workspaces». Το μέλος-δημιουργός έχει τη δυνατότητα να επιλέξει αν το «Workspace» του θα είναι ανοιχτό προς όλους ή θα απαιτείται έγκριση από τον ίδιο, για να μπορεί ένα άλλο μέλος να συμμετέχει στο συγκεκριμένο «Workspace».

Εκτός από την ανταλλαγή απόψεων και τη βοήθεια στη λήψη μιας απόφασης, το CoPe_it! μπορεί να χρησιμοποιηθεί και για τη δημιουργία προσωπικού χώρου συλλογής, αποθήκευσης και οργάνωσης δεδομένων (σκέψεων, αναφορών, εγγράφων και αρχείων γενικότερα) που θα μπορούμε να βλέπουμε και να επεξεργαζόμαστε μόνο εμείς.



***Εικόνα 3.12**: Βασική προβολή χώρου εργασίας CoPe_it! [Πηγή: http://copeit.cti.gr].*

Το σύστημα CoPe_it! αλλά και αναλυτικές πληροφορίες για το σχεδιασμό και λειτουργία του θα αναλυθούν εκτενέστερα στο παρακάτω κεφάλαιο.

Το **bCisive Online** [URL: bCisive Online] είναι ένα Web-Based εργαλείο οπτικοποίησης που υποστηρίζει συνεργασία σε πραγματικό χρόνο, λήψη αποφάσεων και σχεδιασμό, ομαδική επίλυση προβλημάτων και επιτακτικές παρουσιάσεις. Αποτελεί τον online διάδοχο των bCisive 2 (desktop) [URL: bCisive 2] και Rationale [URL: Rationale], έχοντας δημιουργούς τους Daniel Prager και Ben Loft της Austhink Software.

Η ομαδική επίλυση προβλημάτων μέσω του bCisive Online επιτυγχάνεται κάνοντας ένα βήμα παραπέρα από τον επιμερισμό οθόνης και τους εικονικούς πίνακες με μεγαλύτερη αποτελεσματικότητα, διευκολύνοντας τις τηλεδιασκέψεις. Συγκριμένα:

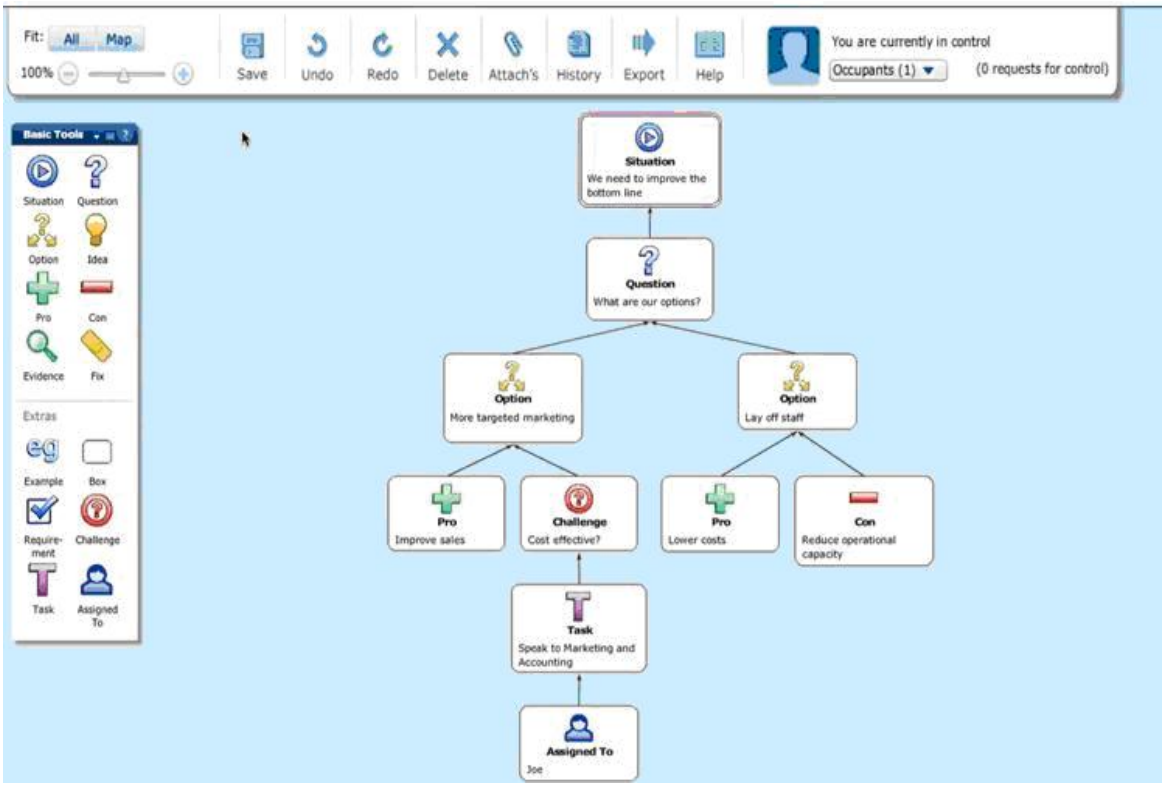
- Στην αίθουσα ή σε ολόκληρο τον κόσμο, ο καθένας βλέπει τον ίδιο χώρο εργασίας
- Ένα άτομο έχει τον έλεγχο σε μία χρονική στιγμή, προσθέτοντας νέα κουτιά και συνδέσεις.
- Άλλοι συμμετέχοντες μπορούν να χρησιμοποιούν τους εικονικούς τους δείκτες, αξιοποιώντας καλύτερα τα σημεία ή κάνοντας ερωτήσεις.
- Τα οπτικά εφέ είναι αποθηκευμένα στο Διαδίκτυο για εύκολη πρόσβαση και αναθεώρηση

Η λήψη αποφάσεων και ο σχεδιασμός υποστηρίζεται στο συγκεκριμένο online εργαλείο, μέσα από τα παρακάτω χαρακτηριστικά:

- Μια κοινή οπτική γλώσσα προωθεί μια κοινή αντίληψη και βοηθά στην εξάλειψη του εγώ και των πολιτικών.
- Τα σημεία του κάθε χρήστη σημειώνονται γρήγορα και βρίσκουν τη θέση τους στην «μεγάλη εικόνα».
- Υπάρχουν πολλά από τα οφέλη των Mind-Mapping, αλλά είναι πιο εύκολο να τα ακολουθήσουν
- Δυνατότητα ανεβάσματος αρχείων ως υποστηρικτικό υλικό από τον υπολογιστή στο bCisive on-line και δημιουργίας συνδέσμων προς άλλους δικτυακούς τόπους

Η δημιουργία επιτακτικών (που καθηλώνουν τον θεατή) παρουσιάσεων γίνεται πιο εύκολη μέσα από τις παρακάτω δυνατότητες που δίνει το bCisive Online:

- Δυνατότητα αποκοπής και εξαγωγής ολοκληρωμένων χαρτών ώστε να εισαχθούν στο PowerPoint.
- Εκτός από τις κουκκίδες προσθέτει και ενδιαφέρον οπτικό υλικό κάνοντας πιο σαφή και πειστική τη παρουσίαση.



***Εικόνα 3.12**: Βασική προβολή χώρου εργασίας bCisive Online. [Πηγή: http://www.bcisiveonline.com]*

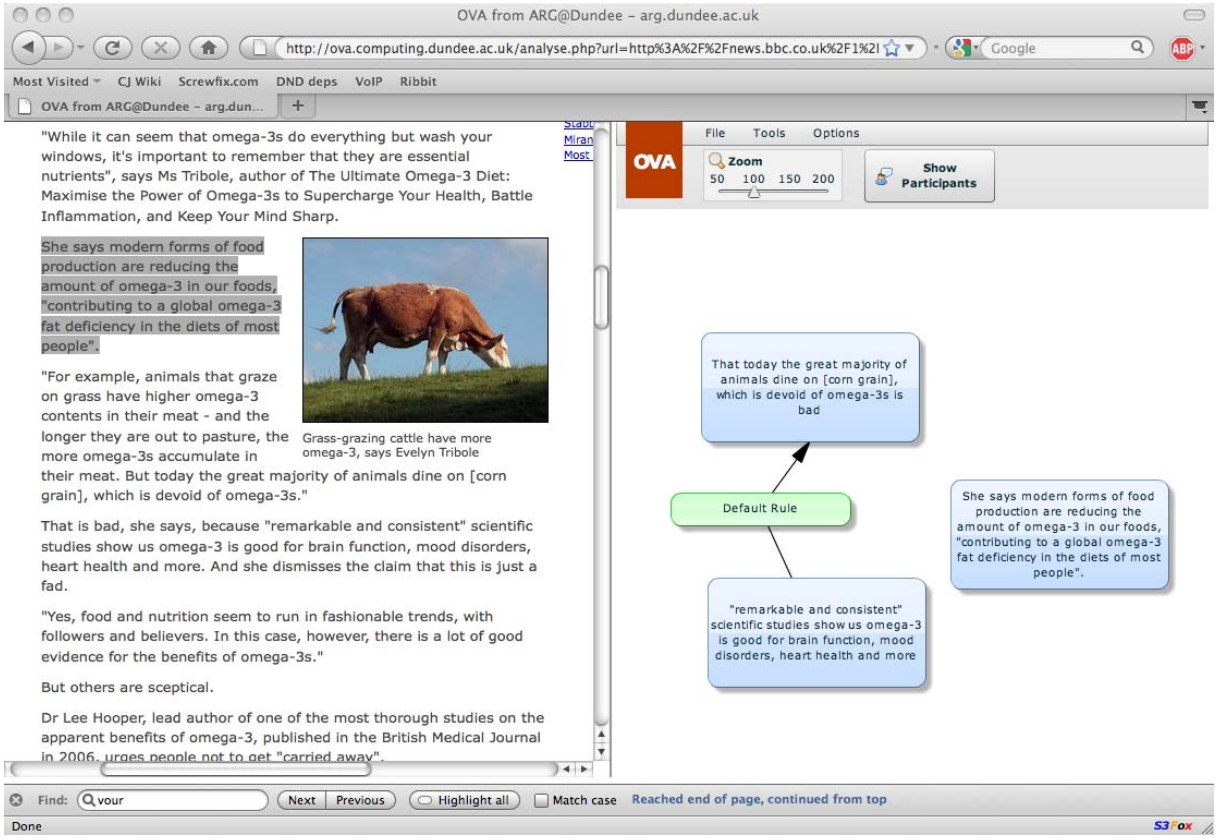
Τέλος αξίζει να αναφέρουμε πως από τεχνολογικής άποψης το συγκεκριμένο Web-Based εργαλείο χρησιμοποιεί Flash/Flex και JavaScript/jQuery για το Front-End, και Python / Django για το Back-End.

Το OVA (Online Visualizations of Argument) [URL: OVA] δημιουργήθηκε στο πανεπιστήμιο του Dundee και αποτελείται από μια σειρά Web-Based εργαλείων που σχεδιαστήκαν έχοντας ως στόχο να διευκολύνουν την ανάλυση και τον χειρισμό επιχειρημάτων σε online περιβάλλοντα, δίνοντας ιδιαίτερη έμφαση στις διαγραμματικές απεικονίσεις. Υπάρχουν διάφορα μέρη:

- **OVA**: Παρέχει ένα «drag and drop» περιβάλλον για ανάλυση επιχειρημάτων που είναι σε μορφή κειμένου. Θυμίζει το Araucaria Software αλλά αρκετά πιο απλοποιημένο. Είναι σχεδιασμένο να λειτουργεί με ιστοσελίδες, τρέχοντας από ένα Online Flash Widget και όχι ως αυτόνομο πρόγραμμα. Επίσης εγγενώς μπορεί να χειριστεί τις AIF (Argument Interchange Format) [Chesnevar et al., 2007] δομές.

- **OVAview**: Είναι ένας μετατροπέας βασισμένος στο CSS και στο Flash (CSS/Flash Based) που μπορεί να χρησιμοποιηθεί για τη μετατροπή αυθαίρετων AIF δομών σε γραφικές και βασισμένες σε κείμενο περιλήψεις.

- **OVAgen**: είναι ένα εύχρηστο online gizmo για τη σύνταξη περίληψης πλαισίων επιχειρηματολογίας (που είναι απλά διαγράμματα) και αποθηκεύει το αποτέλεσμα είτε σε κάθε AIF είτε σε γλώσσα DOT (για ομαλή ένταξη σε LaTeX ή χρήση σε GraphViz).



***Εικόνα 3.13**: Βασική προβολή χώρου εργασίας OVA. [Πηγή: http://www.arg.dundee.ac.uk]*

3.2.2 Συστήματα υποστήριξης επιχειρηματολογίας στο Mobile-Web

Σύμφωνα με την έρευνα που κάναμε, κανένα από τα προαναφερθέντα εργαλεία δεν διαθέτει έκδοση για κινητά είτε μέσω ενός Mobile Site, είτε μέσω μιας Mobile εφαρμογής. Παρόλο τη ραγδαία ανάπτυξη του Web και κατ' επέκταση του Mobile Web και την ανάπτυξη εκατομμυρίων εφαρμογών δεν υπάρχει ακόμα κάποια ολοκληρωμένη προσπάθεια Mobile Web-Based εφαρμογής υποστήριξης συστημάτων επιχειρηματολογίας.

Στο χώρο του Mobile Web συναντήσαμε διάφορες εφαρμογές σύγχρονου (Chat, Messengers) και ασύγχρονου (Forums) διαλόγου αλλά καμία εφαρμογή δομημένου διαλόγου με τη χρησιμοποίηση επιχειρημάτων και εργαλείων υποστήριξης αυτών. Μία καλή προσπάθεια που μπορεί να θεωρηθεί παρεμφερής με το γνωστικό αντικείμενο των συστημάτων υποστήριξης επιχειρηματολογίας έχει γίνει με την εφαρμογή Mindmeister μέσω του iPhone App που υπάρχει διαθέσιμη στο App Store για iPhone, iPod και iPad [URL: MindMeister in iTunes]. Στην συγκεκριμένη εφαρμογή γίνεται on-line και off-line καταγραφή των σκέψεων (μπορεί και επιχειρημάτων) μέσω ενός Mind-Map, που ο χρήστης-δημιουργός έχει τη δυνατότητα να μοιράζεται τις σκέψεις του, αλλά και να δέχεται και εξωγενής παρεμβάσεις από χρήστες που ο ίδιος επιθυμεί. Αναλυτικότερα παρακάτω παρουσιάζεται η συγκεκριμένη εφαρμογή:

MindMeister iPhone App

Το MindMeister [URL: Mindmeister for iPhone] φέρνει την έννοια της χαρτογράφησης μυαλού (Mind Mapping) στο Web, χρησιμοποιώντας τις δυνατότητες του για συνεργασία σε πραγματικό χρόνο και επιτρέποντας πραγματικά παγκόσμια συνεδρίες «καταιγισμού» ιδεών. Οι χρήστες μπορούν on-line να δημιουργούν, να διαχειρίζονται, να μοιράζονται Mind-Maps και να έχουν πρόσβαση σε αυτούς οποιαδήποτε στιγμή από οπουδήποτε. Στο «brainstorming mode», μπορούν να εργάζονται ταυτόχρονα διάφοροι χρήστες στο ίδιο Mind-Map και οι αλλαγές να εμφανίζονται άμεσα, όποτε αυτές συμβαίνουν.

Οι τρείς βασικότερες λειτουργίες του MindMeister App είναι οι εξής:

- Δημιουργία Mind-Map: Επιτυγχάνεται γρήγορη δημιουργία Mind-Map με on-line πρόσβαση ανεξαρτήτου χώρου και χρόνου.

- Διαμοιρασμός Mind-Map: Επιτυγχάνεται άμεσος διαμοιρασμός οποιουδήποτε Mind-Map στους φίλους και τους συναδέλφους. Ο χρήστης-δημιουργός μπορεί να προσκαλέσει κάποιον (μέσω e-mail) και ανάλογα με τι πρόσβαση του δίνετε θα είναι σε θέση είτε να συμβάλλει ή απλά να δει τον χάρτη. Για διαμοιρασμό ενός χάρτη με ολόκληρο τον κόσμο, απλά χρειάζεται ένα κλικ στο κουμπί 'Δημοσίευση' και ο χάρτης θα ενσωματωθεί στο ιστολόγιο ή την ιστοσελίδα της αρεσκείας μας.

- Συνεργασία σε πραγματικό χρόνο: Όταν δύο ή περισσότεροι χρήστες ανοίγουν τον ίδιο χάρτη την ίδια στιγμή εισέρχονται σε λειτουργία «brainstorming». Κάθε αλλαγή που κάνει ο Mobile χρήστης και σώζεται θα εμφανίζεται αμέσως στις οθόνες των άλλων Web χρηστών μέσω του κεντρικού server του MindMeister, χωρίς να χρειάζεται ανανέωση (reload).



***Εικόνα 3.14**: Βασική προβολή χώρου εργασίας MindMeister iPhone App. [Πηγή: iPhone App]*

ΚΕΦΑΛΑΙΟ 4: Υλοποιώντας μια Mobile Εφαρμογή Υποστήριξης Επιχειρηματολογίας

Μέχρι σήμερα οι προσπάθειες υποστήριξης συζητήσεων με επιχειρήματα μέσω κινητών τηλεφώνων είναι ελάχιστες. Σε αντίθεση με διάφορες εφαρμογές κινητών τηλεφώνων οι οποίες επιτρέπουν την «χαλαρή» (μη αυστηρώς δομημένη) επικοινωνία, η δυνατότητα αποστολής επιχειρημάτων σε αυστηρά δομημένους διαλόγους δεν υποστηρίζεται ως στιγμής.

Στο πλαίσιο της διπλωματικής εργασίας προσπαθήσαμε να εκμεταλλευτούμε τα πλεονεκτήματα που δίνει ένα κινητό τηλέφωνο, όσον αφορά στην ευχρηστία, στην ευκολία πρόσβασης και στη χρήση του Web2.0, δημιουργώντας μία Mobile εφαρμογή που θα παρέχει τη δυνατότητα σε χρήστες κινητού τηλεφώνου να συμμετέχουν σε ένα συγκεκριμένο σύστημα υποστήριξης επιχειρηματολογίας, το CoPe_it!. Με αυτόν τον τρόπο επιχειρούμε να συνεισφέρουμε στην προσπάθεια διεύρυνσης της χρήσης του συγκεκριμένου συστήματος.

Στο συγκεκριμένο κεφάλαιο παρουσιάζονται τα σταδία σχεδιασμού και ανάπτυξης του Web Service αλλά και της εφαρμογής για κινητά, αφού πρώτα δίδεται όλο το απαραίτητο θεωρητικό υπόβαθρο για την κατανόηση των πεπραγμένων. Έτσι, αρχικά γίνεται μία εκτενής παρουσίαση του CoPe_it! και έπειτα αναφέρονται οι απαιτήσεις και οι προδιαγραφές για την παρεχόμενη λειτουργικότητα της εφαρμογής μας. Κατόπιν παρουσιάζεται ο σχεδιασμός και η ανάπτυξη που έλαβε χώρα τόσο στη μεριά του συστήματος όσο και στην υλοποίηση της εφαρμογής, ενώ στη συνέχεια παρουσιάζεται η εφαρμογή κινητού που αναπτύχθηκε (λειτουργίες και τρόπο χρήσης). Τέλος αναφέρονται ζητήματα που μπορεί να αποτελέσουν μελλοντικές προκλήσεις για περαιτέρω συνέχιση της εργασίας μας.

4.1 Το σύστημα CoPe_it!

4.1.1 Γενικά

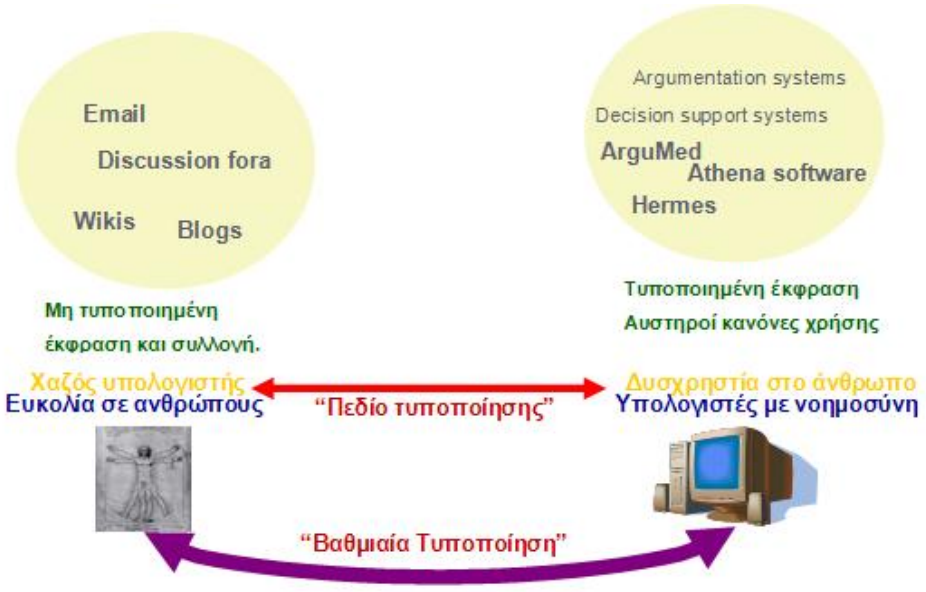
Το CoPe_it! [URL: CoPe_it!] [Karacapilidis et al., 2007] αποτελεί ένα εργαλείο του παγκόσμιου ιστού, που έχει ως στόχο την υποστήριξη και ενθάρρυνση συνεργατικών δραστηριοτήτων ανθρώπων, οργανωμένων σε κοινότητες ή κοινότητες πρακτικής (Communities of Practice – CoPs [Lave et al., 1991]). Παρέχει ένα συνεργατικό περιβάλλον, το οποίο ενθαρρύνει την συζήτηση μεταξύ των συμμετεχόντων, δίνοντας έμφαση στην παροχή διαφόρων τρόπων απεικόνισης (με απόψεις και προβολές) των συζητήσεων με επιχειρήματα, και στη χρήση μηχανισμών επιχειρηματολογικού λογισμού σε ένα αυστηρά τυπικό επίπεδο. Στους χρήστες του CoPe_it! παρέχετε η δυνατότητα συμμετοχής και δημιουργίας νέων κοινοτήτων, όπως επίσης και δυνατότητα συνεργασίας σε διάφορους (ιδιωτικούς ή δημόσιους) χώρους εργασίας (workspaces). Το σύστημα επιτρέπει στους χρήστες να οργανώνουν και να απεικονίζουν τις δικές τους προσεγγίσεις σε ενδιαφέροντα θέματα μέσα από μία διαρκώς εξελισσόμενη διαδικασία ανταλλαγής και δημιουργίας γνώσης. Τα μέλη του CoPe_it! μπορούν επίσης να στηρίζουν και να επεξεργάζονται τα επιχειρήματά τους σε διαφορετικά επίπεδα τυποποίησης: οι χρήστες μπορούν να ξεκινήσουν με ένα πολύ «ανεπίσημο» και μη αυστηρό τρόπο έκφρασης (που είναι κατεξοχήν κατανοητός από τους ανθρώπους) θέτοντας τις απόψεις τους και τα επιχειρήματά τους, ενώ μπορούν παράλληλα να αναδιαμορφώσουν τις ίδιες τις απόψεις τους, να τις αναδομήσουν συνδέοντάς τες με άλλες απόψεις (των ιδίων ή άλλων συμμετεχόντων), χρησιμοποιώντας μία σημασιολογία η οποία έπειτα θα είναι κατανοητή και από το σύστημα. Αυτή ακριβώς η ιδιότητα επιτρέπει την ενεργοποίηση μηχανισμών υποστήριξης λήψης αποφάσεων.

4.1.2 Θεωρητική τεκμηρίωση της λειτουργίας του συστήματος και μοντέλο δεδομένων

Η ιδέα της δημιουργίας ενός συστήματος υποστήριξης επιχειρηματολογίας σε συνεργατικό επίπεδο δεν είναι κάτι το καινούριο. Η συνεργατική επιχειρηματολογία έχει αποδειχθεί από ερευνητές ως αρκετά επιτυχημένη μέθοδος [Moor et al., 2006]. Οι συνεργατικές δράσεις (που έχουν ως στόχο την επίλυση ενός προβλήματος) των κοινοτήτων πρακτικής απλώνονται σε ένα μεγάλο εύρος, από την αρχική εύρεση και συλλογή πληροφορίας, στη δόμησή της, στο φιλτράρισμα επιλεγμένων πηγών και τέλος, στην αξιολόγηση της γνώσης και τη λήψη κάποιας

απόφασης. Έτσι, οι υποδομές και τα συστήματα για την υποστήριξη συνεργασίας με επιχειρήματα είναι αναγκαίο να μπορούν να υποστηρίζουν όλες τις πρακτικές και τους τρόπους συνεργασίας [Andriessen et al., 2003]. Όμως, η διαφορετική τυποποίηση της πληροφορίας που απαιτείται για την υποστήριξη κάθε διαφορετικού τρόπου συνεργασίας είναι ένας παράγοντας ιδιαιτέρα κρίσιμος και αποτελεί σοβαρή δυσκολία στην προσπάθεια αυτή.

