



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων

Πτυχιακή Εργασία

Ανάπτυξη Συστήματος για on line συμπλήρωση ερωτηματολογίων



Φοιτητής: Γρεβενίτης Κωνσταντίνος AM 1123

Επιβλέπων Καθηγητής: Κωνσταντίνος Βασιλάκης

Ηράκλειο, Μάιος 2009

Εισαγωγή

Αντικείμενο της παρούσας πτυχιακής είναι η ανάπτυξη μίας on line εφαρμογής η οποία θα επιτρέπει τη δημιουργία ηλεκτρονικών ερωτηματολογίων και την ανάρτηση αυτών, για συμπλήρωση.

Η λογική της εφαρμογής τούτης, δεν επαφίεται μόνο στην ανάπτυξη ερωτηματολογίων αλλά και στη διαχείριση αυτών. Κάθε ερωτηματολόγιο θα αντιστοιχεί σε ένα μάθημα, που ανήκει σε κάποιο τμήμα και αυτό το μάθημα έχει και κάποιο διδάσκοντα. Οπότε αυτόματα γεννάται η ανάγκη ευέλικτης διαχείρισης, ταξινόμησης και ανάκτησης των ερωτηματολογίων ανά πάσα στιγμή για να συμπληρωθούν ακολούθως.

Επιπλέον από τη στιγμή που η εφαρμογή θα «βρίσκεται» στο διαδίκτυο θα πρέπει να υπάρχει ένας διαχωρισμός σε front end και σε back end. Στο back end θα έχει πρόσβαση ο διαχειριστής του συστήματος, ενώ το front end θα χωρίζεται σε 2 μέρη, στο ένα θα είναι ταξινομημένα τα ερωτηματολόγια ανά σχολή τμήμα και εξάμηνο προς ανάκτηση και συμπλήρωση, ενώ στο δεύτερο θα υπάρχει μία φόρμα υποβολής του ερωτηματολογίου από τον εκάστοτε καθηγητή του μαθήματος που έχει φτιάξει το ερωτηματολόγιο.

Η ορθή διαχείριση περιεχομένου είναι μια διαδικασία που περιλαμβάνει επιμέρους εργασίες όπως η συλλογή, η εισαγωγή, η οργάνωση, η κατηγοριοποίηση, η δόμηση, η επεξεργασία και η δημοσίευση του περιεχομένου. Κάποιοι από αυτούς τους όρους αναφέρθηκαν παραπάνω. Άρα η εφαρμογή διαχείρισης των ερωτηματολογίων θα είναι ουσιαστικά ένα σύστημα διαχείρισης περιεχομένου (content management system, CMS).

Το CMS, γενικότερα είναι ένα σύστημα που αναλαμβάνει να διεκπεραιώσει τις εργασίες αυτές. Σε λογικό επίπεδο, κάθε σύστημα διαχείρισης περιεχομένου διαθέτει μια οντολογία και ένα σύνολο από ταξινομίες (taxonomies) για την κατηγοριοποίηση του περιεχομένου του σε κλάσεις αντικειμένων. Η αποθήκευση του περιεχομένου σε φυσικό επίπεδο γίνεται σε συστατικά στοιχεία που επιλέγονται από τα εκάστοτε CMS, όπως είναι συνήθως πίνακες σε μια σχεσιακή βάση δεδομένων.

Μεθοδολογία

Για την υλοποίηση της πτυχιακής εργασίας και την επίτευξη των στόχων της, ακολουθήθηκαν επιμέρους βήματα τα οποία συνοψίζονται παρακάτω και τα οποία συνιστούν και την μεθοδολογία που χρησιμοποιήθηκε στην παρούσα εργασία.

Πρώτα από όλα, στο μεθοδολογικό σκέλος της πτυχιακής μελετήθηκαν θέματα που αφορούν τα χαρακτηριστικά τα οποία θα πρέπει να έχει μία on line εφαρμογή διαχείρισης ερωτηματολογίων. Τέτοια θέματα αφορούν μοντέλα ανάπτυξης μιας τέτοιας εφαρμογής, απαιτήσεις ευχρηστίας και συμμετοχής των εκάστοτε χρηστών (administrator, φοιτητές, καθηγητές).

Ακολούθως, εξειδικεύτηκε ο επιμέρους τεχνικός στόχος της εργασίας και διερευνήθηκαν ζητήματα που αφορούν την διαχείριση (συλλογή και δημοσίευση) περιεχομένου, εργαλεία και τεχνολογίες που μπορούν να υποστηρίξουν την διαχείριση περιεχομένου, διαθεσιμότητα εργαλείων (υπό μορφή ανοικτού κώδικα). Αποτέλεσμα της διερεύνησης αυτής ήταν η επιλογή των εργαλείων Joomla (CMS) και Lime Survey (Survey Tool), τα οποία αποτέλεσαν την βάση ανάπτυξης της εφαρμογής.