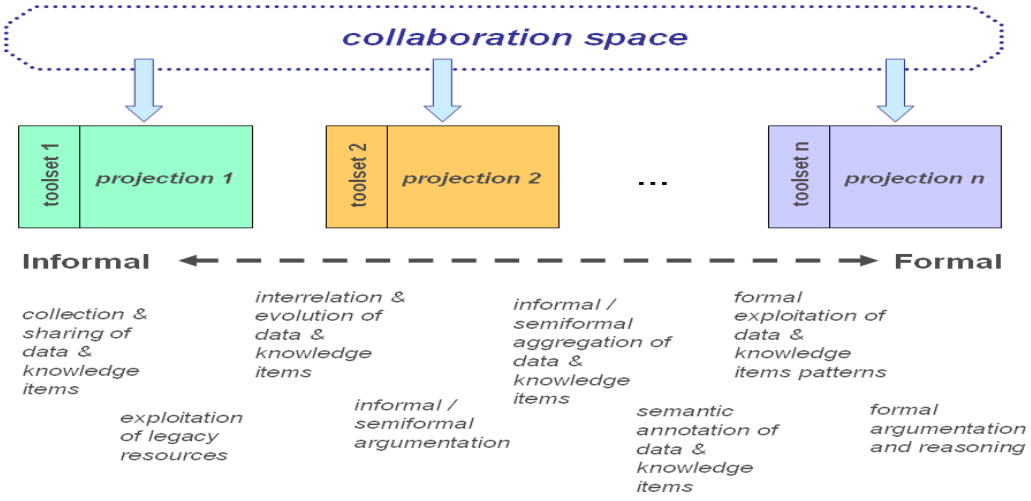
Για παράδειγμα, το επίπεδο τυποποίησης που απαιτείται να υπάρχει στα συστήματα που υποστηρίζουν την επιχειρηματολογία και τη λήψη απόφασης είναι υψηλό και ταυτόχρονα σταθερό - μη μεταβαλλόμενο. Το σταθερό επίπεδο της τυποποίησης στα συνεργατικά συστήματα όχι μόνο εμποδίζει την ευελιξία των συστημάτων σε σχέση με τις συνεχώς μεταβαλλόμενες ανάγκες των κοινοτήτων πρακτικής, αλλά αποτελεί και βασική αιτία ύπαρξης λαθών και ελλείψεων λόγω του υπερβολικού φόρτου επεξεργασίας πληροφοριών σε περιπτώσεις ελέγχου για τη συγκεκριμένη τυποποίηση που ακολουθείται [Marshall et al., 1997; Shipman et., 1993].



Εικόνα 4.1: *Βαθμιαία τυποποίηση στο CoPe_it! [Πηγή: Karacapilidis et al., 2009b]*

Το CoPe_it! επιχειρεί να προσεγγίσει αυτό το ζήτημα, μέσα από την υιοθέτηση ενός ευέλικτου τρόπου τυποποίησης έτσι ώστε να ικανοποιεί τις όποιες απαιτήσεις των κοινοτήτων πρακτικής. Η ποικιλία τυποποίησης που διαθέτει, δύναται να χρησιμοποιηθεί διαφορετικά, ανάλογα με τη συνεργατική δράση που λαμβάνει χώρα (π.χ. λήψη απόφασης, διεξαγωγή «χαλαρής» συζήτησης, επιχειρηματολογία, συλλογή), αλλά και με την επιλογή των ίδιων των συμμετεχόντων σε μία

συνεργασία. Αυτός ο ευέλικτος τρόπος τυποποίησης υλοποιείται από το σύστημα υιοθετώντας την προσέγγιση της βαθμιαίας τυποποίησης (Incremental Formalization), η οποία έχει προταθεί παλαιότερα από ερευνητές του υπερκειμένου [Shipman & McCall, 1994]. Η βαθμιαία τυποποίηση στο CoPe_it!, υποστηρίζεται με τη χρήση διαφορετικών προβολών πάνω στον ίδιο χώρο εργασίας κάθε μία από τις οποίες υποστηρίζει ένα συγκεκριμένο βαθμό τυποποίησης (Εικόνα 4.1). Οι διαφορετικές προοπτικές απεικόνισης των χώρων εργασίας εμπλέκουν ένα σύνολο από αφαιρετικά αντικείμενα που συνδυάζονται με τέτοιο τρόπο, ώστε να μπορέσουν να συνεισφέρουν στην εύρεση λύσης για ένα συγκεκριμένο πρόβλημα. Τα αφαιρετικά αντικείμενα μπορεί να είναι κόμβοι γνώσης, συσχετίσεις και δράσεις που υποστηρίζονται από τη συγκεκριμένη προβολή. Στο CoPe_it! είναι δυνατή η εναλλαγή προβολών στο χώρο εργασίας κατά τη διάρκεια της συνεργασίας. Έτσι σε κάθε εναλλαγή εκτελούνται ένα σύνολο από μετασχηματισμοί, που είναι ικανοί να απεικονίσουν το σύνολο (ή υποσύνολο) της δομής και των πληροφοριών ενός χώρου εργασίας σε μορφή που υποστηρίζεται από την εκάστοτε προβολή. Αυτοί οι μετασχηματισμοί βασίζονται σε κανόνες, οι οποίοι είναι δυνατόν να επαναπροσδιοριστούν ανάλογα με τις ανάγκες χρήσης.

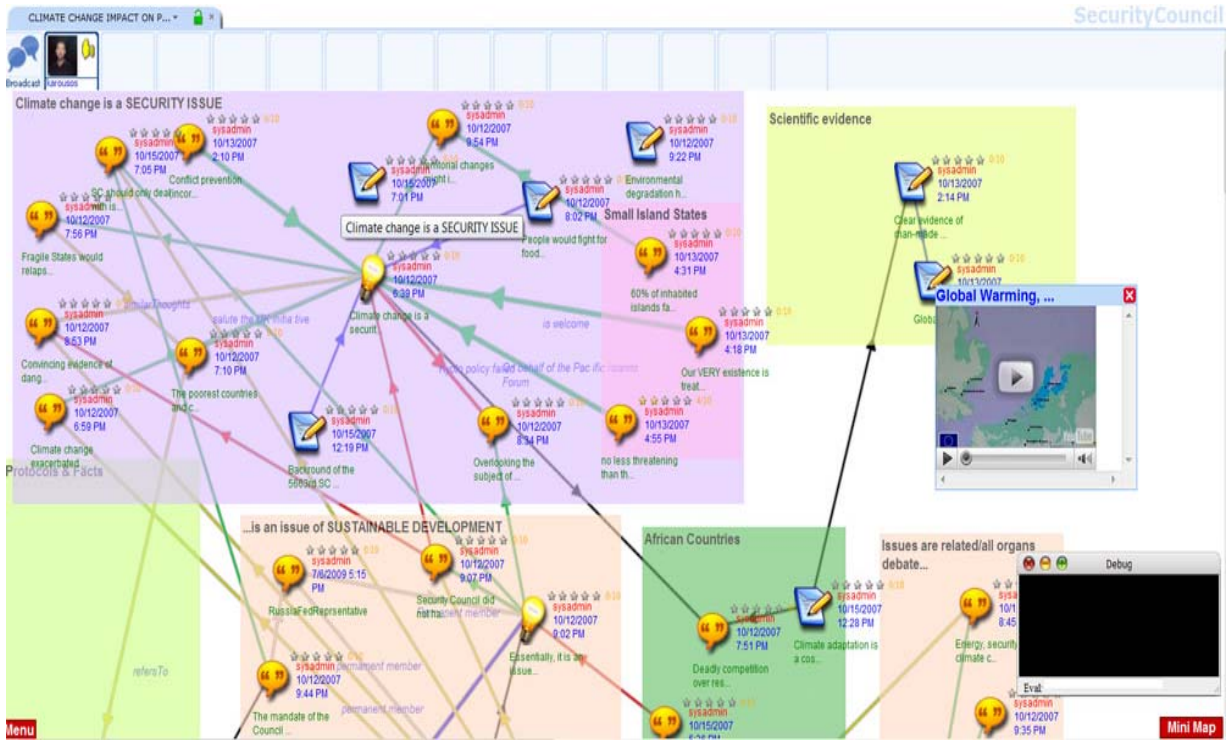


***Εικόνα 4.1**: Η εξέλιξη της τυποποίησης στο CoPe_it! με χρήση προβολών διαφορετικών σκοπών. [Πηγή: Karacapilidis et al., 2009b]*

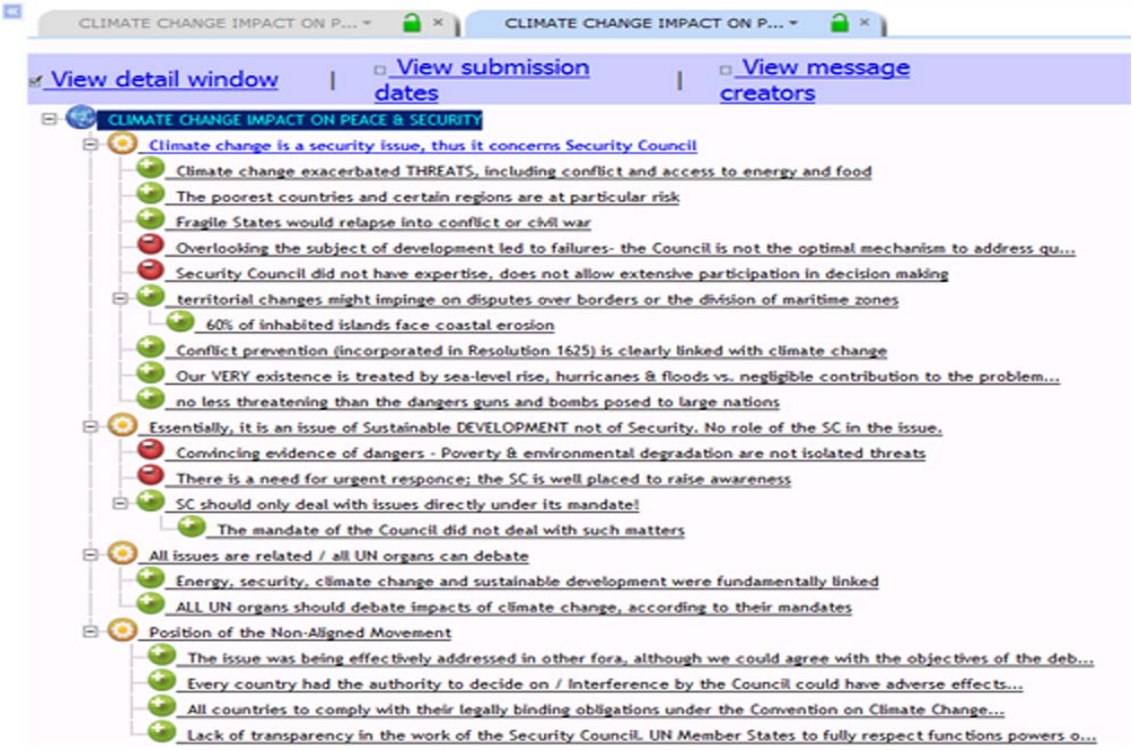
Κάθε μία από τις προβολές για ένα χώρο εργασίας παρέχει ένα μηχανισμό υποστήριξης συγκεκριμένου επιπέδου τυποποίησης. Για παράδειγμα (Εικόνα 4.2) η $1^{η}$ προβολή μπορεί να καλύπτει ανάγκες για τη συλλογή και διαμοίραση γνωστικών αντικειμένων, ενώ $n^{η}$ προβολή μπορεί να καλύπτει ένα σύνολο από λειτουργίες ικανές για υποστήριξη λήψης απόφασης. Όσο

πιο χαμηλό επίπεδο τυποποίησης υποστηρίζεται από μία προβολή τόσο πιο εύκολο στη χρήση αλλά και πιο κατανοητό προς τον άνθρωπο γίνεται. Αντίθετα, σε μία προβολή με χαμηλό επίπεδο τυποποίησης η συγκεντρωμένη γνώση θα είναι πολύ δύσκολο να επεξεργασθεί από τον υπολογιστή, έτσι ώστε να είναι δυνατή η υποστήριξη λήψης απόφασης, αφού τα υποστηριζόμενα αντικείμενα και συσχετίσεις παίρνουν πολύ γενική μορφή χωρίς καμία σημασιολογία. Τα αυστηρά επίπεδα τυποποίησης δυσκολεύουν τον άνθρωπο να εκφραστεί ελεύθερα, περιορίζοντας αρκετά τη δράση του, όμως επιτρέπουν στον υπολογιστή να κατανοεί ολόκληρο το χώρο εργασίας αφού κάθε αντικείμενο έχει πολύ συγκεκριμένη σημασιολογία κατανοητή από τον άνθρωπο αλλά κυρίως από τον υπολογιστή.

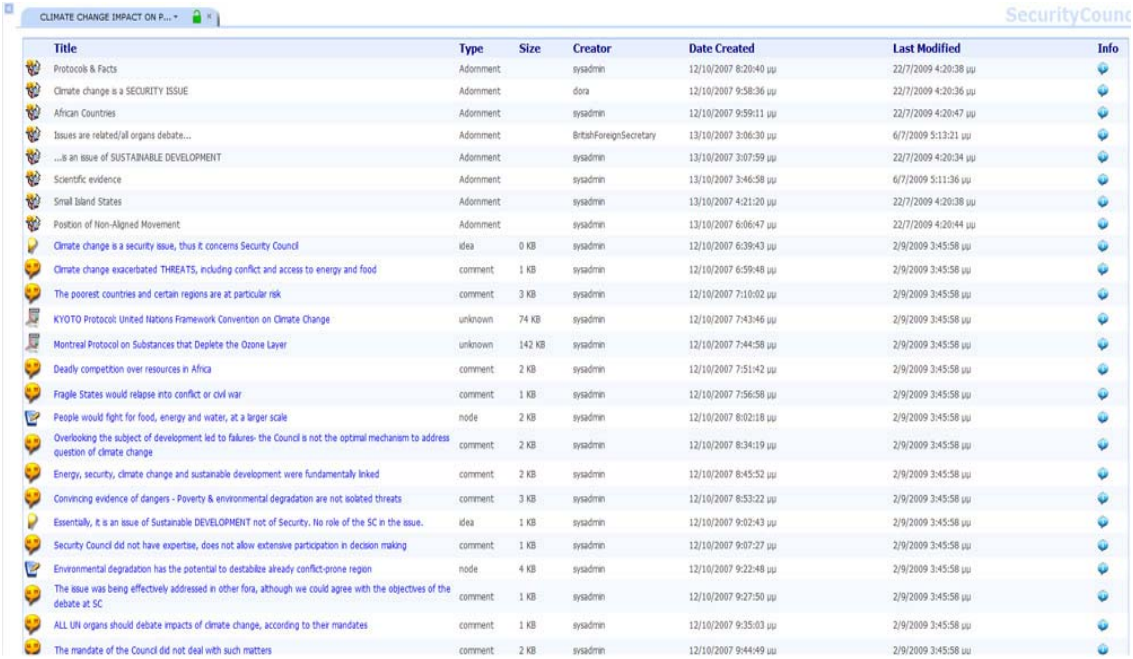
Όπως προκύπτει από τα παραπάνω, το CoPe_it! εστίασε στη δημιουργία πολλών προβολών (Εικόνες 4.3, 4.4, 4.5, 4.6, 4.7), έτσι ώστε να είναι δυνατή η υποστήριξη συνεργασίας των κοινοτήτων πρακτικής με κάθε δυνατό επίπεδο τυποποίησης χρησιμοποιώντας τεχνολογίες που επιτρέπουν τόσο «χαλαρή» (chat, mail, συλλογή και τοποθέτηση πληροφορίας, καταγραφή άποψης) όσο και «αυστηρή» (καταγραφή επιχειρημάτων, ρητή σύνδεση πληροφοριών και αυστηρή σημασιολογία) τυποποίηση.



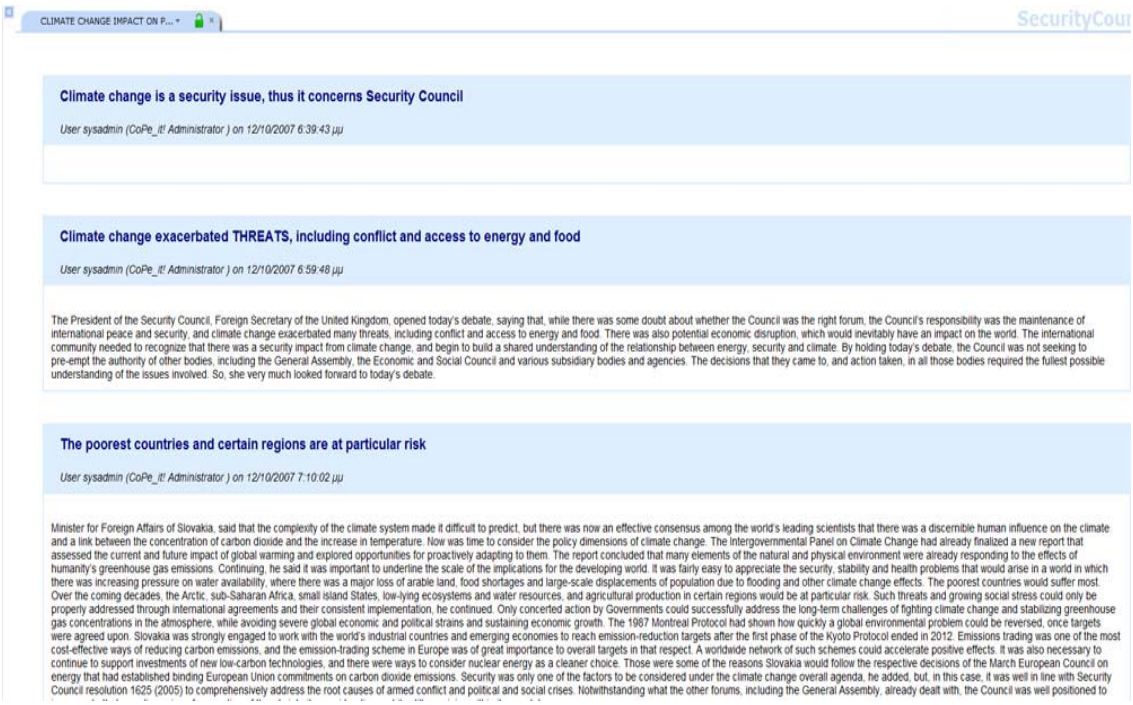
Σχήμα 4.3: *Βασική προβολή χώρου εργασίας: χαλαρή τυποποίηση, εργασία σε χώρο, σύγχρονη συνεργασία, δυνατότητα άμεσης επικοινωνίας. [Πηγή: http://copeit.cti.gr]*



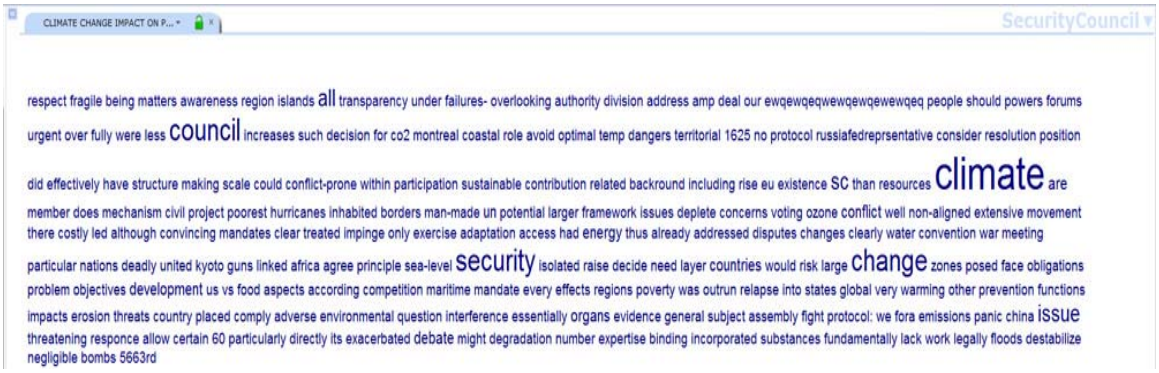
Εικόνα 4.4: *Προβολή αυστηρής τυποποίησης: υποστήριξη λήψης απόφασης, ασύγχρονη συνεργασία, δυνατότητα δημιουργίας αναφορών σκοπών. [Πηγή: Karacapilidis et al., 2009b]*



Εικόνα 4.5*: Προβολή Λίστας: Δυνατότητα ομαδοποίησης και εποπτικής απεικόνισης περιεχομένου σκοπών. [Πηγή: Karacapilidis et al., 2009b]*



Εικόνα 4.6: *Προβολή φόρουμ: Δυνατότητα ασύγχρονης εργασίας σε μορφή φόρουμ σκοπών. [Πηγή: Karacapilidis et al., 2009b]*



Εικόνα 4.7: *Προβολή tag-cloud: Δυνατότητα άμεσης απεικόνισης των λέξεων κλειδιών στο περιεχόμενο της συνεργασίας σκοπών. [Πηγή: Karacapilidis et al., 2009b]*

4.1.3 Χαρακτηριστικές λειτουργίες

Το CoPe_it!, ακολουθώντας τις προδιαγραφές που τέθηκαν στο πλαίσιο του συγκεκριμένου έργου, υποστηρίζει ένα σύνολο λειτουργιών που σχετίζονται με: χώρους συνεργασίας, διαφορετικές απεικονίσεις πληροφοριών, πρωτότυπες τεχνικές οπτικοποίησης δεδομένων, μηχανισμούς επιχειρηματολογίας, διαχείριση διαφορετικών τύπων αντικειμένων που

αποθηκεύονται στο σύστημα, δυνατότητα αναζήτησης, διαδικασίες λήψης απόφασης, πιστοποίηση, εξουσιοδότηση και διαχείριση χρηστών και κοινοτήτων, διαχείριση ρόλων κ.α. [Karacapilidis et al. 2009a]. Παρακάτω περιγράφονται οι χαρακτηριστικές λειτουργίες του συστήματος κατ' αντιστοίχιση με τις λειτουργικές προδιαγραφές του:

- Δημιουργία χώρων συνεργασίας - Το σύστημα υποστηρίζει τη δημιουργία χώρων συνεργασίας (collaboration spaces), στους οποίους μπορεί κανείς να αποθηκεύσει ιδέες και πληροφορίες, να καταγράψει συζητήσεις και προσωπικές σκέψεις, αλλά και να συσχετίσει τις πληροφορίες διαμορφώνοντας έτσι μία οπτική αναπαράσταση χρήσιμη για την κατανόηση όλου του χώρου και για τη διεξαγωγή αποφάσεων. Οι χώροι αυτοί μπορούν να χρησιμοποιηθούν από τα μέλη της κοινότητας είτε για προσωπική είτε για δημόσια χρήσης. Κάθε χρήστης έχει δικαίωμα να δημιουργήσει όσους χώρους επιθυμεί, αλλά και να μοιραστεί τις πληροφορίες στο χώρο με ένα ή περισσότερα πρόσωπα της επιλογής του.

- Μεταβολή αντικειμένων - Τα αντικείμενα, τα οποία εισάγονται στους χώρους εργασίας, μπορούν εύκολα να μεταβληθούν και ως προς τον τύπο τους και ως προς το περιεχόμενό τους. Η αφαιρετική τους ιδιότητα παρέχει τη δυνατότητα ανάλογα, με την προβολή εργασίας που έχει επιλέξει κάθε χρήστης, να μπορούν να μετασχηματίζονται (για παράδειγμα από μία απλή πληροφορία σε μία ιδέα και κατόπιν σε ένα ισχυρό επιχείρημα), να μετακινούνται στο χώρο, να συμμετέχουν σε συσχετίσεις, να ομαδοποιούνται σε σύνθετα αντικείμενα κλπ.

- Παροχή μονάδας αποθήκευσης - Ανεξάρτητα με τη επιλογή της προβολής ή ακόμα και του χώρου εργασίας το σύστημα παρέχει μία βασική μονάδα αποθήκευσης όλων των. Τα περιεχόμενα της μονάδας αποθήκευσης δομούνται σύμφωνα με μία συγκεκριμένη δομική αφαίρεση, έτσι ώστε, ανεξάρτητα του τύπου πληροφορίας που αποθηκεύεται, η μονάδα αποθήκευσης να χρησιμοποιεί πάντα τον ίδιο μηχανισμό αποθήκευσης και προσπέλασης στα δεδομένα. Με αυτόν τον τρόπο η μονάδα αποθήκευσης ανεξαρτητοποιείται από το τύπο και το περιεχόμενο των αντικειμένων που αποθηκεύονται, κάνοντας έτσι δυνατή τη δυναμική προσθήκη νέων τύπων πληροφορίας στο σύστημα χωρίς καμία αλλαγή στην αποθηκευτική συμπεριφορά.

- Αναζήτηση πληροφοριών - Η αναζήτηση των πληροφοριών υποστηρίζεται τόσο σε επίπεδο χώρου συνεργασίας όσο και σε σφαιρικό επίπεδο. Με τον τρόπο αυτό είναι δυνατή η αναζήτηση πληροφορίας από οποιοδήποτε δημόσιο χώρο συνεργασίας στο CoPe_it!

- Υποστήριξη βαθμιαίας τυποποίησης - Ανάλογα με την επιλεγμένη προβολή και τα επιλεγμένα εργαλεία για συνεργασία, το σύστημα υποστηρίζει τη βαθμιαία τυποποίηση και επιτρέπει στους χρήστες να συνεργάζονται είτε κάτω από αυστηρούς κανόνες είτε όχι. Ένα συνηθισμένο σενάριο εργασίας είναι αυτό κατά το οποίο οι χρήστες χρησιμοποιούν τη χωρική προβολή για να συλλέξουν πληροφορίες και να συζητήσουν σύγχρονα ή και ασύγχρονα σε πολύ χαμηλό επίπεδο τυποποίησης και κατόπιν να φιλτράρουν τις πληροφορίες και να τις συσχετίσουν. Ύστερα μπορούν να μεταβούν στην προβολή επιχειρηματολογίας και λήψης απόφασης, έτσι ώστε να συνεχίσουν τη συνεργασία τους σε πολύ αυστηρή μορφή (εισάγοντας μόνο επιχειρήματα και λύσεις) και, τέλος, να λάβουν μία απόφαση.

- Υποστήριξη λήψης απόφασης - Η διαδικασία υποστήριξης λήψης απόφασης διεξάγεται στην προβολή αυστηρής τυποποίησης για επιχειρηματολογία και υποστηρίζεται πλήρως από αντίστοιχο μηχανισμό, ο οποίος με βάση συγκεκριμένους αλγορίθμους μπορεί να αποφανθεί για την πιο «σωστή» λύση μέχρι στιγμής για κάποιο συγκεκριμένο πρόβλημα. Οι αλγόριθμοι αυτοί μπορούν να παραμετροποιηθούν ή και να αντικατασταθούν σε διαφορετικούς τύπους προβλημάτων (π.χ. ιατρικά θέματα, θέματα οικονομικής διαχείρισης κλπ).

- Παροχή επιπλέον μηχανισμού ενημέρωσης - Το σύστημα παρέχει επιπλέον μηχανισμό ενημέρωσης χρηστών και διαχειριστών για τις συνεργασίες που λαμβάνουν χώρο στο ίδιο το σύστημα, μηχανισμό παραμετροποίησης σε σχέση με προσωπικές προτιμήσεις (Personalization) αλλά και ένα υψηλό επίπεδο ασφάλειας και προστασίας προσωπικών πληροφοριών. Όλα αυτά γίνονται κάτω από ένα πρωτότυπο μηχανισμό μοντελοποίησης χρήστη [Evangelou et al., 2006] ο οποίος υποστηρίζει το υποσύστημα αναγνώρισης και εξουσιοδότησης αλλά και αναζήτησης πληροφορίας υψηλού ενδιαφέροντος για κάθε χρήστη ξεχωριστά. Επιπλέον το υποσύστημα διαχείρισης χρηστών και κοινοτήτων πρακτικής παρέχει δυνατότητες δημιουργίας και διαχείρισης μίας κοινότητας από έναν χρήστη, ενισχύοντας έτσι την ελεύθερη δημιουργία ομάδων στο πλαίσιο των δράσεων του Web2.0.

4.1.4 Στοιχεία λειτουργίας

Η καινοτομία του CoPe_it!, σε σχέση με τα υπάρχοντα συστήματα συνεργασίας με επιχειρήματα, έγκειται στα εξής σημαντικά στοιχεία:

- Τη χρήση του Διαδικτύου
- Την υποστήριξη της βαθμιαίας τυποποίησης με ταυτόχρονη λειτουργία
- Την αναπαράσταση υπερκειμένου από δύο διαφορετικά πεδία.

Οι πρωτοτυπίες αυτές κατατάσσουν το CoPe_it! ως σύγχρονο εργαλείο υποστήριξης συνεργασίας και επιτρέπουν την χρησιμοποίηση του για ποικίλους σκοπούς.

Το CoPe_it! είναι μία εφαρμογή του Παγκόσμιου Ιστού, πράγμα που σημαίνει πως οι χρήστες του χρειάζονται μόνο ένα συνηθισμένο φυλλομετρητή για να μπορέσουν το χρησιμοποιήσουν. Μάλιστα, το CoPe_it! έχει βασιστεί στο θεωρητικό υπόβαθρο του Web2.0 και γι'αυτό το λόγο θεωρείται εφαρμογή Web2.0. Χρησιμοποιεί τεχνολογίες που ανήκουν στο Web2.0, ενώ η χρήση του αποβλέπει στη δημιουργία και υποστήριξη κοινωνικών δικτύων. Έτσι, επιτρέπει τόσο τη σύγχρονη όσο και την ασύγχρονη επικοινωνία μεταξύ χρηστών και χρησιμοποιεί πληθώρα τρόπων επικοινωνίας όπως το Chat, τη διαδικασία ομιλίας και εργασίας σε στάδια με τακτικές ζήτησης και παραχώρησης λόγου, την αποστολή ασύγχρονων Posts σε προβολές (όπως στα Forums) και τέλος την αυστηρή επιχειρηματολογία. Παράλληλα, η ελεύθερη πρόσβαση σε αυτό και η δημιουργία και διαχείριση κοινοτήτων πρακτικής, από τους ίδιους τους χρήστες, έχει δώσει ιδιαίτερη ώθηση στη χρήση του.

Η βαθμιαία τυποποίηση αλλά και ο συνδυασμός λειτουργικότητας από δύο διαφορετικά πεδία υπερκειμένου, το χωρικό και το πεδίο επιχειρηματολογίας, εκτός του ότι παρέχει μεγάλη ευελιξία στη χρήση του συστήματος καθιστά το CoPe_it! ως μία εφαρμογή γενικής χρήσης στον τομέα συνεργασίας με επιχειρήματα, ενώ, παράλληλα, ανοίγει ένα μεγάλο φάσμα διαφορετικών εφαρμογών τις οποίες μπορεί να υποστηρίξει. Ως παράδειγμα αναφέρεται η χρήση του CoPe_it! από κοινότητες πρακτικής παιδαγωγών, ενώ το υποσύστημα επιχειρηματολογίας έχει φανεί χρήσιμο τόσο για ιατρικά θέματα όσο και για θέματα διπλωματίας. Επίσης, παρατηρείται χρήση του CoPe_it! ως εργαλείο ανταλλαγής και αποθήκευσης αρχείων κάθε τύπου αλλά, και ως σύστημα κράτησης και απεικόνισης προσωπικών σκέψεων και παρατηρήσεων πέρα από τα συνεργατικά πλαίσια.

Σε τέσσερα περίπου χρόνια λειτουργίας, στο CoPe_it! έχουν δημιουργηθεί 149 (85 ενεργές) κοινότητες χρηστών με συνολικά 721 (687 ενεργοί) χρήστες και 1405 (1079 ενεργοί) χώροι εργασίας.

4.2 Βασικές απαιτήσεις της εφαρμογής

Αυτό που σε πρώτη φάση θέλαμε να επιτύχουμε είναι το τελικό αποτέλεσμα στα μάτια του χρήστη να είναι κάτι πολύ εύχρηστο, γρήγορο και να καλύπτει απόλυτα τις βασικές ανάγκες των μελών του CoPe_it! Σκοπός μας είναι ο χρήστης, ανεξαρτήτως του που βρίσκεται, να έχει τη δυνατότητα να συνεχίσει απρόσκοπτα την δραστηριότητα του στο CoPe_it!.

Για την λειτουργία του Mobile CoPe_it! απαιτείται πρόσβαση στο Διαδίκτυο. Η πρόσβαση αυτή μπορεί να επιτευχθεί είτε μέσω Wi-Fi, είτε χρησιμοποιώντας τα δίκτυα κινητής τηλεφωνίας και κατά συνέπεια τις τεχνολογίες GPRS, 3G, HSDPA, 4G ανάλογα με τον υποστηριζόμενο τύπο υπηρεσίας δεδομένων της κινητής μας συσκευής και της διαθεσιμότητας του παρόχου.

Οι λειτουργίες του Mobile CoPe_it! που θα καλύπτουν κάποιες από τις βασικές ανάγκες των μελών του είναι οι εξής:

- Είσοδος στο CoPe_it!, δίνοντας το όνομα χρήστη (Username) και τον κωδικό πρόσβασης του μέλους στην υπηρεσία (Password).
- Αποθήκευση στη μνήμη των στοιχείων εισόδου του μέλους ώστε να μην χρειάζεται με κάθε άνοιγμα της εφαρμογής επαναπληκτρολόγηση των στοιχείων.
- Εμφάνιση και επιλογή των κοινοτήτων (Communities). Δηλαδή να εμφανίζεται η λίστα των Communities που ο χρήστης έχει τη δυνατότητα-εξουσιοδότηση να εισέλθει και στην συνέχεια αφού επιλέξει να εισέρχεται.
- Εμφάνιση και επιλογή των χώρων εργασίας (Workspaces). Αντίστοιχα με την παραπάνω λειτουργία αλλά αυτή τη φορά θα έχει εφαρμογή στα Workspaces.
- Ανάγνωση και εμφάνιση οποιουδήποτε αντικειμένου βρίσκεται μέσα στο Workspace της επιλογής μας.
- Δημιουργία Post, χρησιμοποιώντας τον αντίστοιχο τύπο του γνωστικού αντικειμένου (Knowledge Type) που επιθυμεί ο χρήστης, δηλαδή ιδέα (Idea), σημείωση (Note), σχόλιο (Comment), άλλο (Other) και τον τύπο περιεχομένου (Content Type) της αρεσκείας του (Text, URL, File).
- Δυνατότητα δημιουργίας συνημμένου αρχείου μέσα σε ένα Post (φωτογραφία, έγγραφο κλπ)

- Αποθήκευση στη μνήμη του κινητού του τελευταίου χώρου εργασίας που επισκέφθηκε το μέλος, ώστε στην επόμενη είσοδο του στην εφαρμογή να βλέπει άμεσα τις ανανεώσεις που έγιναν στον συγκεκριμένο χώρο εργασίας.

Ως προς το τύπο κατασκευής του Mobile CoPe_it! είχαμε δύο επιλογές. Θα μπορούσαμε να κατασκευάσουμε μια Mobile Webpage ή ένα Mobile Application, όπως και κάναμε. Κάποια βασικά πλεονεκτήματα της εφαρμογής έναντι της σελίδας, που μας ώθησαν να επιλέξουμε την πρώτη είναι:

- Η μελλοντική δυνατότητα που μας δίνεται για off-line εργασία
- Η δυνατότητα ενσωμάτωσης στην εφαρμογή των προσφερόμενων υπηρεσιών του κινητού (πχ να χρησιμοποιηθούν φωτογραφίες από το κινητό για upload)
- Το ωραιότερο και εξυπηρετικότερο interface, καθώς είναι σχεδιασμένο με βάση τις ακριβείς δυνατότητες του κινητού
- Η μεγαλύτερη ταχύτητα

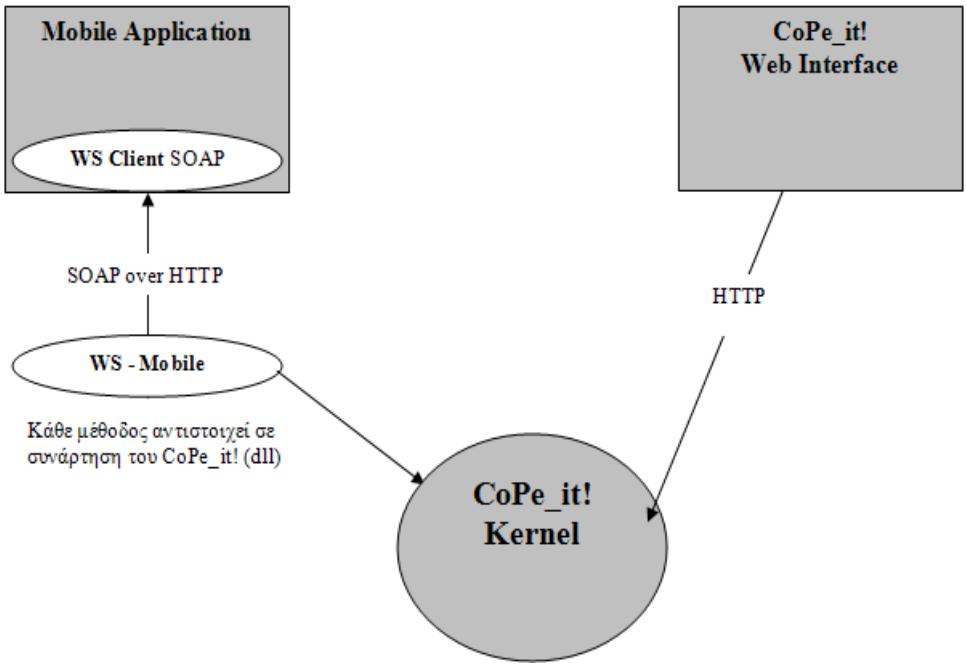
Οι βασικές απαιτήσεις που είχαμε ως προς την επιλογή του λειτουργικού συστήματος του κινητού, για το οποίο θα φτιάξουμε την εφαρμογή, ήταν τρείς:

- Γρήγορος χρόνος σχεδίασης και δημιουργίας του πρώτου πρωτοτύπου
- Εύκολη και γρήγορη παραμετροποίηση
- Δυνατότητα επέκτασης της λειτουργικότητας του

Λαμβάνοντας υπόψη τις τρεις παραπάνω απαιτήσεις αποφασίσαμε να κατασκευάσουμε την εφαρμογή έτσι ώστε να είναι συμβατή με συσκευές που λειτουργούν με Windows Mobile OS [URL: Windows Mobile OS]. Το συγκεκριμένο λειτουργικό σύστημα είναι απολύτως συμβατό με τις εφαρμογές που δημιουργεί το Visual Studio [URL: Visual Studio], το οποίο φαντάζει το ιδανικό εργαλείο για να εκπληρώσουμε τις παραπάνω απαιτήσεις.

4.3 Σχεδιασμός

Με βάση τις προαναφερθείσες απαιτήσεις καθώς και την υπάρχουσα σχεδιαστική προσέγγιση του συστήματος CoPe_it! σχεδιάσθηκε η παρακάτω αρχιτεκτονική (Εικόνα 4.7), βάση της οποίας προχώρησε η ανάπτυξη της εφαρμογής μας.



Εικόνα 4.7: *Αρχιτεκτονική - Εννοιολογικό διάγραμμα του Mobile CoPe_it!*

Το σχήμα της Εικόνας 4.7 αποτελεί εννοιολογικό σχεδιασμό, ο οποίος απαρτίζεται από τα εξής τμήματα:

- CoPe_it! Kernel - Είναι ο πυρήνας του συστήματος και αποτελεί τον μηχανισμό που είναι υπεύθυνος για την ερμηνεία και εκτέλεση όλων των αιτούμενων πράξεων από τους χρήστες. Επίσης ο CoPe_it! Kernel περιέχει και τον μηχανισμό αποθήκευσης στον οποίο καταχωρεί και οργανώνει όλα τα δεδομένα που εισέρχονται στο σύστημα.

- CoPe_it! Web Interface - Αποτελεί το γραφικό περιβάλλον του συστήματος και είναι ένα σύνολο από δυναμικές σελίδες, που είναι υπεύθυνες για την επικοινωνία του χρήστη με το σύστημα.

- Web Service Mobile - Αποτελεί το ενδιάμεσο τμήμα που υλοποιήθηκε στο πλαίσιο της εργασίας μας και επιτρέπει την επικοινωνία του CoPe_it! με εφαρμογές κινητών. Το συγκεκριμένο Web Service υποστηρίζει ένα σύνολο από μεθόδους που προδιαγράφηκαν για αυτό τον σκοπό.

- Mobile Application - Περιέχει το Web Service Client SOAP [SOAP - W3C Simple Object Access Protocol], το οποίο είναι ένας πελάτης που δημιουργήθηκε από το πρωτόκολλο SOAP ώστε να επικοινωνεί με το Web Service – Mobile, μέσω του HTTP πρωτοκόλλου.

4.4 Ανάπτυξη

Για να επιτύχουμε το στόχο μας, δηλαδή τη δημιουργία της πρώτης κινητής εφαρμογής συστήματος επιχειρηματολογίας / συνεργασίας, χρειάστηκε μία παράπλευρη ανάπτυξη λογισμικού. Τα δύο μέρη-πλευρές που αναπτύχθηκαν έχουν να κάνουν με τη μεριά του Server (Server Side) και την μεριά της κάθε αυτού εφαρμογής (Mobile Application) που θα εγκαθιστάτε στην κινητή συσκευή.

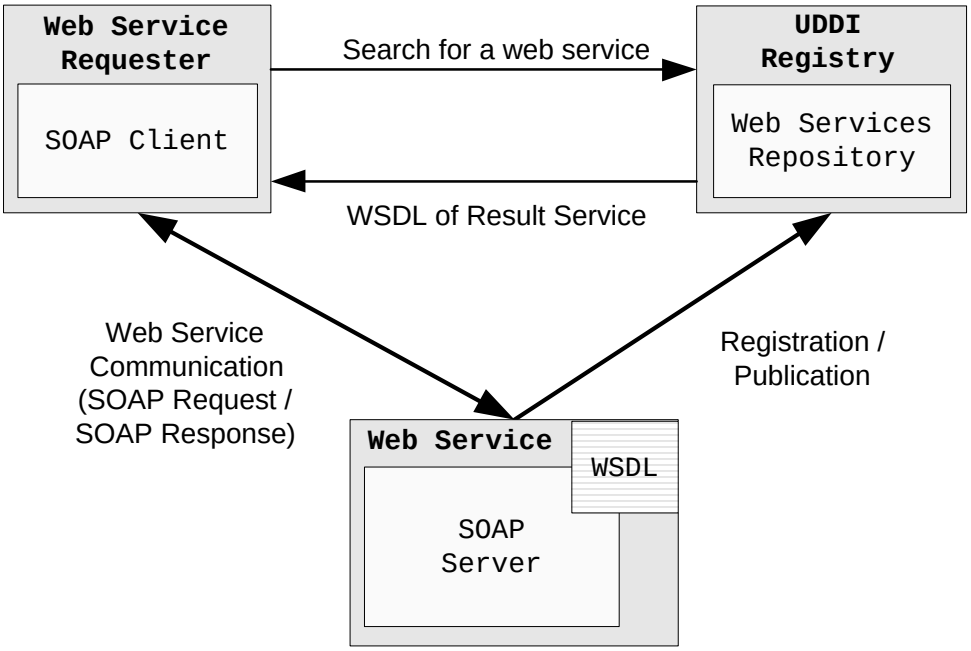
4.4.1 Server Side

Στον Server είναι απαραίτητο να υπάρχουν οι συναρτήσεις που θα καλούμε ώστε να ικανοποιείται κάθε φορά η εντολή που δίνει ο χρήστης στην εφαρμογή. Αποφασίσαμε να χρησιμοποιήσουμε την τεχνολογία των Web Services και να δημιουργήσουμε ένα καινούριο Web Service, βασισμένο σε οκτώ μεθόδους. Οι περισσότερες μέθοδοι χρησιμοποιούν συναρτήσεις που ήδη υπήρχαν στο CoPe_it! και κάποιες που δεν υπήρχαν, τις δημιούργησαν οι προγραμματιστές του CoPe_it!, ύστερα από συνεργασία μας.

4.4.1.1 Τι είναι τα Web Services;

Σύμφωνα με τον ορισμό του W3C ως Web Service αναφέρεται: «Μία εφαρμογή λογισμικού η οποία είναι δυνατό να αναγνωριστεί από ένα URI [URL: IETF RFC 2396], και τόσο το Interface του όσο και οι δεσμοί του είναι δυνατόν να καθορισθούν, να περιγραφούν και να ανακαλυφθούν μέσω κειμένων XML. Επιπλέον, υποστηρίζει άμεσες διαδράσεις με άλλες εφαρμογές λογισμικού, με χρήση μηνυμάτων βασισμένα σε XML και πρωτόκολλα του Διαδικτύου» [URL: W3C Web Services Architecture Domain].

Ένα τυπικό μοντέλο Web Service αποτελείται από έναν εξυπηρετητή SOAP (Simple Object Access Protocol), ο οποίος αλληλεπιδρά με το περιβάλλον του, μέσω ενός συνόλου από πράξεις προσβάσιμες από δίκτυο με χρήση μηνυμάτων (σύμφωνα με αποδεκτά πρότυπα – standards), βασισμένα σε XML. Για την πλήρη περιγραφή ενός Web Service χρησιμοποιείται μία ειδική γλώσσα περιγραφής Web Service: η WSDL (Web Service Description Language) [URL: WSDL - W3C Web Service Description Language].



Εικόνα 4.8: *Σχέδιο χρήσης των Web Services. [Πηγή: Karousos et al., 2003b]*

Η αποδοχή ενός συνόλου από διεθνή πρότυπα όσο αφορά στα Web Services, οδήγησε τους περισσότερους κατασκευαστές να σχεδιάζουν και να υλοποιούν Web Services με ένα πολύ κοινό τρόπο. Αυτά τα πρότυπα έδωσαν την ευκαιρία στους κατασκευαστές εφαρμογών να χρησιμοποιούν τα Web Services με ευκολία και χωρίς μεγάλο κόπο. Στην Εικόνα 4.8 παρουσιάζεται ένα βασικό σχέδιο της χρήσης των Web Services, το οποίο αναλύεται ως εξής:

- Το πρωτόκολλο SOAP (Simple Object Access Protocol) - Αποτελεί ένα διεθνές πρότυπο από τον οργανισμό του W3C. Το SOAP χρησιμοποιείται για τον καθορισμό του τρόπου αποστολής και λήψης αντικειμένων δομημένα σε μορφή XML. Έτσι, κάθε υπηρεσία που λειτουργεί με SOAP μπορεί να λαμβάνει κλήσεις για μεθόδους, περνώντας σε αυτή αντίστοιχα αντικείμενα σε XML. Το πιο συνηθισμένο πρωτόκολλο, το οποίο χρησιμοποιείται για την ανταλλαγή μηνυμάτων σε SOAP είναι το Hypertext Transfer Protocol (HTTP). Παρόλα αυτά, το SOAP μπορεί να χρησιμοποιηθεί και με διαφορετικά πρωτόκολλα, όπως το File Transfer Protocol (FTP) [URL: FTP - IETF RFC 2821] και το Simple Mail Transfer (SMTP) [URL: SMTP - IETF RFC 2821].

- Η γλώσσα WSDL (Web Service Description Language) - Είναι μία γλώσσα βασισμένη στις αρχές της XML, και χρησιμοποιείται για να προσδιορίσει και να περιγράψει πλήρως το interface που παρέχεται από ένα Web Service. Μετά τον ορισμό της γλώσσας αυτής ως διεθνές πρότυπο

από το W3C, είναι δυνατή η χρήση της, με στόχο τη δημιουργία αντίστοιχων WSDL αρχείων περιγραφής Web Services, τα οποία μπορούν να δημοσιευθούν στο Διαδίκτυο, και οι σύνδεσμοι προς αυτά να οδηγούν έναν άνθρωπο ή ένα πρόγραμμα στην κατανόηση της υπηρεσίας και, ενδεχομένως, την αυτόματη δημιουργία λογισμικού-πελάτη για τη χρήση της υπηρεσίας αυτής.

- Το UDDI (Universal Description, Discovery, and Integration) [URL: Universal Description, Discovery and Integration of Web Services] είναι ένα πρωτόκολλο καταχώρησης των Web Services. Χρησιμοποιείται για να παρέχει πληροφορίες για τα Web Services και μπορεί να λειτουργεί για την καταχώρηση δημόσιων, ιδιωτικών ή και υβριδικών τύπων πληροφορίας. Η καταχώρηση στο UDDI θα πρέπει να περιέχει συγκεκριμένες πληροφορίες για το Web Service. Αρχικά, θα πρέπει να έχει αναφορά στο αρχείο WSDL που θα περιγράφει το Web Service, αλλά και στην τοποθεσία στην οποία λειτουργεί το Web Service. Επιπρόσθετα, μία καταχώρηση θα πρέπει να έχει και άλλες χρήσιμες πληροφορίες (ακόμα και μη τεχνικές) για το Web Service και τον πάροχο του, όπως το κόστος του, τα στοιχεία των ιδιοκτητών του και πολιτικές λειτουργίας και παροχής υπηρεσίας.

- Οι καταχωρητές στου οποίους καταχωρείται η ύπαρξη των διαθέσιμων υπηρεσιών αλλά και η πλήρης περιγραφή τους μπορεί να είναι δημόσιοι, ιδιωτικοί ή και υβριδικοί. Ένας δημόσιος καταχωρητής παρέχει στους κατασκευαστές τη δυνατότητα να καταχωρήσουν και, με αυτό τον τρόπο, να κάνουν γνωστά τα Web Services τους. Κατόπιν, οι καταχωρητές αυτοί, είναι διαθέσιμοι στο κοινό για αναζήτηση και μεταφόρτωση πληροφορίας για τα διαθέσιμα Web Services. Ένας ιδιωτικός καταχωρητής δεν παρέχει υπηρεσίες στο ευρύ κοινό. Αντίθετα, είναι αφιερωμένος σε ένα μικρό κύκλο ενδιαφερομένων – συνήθως εταιριών – και παρέχει δυνατότητα διαμοίρασης των Web Services μόνο μεταξύ των συγκεκριμένων πελατών που ανήκουν στον συγκεκριμένο επιχειρηματικό κύκλο. Τέλος, ένας υβριδικός καταχωρητής λειτουργεί με ευελιξία: παρέχει τις πληροφορίες στο ευρύ κοινό, αλλά με αρκετούς περιορισμούς που τίθενται ανά περίπτωση.

Μερικά από τα χαρακτηριστικά πλεονεκτήματα της χρήσης των Web Services [Fremantle et al., 2002] είναι τα εξής:

- Διαλειτουργικότητα - Ένα Web Service παρέχει ανεξαρτησία πλατφόρμας υλικού αλλά και λογισμικού. Όποιος πελάτης χρησιμοποιεί τα διεθνή Πρότυπα (Standards) της τεχνολογίας

αυτής, μπορεί πολύ εύκολα να αποκτήσει πρόσβαση σε ένα Web Service. Επιπλέον, χρησιμοποιώντας τα αρχεία WSDL είναι δυνατή η αυτόματη δημιουργία ενός πελάτη SOAP.

- Ολοκλήρωση - Ένα Web Service μπορεί εύκολα να ολοκληρωθεί με οποιαδήποτε δικτυακή εφαρμογή, ανεξάρτητα από τις εσωτερικές προγραμματιστικές λεπτομέρειες της εφαρμογής.

- Διαθεσιμότητα - Ο κατασκευαστής ενός Web Service παρέχει ικανή πληροφόρηση, έτσι ώστε κάθε ένας κατασκευαστής να μπορέσει να χρησιμοποιήσει το Web Service αυτό φτιάχνοντας πρώτα έναν πελάτη.

- Συνθεσιμότητα - Τα Web Services μπορούν να αποτελέσουν τμήμα ενός πιο σύνθετου Web Service.

- Αυτό-περιγραφή - Τα Web Services έχουν τη δυνατότητα να περιγράφουν τον εαυτό τους με τέτοιο τρόπο, έτσι ώστε να μπορούν εύκολα να ανακαλυφθούν και να αναγνωριστούν. Έτσι, το interface, η τοποθεσία και οι πληροφορίες προσπέλασης του Web Service είναι αναγνωρίσιμες από οποιαδήποτε εφαρμογή.

- Δημοσιότητα - Η παροχή των περιγραφών των Web Services σε δημόσιους καταχωρητές τα κάνει εύκολα αναζητήσιμα και χρησιμοποιήσιμα από οποιοδήποτε χρήστη.

Ολοκληρώνοντας, σε αντίθεση με τη χρήση του HTTP πρωτοκόλλου, στο οποίο επισυνάπτεται όλη η επιπλέον επιθυμητή σημασιολογία, η έννοια των Web Services επιτρέπει στη σημασιολογία να είναι συγκεκριμένη για κάθε διαθέσιμη μέθοδο που παρέχεται ώστε να είναι δυνατή η επεξεργασία της από οποιαδήποτε λογισμικά συστήματα με αυτόματο τρόπο. Σε συνδυασμό με τη δυνατότητα της εναλλαγής του βασικού πρωτοκόλλου μεταφοράς (HTTP, FTP κ.α.) και την ευρεία διαθεσιμότητα όλων των Web Services, η χρήση της συγκεκριμένης τεχνολογίας θέτει τις βάσεις για την κατασκευή συστημάτων Διαδικτύου υψηλής ποιότητας.

4.4.1.2 Mobile CoPe_it! Web Service

Όπως προαναφέραμε το δικό μας Web Service, που δημιουργήσαμε για τις ανάγκες της εφαρμογής αποτελείται από οκτώ μεθόδους. Παρακάτω περιγράφονται αναλυτικά τις λειτουργίες αυτών των οκτώ μεθόδων:

- LoadCops - Χρησιμοποιώντας την μέθοδο «LoadCops» [URL: LoadCops of mobilews], δίνοντας ως είσοδο το όνομα χρήστη (Username) και τον κωδικό πρόσβασης στην υπηρεσία (Password), μας επιστρέφεται μία λίστα Κοινοτήτων (Communities), τις οποίες έχει τη δυνατότητα (άδεια-εξουσιοδότηση) ο συγκεκριμένος χρήστης να επισκεφθεί.

- LoadWorkSpaces - Χρησιμοποιώντας την μέθοδο «LoadWorkSpaces» [URL: LoadWorkSpaces of mobilews], δίνοντας ως είσοδο το id του Community, μας επιστρέφονται οι χώροι εργασίας (Workspaces) της επιλεγμένης από εμάς Κοινότητας.

- LoadDocuments - Μέσω της μεθόδου «LoadDocuments» [URL: LoadDocuments of mobilews], διαλέγοντας ένα Workspace φορτώνεται το περιεχόμενο του.

- Login: Χρησιμοποιώντας την μέθοδο «Login» [URL: Login of mobilews], δίνοντας σαν είσοδο το Username και το Password του χρήστη, αυτή μας κάνει Log-in στο Cope_it!

- RetrieveDocByName: Με τη μέθοδο «RetrieveDocByName» [URL: RetrieveDocByName of mobilews] μας δίνεται η δυνατότητα να έχουμε πρόσβαση σε ένα καταχωρημένο στοιχείο σύμφωνα με το μοναδικό του όνομά.

- RetrieveDocbyID: Με τη μέθοδο «RetrieveDocbyID» [URL: RetrieveDocbyID of mobilews] μας δίνεται η δυνατότητα να έχουμε πρόσβαση σε ένα καταχωρημένο στοιχείο σύμφωνα με το μοναδικό id του.

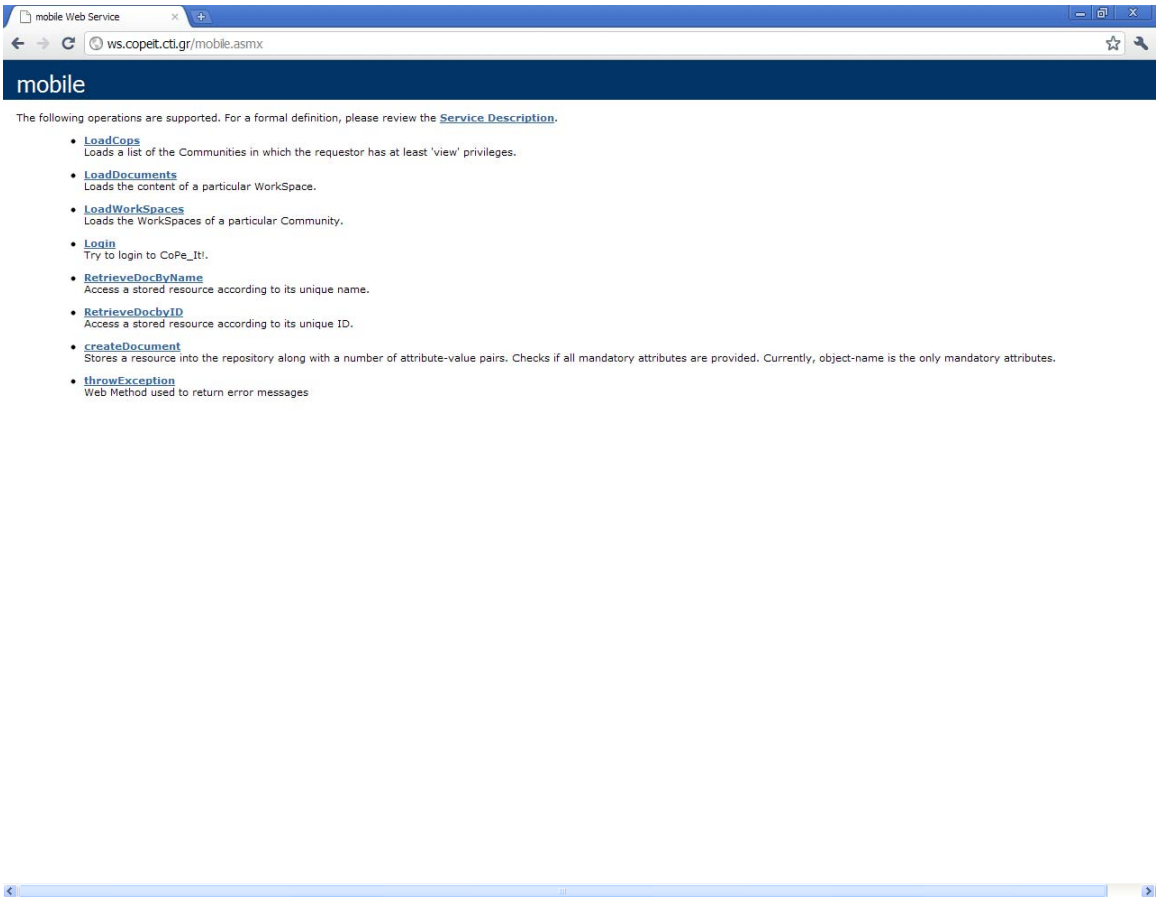
- CreateDocument: Μέσω της μεθόδου «CreateDocument» [URL: CreateDocument of mobilews], καταχωρείται ένα γνωστικό αντικείμενο συμπεριφοράς μαζί με ένα σύνολο από ζευγάρια τύπου όνομα-τιμή. Στην συνέχεια ελέγχει εάν παρέχονται όλα τα υποχρεωτικά στοιχεία.

- ThrowException: Η συγκεκριμένη μέθοδος [URL: ThrowException of mobilews], χρησιμοποιείται από το Web Service για να μας επιστρέψει κάποιο μήνυμα λάθους σε περίπτωση σφάλματος εισόδου δεδομένων ή λάθος εκτέλεσης από το σύστημα.

Αξίζει να αναφέρουμε πως, το Username και το Password περνάνε πάντα στη κεφαλίδα (Header) του κάθε SOAP μηνύματος και σε κάθε μέθοδο που καλείται, πρώτα γίνεται πιστοποίηση του χρήστη και των δικαιωμάτων πρόσβασης που έχει.

Η ανάπτυξη του Web Service έγινε με χρήση του Visual Studio και της γλώσσας C# [URL: C# Developer Center], δημιουργώντας και δημοσιεύοντας ένα Web Service Project με τίτλο

«mobilews» το οποίο βρίσκεται στη διεύθυνση http://ws.copeit.cti.gr/mobile.asmx. Το αντίστοιχο WSDL αρχείο περιγραφής είναι στη διεύθυνση http://ws.copeit.cti.gr/mobile.asmx?wsdl. Η μέχρι σήμερα ανάπτυξη του συστήματος CoPe_it! με χρήση τεχνολογίας .NET και το Visual Studio μας έδωσε τη δυνατότητα να χρησιμοποιήσουμε με αρκετά εύκολο τρόπο την υπάρχουσα υποδομή του CoPe_it! και να μπορέσουμε να ενσωματώσουμε μέσα στο Web Service όλες τις απαραίτητες συναρτήσεις του CoPe_it! ώστε να τις καλούμε και να επιστρέφουμε στους SOAP πελάτες τα επιθυμητά αποτελέσματα.



Εικόνα 4.9: *Προβολή ιστοσελίδας Web Service Project «mobilews». [Πηγή: http://ws.copeit.cti.gr/mobile.asmx]*

4.4.2 Mobile Application Side

Για να δημιουργήσουμε το Mobile Application αποφασίσαμε να χρησιμοποιήσουμε Visual Studio 2005 [URL: Visual Studio] και να γράψουμε κώδικα σε C# [URL: C# Developer Center] (βλέπε παράρτημα). Για να καταλήξουμε όμως στην απόφαση πως θα χρησιμοποιήσουμε Visual

Studio προσμετρήσαμε κάποια βασικά πλεονεκτήματα του συγκεκριμένου περιβάλλοντος. Τα πλεονεκτήματα αυτά είναι τα εξής:

- Συμβατότητα - Είναι η απόλυτα συμβατό με τα Windows Mobile
- Εξομοιωτές κινητών - Η ευκολία που παρέχει στους κατασκευαστές εφαρμογών κινητών, με τη χρήση ειδικών εξομοιωτών κινητού (Mobile Emulator) κατά τη σχεδίαση (design) και κατά τη διάρκεια του χρόνου εκτέλεσης (run-time).
- Web References - Αξίζει να αναφερθεί η πολύ εύκολη αξιοποίηση των Web Services από το Visual Studio, μέσω των Web References που έχουμε τη δυνατότητα να βάλουμε. Δηλαδή βάζοντας μόνο ως αναφορά τη σελίδα http://ws.copeit.cti.gr/mobile.asmx το Visual Studio μπορεί να χρησιμοποιήσει όλες τις μεθόδους που περιέχει το Web Service που φτιάξαμε ως τοπικές συναρτήσεις.

Χρησιμοποιώντας το Visual Studio δημιουργήσαμε αρκετές φόρμες και με χρήση «event programming», δώσαμε τα επιθυμητά αποτελέσματα στην εμφάνιση και στη λειτουργία της εφαρμογής.

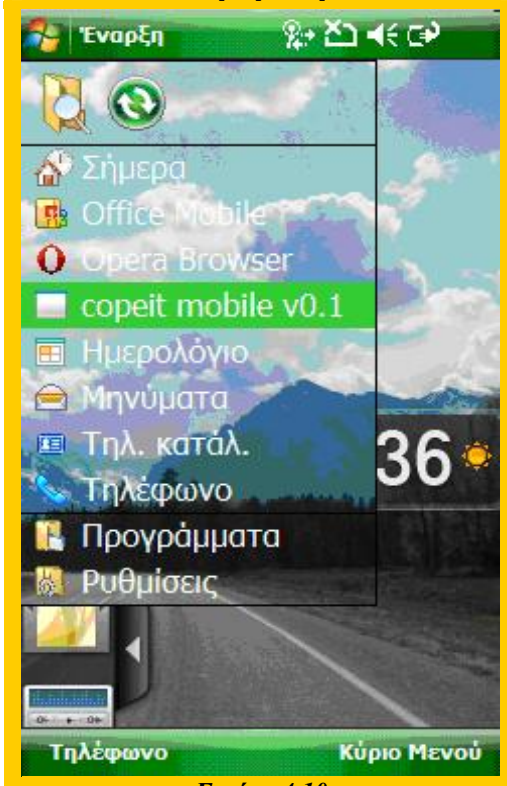
Έτσι δημιουργήσαμε το αρχείο εγκατάστασης σε Windows Mobile OS, MobileCopeit.cab. Το συγκεκριμένο αρχείο εγκαθιστά το MobileCopeit.exe στο κινητό και τοποθετεί ένα άλλο αρχείο με τίτλο Acount.txt στον φάκελο εγκατάστασης.

Το Acount.txt είναι ένα αρχείο κειμένου που εκεί καταγράφεται το Username και το Password του χρήστη και συγκρατεί το Community και το Workspace, που επισκέφτηκε τελευταία φορά. Έτσι ο χρήστης αποφεύγει να κάνει Login κάθε φορά που ανοίγει την εφαρμογή και πατώντας μόνο ένα link «Click to open workspace» μεταφέρεται στο Workspace που είχε επισκεφθεί τη τελευταία φορά.

Η χρήση της εφαρμογής γίνεται με ένα απλό κλικ στο εικονίδιο της και απαιτεί μόνο την ύπαρξη σύνδεσης με το Internet. Στη παρακάτω ενότητα θα περιγραφεί με εκτενή τρόπο η λειτουργία της εφαρμογής μέσα από αναλυτικά παραδείγματα.

4.5 Περιγραφή λειτουργίας – Εγχειρίδιο χρήσης

Στη συγκεκριμένη ενότητα περιγράφεται αναλυτικά η λειτουργία του Mobile CoPe_it! μέσα από ανάλυση βήμα-βήμα ενός πραγματικού σεναρίου (βλέπε εικόνες 4.10, 4.11, 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19, 4.20, 4.21, 4.22, 4.23, 4.24, 4.25) που υλοποιείται από έναν Mobile χρήστη της εφαρμογής, ο οποίος είναι κάτοχος κινητής συσκευής Samsung OMNIA SGH-i900 [URL: Samsung OMNIA SGH-i900], λειτουργικού συστήματος Microsoft Windows Mobile 6.1 Professional [URL: Windows Mobile OS].

Προβολή	**Σενάριο**
 ***Εικόνα 4.10***	*Επιλέγουμε την εφαρμογή από το μενού Έναρξη*



Εικόνα 4.11

Εισάγουμε το Username και το Password και πατάμε 'Login'.



Εικόνα 4.12

Εφόσον το Username και το Password είναι σωστά εμφανίζεται το μήνυμα 'Login Succeeded' και πατάμε το 'ν' για να προχωρήσουμε.



Εικόνα 4.13

Σε περίπτωση που δώσαμε λάθος στοιχεία τότε εμφανίζεται μήνυμα λάθους 'Login Failed' και ξαναπροσπαθούμε.



Εικόνα 4.14

Εφόσον κάναμε επιτυχή είσοδο μας βγάζει στο συγκεκριμένο περιβάλλον και μας δίνει δύο επιλογές:

1. Να επιλέξουμε το 'My Account' (ώστε να αλλάξουμε χρήστη, πράγμα σπάνιο για Mobile χρήστη).

2. Να επιλέξουμε το 'Communities', ώστε να επιλέξουμε σε ποια κοινότητα θέλουμε να συνδεθούμε.

Το συγκεκριμένο περιβάλλον θα εμφανίζεται από εδώ και στο εξής κάθε φορά που ανοίγουμε την εφαρμογή. Αφού θα έχει αποθηκευτεί το Username και το Password.



Εικόνα 4.15

Επιλέγουμε το 'Communities'.



Εικόνα 4.16

Εμφανίζονται τα Communities που είμαστε εγγεγραμμένοι.

Στη συγκεκριμένη περίπτωση το Mobile Phones και το Earth.

Το Earth είναι ένα κεντρικό community, όπου είναι φτιαγμένο από τον Administrator του Copeit, ώστε να εισάγεσαι αυτόματα κατά την εγγραφή σου ως καινούριος χρήστης του Cope_it!

π.χ. Εμείς αποφασίζουμε να μπούμε στη κοινότητα 'Mobile Phones'.



Εικόνα 4.17

Μας εμφανίζεται το συγκεκριμένο περιβάλλον όπου επιλέγουμε το κουμπί 'Workspaces'.



Εικόνα 4.18

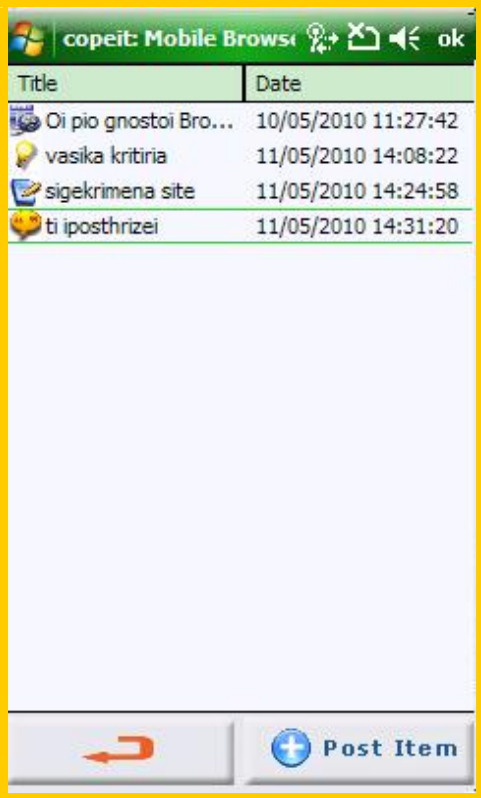
Βλέπουμε τη λίστα με τα Workspaces που ανήκουν στη κοινότητα Mobile Web και έχουμε τη δυνατότητα να συμμετέχουμε.

π.χ. επιλέγουμε το 'Mobile Browsers' και πατάμε το 'ν'



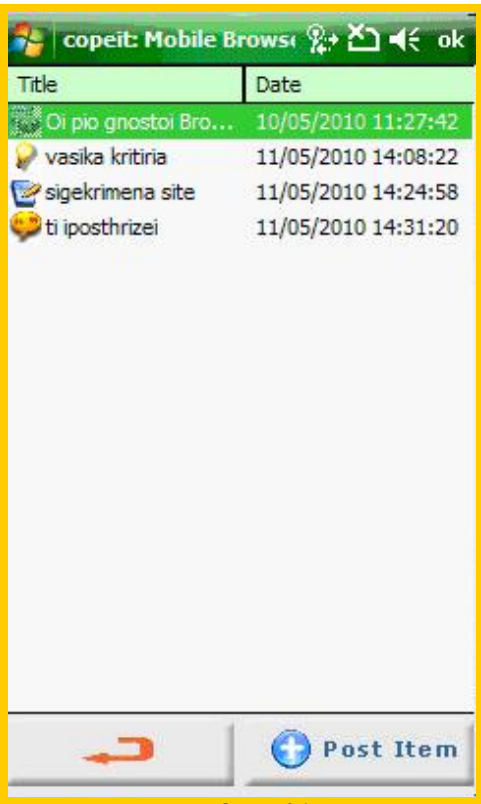
Εικόνα 4.19

Μας εμφανίζει την αρχική οθόνη και αναμένουμε να ανοίξει το Workspace που επιλέξαμε.



Εικόνα 4.20

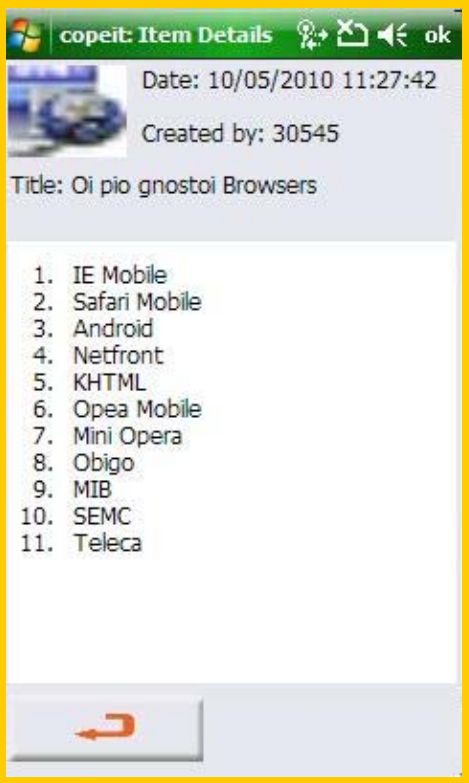
Εμφανίζεται μια λίστα με τα αντικείμενα που βρίσκονται εντός του Workspace Mobile Browsers.



Εικόνα 4.21

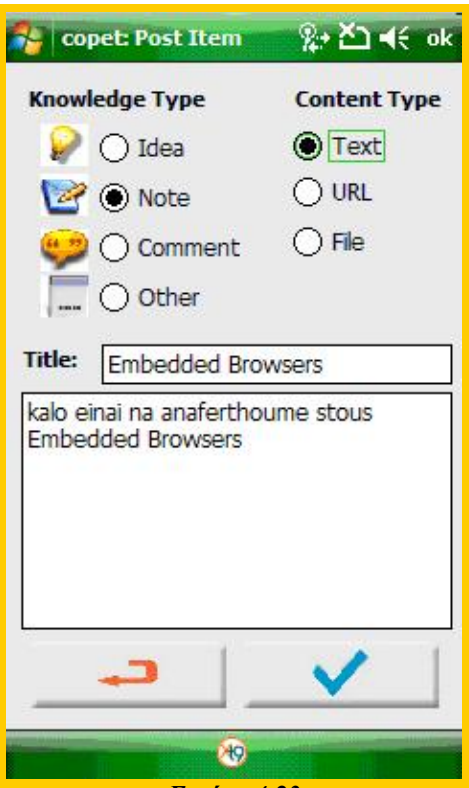
Έχουμε τη δυνατότητα να επιλέξουμε προς εμφάνιση όποιο αντικείμενο επιθυμούμε.

Π.χ. Επιλέγουμε το 'Oi pio gnostoi Browsers'.



Εικόνα 4.22

Ανοίγει το επιλεγμένο αντικείμενο και βλέπουμε το περιεχόμενο του.

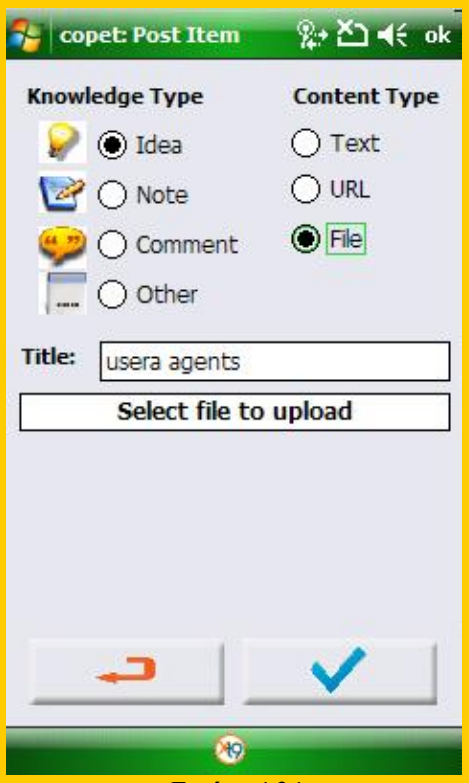


Εικόνα 4.23

Εκτός από το να δούμε κάποιο αντικείμενο έχουμε τη δυνατότητα και να καταχωρήσουμε (Post) ένα αντικείμενο.

π.χ. Εδώ αποφασίσαμε να καταχωρήσουμε (Post) μια σημείωση (Note) σε μορφή text.

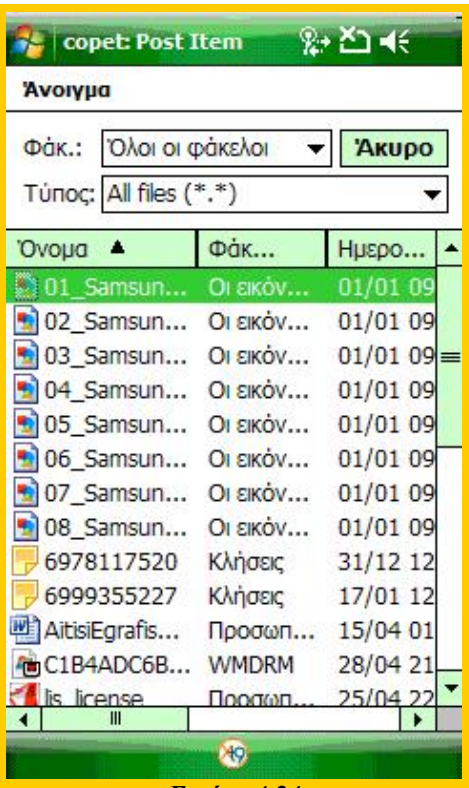
Ορίζουμε τίτλο και περιεχόμενο και πατάμε 'v'.



Εικόνα 4.24

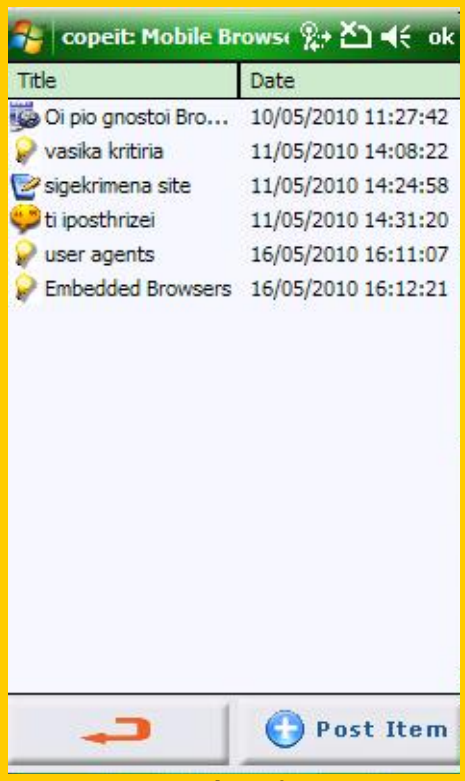
Εδώ αποφασίσαμε να καταχωρήσουμε μια ιδέα (Idea) σε μορφή File, δηλαδή να ανεβάσουμε ένα αρχείο.

Ορίζουμε τίτλο και επιλέγουμε το 'Select file to upload', ώστε να επιλέξουμε το αρχείο που θέλουμε να καταχωρήσουμε (Post).



Εικόνα 4.24

Επιλέγουμε το αρχείο που επιθυμούμε.



Εικόνα 4.25

Εμφανίζεται το ανανεωμένο Workspace που όπως βλέπουμε περιέχει και τα 2 καινούρια αντικείμενα που δημιουργήσαμε μέσω του mobile application, το 'user agents' και το 'Embedded Browsers'.

4.6 Θέματα εξέλιξης

Υλοποιώντας τη συγκεκριμένη εφαρμογή, είναι κατανοητό πως καλύψαμε μόνο κάποιες βασικές ανάγκες ενός χρήστη του Mobile CoPe_it!, ώστε να μπορεί με ευκολία να συμμετέχει στο σύστημα επιχειρηματολογίας ανεξαρτήτως χώρου, χρόνου και χωρίς τη βοήθεια του υπολογιστή του.

Στα άμεσα αλλά και μελλοντικά σχέδια μας είναι η εξέλιξη της συγκεκριμένης εφαρμογής ώστε να ικανοποιεί ακόμα μεγαλύτερη γκάμα λειτουργιών που επιθυμεί ο χρήστης, χρησιμοποιώντας ως «κύρια» εφαρμογή το Mobile CoPe_it! και όχι ως απλά μια «βοηθητική» εφαρμογή έως ότου βρεθούμε μπροστά σε έναν υπολογιστή. Επίσης στοχεύουμε στη δημιουργία της αντίστοιχης εφαρμογής για μεγαλύτερη γκάμα κινητών τηλεφώνων και λειτουργικών συστημάτων.

Στα άμεσα σχέδια μας με την έκδοση του αναβαθμισμένου λογισμικού Mobile CoPe_it! V.2 οι προτεραιότητες μας θα είναι να καλύψουμε τις τρεις βασικές παρακάτω λειτουργίες:

- Άμεση συσχέτιση αντικειμένων - Δυνατότητα άμεσης συσχέτισης αντικειμένου με άλλο αντικείμενο την ώρα του post. Δηλαδή όταν ο Mobile χρήστης πραγματοποιεί ένα post να έχει τη δυνατότητα να προαποφασίσει με ποιο ήδη υπάρχων αντικείμενο θέλει να συσχετίσει το νεοεισερχόμενο αντικείμενο του, ώστε στη προβολή χάρτη να είναι απόλυτα κατανοητή η σχέση μεταξύ των δύο συσχετιζόμενων αντικειμένων.

- Off-line εργασία – Δυνατότητα εργασίας και δημιουργίας ενός επιχειρήματος τοπικά στο κινητό (χωρίς χρήση Mobile Internet). Αυτή η λειτουργία θα χρησιμεύει είτε για καθαρά προσωπική χρήση, είτε μετέπειτα θα γίνεται εισαγωγή (import) σε ένα Community ως νέο Workspace, είτε σε ήδη υπάρχων εφόσον κατά τη διάρκεια της τοπικής επεξεργασίας αυτό το Workspace δεν έχει υποστεί αλλαγές από κάποιον άλλο χρήστη.

- Επεξεργασία υπαρχόντων αντικειμένων – Δυνατότητα επεξεργασίας αντικειμένων που είναι ήδη τοποθετημένα σε ένα Workspace από τον χρήστη σε προηγούμενη επίσκεψη του.

- Αναζήτηση αντικειμένων - Δυνατότητα αναζήτησης αντικειμένων από το κινητό, όπως ακριβώς προσφέρεται η συγκεκριμένη υπηρεσία στο CoPe_it!

Στην συνέχεια και αφού επιτευχθεί με επιτυχία το παραπάνω πλάνο μας, με την έκδοση της δεύτερης έκδοσης του Mobile CoPe_it!, θα προσπαθήσουμε να κάνουμε ένα ακόμα σημαντικό

βήμα εξέλιξης προσπαθώντας να υλοποιήσουμε αν όχι όλους κάποιους από τους παρακάτω στόχους:

- Εναλλακτικές μορφές απεικόνισης - Εκτός της υποστήριξης προβολής σε μορφή λίστας, να υπάρχει δυνατότητα υποστήριξης εναλλακτικών μορφών απεικόνισης των διαλόγων και της έκθεσης επιχειρημάτων στα Workspaces, με απώτερο σκοπό την κάλυψη όλων των προβολών που καλύπτει το CoPe_it! μέσω ηλεκτρονικού υπολογιστή (προβολή χάρτη – χαλαρή τυποποίηση, προβολή αυστηρής τυποποίησης και προβολή φόρουμ).

- Ολοκλήρωση με το λειτουργικό – Με την εγκατάσταση της εφαρμογής θα εμφανίζεται αυτόματα σε κάθε τι συμβατό με το Mobile CoPe_it!, που θα υπάρχει στη μνήμη του κινητού (π.χ. φωτογραφία, link, κείμενο κλπ), επιλογή για χρησιμοποίηση του συγκεκριμένου αντικειμένου στο CoPe_it!. (π.χ. βγάζοντας μία φωτογραφία να έχει επιλογή → αποστολή → σε CoPe_it!, όπως ακριβώς συμβαίνει με το MMS και το e-mail)

- Σύστημα ειδοποιήσεων – Μετά από κάθε αλλαγή σε ένα Workspace άμεσου ενδιαφέροντος του χρήστη να του αποστέλλεται μία ειδοποίηση είτε με την έναρξη της εφαρμογής, είτε μέσω mail ή SMS. Βέβαια η συγκεκριμένη λειτουργία πρέπει αρχικά να υλοποιηθεί από την ομάδα εργασίας του CoPe_it!, ώστε να εκμεταλλευτούμε τη συγκεκριμένη υπηρεσία και στο κινητό.

- Περισσότερα λειτουργικά συστήματα – Μετά την ολοκλήρωση της 2^{ης} έκδοσης του Mobile CoPe_it! και αφού έχουμε φέρει εις πέρας κάποιες από τις παραπάνω προκλήσεις – στόχους, θα γίνει μία προσπάθεια επέκτασης της συγκεκριμένης εφαρμογής και σε άλλα λειτουργικά συστήματα κινητών συσκευών, με πρώτο μας μέλημα τη δημιουργία του Mobile CoPe_it! App για συσκευές που υποστηρίζουν iOS (iPhone/iPod/iPad) [URL: iOS] και στη συνέχεια σε Android OS [URL: Android OS] (κινητά και Tablets).

ΚΕΦΑΛΑΙΟ 5: Σύνοψη

Το παρόν κεφάλαιο συνοψίζει τη Διπλωματική Εργασία και καταλήγει σε ένα σύνολο από χρήσιμα συμπεράσματα που διεξάχθηκαν από την μελέτη και την έρευνα που έγινε κατά την διάρκεια εκπόνησης της.

5.1 Σύνοψη

Η παρούσα Διπλωματική Εργασία πραγματεύεται τα Συστήματα Υποστήριξης Επιχειρηματολογίας και το κατά πόσον αυτά έχουν θέση στον Κινητό Παγκόσμιο Ιστό (Mobile Web). Η ραγδαία ανάπτυξη των τεχνολογιών και προσφερόμενων υπηρεσιών στον Παγκόσμιο Ιστό, φαίνεται πως συνεπαίρνει και τα Συστήματα Υποστήριξης Επιχειρηματολογίας όπου γίνεται μια σημαντική προσπάθεια από τους δημιουργούς των αντίστοιχων εργαλείων για την μεταφορά τους σε αντίστοιχες υπηρεσίες του Web2.0. Αυτή τη στιγμή βρίσκονται on-line στο Web οχτώ τέτοια εργαλεία. Όμως οι περισσότερες εταιρίες-δημιουργοί υπηρεσιών-εφαρμογών του Web2.0, βλέποντας τα γρήγορα και μεγάλα τεχνολογικά άλματα που συντελούνται στο χώρο του Mobile Internet, μέσα από την ανάπτυξη των υποδομών αλλά και τη χρήση όλο και πιο ισχυρών και ικανών κινητών συσκευών (smartphones, tablets), μετατοπίζουν ένα μεγάλο κομμάτι του ενδιαφέροντος τους στη «μεταφορά» των προσφερόμενων υπηρεσιών τους στο χώρο του Mobile Web. Εδώ ακριβώς είναι που εντοπίζεται το κενό που υπάρχει στα Συστήματα Υποστήριξης Επιχειρηματολογίας. Σύμφωνα με την παραπάνω έρευνα δεν υπάρχει κάποια αντίστοιχη έκδοση για κινητά, είτε μέσω ενός Mobile Site, είτε μέσω κάποιας Mobile εφαρμογής. Μία πρώτη ολοκληρωμένη προσπάθεια έγινε για τη κάλυψη ενός μικρού μέρους αυτού του κενού με τη δημιουργία μίας Mobile εφαρμογής που παρέχει σε χρήστες κινητού τηλεφώνου, με λειτουργικό σύστημα Windows Mobile, τη δυνατότητα να συμμετέχουν σε ένα συγκεκριμένο Σύστημα Υποστήριξης Επιχειρηματολογίας: το CoPe_it!.

5.2 Συμπεράσματα

Από την παραπάνω έρευνα και μελέτη της Διπλωματικής Εργασίας βγαίνουν αρκετά συμπεράσματα που μπορούν να αποτελέσουν χρήσιμα εργαλεία στα χέρια των μελών των Κοινοτήτων ενασχόλησης με τα Συστήματα Υποστήριξης Επιχειρηματολογίας και με το Mobile Web, είτε ανεξάρτητα η μία Κοινότητα με την άλλη, είτε συνδυαστικά.

Όπως ειπώθηκε από τη Mary Meeker είναι πλέον ξεκάθαρο πως η δεκαετία που διανύουμε είναι δεκαετία του Mobile Web. Μετά την εποχή του Main Frame, την εποχή των μικροϋπολογιστών, την εποχή των προσωπικών υπολογιστών, του desktop Διαδικτύου και του Παγκόσμιου Ιστού, βρισκόμαστε ήδη στην εποχή του Κινητού Παγκόσμιου Ιστού (Mobile Web). Η μετατόπιση-εξέλιξη αυτή από το κλασικό Web στο Mobile Web οδήγησε στην κατασκευή χιλιάδων Mobile

Sites αλλά και ακόμη περισσότερων χιλιάδων Mobile εφαρμογών. Τα σημαντικότερα στηρίγματα προς αυτή την κατεύθυνση δόθηκαν από τα παρακάτω στοιχεία:

- Ανάπτυξη προτύπων από τη W3C κοινότητα για το περιεχόμενο του Mobile Web
- Δημιουργία κάποιων βασικών προτύπων κινητών συσκευών για χρήση Mobile Web
- Ανάπτυξη της XHTML Mobile Profile, που ουσιαστικά είναι η XHTML Basic με κάποια επιπρόσθετα στοιχεία και ιδιότητες της XHTML
- Ραγδαία ανάπτυξη των δικτύων κινητής τηλεφωνίας (3G, HSDPA, 4G), όπου ουσιαστικά λύνεται κατά μεγάλο βαθμό το πρόβλημα του bandwidth και της ταχύτητας
- Η κυκλοφορία των Smartphones που εκμεταλλεύονται την μεγάλη επεξεργαστική τους ισχύ, την ύπαρξη Wi-Fi, το μεγαλύτερο μέγεθος οθόνης σε σχέση με τα cell phones, την καλύτερη ανάπτυξη των browser τους και γενικότερα τον πιο δομημένο τρόπο ανάπτυξης εγγενών εφαρμογών.

Η μεγάλη ανάπτυξη του Mobile Web επιβεβαιώνεται στατιστικά και από τρείς μεγάλες έρευνες που παρουσιαστήκαν για λογαριασμό της Morgan Stanley (δύο) και της informa Telecoms & Media (μία). Τα σημαντικότερα ευρήματα έχουν να κάνουν με το ότι:

- Οι χρήστες του Mobile Internet στα μέσα του 2014, θα ξεπεράσουν σε αριθμό αυτούς του Desktop Internet [Meeker, 2010a].
- Στα μέσα του 2012 ο αριθμός των χρηστών Smartphones θα ξεπερνά τον αριθμό των χρηστών προσωπικών υπολογιστών (σταθερών και Netbook) [Meeker, 2010b].
- Ο αριθμός των χρηστών smartphones αναμένεται να ξεπεράσει το 1 δισεκατομμύριο έως το 2013 [URL: informa Telecoms & Media].

Τα Συστήματα Υποστήριξης Επιχειρηματολογίας έχουν κάνει την εμφάνιση εδώ και αρκετά χρόνια και έχουν τόσο ευρύ πεδίο εφαρμογής σε τομείς όπως η εκπαίδευση, η Ιατρική, η Τεχνητή Νοημοσύνη, στο δημόσιο και ιδιωτικό τομέα όπου λαμβάνουν χώρα διοικητικά συμβούλια αλλά και σε πολλές άλλες περιπτώσεις. Υπάρχει μεγάλος αριθμός τέτοιων ηλεκτρονικών εφαρμογών και αντίστοιχων εργαλείων. Συγκεκριμένα στην έρευνα που έγινε στη Διπλωματική Εργασία καταγράφηκαν τα εξής σημαντικά στατιστικά και όχι μόνο

συμπεράσματα:

- Υπάρχουν διαθέσιμες τριάντα εννέα ηλεκτρονικές εφαρμογές Συστημάτων Υποστήριξης Επιχειρηματολογίας (απεικόνισης επιχειρημάτων, σκέψης και διαλόγων)
- Από τις τριάντα εννέα αυτές εφαρμογές οι δέκα υπήρξαν ή υπάρχουν διαθέσιμες on-line και βασισμένες στο Web
- Από αυτές τις δέκα, οι οχτώ είναι διαθέσιμες σήμερα
- Δεν υπάρχει καμία έκδοση των συγκεκριμένων εφαρμογών για κινητά, είτε ως ιστοσελίδα κινητού (mobile site), είτε ως εφαρμογή για κινητά (mobile application)

Από τα παραπάνω συμπεραίνεται η ύπαρξη ενός κενού στον χώρο του Mobile Web και των Συστημάτων Υποστήριξης Επιχειρηματολογίας. Από την μεριά του Mobile Web είναι φανερό πως έχουν ωριμάσει οι συνθήκες ώστε οι περισσότερες εφαρμογές του Web2.0 να είναι προσβάσιμες και από τις κινητές συσκευές, μέσω mobile sites ή μέσω mobile applications. Από τη μεριά των Συστημάτων Υποστήριξης Επιχειρηματολογίας είναι επίσης σαφές ότι οι οχτώ on-line web-based εφαρμογές βοηθούν πολύ στην επίλυση προβλημάτων, στην έκφραση σκέψεων και διαλόγων, στην μάθηση και στην συνεργασία μέσω επιχειρημάτων, αλλά περιορίζονται στο χώρο και στο χρόνο.

Η δυνατότητα διαδραστικότητας ανθρώπου και Παγκόσμιου Ιστού με βάση τις σημερινές συνθήκες πρέπει να είναι ανεξάρτητη χώρου, χρόνου και μέσου. Έτσι και κάθε «ιδιόκτητης» μιας υπηρεσίας του Web2.0 που σέβεται το κοινό του, οφείλει να δώσει πρόσβαση στην προσφερόμενη υπηρεσία του και μέσω του Mobile Web. Είναι απαραίτητο να γίνει η αρχή και στο πεδίο των Συστημάτων Υποστήριξης Επιχειρηματολογίας με τη δημιουργία αντίστοιχων Mobile υπηρεσιών.

Στην Διπλωματική Εργασία αναπτύχθηκε μία Mobile εφαρμογή που παρέχει σε χρήστες κινητού τηλεφώνου, με λειτουργικό σύστημα Windows Mobile, τη δυνατότητα να συμμετέχουν σε ένα συγκεκριμένο Σύστημα Υποστήριξης Επιχειρηματολογίας: το CoPe_it!. Μέσα από την ανάπτυξη της πρώτης έκδοσης της συγκεκριμένης εφαρμογής καλύφθηκαν κάποιες βασικές ανάγκες ενός Mobile χρήστη του CoPe_it!, που θα βοηθήσουν στην περεταίρω χρήση της συγκεκριμένης υπηρεσίας αλλά και στην αναπόσπαστη επικοινωνία των χρηστών του, σε μια πρώτη προσπάθεια για μία μικρή κάλυψη του κενού που υπάρχει στο χώρο του Mobile Web και

των Συστημάτων Υποστήριξης Επιχειρηματολογίας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Anderson, R. C., & Pearson, P. D. (1984). A schema-theoretic view of basic processes in reading comprehension. In P. D. Pearson, R. Barr, M. L. Kamil, & P. Mosenthal (Eds.), *Handbook of reading research* (pp. 255-291). New York: Longman.

Ahmed, M.A (2007). The Web 2.0 Culture. *Ezine Article. R*etrieved on August 22, 2007, from http://ezinearticles.com/?The-Web-2.0-Culture&id=693129

Andriessen, J., Baker, M., & Suthers, D. (Eds.). (2003). *Arguing to learn: Confronting* cognitions in computersupported collaborative learning environments. Dordrecht: Kluwer Academic Publishers.

Baker, M. J. (2003). Computer-mediated argumentative interactions for the co-elaboration of scientific notions. In J. Andriessen, M. J. Baker & D. D. Suthers (Eds.), *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments* (pp. 47–78). Dordrecht, The Netherlands: Kluwer Academic Publishers.

Barth, E. M., & Krabbe, E. C. W. (Eds.). (1982). *From axiom to dialogue: A philosophical study of logics and argumentation*. Berlin: Walter De Gruyter.

Bell, P. (1997). Using argument representations to make thinking visible for individuals and groups. In R. Hall, N. Miyake & N. Enyedy (Eds.), *Proceedings of the 2nd International Conference on Computer Support for Collaborative Learning* (CSCL 1997) (pp. 10–19). Toronto, Ontario, Canada: University of Toronto Press.

Bell, P., & Linn, M. C. (2000). Scientific arguments as learning artefacts: Designing for learning from the web with KIE. *International Journal of Science Education, 22*, 797–817.

Berners-Lee, T. J., Cailliau R., Groff, J. F., Pollermann B. (1992). World-Wide Web: The information universe. *Electronic Networking: Research, Applications and Policy* 2 (1), 52-58.

Berners-Lee, T. (2002). *Υφαίνοντας τον παγκόσμιο ιστό*, μτφρ. Καλαϊτζής, Ν. Αθήνα: Γκοβόστης.

Bex, F., van den Braak, S., Oostendorp, H., Prakken, H., Verheij, B., & Vreeswijk, G. A. W. (2007). Sense-making software for crime investigation: How to combine stories and arguments? *Law, Probability and Risk*, *6*(1–4), 145–168.

Billig, M. (1987). *Arguing and Thinking: A Rhetorical Approach to Social Psychology*. Cambridge: Cambridge University Press.

Boehm, B., Egyed, A., Kwan, J., Port, D., Shah, A., & Madachy, R. (1998). Using the WinWin Spiral Model: A Case Study, *Computer*, vol.31, no. 7, (pp. 33-44).

Buckingham Shum, S. J., Selvin, A. M., Sierhuis, M., Conklin, J., Haley, C. B., & Nuseibeh, B. (2006). Hypermedia support for argumentation-based rationale: 15 years on from gIBIS and QOC. In A.H. Dutoit, R. McCall, I. Mistrik & B. Paech (Eds.), *Rationale management in software engineering* (pp. 111–132). Berlin, Germany: Springer.

Buckingham Shum, S.J., Uren, V., Li, G., Domingue, J., and Motta, E. (2003). Visualizing Internetworked Argumentation. In: *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. Paul A. Kirschner, Simon J. Buckingham Shum and Chad S. Carr (Eds), (pp.185-204). Springer-Verlag: London.

Burge, J. E. (2005). *Software Engineering Using design RATionale*. PhD thesis, Worcester Polytechnic Institute, Department of Computer Science, 100 Institute Road - Worcester, MA, USA.

Bush, V. (1945). As We May Think. *Atlantic Monthly*, *176*(1), 101-108.

Chesñevar, C., McGinnis, J., Modgil, S., Rahwan, I., Reed, C., Simari, G., South, M., Vreeswijk, G., & Willmott, S. (2007). Towards an argument interchange format. *The Knowledge Engineering Review, 21*(4), 293–316.

Carr, C. S. (2003). Using computer supported argument visualization to teach legal argumentation. In P. A. Kirschner, S. J. Buckingham Shum & C. S. Carr (Eds.), *Visualizing argumentation: Software tools for collaborative and educational sense-making* (pp. 75–96), London, UK: Springer.

Evangelou, C.E., Karousos, N., Tzagarakis, M., Gkotsis, G., & Nousia, D. (2006). Personalization Services in Argumentation Tools: a Catalyst for Learning. *Workshop of "TEL Communities of Practice" in the First European Conference on Technology Enhanced Learning*, Crete, Greece.

Conklin, J., Selvin, A., Buckingham Shum, S. and Sierhuis, M. (2003) Facilitated Hypertext for Collective Sensemaking: 15 Years on from gIBIS. Keynote Address, *Proceedings LAP'03: 8th International Working Conference on the Language-Action Perspective on Communication Modelling*, (Eds.) H. Weigand, G. Goldkuhl and A. de Moor. Tilburg, Netherlands.

Conklin, J., & Begeman, M. L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *Proceedings of the ACM Conference on Computer-supported Cooperative Work (CSCW '88)* (pp. 140–152). New York: ACM Press.

de Groot, R., Drachman, R., Hever, R., Schwarz, B., Hoppe, U., Harrer, A., De Laat, M., Wegerif, R., McLaren, B. M., & Baurens, B. (2007). Computer supported moderation of ediscussions: The ARGUNAUT approach. In C. Chinn, G. Erkens & S. Puntambekar (Eds.), *Mice, Minds, and Society – The Computer Supported Collaborative Learning (CSCL) Conference 2007,* (pp. 165–167). International Society of the Learning Sciences.

Driver, R., Newton, P., & Osborne, J. (2000). Establishing the norms of scientific argumentation in classrooms. *Science Education*, 84(3), 287-313.

Engelbart, D.C. (1962). Augmenting Human Intellect: A Conceptual Framework. Summary

Report AFOSR-3223 under Contract AF 49(638)-1024, SRI Project 3578 for Air Force Office

of Scientific Research, Stanford Research Institute, Menlo Park, Ca.

Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*, Ph.D. thesis, University of California, Irvine.

Garrett, J. J. (2005). Ajax: A New Approach to Web Applications. Online resource, http://www.adaptivepath.com/publications/essays/archives/ 000385.php. Created on February 18, 2005. Accessed on March 5, 2009.

Gordon, T. F., & Karacapilidis, N. (1997). The Zeno argumentation framework. In *Proceedings of the 6th International Conference on AI and Law (ICAIL 1997)* (pp. 10–18), New York: ACM Press.

Gordon, T. F., Prakken, & H., Walton, D. (2007). The Carneades model of argument and burden of proof. *Artificial Intelligence, 171*(10–15), 875–896.

Hair, D. C. (1991): Legalese: A legal argumentation tool. *SIGCHI Bulletin, 23*(1), 71–74.

Karacapilidis, N., & Papadias, D. (1998). HERMES: Supporting Argumentative Discourse in Multi-Agent Decision Making. *In Proceedings of the 15th National Conference on Artificial Intelligenc*e (AAAI-98), (pp. 827-832). Madison, WI, AAAI/MIT Press.

Karacapilidis, N., Tzagarakis, M., Karousos, N., Gkotsis, G., Kallistros, V., Christodoulou, S., Mettouris, C., & Nousia, D. (2009a). Tackling cognitively-complex collaboration with CoPe_it!. *International Journal of Web-Based Learning and Teaching Technologies*, *4*(3), 22–38.

Karacapilidis, N., Tzagarakis, M., Karousos, N., Gkotsis, G., Kallistros, V., Christodoulou, S., Nousia, D., Mettouris, C. & Kyriakou, P. (2009b). CoPe_it! - Supporting collaboration, enhancing learning. In: G. Ditsa, S. Conger and L. Janczewski (eds.), *Proceedings of the 2009 International Conference on Information Resources Management* (Conf-IRM 2009b), Al Ain, UAE.

Klein, M. & Iandoli, L., (2008). Supporting Collaborative Deliberation Using a Large-Scale Argumentation System: The MIT Collaboratorium, MIT Sloan Research Paper No. 4691-08.

Kuhn, D. (1991). *The skills of argument*. Cambridge, UK: Cambridge University Press.

Kuhn, D., & Udell, W. (2003). The development of argument skills. *Child Development, 74*(5), 1245-1260.

Kuhn, D., Shaw, V., & Felton, M. (1997). Effects of dyadic interaction on argumentive reasoning. *Cognition and Instruction, 15*(3), 287-315.

Kunz, W., & Rittel, H. (1970). *Issues as elements of information systems. Working paper #131*. Institut für Grundlagen der Planung I.A. University of Stuttgart, Germany.

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Cambridge: Cambridge University Press. ISBN 0521423740; first published in 1990 as Institute for Research on Learning report 90-0013

Lee, J. (1991), Extending the Potts and Bruns model for recording design rationale. *Proceedings of the 13th International Conference on Software Engineering* (ICSE '13) (pp. 114-125). IEEE Computer Society Press, Los Alamitos, CA.

Leitão, S. (2000). The potential of argument in knowledge building. *Human Development, 43* (6), 332–360.

Linn, M. C., Clark, D., & Slotta, J. D. (2003). Wise design for knowledge integration. *Science Education*, *87*(4), 517–538.

Lowrance, J. D. (2007). Graphical manipulation of evidence in structured arguments. *Law, Probability & Risk, 6*(1–4), 225–240.

Lowrance, J., Harrison, I., Rodriguez, A., Yeh, E., Boyce, T., Murdock, J., Thomere, J., & Murray, K. (2008). Template-based structured argumentation. In A. Okada, S. Buckingham Shum & T. Sherborne (Eds.), *Knowledge cartography: Software tools and mapping techniques* (pp. 307–333), London, UK: Springer.

MacLean, A., Young, R. M., Bellotti, V. M. E. & Moran, T. P. (1991). Questions, Options and Criteria: Elements of Design Space Analysis. To appear in *Human-Computer Interaction*, Vol 6 (3&4). 1991 {Special Issue on Design Rationale: J. M. Carroll & T. P. Moran (Eds.)}.

Maclean, A., Young, R.M., Bellotti, V.M.E., Moran, T. (1996). Questions, Options, and Criteria: Elements of Design Space Analysis. In Moran, T., & Carroll, J., *Design Rationale Concepts, Techniques, and Use, Lawrence Erlbaum Associates* (pp. 53-106).

Malone, T. W., & Klein, M. (2007). Harnessing Collective Intelligence to Address Global Climate Change. *Innovations 2*(3), 15–26.

Marshall, C., & Shipman, F. (1997). Spatial Hypertext and the Practice of Information Triage. In *Proceedings of the 8th ACM Conference on Hypertext* (pp. 124-133). Southampton, UK.

Marshall, C. C., Halasz, F. G., Rogers, R. A., & Janssen, W. C. (1991). Aquanet: A hypertext tool to hold your knowledge in place. In *Proceedings of the 3rd annual ACM conference on Hypertext (HYPERTEXT '91)* (pp. 261–275). New York: ACM Press.

McAlister, S., Ravenscroft, A., & Scanlon, E. (2004). Combining interaction and context design to support collaborative argumentation using a tool for synchronous CMC. *Journal of Computer Assisted Learning: Special Issue: Developing Dialogue for Learning, 20*(3), 194–204.

McCall, R.J. (1991). PHI: A Conceptual Foundation for Design Hypermedia. *Design Studies 12*(1), 30–41.

McLaren, B. M., Scheuer, O., & Mikšátko, J. (in press). Supporting collaborative learning and ediscussions using artificial intelligence techniques. *International Journal of Artificial Intelligence in Education*.

Means M.L., & Voss, J.F. (1996). Who reasons well? Two studies of informal reasoning among children of different grade, ability, and knowledge levels. *Cognition and Instruction, 14(2),* 139-178.

Meeker, M., Devitt, S., & Wu, L. (2010a). Internet Trends. *In Morgan Stanley Research*, April 12. Retrieved from http://www.morganstanley.com/institutional/techresearch/pdfs/Internet_Trends_041210.pdf

Meeker, M., Devitt, S., & Wu, L. (2010b). Internet Trends. *In CM Summit, Morgan Stanley Research*. June 7, New York City, USA. Retrieved from http://www.morganstanley.com/institutional/techresearch/cm_summit_june2010.html

De Moor, A., & Aakhus, M. (2006). Argument support: From technologies to tools. Communications of the ACM, 49(3), 93-98

Munneke, L., van Amelsvoort, M., & Andriessen, J. (2003). The role of diagrams in collaborative argumentation-based learning. *International Journal of Educational Research, 39*(1–2), 113–131.

Nelson, T. (1965). A File Structure for the Complex, the Changing, and the Indeterminate. In *Proceedings of the 20th National Conference* (pp. 84-100). New York, NY: Association for Computing Machinery.

Nelson, T.H. (1981). *Literary Machines*. Swarthmore, Pa.: Self-published.

Pinkwart, N., Aleven, V., Ashley, K., & Lynch, C. (2006a). Toward legal argument instruction with graph grammars and collaborative filtering techniques. In M. Ikeda, K. Ashley, & T. W. Chan (Hrsg.), *Proceedings of the 8th International Conference on Intelligent Tutoring Systems (ITS 2006)* (pp. 227–236). Berlin, Germany: Springer.

Rahwan, I., Zablith, F. and Reed, C. (2007) Towards Large Scale Argumentation Support on the Semantic Web, *The 22nd Conference on Artificial Intelligence (AAAI)* (pp. 1446-1451). California, USA: AAAI Press.

Ranney, M., & Schank, P. (1998). Toward an integration of the social and the scientific: Observing, modeling, and promoting the explanatory coherence of reasoning. In S. Read & L. Miller (Eds.), *Connectionist models of social reasoning and social behaviour* (pp. 245–274). Mahwah, NJ: Lawrence Erlbaum.

Ravenscroft, A., Sagar, M., Baur, E., & Oriogun. P. (2008). Ambient pedagogies, meaningful learning and social software. In S. Hatzipanagos & S. Warburton (Eds.), *Social software & developing community ontologies* (pp. 432–450). Hershey, PA: IGI Global Publishing.

Reed, C., & Rowe, G. (2004). Araucaria: Software for argument analysis, diagramming and representation. *International Journal of AI Tools, 14*(3–4), 961–980.

Reznitskaya, A., Anderson, R. C., McNurlen, B., Nguyen-Jahiel, K., Archodidou, A., & Kim, S. (2001). Influence of oral discussion on Written Argument. *Discourse Processes*, *32*(2-3), 155-175.

Rolf, B., & Magnusson, C. (2002). Developing the art of argumentation. A software approach. In: Proceedings of *the 5th International Conference on Argumentation. International Society for the Study of Argumentation (ISSA-2002)* (pp. 919–926).

Sandoval, W. A., & Millwood, K. A. (2005). The quality of student's use of evidence in written scientific explanations. *Cognition and Instruction, 23*(1), 23-55.

Scheuer, O., Loll, F., Pinkwart, N. & McLaren, B.M. (2010). Computer- Supported Argumentation: A Review of the State of the Art. International Journal of Computer-Supported Collaborative Learning. 5(1).

Schneider, D. C., Voigt, C., & Betz, G. (2007). ArguNet - a software tool for collaborative argumentation analysis and research. *Paper presented at the 7th Workshop on Computational Models of Natural Argument* (*CMNA VII).*

Schwarz, B. B., & Glassner, A. (2007). The role of floor control and of ontology in argumentative activities with discussion-based tools. *International Journal of Computer-Supported Collaborative Learning (ijCSCL)*, *2*(4), 449–478.

Shipman, F.M., & McCall, R. (1994). Supporting knowledge-base evolution with incremental formalization. In *Proceedings of CHI'94 Conference* (pp. 285 291). Boston, MA.

Shipman, F.M. (1993). Supporting Knowledge-Base Evolution with Incremental Formalization. PhD Dissertation, Univ. of Colorado, Department of Computer Science, Boulder, CO.

Suthers, D. D., Weiner, A., Connelly, J., & Paolucci, M. (1995). Belvedere: Engaging students in critical discussion of science and public policy issues. In J. Greer (Ed.), *Proceedings of the 7th World Conference on Artificial Intelligence in Education (AI-ED 1995)* (pp. 266–273). Charlottesville, VA: Association for the Advancement of Computing in Education.

O'Reilly, T. (2005). What is Web 2.0 - Design patterns and business models for the next generation of software. *O'Reilly*. Retrieved on November 10, 2007, from http:// www.oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

Toulmin, S. E. (1958). *The uses of argument.* Cambridge, UK: Cambridge University Press.

Tsovaltzi, D., Rummel, N., McLaren, B. M., Pinkwart, N., Scheuer, O., Harrer, A., & Braun, I. (in press). Extending a virtual chemistry laboratory with a collaboration script to promote conceptual learning. *International Journal of Technology Enhanced Learning (IJTEL)*.

van den Braak, S., & Vreeswijk, G. (2006). AVER: Argument visualization for evidential reasoning. In T. M. van Engers (Ed.), *Proceedings of the 19th Conference on Legal Knowledge and Information Systems (JURIX 2006)* (pp. 151–156). Amsterdam, The Netherlands: IOS Press.

Van Eemeren, F., & Grootendorst, R. (1992). Argumentation, communication, and fallacies: a pragma-dialectical perspective. *Hillsdale,* NJ: Lawrence Erlbaum Associates.

Van Eemeren, F., & Grootendorst, R. (1999). Developments in Argumentation Theory. In J. Andriessen & P. Coirier (Eds.). *Foundations of argumentative text processing* (pp. 43-57). Amsterdam: Amsterdam University Press.

van Gelder, T. (2002). Argument mapping with Reason!Able. *The American Philosophical Association Newsletter on Philosophy and Computers, 2* (1), 85–90.

van Gelder, T. (2003). Enhancing deliberation through computer-supported argument visualization. In P. A. Kirschner, S. J. Buckingham-Shum & C. S. Carr (Eds.), *Visualizing argumentation: Software tools for collaborative and educational sense-making* (pp. 97–115). London, UK: Springer.

van Gelder, T. (2007). The rationale for Rationale. *Law, Probability and Risk, 6*(1–4), 23–42.

Verheij, B. (2003). Artificial argument assistants for defeasible argumentation. *Artificial Intelligence, 150*(1–2), 291–324.

Voss, J. F., Tyler, S. W., & Yengo, L. A. (1983). Individual differences in the solving of social science problems. In R. F. Dillon & R. R. Schmeck (Eds). *Individual differences in cognition* (pp. 204-232). New York, Academic Press.

Voss, J., & Means, M. (1991). Learning to reason via instruction in Argumentation. *Learning & instruction*, 1, 337-350.

Weinberger, A., & Fischer, F. (2006). A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers & Education, 46*(1), 71-95.

Ηλεκτρονική Βιβλιογραφία

[AcademicTalk] at http://www.londonmet.ac.uk/ltri/research/projects/at.htm

[Amazon] at http://www.amazon.com

[Android Developers] at http://developer.android.com/index.html

[Android OS] at http://www.android.com/

[App Store] at http://www.apple.com/iphone/features/app-store.html

[Araucaria software] at http://Araucaria.computing.dundee.ac.uk

[ArgDF] at http://argdf.org/

[Argue / ArguMed] at http://www.ai.rug.nl/~verheij/aaa/argumed3.htm

[Argumentum] at http://arg.umentum.com/

[ArguNet] at http://www.argunet.org

[Athena] at http://www.athenasoft.org

[bCisive 2] at http://bcisive.austhink.com/

[bCisive online] at http://www.bcisiveonline.com

[Belvedere v3 & v4] at http://lilt.ics.hawaii.edu/lilt/software/belvedere

[Blogspot] at http://www.blogspot.com

[C# Developer Center] at http://msdn.microsoft.com/en-us/vcsharp

[Carneades] at http://carneades.berlios.de

[CDMA Development Group] at http://www.cdg.org/

[changingminds: argument] at
http://www.changingminds.org/disciplines/argument/argument.htm

[ClaiMaker / ClaimFinder] at http://kmi.open.ac.uk/projects/scholonto

[CoFFEE] at http://www.coffee-soft.org/

[cohere] at http://cohere.open.ac.uk/index.php

[Collaboratorium] at http://cci.mit.edu/research/climate.html

[Convince Me] at http://www.soe.berkeley.edu/~schank/convinceme

[CreateDocument of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=createDocument

[Creative Commons] at http://creativecommons.org

[DebateGraph] at http://www.debategraph.org

[Debatepedia] at http://wiki.idebate.org

[Del.icio.us] at http://www.delicious.com

[Digalo] at http://www.dunes.gr

[ETSI, EDGE] at http://www.etsi.org/WebSite/Technologies/edge.aspx

[ETSI, General Packet Radio Service GPRS] at
http://www.etsi.org/WebSite/Technologies/gprs.aspx

[ETSI, HSCSD] at http://www.etsi.org/WebSite/Technologies/hscsd.aspx

[ETSI, Mobile technologies GSM] at http://www.etsi.org/WebSite/Technologies/gsm.aspx

[ETSI, UMTS] at http://www.etsi.org/WebSite/Technologies/UMTS.aspx

[ETSI, W-CDMA] at http://www.etsi.org/WebSite/Technologies/wcdma.aspx

[Facebook] at http://www.facebook.com

[Flickr] http://www.flickr.com

[FTP - IETF RFC 2821] at http://www.ietf.org/rfc/rfc959.txt

[Google Maps] at http://maps.google.com

[Hermes] at http://www-sop.inria.fr/aid/hermes

[IETF RFC 2396] at http://www.ietf.org/rfc/rfc2396.txt

[Interloc] at http://www.interloc.org

[International Telecommunications Union 2009] at http://www.itu.int/ITU-D/ict/material/Telecom09_flyer.pdf

[Ios Dev Center] at http://developer.apple.com/devcenter/ios/index.action

[iPad] at http://www.apple.com/ipad/

[iPhone 2G] at http://www.apple.com/pr/library/2007/01/09iphone.html

[iPhone] at http://www.apple.com/iphone/

[iTunes] at www.apple.com/itunes/

[Java] at http://www.java.com

[KIE/SenseMaker, WISE] at http://tels.sourceforge.net/sensemaker

[LoadCops of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=LoadCops

[LoadDocuments of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=LoadWorkSpaces

[LoadWorkSpaces of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=LoadDocuments

[Login of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=Login

[Macromedia] at http://www.macromedia.com

[MapQuest] at http://www.mapquest.com

[Microsoft] at http://www.microsoft.com

[Mindmeister in iTunes] at http://itunes.apple.com/app/mindmeister-for-ipad/id381073026?mt=8

[Mindmeister for iPhone] at http://www.mindmeister.com/iphone

[Mindmeister] at http://www.mindmeister.com

[MobiReady Report] http://mobiforge.com/

[Morgan Stanley] at http://www.morganstanley.com

[MySpace] at http://www.myspace.com

[Netscape] at http://www.aol.netscape.com

[O'reilly Media, What Is Web 2.0] at http://oreilly.com/pub/a/web2/archive/what-is-web-20.html

[OVA - Online visualization of argument] at http://ova.computing.dundee.ac.uk/

[Passani 2010 - Global Authoring Practices for the Mobile Web] at http://www.passani.it/gap/

[Rationale] at http://rationale.austhink.com/

[Reason] at http://rationale.austhink.com/reasonable

[RetrieveDocbyID of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=RetrieveDocbyID

[RetrieveDocByName of mobilews] at
http://ws.copeit.cti.gr/mobile.asmx?op=RetrieveDocByName

[Samsung Galaxy Tab] at http://www.samsung.com/us/mobile/galaxy-tab

[Samsung OMNIA SGH-i900] at http://omnia.samsungmobile.com/omnia/

[Samsung SPH-M4650 Review] at
http://pdadb.net/index.php?m=specs&id=932&c=samsung_sph-m4650_multi-touch

[SEAS] at http://www.ai.sri.com/~seas

[SMTP - IETF RFC 2821] at http://www.ietf.org/rfc/rfc2821.txt

[SOAP - W3C Simple Object Access Protocol] at http://www.w3.org/tr/SOAP

[TiVo] at http://www.tivo.com

[Twitter] at http://www.twitter.com

[Universal Description, Discovery and Integration of Web Services (UDDI)] at http://www.uddi.org

[Visual Studio] at http://www.microsoft.com/visualstudio/

[W3C Web Services Architecture Domain] at http://www.w3.org/2002/ws/

[W3C, Mobile Web Best Practices 1.0, 2008] at http://www.w3.org/TR/mobile-bp/

[W3C, mobileOK Scheme 1.0, 2009] at http://www.w3.org/TR/mobileOK/

[Windows Mobile OS] at http://www.microsoft.com/windowsmobile

[World Wide Web Consortium] at http://www.w3.org

[WSDL - W3C Web Services Description Language] at http://www.w3.org/tr/WSDL

[ThrowException of mobilews] at http://ws.copeit.cti.gr/mobile.asmx?op=throwException

ΠΑΡΑΡΤΗΜΑ

1. Web Service

1.1 ToWSDL αρχείο περιγραφής του mobile_ws (Mobile Web Service)

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:tns="http://copeit.cti.gr/"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/" targetNamespace="http://copeit.cti.gr/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://copeit.cti.gr/">
      <s:element name="Login">
        <s:complexType/>
      </s:element>
      <s:element name="LoginResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="LoginResult" type="s:int"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="AuthHeader" type="tns:AuthHeader"/>
      <s:complexType name="AuthHeader">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="Username" type="s:string"/>
          <s:element minOccurs="0" maxOccurs="1" name="password" type="s:string"/>
        </s:sequence>
      </s:complexType>
      <s:element name="LoadCops">
        <s:complexType/>
      </s:element>
      <s:element name="LoadCopsResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="LoadCopsResult" type="tns:ArrayOfCop"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="ArrayOfCop">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="Cop" nillable="true" type="tns:Cop"/>
        </s:sequence>
      </s:complexType>
      <s:complexType name="Cop">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="cop_id" type="s:int"/>
          <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string"/>
          <s:element minOccurs="0" maxOccurs="1" name="description" type="s:string"/>
          <s:element minOccurs="0" maxOccurs="1" name="logo" type="s:base64Binary"/>
        </s:sequence>
```

```
    </s:complexType>
    <s:element name="LoadWorkSpaces">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="id" type="s:int"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="LoadWorkSpacesResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="LoadWorkSpacesResult"
type="tns:ArrayOfWorkSpace"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:complexType name="ArrayOfWorkSpace">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="unbounded" name="WorkSpace" nillable="true"
type="tns:WorkSpace"/>
      </s:sequence>
    </s:complexType>
    <s:complexType name="WorkSpace">
      <s:sequence>
        <s:element minOccurs="1" maxOccurs="1" name="id" type="s:int"/>
        <s:element minOccurs="1" maxOccurs="1" name="creator_id" type="s:int"/>
        <s:element minOccurs="1" maxOccurs="1" name="creationDate" type="s:dateTime"/>
        <s:element minOccurs="1" maxOccurs="1" name="modificationDate" type="s:dateTime"/>
        <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string"/>
        <s:element minOccurs="1" maxOccurs="1" name="prive" type="s:int"/>
      </s:sequence>
    </s:complexType>
    <s:element name="LoadDocuments">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="id" type="s:int"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="LoadDocumentsResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="LoadDocumentsResult" type="tns:ArrayOfDocument"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:complexType name="ArrayOfDocument">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="unbounded" name="Document" nillable="true"
type="tns:Document"/>
      </s:sequence>
    </s:complexType>
    <s:complexType name="Document">
      <s:sequence>
        <s:element minOccurs="1" maxOccurs="1" name="id" type="s:int"/>
        <s:element minOccurs="1" maxOccurs="1" name="creator_id" type="s:int"/>
        <s:element minOccurs="1" maxOccurs="1" name="creationDate" type="s:dateTime"/>
```

```
      <s:element minOccurs="1" maxOccurs="1" name="modificationDate" type="s:dateTime"/>
      <s:element minOccurs="0" maxOccurs="1" name="title" type="s:string"/>
      <s:element minOccurs="0" maxOccurs="1" name="mimetype" type="s:string"/>
      <s:element minOccurs="0" maxOccurs="1" name="style" type="s:string"/>
      <s:element minOccurs="1" maxOccurs="1" name="x" type="s:int"/>
      <s:element minOccurs="1" maxOccurs="1" name="y" type="s:int"/>
      <s:element minOccurs="1" maxOccurs="1" name="height" type="s:int"/>
      <s:element minOccurs="1" maxOccurs="1" name="width" type="s:int"/>
      <s:element minOccurs="0" maxOccurs="1" name="cnt" type="s:base64Binary"/>
    </s:sequence>
  </s:complexType>
  <s:element name="createDocument">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="pdoc" type="tns:Document"/>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="createDocumentResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="1" maxOccurs="1" name="createDocumentResult" type="s:int"/>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="RetrieveDocByName">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="objectname" type="s:string"/>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="RetrieveDocByNameResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="RetrieveDocByNameResult" type="tns:Resource"/>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:complexType name="Resource">
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="cnt" type="s:base64Binary"/>
      <s:element minOccurs="0" maxOccurs="1" name="metadata" type="tns:ArrayOfTAttributeValue"/>
    </s:sequence>
  </s:complexType>
  <s:complexType name="ArrayOfTAttributeValue">
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="unbounded" name="TAttributeValue" nillable="true"
type="tns:TAttributeValue"/>
    </s:sequence>
  </s:complexType>
  <s:complexType name="TAttributeValue">
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="name" type="s:string"/>
      <s:element minOccurs="0" maxOccurs="1" name="value" type="s:string"/>
    </s:sequence>
  </s:complexType>
```

```
    <s:element name="RetrieveDocbyID">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="id" type="s:int"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="RetrieveDocbyIDResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="RetrieveDocbyIDResult" type="tns:Resource"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="throwException">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="errorMsg" type="s:string"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="throwExceptionResponse">
      <s:complexType/>
    </s:element>
  </s:schema>
</wsdl:types>
<wsdl:message name="LoginSoapIn">
  <wsdl:part name="parameters" element="tns:Login"/>
</wsdl:message>
<wsdl:message name="LoginSoapOut">
  <wsdl:part name="parameters" element="tns:LoginResponse"/>
</wsdl:message>
<wsdl:message name="LoginAuthHeader">
  <wsdl:part name="AuthHeader" element="tns:AuthHeader"/>
</wsdl:message>
<wsdl:message name="LoadCopsSoapIn">
  <wsdl:part name="parameters" element="tns:LoadCops"/>
</wsdl:message>
<wsdl:message name="LoadCopsSoapOut">
  <wsdl:part name="parameters" element="tns:LoadCopsResponse"/>
</wsdl:message>
<wsdl:message name="LoadCopsAuthHeader">
  <wsdl:part name="AuthHeader" element="tns:AuthHeader"/>
</wsdl:message>
<wsdl:message name="LoadWorkSpacesSoapIn">
  <wsdl:part name="parameters" element="tns:LoadWorkSpaces"/>
</wsdl:message>
<wsdl:message name="LoadWorkSpacesSoapOut">
  <wsdl:part name="parameters" element="tns:LoadWorkSpacesResponse"/>
</wsdl:message>
<wsdl:message name="LoadWorkSpacesAuthHeader">
  <wsdl:part name="AuthHeader" element="tns:AuthHeader"/>
</wsdl:message>
<wsdl:message name="LoadDocumentsSoapIn">
  <wsdl:part name="parameters" element="tns:LoadDocuments"/>
</wsdl:message>
<wsdl:message name="LoadDocumentsSoapOut">
```

```
  <wsdl:part name="parameters" element="tns:LoadDocumentsResponse"/>
 </wsdl:message>
 <wsdl:message name="LoadDocumentsAuthHeader">
  <wsdl:part name="AuthHeader" element="tns:AuthHeader"/>
 </wsdl:message>
 <wsdl:message name="createDocumentSoapIn">
  <wsdl:part name="parameters" element="tns:createDocument"/>
 </wsdl:message>
 <wsdl:message name="createDocumentSoapOut">
  <wsdl:part name="parameters" element="tns:createDocumentResponse"/>
 </wsdl:message>
 <wsdl:message name="createDocumentAuthHeader">
  <wsdl:part name="AuthHeader" element="tns:AuthHeader"/>
 </wsdl:message>
 <wsdl:message name="RetrieveDocByNameSoapIn">
  <wsdl:part name="parameters" element="tns:RetrieveDocByName"/>
 </wsdl:message>
 <wsdl:message name="RetrieveDocByNameSoapOut">
  <wsdl:part name="parameters" element="tns:RetrieveDocByNameResponse"/>
 </wsdl:message>
 <wsdl:message name="RetrieveDocbyIDSoapIn">
  <wsdl:part name="parameters" element="tns:RetrieveDocbyID"/>
 </wsdl:message>
 <wsdl:message name="RetrieveDocbyIDSoapOut">
  <wsdl:part name="parameters" element="tns:RetrieveDocbyIDResponse"/>
 </wsdl:message>
 <wsdl:message name="throwExceptionSoapIn">
  <wsdl:part name="parameters" element="tns:throwException"/>
 </wsdl:message>
 <wsdl:message name="throwExceptionSoapOut">
  <wsdl:part name="parameters" element="tns:throwExceptionResponse"/>
 </wsdl:message>
 <wsdl:portType name="mobileSoap">
  <wsdl:operation name="Login">
   <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Try to login to
CoPe_It!.</wsdl:documentation>
   <wsdl:input message="tns:LoginSoapIn"/>
   <wsdl:output message="tns:LoginSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="LoadCops">
   <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Loads a list of the Communities in
which the requestor has at least 'view' privileges.</wsdl:documentation>
   <wsdl:input message="tns:LoadCopsSoapIn"/>
   <wsdl:output message="tns:LoadCopsSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="LoadWorkSpaces">
   <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Loads the WorkSpaces of a particular
Community.</wsdl:documentation>
   <wsdl:input message="tns:LoadWorkSpacesSoapIn"/>
   <wsdl:output message="tns:LoadWorkSpacesSoapOut"/>
  </wsdl:operation>
  <wsdl:operation name="LoadDocuments">
   <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Loads the content of a particular
WorkSpace.</wsdl:documentation>
   <wsdl:input message="tns:LoadDocumentsSoapIn"/>
   <wsdl:output message="tns:LoadDocumentsSoapOut"/>
  </wsdl:operation>
```

<wsdl:operation name="createDocument">
<wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Stores a resource into the repository along with a number of attribute-value pairs. Checks if all mandatory attributes are provided. Currently, object-name is the only mandatory attributes.</wsdl:documentation>
<wsdl:input message="tns:createDocumentSoapIn"/>
<wsdl:output message="tns:createDocumentSoapOut"/>
</wsdl:operation>
<wsdl:operation name="RetrieveDocByName">
<wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Access a stored resource according to its unique name.</wsdl:documentation>
<wsdl:input message="tns:RetrieveDocByNameSoapIn"/>
<wsdl:output message="tns:RetrieveDocByNameSoapOut"/>
</wsdl:operation>
<wsdl:operation name="RetrieveDocbyID">
<wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Access a stored resource according to its unique ID.</wsdl:documentation>
<wsdl:input message="tns:RetrieveDocbyIDSoapIn"/>
<wsdl:output message="tns:RetrieveDocbyIDSoapOut"/>
</wsdl:operation>
<wsdl:operation name="throwException">
<wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">Web Method used to return error messages</wsdl:documentation>
<wsdl:input message="tns:throwExceptionSoapIn"/>
<wsdl:output message="tns:throwExceptionSoapOut"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="mobileSoap" type="tns:mobileSoap">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="Login">
<soap:operation soapAction="http://copeit.cti.gr/Login" style="document"/>
<wsdl:input>
<soap:body use="literal"/>
<soap:header message="tns:LoginAuthHeader" part="AuthHeader" use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="LoadCops">
<soap:operation soapAction="http://copeit.cti.gr/LoadCops" style="document"/>
<wsdl:input>
<soap:body use="literal"/>
<soap:header message="tns:LoadCopsAuthHeader" part="AuthHeader" use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="LoadWorkSpaces">
<soap:operation soapAction="http://copeit.cti.gr/LoadWorkSpaces" style="document"/>
<wsdl:input>
<soap:body use="literal"/>
<soap:header message="tns:LoadWorkSpacesAuthHeader" part="AuthHeader" use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>

```
  </wsdl:operation>
  <wsdl:operation name="LoadDocuments">
    <soap:operation soapAction="http://copeit.cti.gr/LoadDocuments" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
      <soap:header message="tns:LoadDocumentsAuthHeader" part="AuthHeader" use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="createDocument">
    <soap:operation soapAction="http://copeit.cti.gr/createDocument" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
      <soap:header message="tns:createDocumentAuthHeader" part="AuthHeader" use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="RetrieveDocByName">
    <soap:operation soapAction="http://copeit.cti.gr/RetrieveDocByName" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="RetrieveDocbyID">
    <soap:operation soapAction="http://copeit.cti.gr/RetrieveDocbyID" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="throwException">
    <soap:operation soapAction="http://copeit.cti.gr/throwException" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="mobileSoap12" type="tns:mobileSoap">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="Login">
    <soap12:operation soapAction="http://copeit.cti.gr/Login" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
      <soap12:header message="tns:LoginAuthHeader" part="AuthHeader" use="literal"/>
    </wsdl:input>
```

```
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="LoadCops">
  <soap12:operation soapAction="http://copeit.cti.gr/LoadCops" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
    <soap12:header message="tns:LoadCopsAuthHeader" part="AuthHeader" use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="LoadWorkSpaces">
  <soap12:operation soapAction="http://copeit.cti.gr/LoadWorkSpaces" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
    <soap12:header message="tns:LoadWorkSpacesAuthHeader" part="AuthHeader" use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="LoadDocuments">
  <soap12:operation soapAction="http://copeit.cti.gr/LoadDocuments" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
    <soap12:header message="tns:LoadDocumentsAuthHeader" part="AuthHeader" use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="createDocument">
  <soap12:operation soapAction="http://copeit.cti.gr/createDocument" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
    <soap12:header message="tns:createDocumentAuthHeader" part="AuthHeader" use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RetrieveDocByName">
  <soap12:operation soapAction="http://copeit.cti.gr/RetrieveDocByName" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="RetrieveDocbyID">
  <soap12:operation soapAction="http://copeit.cti.gr/RetrieveDocbyID" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
```

```
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="throwException">
    <soap12:operation soapAction="http://copeit.cti.gr/throwException" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
 </wsdl:binding>
 <wsdl:service name="mobile">
  <wsdl:port name="mobileSoap" binding="tns:mobileSoap">
    <soap:address location="http://ws.copeit.cti.gr/mobile.asmx"/>
  </wsdl:port>
  <wsdl:port name="mobileSoap12" binding="tns:mobileSoap12">
    <soap12:address location="http://ws.copeit.cti.gr/mobile.asmx"/>
  </wsdl:port>
 </wsdl:service>
</wsdl:definitions>
```

1.2 Το XML αρχείο περιγραφής των 8 μεθόδων του mobilews (Mobile Web Service)

Μέθοδος LoadCops (Δίνοντας ως είσοδο το Username και τον το Password, μας επιστρέφεται μία λίστα από Communities, τα οποία έχει δικαίωμα να επισκεφθεί)

REQUEST MESSAGE
SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/LoadCops"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <LoadCops xmlns="http://copeit.cti.gr/" />
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoadCopsResponse xmlns="http://copeit.cti.gr/">
      <LoadCopsResult>
        <Cop>
          <cop_id>int</cop_id>
          <title>string</title>
          <description>string</description>
          <logo>base64Binary</logo>
        </Cop>
        <Cop>
          <cop_id>int</cop_id>
          <title>string</title>
          <description>string</description>
          <logo>base64Binary</logo>
        </Cop>
      </LoadCopsResult>
    </LoadCopsResponse>
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap12:Header>
  <soap12:Body>
    <LoadCops xmlns="http://copeit.cti.gr/" />
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoadCopsResponse xmlns="http://copeit.cti.gr/">
      <LoadCopsResult>
        <Cop>
          <cop_id>int</cop_id>
          <title>string</title>
          <description>string</description>
          <logo>base64Binary</logo>
        </Cop>
        <Cop>
          <cop_id>int</cop_id>
          <title>string</title>
          <description>string</description>
          <logo>base64Binary</logo>
        </Cop>
      </LoadCopsResult>
    </LoadCopsResponse>
  </soap12:Body>
</soap12:Envelope>
```

Μέθοδος LoadDocuments (διαλέγοντας ένα Workspace φορτώνεται το περιεχόμενο του)

REQUEST MESSAGE

SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/LoadDocuments"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <LoadDocuments xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </LoadDocuments>
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoadDocumentsResponse xmlns="http://copeit.cti.gr/">
      <LoadDocumentsResult>
        <Document>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <mimetype>string</mimetype>
          <style>string</style>
          <x>int</x>
          <y>int</y>
          <height>int</height>
          <width>int</width>
          <cnt>base64Binary</cnt>
        </Document>
        <Document>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <mimetype>string</mimetype>
          <style>string</style>
          <x>int</x>
          <y>int</y>
          <height>int</height>
          <width>int</width>
          <cnt>base64Binary</cnt>
        </Document>
      </LoadDocumentsResult>
    </LoadDocumentsResponse>
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap12:Header>
```

```
  <soap12:Body>
    <LoadDocuments xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </LoadDocuments>
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoadDocumentsResponse xmlns="http://copeit.cti.gr/">
      <LoadDocumentsResult>
        <Document>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <mimetype>string</mimetype>
          <style>string</style>
          <x>int</x>
          <y>int</y>
          <height>int</height>
          <width>int</width>
          <cnt>base64Binary</cnt>
        </Document>
        <Document>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <mimetype>string</mimetype>
          <style>string</style>
          <x>int</x>
          <y>int</y>
          <height>int</height>
          <width>int</width>
          <cnt>base64Binary</cnt>
        </Document>
      </LoadDocumentsResult>
    </LoadDocumentsResponse>
  </soap12:Body>
</soap12:Envelope>
```

<u>Μέθοδος LoadWorkSpaces</u> (δίνοντας ως είσοδο το id του Community, μας επιστρέφεται μία λίστα των Workspaces του συγκεκριμένου Community)

<u>REQUEST MESSAGE</u>
SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/LoadWorkSpaces"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <LoadWorkSpaces xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </LoadWorkSpaces>
  </soap:Body>
</soap:Envelope>
```

<u>RESPONSE MESSAGE</u>

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoadWorkSpacesResponse xmlns="http://copeit.cti.gr/">
      <LoadWorkSpacesResult>
        <WorkSpace>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <prive>int</prive>
        </WorkSpace>
        <WorkSpace>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <prive>int</prive>
        </WorkSpace>
      </LoadWorkSpacesResult>
    </LoadWorkSpacesResponse>
```

```
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap12:Header>
  <soap12:Body>
    <LoadWorkSpaces xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </LoadWorkSpaces>
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoadWorkSpacesResponse xmlns="http://copeit.cti.gr/">
      <LoadWorkSpacesResult>
        <WorkSpace>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <prive>int</prive>
        </WorkSpace>
        <WorkSpace>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <prive>int</prive>
        </WorkSpace>
      </LoadWorkSpacesResult>
    </LoadWorkSpacesResponse>
  </soap12:Body>
</soap12:Envelope>
```

<u>Μέθοδος Login</u> (δίνοντας σαν είσοδο το Username και το Password, μας κάνει Log-in στο Cope_it!)

<u>REQUEST MESSAGE</u>
SOAP 1.1
The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/Login"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <Login xmlns="http://copeit.cti.gr/" />
  </soap:Body>
</soap:Envelope>
```

<u>RESPONSE MESSAGE</u>

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LoginResponse xmlns="http://copeit.cti.gr/">
      <LoginResult>int</LoginResult>
    </LoginResponse>
  </soap:Body>
</soap:Envelope>
```

<u>REQUEST MESSAGE</u>
SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap12:Header>
  <soap12:Body>
    <Login xmlns="http://copeit.cti.gr/" />
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <LoginResponse xmlns="http://copeit.cti.gr/">
      <LoginResult>int</LoginResult>
    </LoginResponse>
  </soap12:Body>
</soap12:Envelope>
```

Μέθοδος RetrieveDocByName (δυνατότητα να έχουμε πρόσβαση σε ένα καταχωρημένο στοιχείο σύμφωνα με το μοναδικό του όνομά)

REQUEST MESSAGE

SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/RetrieveDocByName"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <RetrieveDocByName xmlns="http://copeit.cti.gr/">
      <objectname>string</objectname>
    </RetrieveDocByName>
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
```

```
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <RetrieveDocByNameResponse xmlns="http://copeit.cti.gr/">
      <RetrieveDocByNameResult>
        <cnt>base64Binary</cnt>
        <metadata>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
        </metadata>
      </RetrieveDocByNameResult>
    </RetrieveDocByNameResponse>
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <RetrieveDocByName xmlns="http://copeit.cti.gr/">
      <objectname>string</objectname>
    </RetrieveDocByName>
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <RetrieveDocByNameResponse xmlns="http://copeit.cti.gr/">
      <RetrieveDocByNameResult>
        <cnt>base64Binary</cnt>
        <metadata>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
```

```
          </TAttributeValue>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
        </metadata>
      </RetrieveDocByNameResult>
    </RetrieveDocByNameResponse>
  </soap12:Body>
</soap12:Envelope>
```

Μέθοδος RetrieveDocbyID (δίνεται πρόσβαση σε ένα καταχωρημένο στοιχείο σύμφωνα με το μοναδικό του id)

REQUEST MESSAGE

SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/RetrieveDocbyID"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <RetrieveDocbyID xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </RetrieveDocbyID>
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length


<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <RetrieveDocbyIDResponse xmlns="http://copeit.cti.gr/">
      <RetrieveDocbyIDResult>
        <cnt>base64Binary</cnt>
        <metadata>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
```

```
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
        </metadata>
      </RetrieveDocbyIDResult>
    </RetrieveDocbyIDResponse>
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <RetrieveDocbyID xmlns="http://copeit.cti.gr/">
      <id>int</id>
    </RetrieveDocbyID>
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <RetrieveDocbyIDResponse xmlns="http://copeit.cti.gr/">
      <RetrieveDocbyIDResult>
        <cnt>base64Binary</cnt>
        <metadata>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
          <TAttributeValue>
            <name>string</name>
            <value>string</value>
          </TAttributeValue>
        </metadata>
      </RetrieveDocbyIDResult>
    </RetrieveDocbyIDResponse>
  </soap12:Body>
</soap12:Envelope>
```

Μέθοδος CreateDocument (καταχωρείται ένα γνωστικό αντικείμενο συμπεριφοράς μαζί με ένα σύνολο από ζευγάρια τύπου όνομα-τιμή. Στην συνέχεια ελέγχει εάν παρέχονται όλα τα υποχρεωτικά στοιχεία)

REQUEST MESSAGE

SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/createDocument"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <AuthHeader xmlns="http://copeit.cti.gr/">
      <Username>string</Username>
      <password>string</password>
    </AuthHeader>
  </soap:Header>
  <soap:Body>
    <createDocument xmlns="http://copeit.cti.gr/">
      <pdoc>
        <id>int</id>
        <creator_id>int</creator_id>
        <creationDate>dateTime</creationDate>
        <modificationDate>dateTime</modificationDate>
        <title>string</title>
        <mimetype>string</mimetype>
        <style>string</style>
        <x>int</x>
        <y>int</y>
        <height>int</height>
        <width>int</width>
        <cnt>base64Binary</cnt>
      </pdoc>
    </createDocument>
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <createDocumentResponse xmlns="http://copeit.cti.gr/">
      <createDocumentResult>int</createDocumentResult>
    </createDocumentResponse>
```

```
    </soap:Body>
  </soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
  POST /mobile.asmx HTTP/1.1
  Host: ws.copeit.cti.gr
  Content-Type: application/soap+xml; charset=utf-8
  Content-Length: length

  <?xml version="1.0" encoding="utf-8"?>
  <soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap12:Header>
      <AuthHeader xmlns="http://copeit.cti.gr/">
        <Username>string</Username>
        <password>string</password>
      </AuthHeader>
    </soap12:Header>
    <soap12:Body>
      <createDocument xmlns="http://copeit.cti.gr/">
        <pdoc>
          <id>int</id>
          <creator_id>int</creator_id>
          <creationDate>dateTime</creationDate>
          <modificationDate>dateTime</modificationDate>
          <title>string</title>
          <mimetype>string</mimetype>
          <style>string</style>
          <x>int</x>
          <y>int</y>
          <height>int</height>
          <width>int</width>
          <cnt>base64Binary</cnt>
        </pdoc>
      </createDocument>
    </soap12:Body>
  </soap12:Envelope>
```

RESPONSE MESSAGE

```
  HTTP/1.1 200 OK
  Content-Type: application/soap+xml; charset=utf-8
  Content-Length: length

  <?xml version="1.0" encoding="utf-8"?>
  <soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap12:Body>
      <createDocumentResponse xmlns="http://copeit.cti.gr/">
        <createDocumentResult>int</createDocumentResult>
      </createDocumentResponse>
    </soap12:Body>
  </soap12:Envelope>
```

Μέθοδος ThrowException (μας επιστρέψει μήνυμα λάθους σε περίπτωση σφάλματος εισόδου δεδομένων ή λάθος εκτέλεσης από το σύστημα)

REQUEST MESSAGE

SOAP 1.1

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://copeit.cti.gr/throwException"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <throwException xmlns="http://copeit.cti.gr/">
      <errorMsg>string</errorMsg>
    </throwException>
  </soap:Body>
</soap:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <throwExceptionResponse xmlns="http://copeit.cti.gr/" />
  </soap:Body>
</soap:Envelope>
```

REQUEST MESSAGE

SOAP 1.2

```
POST /mobile.asmx HTTP/1.1
Host: ws.copeit.cti.gr
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <throwException xmlns="http://copeit.cti.gr/">
      <errorMsg>string</errorMsg>
    </throwException>
  </soap12:Body>
</soap12:Envelope>
```

RESPONSE MESSAGE

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <throwExceptionResponse xmlns="http://copeit.cti.gr/" />
  </soap12:Body>
</soap12:Envelope>
```

1.3 Ο κώδικας υλοποίησης του mobilews σε C# (.cs αρχεία)

mobile.cs

```
using System;
using System.Web;
using System.Collections;
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Collections.Generic;
using System.Xml;
using System.Xml.Serialization;

using gr.cti.tel.copeitWebService;

/// <summary>
/// Summary description for userManagement
/// </summary>
[WebService(Namespace = "http://copeit.cti.gr/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class mobile : System.Web.Services.WebService
{

    public AuthHeader sHeader;

    public mobile()
    {

        //Uncomment the following line if using designed components
        //InitializeComponent();
    }

    [WebMethod(Description = "Try to login to CoPe_It!.")]
    [XmlInclude(typeof(AuthHeader))]
    [SoapHeader("sHeader")]
    public int Login()
    {
        if (sHeader == null)
            throwException("no credentials given");

        serviceHandler SH = new serviceHandler();
        int uid = SH.authenticateUser(sHeader.Username, sHeader.password);
        if (uid < 0)
```

```
            throwException("Wrong username and password");

        return uid;
    }

    [WebMethod(Description = "Loads a list of the Communities in which the requestor has at least 'view'
privileges.")]
    [XmlInclude(typeof(AuthHeader))]
    [SoapHeader("sHeader")]
    public CopList LoadCops()
    {
        if (sHeader == null)
            throwException("no credentials given");

        serviceHandler SH = new serviceHandler();
        int uid = SH.authenticateUser(sHeader.Username, sHeader.password);
        if (uid < 0)
            throwException("Wrong username and password");

        CopList rs = new CopList();

        return (SH.LoadCops(uid));

        //return null;

    }

    [WebMethod(Description = "Loads the WorkSpaces of a particular Community.")]
    [XmlInclude(typeof(AuthHeader))]
    [SoapHeader("sHeader")]
    [XmlInclude(typeof(WorkSpace))]
    [XmlInclude(typeof(WorkSpaceList))]
    public WorkSpaceList LoadWorkSpaces(int id)
    {
        if (sHeader == null)
            throwException("no credentials given");

        serviceHandler SH = new serviceHandler();
        int uid = SH.authenticateUser(sHeader.Username, sHeader.password);
        if (uid < 0)
            throwException("Wrong username and password");
        WorkSpaceList workspaces = new WorkSpaceList();
        workspaces = SH.loadWorkSpaces(id, uid);

        return workspaces;
    }

    [WebMethod(Description = "Loads the content of a particular WorkSpace.")]
    [XmlInclude(typeof(AuthHeader))]
    [SoapHeader("sHeader")]
    [XmlInclude(typeof(Document))]
    [XmlInclude(typeof(DocumentList))]
    public DocumentList LoadDocuments(int id)
    {
        if (sHeader == null)
            throwException("no credentials given");
```

```
        serviceHandler SH = new serviceHandler();
        int uid = SH.authenticateUser(sHeader.Username, sHeader.password);
        if (uid < 0)
            throwException("Wrong username and password");
        DocumentList docs = new DocumentList();
        docs = SH.loadDocuments(id, uid);

        return docs;
    }

        [WebMethod(Description = "Stores a resource into the repository along with a number of attribute-value
pairs. Checks if all mandatory attributes are provided. Currently, object-name is the only mandatory attributes.")]
    [XmlInclude(typeof(AuthHeader))]
    [XmlInclude(typeof(Document))]
    [SoapHeader("sHeader")]
    public int PostDocument(Document doc)
    {
        if (sHeader == null)
            throwException("no credentials given");
        if (content==null)
            throwException("");

        serviceHandler SH = new serviceHandler();
        int uid = SH.authenticateUser(sHeader.Username, sHeader.password);
        if (uid < 0)
            throwException("Wrong username and password");

        doc.creator_id = uid;
        int rs = SH.Store(doc.cnt[], doc);

        return rs;
    }

    [WebMethod(Description = "Access a stored resource according to its unique name.")]
    public Resource RetrieveDocByName(string objectname)
    {
        return null;
    }

    [WebMethod(Description = "Access a stored resource according to its unique ID.")]
    public Resource RetrieveDocbyID(int id)
    {
        return null;
    }


    [WebMethod(Description = "Web Method used to return error messages")]
    public void throwException(string errorMsg)
    {
        SoapException se;
        if (errorMsg == null)
            se = new SoapException("Fault occurred", SoapException.ClientFaultCode,
Context.Request.Url.AbsoluteUri, (XmlNode)null);
        else
            se = new SoapException("Fault occurred: " + errorMsg, SoapException.ClientFaultCode,
Context.Request.Url.AbsoluteUri, (XmlNode)null);
        throw se;
```

```
    }

}
```

serviceHandler.cs

```
using System;
using System.Data;
using System.Configuration;
/*
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
*/
using System.IO;
using System.Collections.Generic;

using gr.cti.tel.CollaborationSpaceNS;
using gr.cti.tel.CoPManagement;
using gr.cti.tel.Domain;
using gr.cti.tel.copeit;
using gr.cti.tel.Common;

namespace gr.cti.tel.copeitWebService
{
    /// <summary>
    /// Summary description for serviceHandler
    /// </summary>
    public class serviceHandler
    {
        private Tcopeit ci;

        public serviceHandler()
        {
            ci = new Tcopeit(@ConfigurationSettings.AppSettings["DBConnectionString"]);
        }

        #region "generic"
        public int authenticateUser(string username, string password)
        {
            TUser u = new TUser();
            u.SetAttributeValue("Name", username);
            u.SetAttributeValue("Password", password);
            ci.IdentifyUser(u);
            return (u.id_sysobject);
        }
        #endregion

       #region "repository"
```

```
public int Store(byte[] content, TDocument doc)
{
   doc.filename = ci.storagePath + "/" + Guid.NewGuid().ToString();

   try
   {
      FileStream fStream = new FileStream(doc.filename, FileMode.CreateNew);
      BinaryWriter bw = new BinaryWriter(fStream);
      bw.Write(content);
      bw.Close();
      fStream.Close();
   }
   catch (Exception)
   {
      return (-5);
   }

   return (0);
}

public byte[] get(int resourceID, int callerID)
{
   TDocument doc = new TDocument();
   doc.id = resourceID;
   int rs = ci.loadDocument(doc, callerID);
   if (rs < 0)
      return null;
   byte[] f;// = new byte[];
   try
   {
      FileStream fs = new FileStream(doc.filename, FileMode.Open, FileAccess.Read);
      BinaryReader r = new BinaryReader(fs);
      int bufferSize = 1024;
      byte[] buffer = new byte[1024];
      f = new byte[fs.Length];
      for (int i = 0; i < (fs.Length / bufferSize); i++)
      {
         buffer = r.ReadBytes(bufferSize);
         Array.Copy(buffer, f, bufferSize);
         //Response.BinaryWrite(buffer);
      }

      if ((fs.Length % bufferSize) > 0)
      {
         buffer = r.ReadBytes((int)fs.Length % bufferSize);
         Array.Copy(buffer, f, (int)fs.Length % bufferSize);
         //Response.BinaryWrite(buffer);
      }
      fs.Close();
   }
   catch (Exception)
   {
      return null;
   }

   Document d = new Document(doc);
   d.cnt = f;
```

```
    return (f);
}

public int importFromCompendium(int spaceID, int copID, int callerId, byte[] thisFile)
{
    MemoryStream memStream = new MemoryStream(thisFile.Length);
    memStream.Write(thisFile, 0, thisFile.Length);
    int rs = ci.importCompendiumSpaceFull(memStream, copID, callerId);
    memStream.Close();
    return (rs);
}

public byte[] ExportToCompendium(int spaceID, int callerId, Options opt)
{
    TCanvas cnv = new TCanvas();
    cnv.id = spaceID;
    ci.LoadCollborationSpace(cnv, 2);
    ci.createCompendiumSpace(cnv, "http://copeit.cti.gr", true, 2);
    byte[] rs = (byte[])ci.LastResponse.iObjectSet[0];
    return (rs);

}

public byte[] export(int spaceID, Options opt, int callerID)
{
    TCanvas cnv = new TCanvas();
    cnv.id = spaceID;
    ci.toXML(cnv, callerID);
    MemoryStream ms = new MemoryStream();
    ms.Write((byte[])ci.LastResponse.iObjectSet[0], 0, ((byte[])ci.LastResponse.iObjectSet[0]).Length);
    ms.Position = 0;
    byte[] data = ms.ToArray();
    return (data);
}

public int import(int spaceID, int copID, byte[] cnt, Options opt, int callerID)
{
    MemoryStream ms = new MemoryStream();
    ms.Read(cnt, 0, cnt.Length);
    ms.Position = 0;
    int rs = ci.fromXML(copID, ms, callerID);
    return (rs);
}

#endregion

#region "mobile"

public WorkSpaceList loadWorkSpaces(int id, int callerID)
{
    TCanvasList cl =new TCanvasList();
    ci.LoadCoPCollborationSpaces(cl, id, callerID, true);

    WorkSpaceList wl = new WorkSpaceList();
    for (int i = 0; i < cl.Count; i++)
    {
```

```
        WorkSpace w = new WorkSpace(cl[i]);
        wl.Add(w);
    }

    return wl;
}

public DocumentList loadDocuments(int id, int callerID)
{
    TCanvas c = new TCanvas();
    c.id = id;
    ci.LoadCollborationSpace(c, callerID);

    TDocumentList dl = new TDocumentList();
    ci.getDocuments(c, dl, callerID);

    DocumentList documents = new DocumentList();
    for (int i = 0; i < dl.Count; i++)
    {
        Document d = new Document(dl[i]);
        documents.Add(d);
    }

    return documents;
}

public int createDocument(Document pdoc, int callerID)
{
                        TDocument doc = new TDocument();
                        doc.style = pdoc.style;
    doc.title = pdoc.title;
    doc.MIMEType = pdoc.mimetype;
    doc.title = System.Web.HttpContext.Current.Server.HtmlEncode(pdoc.title);
    doc.positionX = 1;
    doc.positionY = 1;
    //doc.width = width;
    //doc.height = height;
    doc.parent_id = pdoc.width;
    doc.creatot_id = uid;

    doc.filename = ci.storagePath + "/" + Guid.NewGuid().ToString();

    try
    {
        FileStream fStream = new FileStream(doc.filename, FileMode.CreateNew);
        BinaryWriter bw = new BinaryWriter(fStream);
        bw.Write(pdoc.cnt);
        bw.Close();
        fStream.Close();
    }
    catch (Exception)
    {
        return (-5);
    }

    return (0);
    return ci.createNewDocument(doc, callerID);
```

```
                }

        #endregion
    }
}
```

commonr.cs

```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

using System.Web.Services;
using System.Web.Services.Protocols;
using System.Collections.Generic;

using gr.cti.tel.CollaborationSpaceNS;
using gr.cti.tel.Domain;
using gr.cti.tel.copeit;
using gr.cti.tel.Common;

namespace gr.cti.tel.copeitWebService
{

    public class AuthHeader : SoapHeader
    {
        public string Username;
        public string password;
    }


    public class User
    {
        public int user_id;
        public string name;
        public string password;
        public string FName;
        public string LName;
        public string email;
        public byte[] photo;
        public string address;
    }
    public class UserList : List<User> { }

    public class Cop
    {
```

```
                    public int cop_id;
    public string title;
    public string description;
    public byte[] logo;

    public Cop() { }
}
public class CopList : List<Cop> { }

public class WorkSpace
{
    public int id;
    public int creator_id;
    public DateTime creationDate;
    public DateTime modificationDate;
    public string title;
    public int prive;

    public WorkSpace() { }

    public WorkSpace(TSysObject so)
    {
        TCanvas c = new TCanvas();
        c.SetValues(so);
        id = c.id_sysobject;
        creator_id = c.creator_id;
        creationDate = c.creation_date;
        modificationDate = c.modification_date;
        title = c.title;
        prive = c.prive;
    }
}
public class WorkSpaceList : List<WorkSpace> { }

public class Document
{
    public int id;
    public int creator_id;
    public DateTime creationDate;
    public DateTime modificationDate;
    public string title;
    public string mimetype;
    public string style;
    public int x;
    public int y;
    public int height;
    public int width;
    public byte[] cnt;

    public Document() { }

    public Document(TSysObject so)
    {
        TDocument doc = new TDocument();
        doc.SetValues(so);
        id = doc.id_sysobject;
        creator_id = doc.creator_id;
```

```
            creationDate = doc.creation_date;
            modificationDate = doc.modification_date;
            title = doc.title;
            mimetype = doc.MIMEType;
            style = doc.style;
            x = doc.positionX;
            y = doc.positionY;
            height = doc.height;
            width = doc.width;
        }
    }
    public class DocumentList : List<Document> { }

    public class TAttributeValue
    {
        public string name;
        public string value;
    }
    public class AVList : List<TAttributeValue> { }

    public class Options : AVList { }

    public class Resource
    {
        public byte[] cnt;
        public AVList metadata;
    }

}
```

2. Mobile Application

2.1. Ο κώδικας υλοποίησης της εφαρμογής στο κινητό σε C#

(.cs αρχεία) μέσω Visual Studio με αποτέλεσμα τη δημιουργία του αρχείου εγκατάστασης copeitmobile.CAB

accountStatus.cs

```
using System;
using System.Collections.Generic;
using System.Text;
using System.IO;

namespace copeitmob
{
    public class accountStatus
    {
        private static accountStatus instance;
```

```
        public string username;
        public string password;
        public int currentCopID;
        public string currentCopTitle;
        public int currentWorkspaceID;
        public string currentWorkspaceTitle;
        public string filePath;

        protected accountStatus()
        {
        }

        public static accountStatus Instance()
        {
           if (instance == null)
           {
              instance = new accountStatus();
              instance.init();
           }

           return instance;
        }

         public void init()
        {
           username = "";
           password = "";
           currentCopID = 0;
           currentCopTitle = "";
           currentWorkspaceID = 0;
           currentWorkspaceTitle = "";
        }

        public void saveStatus()
        {
           TextWriter tw = new StreamWriter(filePath);
           tw.WriteLine(username + "*##*" + password + "*##*" + currentCopID.ToString() + "*##*" +
currentCopTitle + "*##*" + currentWorkspaceID.ToString() + "*##*" + currentWorkspaceTitle);
           tw.Close();
        }

        public void loadStatus()
        {
           StreamReader streamReader = new StreamReader(filePath);
           string text = streamReader.ReadToEnd();
           streamReader.Close();
           string[] data = SplitByString(text, "*##*");
           username = data[0];
           password = data[1];
           currentCopID = Int32.Parse(data[2]);
           currentCopTitle = data[3];
           currentWorkspaceID = Int32.Parse(data[4]);
           currentWorkspaceTitle = data[5];
        }


        public static string[] SplitByString(string testString, string split)
```

```
        {
            int offset = 0;
            int index = 0;
            int[] offsets = new int[testString.Length + 1];
             while (index < testString.Length)
            {
                int indexOf = testString.IndexOf(split, index);
                if (indexOf != -1)
                {
                    offsets[offset++] = indexOf;
                    index = (indexOf + split.Length);
                }
                else
                {
                    index = testString.Length;
                }
            }

            string[] final = new string[offset + 1];
            if (offset == 0)
            {
                final[0] = testString;
            }
            else
            {
                offset--;
                final[0] = testString.Substring(0, offsets[0]);
                for (int i = 0; i < offset; i++)
                {
                    final[i + 1] = testString.Substring(offsets[i] + split.Length, offsets[i + 1] - offsets[i] - split.Length);
                }
                final[offset + 1] = testString.Substring(offsets[offset] + split.Length);
            }
            return final;
        }


          }

    }
```

Frm_Account.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Web.Services;
using copeitmob.gr.cti.copeit.ws;
```

```
namespace copeitmob
{
    public partial class Frm_Account : Form
    {
        public Frm_Account()
        {
            Cursor.Current = Cursors.Default;
            Application.DoEvents();
            InitializeComponent();
            tb_username.Text = accountStatus.Instance().username;
            tb_password.Text = accountStatus.Instance().password;
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Cursor.Current = Cursors.WaitCursor;

            lb_test.ForeColor = System.Drawing.Color.Gray;
            lb_test.Text = "Please wait...";
            Application.DoEvents();

            copeitmob.gr.cti.copeit.ws.mobile mobService = new copeitmob.gr.cti.copeit.ws.mobile();

            AuthHeader Credentials = new AuthHeader();
            Credentials.Username = tb_username.Text;
            Credentials.password = tb_password.Text;
            mobService.AuthHeaderValue = Credentials;
            try
            {
                int login = mobService.Login();
                lb_test.ForeColor = System.Drawing.Color.Green;
                lb_test.Text = "Login Succeded";
            }
            catch
            {
                lb_test.ForeColor = System.Drawing.Color.Red;
                lb_test.Text = "Login Failed";
            }
            Cursor.Current = Cursors.Default;
            Application.DoEvents();

        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void pictureBox2_Click_1(object sender, EventArgs e)
        {
            Cursor.Current = Cursors.WaitCursor;
            Application.DoEvents();

            lb_test.ForeColor = System.Drawing.Color.Gray;
            lb_test.Text = "Trying to save. Please wait...";
            Application.DoEvents();
```

```
            copeitmob.gr.cti.copeit.ws.mobile mobService = new copeitmob.gr.cti.copeit.ws.mobile();

            AuthHeader Credentials = new AuthHeader();
            Credentials.Username = tb_username.Text;
            Credentials.password = tb_password.Text;
            mobService.AuthHeaderValue = Credentials;
            try
            {
               int login = mobService.Login();
               lb_test.ForeColor = System.Drawing.Color.Green;
               lb_test.Text = "Saved!";
               accountStatus.Instance().init();
               accountStatus.Instance().username = tb_username.Text;
               accountStatus.Instance().password = tb_password.Text;
               accountStatus.Instance().saveStatus();
               Cursor.Current = Cursors.Default;
               Application.DoEvents();

                this.Close();
            }
            catch
            {
               lb_test.ForeColor = System.Drawing.Color.Red;
               lb_test.Text = "Cannot Save: Login Failed";
               Cursor.Current = Cursors.Default;
               Application.DoEvents();
            }
         }
      }
   }
```

Frm_Copeit.cs

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

namespace copeitmob
{
      public partial class frm_copeit : Form
      {
         public frm_copeit()
         {
            InitializeComponent();

            update_controls();
```

```
        }

        private void update_controls()
        {
            if (accountStatus.Instance().username == "")
            {
                button3.Enabled = false;
                button4.Enabled = false;
                llb_open_space.Text = "Please Load a Workspace";
                llb_open_space.Enabled = false;
            }
            else
            {
                button3.Enabled = true;
                if (accountStatus.Instance().currentCopID <= 0)
                {
                    button4.Enabled = false;
                    llb_open_space.Text = "Please Load a Workspace";
                    llb_open_space.Enabled = false;
                }
                else
                {
                    button4.Enabled = true;
                    if (accountStatus.Instance().currentWorkspaceID <= 0)
                    {
                        llb_open_space.Text = "Please Load a Workspace";
                        llb_open_space.Enabled = false;
                    }
                    else
                    {
                        llb_open_space.Enabled = true;
                        llb_open_space.Text = "Click to open Workspace: " +
accountStatus.Instance().currentWorkspaceTitle;
                    }

                }
            }
        }


        private void button4_Click(object sender, EventArgs e)
        {
            Cursor.Current = Cursors.WaitCursor;
            Application.DoEvents();
            Frm_Select_Workspace frm_select_workspace = new Frm_Select_Workspace();
            frm_select_workspace.ShowDialog();

            update_controls();
        }

        private void llb_open_space_Click(object sender, EventArgs e)
        {
            llb_open_space.Text = "Openning Workspace. Please Wait...";
            Cursor.Current = Cursors.WaitCursor;
            Application.DoEvents();
            Frm_Open_Workspace frm_open_workspace = new Frm_Open_Workspace();
            frm_open_workspace.ShowDialog();
```

```
            update_controls();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Cursor.Current = Cursors.WaitCursor;
            Application.DoEvents();
            Frm_Account frm_account = new Frm_Account();
            frm_account.ShowDialog();
            update_controls();
        }

        private void button3_Click(object sender, EventArgs e)
        {
            Cursor.Current = Cursors.WaitCursor;
            Application.DoEvents();
            Frm_Select_Community frm_community = new Frm_Select_Community();
            frm_community.ShowDialog();
            update_controls();
        }
    }
}
```

Frm_Open_Document.cs (άνοιγμα αντικειμένου)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using copeitmob.gr.cti.copeit.ws;

namespace copeitmob
{
    public partial class Frm_Open_Document : Form
    {
        public Frm_Open_Document()
        {
            InitializeComponent();
        }

        public void OpenDoc(ListViewItem.ListViewSubItemCollection s)
        {
            webBrowser1.Navigate(new Uri("http://copeit.cti.gr/get_resource.aspx?u=" +
accountStatus.Instance().username + "&p=" + accountStatus.Instance().password + "&item="+s[1].Text.ToString()
));
            lb_date.Text = "Date: "+s[2].Text;
            lb_creator.Text ="Created by: "+s[3].Text;
            pictureBox1.Image = imageList1.Images[Int32.Parse(s[4].Text)];
            lb_title.Text = "Title: "+s[5].Text;
```

```
            Cursor.Current = Cursors.Default;
            Application.DoEvents();
        }

        private void lb_date_ParentChanged(object sender, EventArgs e)
        {

        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {

        }

        private void pictureBox3_Click_1(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

Frm_Open_Workspace.cs (άνοιγμα workspace της επιλογής μας)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Web.Services;
using copeitmob.gr.cti.copeit.ws;

namespace copeitmob
{
    public partial class Frm_Open_Workspace : Form
    {

        private int doc_image_index(Document doc)
        {
            string ext = "png";
            string docExt = doc.mimetype;
            string im = "";
            if (doc.style == "idea")
                return 49;
            else if (doc.style == "node")
                return 50;
            else if (doc.style == "comment")
                return 47;
            else if ((docExt == "doc") || (docExt == "dot") || (docExt == "rtf"))
                return 7;
            else if (docExt == "pdf")
```

```
    return 21;
else if (docExt == "pub")
    return 25;
else if (docExt == "xls")
    return 45;
else if (docExt == "ppt")
    return 23;
else if (docExt == "txt")
    return 37;
else if (docExt == "mdb")
    return 14;
else if (docExt == "ps")
    return 24;
else if (docExt == "htm")
    return 11;
else if (docExt == "html")
    return 11;
else if (docExt == "mov")
    return 16;
else if (docExt == "avi")
    return 2;
else if (docExt == "exe")
    return 8;
else if (docExt == "gif")
    return 10;
else if (docExt == "png")
    return 22;
else if (docExt == "rar")
    return 27;
else if (docExt == "swf")
    return 32;
else if (docExt == "wav")
    return 42;
else if (docExt == "wmv")
    return 44;
else if (docExt == "zip")
    return 46;
else if (docExt == "sxw")
    return 35;
else if (docExt == "stw")
    return 31;
else if (docExt == "sxc")
    return 33;
else if (docExt == "stc")
    return 29;
else if (docExt == "sxi")
    return 34;
else if (docExt == "sti")
    return 30;
else if (docExt == "jpg")
    return 13;
else if (docExt == "mp3")
    return 17;
else if (docExt == "mp4")
    return 18;
else if (docExt == "url")
    return 11;
```

```
   else
      return 51;
}
private string doc_image_name(Document doc)
{
      string ext="png";
      string docExt = doc.mimetype;
      string im = "";
      if (doc.style == "idea")
         return "idea_thumb.png";
      else if (doc.style == "node")
         return "note_thumb.png";
      else if (doc.style == "comment")
         return "comment_thumb.png";
      else if ((docExt == "doc") || (docExt == "dot") || (docExt == "rtf"))
         return "doc." + ext;
      else if (docExt == "pdf")
         return "pdf." + ext;
      else if (docExt == "pub")
         return "pub." + ext;
      else if (docExt == "xls")
         return "xls." + ext;
      else if (docExt == "ppt")
         return "ppt." + ext;
      else if (docExt == "txt")
         return "txt." + ext;
      else if (docExt == "mdb")
         return "mdb." + ext;
      else if (docExt == "ps")
         return "ps." + ext;
      else if (docExt == "htm")
         return "html." + ext;
      else if (docExt == "html")
         return "html." + ext;
      else if (docExt == "mov")
         return "mov." + ext;
      else if (docExt == "avi")
         return "avi." + ext;
      else if (docExt == "exe")
         return "exe." + ext;
      else if (docExt == "gif")
         return "gif." + ext;
      else if (docExt == "png")
         return "png." + ext;
      else if (docExt == "rar")
         return "rar." + ext;
      else if (docExt == "swf")
         return "swf." + ext;
      else if (docExt == "wav")
         return "wav." + ext;
      else if (docExt == "wmv")
         return "wmv." + ext;
      else if (docExt == "zip")
         return "zip." + ext;
      else if (docExt == "sxw")
         return "sxw." + ext;
      else if (docExt == "stw")
```

```
            return "stw." + ext;
        else if (docExt == "sxc")
            return "sxc." + ext;
        else if (docExt == "stc")
            return "stc." + ext;
        else if (docExt == "sxi")
            return "sxi." + ext;
        else if (docExt == "sti")
            return "sti." + ext;
        else if (docExt == "jpg")
            return "jpg." + ext;
        else if (docExt == "mp3")
            return "mp3." + ext;
        else if (docExt == "url")
            return "html." + ext;
        else
            return "unknown." + ext;
}

public Frm_Open_Workspace()
{
    InitializeComponent();

    init();

}

public void init()
{
    this.Text = "copeit: "+accountStatus.Instance().currentWorkspaceTitle;

    copeitmob.gr.cti.copeit.ws.mobile mobService = new copeitmob.gr.cti.copeit.ws.mobile();

    AuthHeader Credentials = new AuthHeader();
    Credentials.Username = accountStatus.Instance().username;
    Credentials.password = accountStatus.Instance().password;
    mobService.AuthHeaderValue = Credentials;


    copeitmob.gr.cti.copeit.ws.Document[] docs;
    docs = mobService.LoadDocuments(accountStatus.Instance().currentWorkspaceID);

    listView1.Items.Clear();


    // Display items in the ListView control

    for (int i = 0; i < docs.Length; i++)
    {
        // Define the list items
        ListViewItem lvi = new ListViewItem(docs[i].title);
        lvi.SubItems.Add(docs[i].id.ToString());
        lvi.SubItems.Add(docs[i].creationDate.ToString());
        lvi.SubItems.Add(docs[i].creator_id.ToString());
        lvi.ImageIndex = doc_image_index(docs[i]);
        lvi.SubItems.Add(lvi.ImageIndex.ToString());
        lvi.SubItems.Add(docs[i].title.ToString());
```

```
            // Add the list items to the ListView
            listView1.Items.Add(lvi);
        }

        Cursor.Current = Cursors.Default;
        Application.DoEvents();

    }


    private void pictureBox1_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    private void pictureBox2_Click(object sender, EventArgs e)
    {
        Cursor.Current = Cursors.WaitCursor;
        Application.DoEvents();
        Frm_Post frm_post = new Frm_Post();
        frm_post.frm_parent = this;
        frm_post.ShowDialog();

    }

    private void listView1_KeyDown(object sender, KeyEventArgs e)
    {
        switch (e.KeyCode)
        {
            case Keys.Return:
                Cursor.Current = Cursors.WaitCursor;
                Application.DoEvents();
                base.OnKeyDown(e);
                Frm_Open_Document frm_document = new Frm_Open_Document();
                ListViewItem li = (ListViewItem)listView1.Items[listView1.SelectedIndices[0]];
                frm_document.OpenDoc(li.SubItems);
                frm_document.ShowDialog();
                break;
        }
        base.OnKeyDown(e);


    }
  }
}
```

<u>Frm_Post.cs</u> (δημιουργία και αποστολή post)

```
using System;
using System.IO;
using System.Collections.Generic;
using System.ComponentModel;
```

```
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using copeitmob.gr.cti.copeit.ws;

namespace copeitmob
{
   public partial class Frm_Post : Form
   {
      Document postdoc;
      string fname;
      public Frm_Open_Workspace frm_parent;

      public Frm_Post()
      {
         InitializeComponent();
         postdoc = new Document();
         fname = "";
         Cursor.Current = Cursors.Default;
         Application.DoEvents();

      }

      private void pictureBox2_Click_1(object sender, EventArgs e)
      {

      }

      private void pictureBox3_Click(object sender, EventArgs e)
      {

      }

      private void pictureBox3_Click_1(object sender, EventArgs e)
      {
         this.Close();
      }

      private void pictureBox2_Click(object sender, EventArgs e)
      {
         Cursor.Current = Cursors.WaitCursor;
         Application.DoEvents();
         //lb_test.ForeColor = System.Drawing.Color.Gray;
         //lb_test.Text = "Please wait...";
         //Application.DoEvents();

         if (rb_idea.Checked)
            postdoc.style = "idea";
         else if (rb_note.Checked)
            postdoc.style = "node";
         else if (rb_comment.Checked)
            postdoc.style = "comment";
         else if (rb_other.Checked)
            postdoc.style = "unknown";

         System.Text.UTF8Encoding encoding = new System.Text.UTF8Encoding();
```

```
    if (rb_text.Checked || rb_url.Checked)
    {
        postdoc.mimetype = "html";
        postdoc.cnt = encoding.GetBytes(tb_content.Text.ToString());
    }
    else if (rb_file.Checked)
    {
        string rs = "unknown";
        int k = fname.LastIndexOf('.');
        if ((k >= 0) && (k < fname.Length + 1))
            rs = fname.Substring(k + 1);
        postdoc.mimetype = rs;
    }

    postdoc.title = tb_title.Text;
    postdoc.x = 10;
    postdoc.y = 50;
    //prosoxh pernao to canvas id mesa apo to width
    postdoc.width = accountStatus.Instance().currentWorkspaceID;

    mobile mobService = new mobile();

    AuthHeader Credentials = new AuthHeader();
    Credentials.Username = accountStatus.Instance().username;
    Credentials.password = accountStatus.Instance().password;
    mobService.AuthHeaderValue = Credentials;

    int t = mobService.createDocument(postdoc);

    frm_parent.init();
    this.Close();
    Cursor.Current = Cursors.Default;
    Application.DoEvents();

}

private void rb_file_CheckedChanged(object sender, EventArgs e)
{
    if (rb_file.Checked)
    {
        tb_content.Visible = false;
        bt_openfile.Visible = true;
    }
    else
    {
        tb_content.Visible = true;
        bt_openfile.Visible = false;
    }
}

private void rb_url_CheckedChanged(object sender, EventArgs e)
{
    if (rb_file.Checked)
    {
        tb_content.Visible = false;
        bt_openfile.Visible = true;
    }
```

```
            else
            {
                tb_content.Visible = true;
                bt_openfile.Visible = false;
            }
        }

        private void rb_text_CheckedChanged(object sender, EventArgs e)
        {
            if (rb_file.Checked)
            {
                tb_content.Visible = false;
                bt_openfile.Visible = true;
            }
            else
            {
                tb_content.Visible = true;
                bt_openfile.Visible = false;
            }
        }

        private void bt_openfile_Click(object sender, EventArgs e)
        {
            OpenFileDialog dialog = new OpenFileDialog();
            dialog.Filter = "All files (*.*)|*.*";
            dialog.InitialDirectory = "";
            if (dialog.ShowDialog() == DialogResult.OK)
            {
                fname=dialog.FileName;
                FileStream fs = new FileStream(fname, FileMode.Open, System.IO.FileAccess.Read);
                System.IO.BinaryReader br = new System.IO.BinaryReader(fs);
                postdoc.cnt = br.ReadBytes((Int32)fs.Length);
                br.Close();
                fs.Close();
            }
        }
    }
}
```

Frm_Select_Community.cs (επιλογή community)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using copeitmob.gr.cti.copeit.ws;

namespace copeitmob
{
```

```
public partial class Frm_Select_Community : Form
{
   public Frm_Select_Community()
   {
      InitializeComponent();

      lb_curren_community.Text = (accountStatus.Instance().currentCopTitle == "" ? "You haven't selected a
community yet" : accountStatus.Instance().currentCopTitle);

      copeitmob.gr.cti.copeit.ws.mobile mobService = new copeitmob.gr.cti.copeit.ws.mobile();

      AuthHeader Credentials = new AuthHeader();
      Credentials.Username = accountStatus.Instance().username;
      Credentials.password = accountStatus.Instance().password;
      mobService.AuthHeaderValue = Credentials;

      Cop[] cops;
      cops = mobService.LoadCops();
      for (int i = 0; i < cops.Length; i++)
      {
         copLItem l = new copLItem(cops[i]);
         listBox1.Items.Add(l);
         if (l.title == accountStatus.Instance().currentCopTitle)
            listBox1.SelectedIndex = i;
      }
      Cursor.Current = Cursors.Default;
      Application.DoEvents();


   }

   private void pictureBox3_Click_1(object sender, EventArgs e)
   {
      this.Close();

   }

   private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
   {
      copLItem l = (copLItem)listBox1.SelectedItem;
      Cop currentcop = (Cop)l.o;
      lb_curren_community.Text = currentcop.title;
   }


   private void pictureBox2_Click(object sender, EventArgs e)
   {
      copLItem l = (copLItem)listBox1.SelectedItem;
      Cop currentcop = (Cop)l.o;
      accountStatus.Instance().currentCopID = currentcop.cop_id;
      accountStatus.Instance().currentCopTitle = currentcop.title;
      lb_curren_community.Text = accountStatus.Instance().currentCopTitle;
      accountStatus.Instance().saveStatus();
      this.Close();

   }
```

```
        private void listBox1_KeyDown(object sender, KeyEventArgs e)
        {
            switch (e.KeyCode)
            {
                case Keys.Return:
                    copLItem l = (copLItem)listBox1.SelectedItem;
                    Cop currentcop = (Cop)l.o;
                    accountStatus.Instance().currentCopID = currentcop.cop_id;
                    accountStatus.Instance().currentCopTitle = currentcop.title;
                    lb_curren_community.Text = accountStatus.Instance().currentCopTitle;
                    accountStatus.Instance().saveStatus();
                    this.Close();
                    break;
            }

            base.OnKeyDown(e);

        }
    }

    public class copLItem
    {
        public object o;
        public string title;

        public copLItem(object x)
        {
            o = x;
            title = ((Cop)(x)).title;
        }

        public override string ToString()
        {
            return title;
        }
    }

}
```

Frm_Select_Workspace.cs (επιλογή workspace)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using copeitmob.gr.cti.copeit.ws;

namespace copeitmob
{
```

```
public partial class Frm_Select_Workspace : Form
{
    public Frm_Select_Workspace()
    {
        InitializeComponent();

        lb_current_workspace.Text = (accountStatus.Instance().currentWorkspaceTitle == "" ? "You haven't
opened a workspace yet" : accountStatus.Instance().currentWorkspaceTitle);

        copeitmob.gr.cti.copeit.ws.mobile mobService = new copeitmob.gr.cti.copeit.ws.mobile();

        AuthHeader Credentials = new AuthHeader();
        Credentials.Username = accountStatus.Instance().username;
        Credentials.password = accountStatus.Instance().password;
        mobService.AuthHeaderValue = Credentials;

        WorkSpace[] wss;
        wss = mobService.LoadWorkSpaces(accountStatus.Instance().currentCopID);
        for (int i = 0; i < wss.Length; i++)
        {
            wsLItem l = new wsLItem(wss[i]);
            listBox1.Items.Add(l);
        }
        Cursor.Current = Cursors.Default;
        Application.DoEvents();
    }

    private void pictureBox3_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    private void pictureBox2_Click(object sender, EventArgs e)
    {
        wsLItem l = (wsLItem)listBox1.SelectedItem;
        WorkSpace currentws = (WorkSpace)l.o;
        accountStatus.Instance().currentWorkspaceID = currentws.id;
        accountStatus.Instance().currentWorkspaceTitle = currentws.title;
        lb_current_workspace.Text = accountStatus.Instance().currentWorkspaceTitle;
        accountStatus.Instance().saveStatus();
        this.Close();
    }

    private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
    {
        wsLItem l = (wsLItem)listBox1.SelectedItem;
        WorkSpace currentws = (WorkSpace)l.o;
        lb_current_workspace.Text = currentws.title;
    }

    private void listBox1_KeyDown(object sender, KeyEventArgs e)
    {
        switch (e.KeyCode)
        {
            case Keys.Return:
                wsLItem l = (wsLItem)listBox1.SelectedItem;
                WorkSpace currentws = (WorkSpace)l.o;
```

```
                accountStatus.Instance().currentWorkspaceID = currentws.id;
                accountStatus.Instance().currentWorkspaceTitle = currentws.title;
                lb_current_workspace.Text = accountStatus.Instance().currentWorkspaceTitle;
                accountStatus.Instance().saveStatus();
                this.Close();
                break;
            }

            base.OnKeyDown(e);

        }

    }

    public class wsLItem
    {
        public object o;
        public string title;

        public wsLItem(object x)
        {
            o = x;
            title = ((WorkSpace)(x)).title;
        }

        public override string ToString()
        {
            return title;
        }
    }
}
```

Program.cs (αποθηκεύεται στο account.txt το username και το Password του χρήστη και συγκρατεί το Community και το Workspace, που επισκέφτηκε τελευταία φορά ώστε να μην χρειάζεται να ξανακάνει Login και τον πηγαίνει με ένα κλικ κατευθείαν στο τελευταίο workspace)

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace copeitmob
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [MTAThread]
        static void Main()
        {
```

```
            accountStatus.Instance().filePath = "\\Program Files\\copeitmob\\account.txt";
            accountStatus.Instance().loadStatus();

            Application.Run(new frm_copeit());
        }
    }
}
```