Στο τελευταίο στάδιο η εφαρμογή αναπτύχθηκε σε πλήρη κλίμακα, εγκαταστάθηκε σε apache server και ελέγχτηκε υπό πραγματικές συνθήκες.

Αποτελέσματα

Το αποτέλεσμα της πτυχιακής ταύτης ήταν η δημιουργία της εφαρμογής διαχείρισης ερωτηματολογίων και συμπλήρωσης αυτών. Στην εφαρμογή αυτή θα μπορούν να έχουν πρόσβαση οι εμπλεκόμενοι από οποιαδήποτε τοποθεσία, ιδιαίτερα εξυπηρετικό για τους καθηγητές και τους διαχειριστές τους συστήματος.

Δομή και Οργάνωση της αναφοράς

Η παρούσα αναφορά αποτελείται από την 2 μέρη:

1. Η πρώτη ενότητα αφορά τεχνολογίες και εργαλεία δημιουργίας διαδικτυακών εφαρμογών, εστιάζοντας στο διαδίκτυο, το λογισμικό ανοικτού κώδικα (Open source) και τα σύγχρονα εργαλεία διαχείρισης περιεχομένου (CMS).
2. Στην δεύτερη ενότητα γίνεται παρουσίαση της εφαρμογής καθώς και τα εργαλεία που χρησιμοποιήθηκαν για την ολοκλήρωσή του.

Περιεχόμενα

Μέρος 1^ο - Γενικές Πληροφορίες

Κεφ. 1ο - Internet και Παγκόσμιος Ιστός

- Εισαγωγή.....7
- Η ιστορία του παγκόσμιου ιστού.....8
 - 1960 – 1969.....8
 - 1970 - 1979.....9
 - 1980 – 1989.....14
 - 1990 - σήμερα.....17

Κεφ. 2ο - Λογισμικό Ανοιχτού Κώδικα

- Εισαγωγή.....19
- Πλεονεκτήματα λογισμικού ανοιχτού κώδικα.....19
- Μειονεκτήματα λογισμικού ανοιχτού κώδικα.....21

Κεφ. 3ο - Εφαρμογές ΛΑΚ

- Εισαγωγή.....22
- Τι είναι το Σύστημα Διαχείρισης Περιεχομένου (CMS)22
 - Τι σημαίνει περιεχόμενο (content)23
 - Το CMS σε βάθος.....23
 - Content Domain.....23
 - Μορφή (format) παρουσίασης.....23
 - Λειτουργίες ενός CMS.....24
 - Σύστημα σύνταξης (authoring).....24
 - Σύστημα διαχείρισης (management).....24
 - Σύστημα αυτοματοποίησης κύκλου εργασιών (workflow automation).....25
 - Σύστημα έκδοσης (publication system)25
 - Άλλες λειτουργίες και δυνατότητες.....26
 - i. Personalization26
 - ii. Διαχείριση Metadata27
 - iii. Επεκτασιμότητα (Scalability)27
 - Λοιπές Τεχνολογίες.....27
 - PHP)27
 - MySQL.....28
 - Apache Server.....29
 - Aptana IDE.....29

Μέρος 2ο – Σχεδίαση και ανάπτυξη

Κεφ. 4ο – Ανάλυση του component

- Εισαγωγή.....32
- Λίγα λόγια για το Joomla.....32
- Component διαχείρισης των ερωτηματολογίων.....33
- Διεπαφή διαχείρισης των ερωτηματολογίων.....33
- Διεπαφή των ρυθμίσεων του component.....35

- Διεπαφή βοήθειας.....35
- Front End διεπαφή.....36

Κεφ. 5ο – Υλοποίηση του component

- Εισαγωγή.....37
- Ανάπτυξη του component.....37
 - ο Δημιουργία των φακέλων στις κατάλληλες θέσεις.....37
 - ο Εισαγωγή δεδομένων στη βάση.....38
 - ο Συγγραφή κώδικα για την επικοινωνία με τη βάση.....40
 - ο Δομές φακέλων.....45
 - ο Σχεδίαση διεπαφών.....46

Κεφ. 6ο – Έλεγχος – Εικονικό Σενάριο

- Εισαγωγή.....115
- Lime Survey.....115
- Joomla.....117

Κεφ. 7ο – Packaging και Distribution

- Packaging.....121
- Distribution.....127

Κεφ. 8ο – Συμπεράσματα.....128

Βιβλιογραφία.....129

Ευρετήριο.....130

Μέρος 1^ο – Γενικές Πληροφορίες

- Κεφ. 1^ο - Internet και Παγκόσμιος Ιστός
- Κεφ. 2^ο - Λογισμικό Ανοιχτού Κώδικα
- Κεφ. 3^ο - Εφαρμογές ΛΑΚ

Κεφάλαιο 1^ο

Internet και Παγκόσμιος Ιστός

Εισαγωγή

Το Internet και ιδιαίτερα η υπηρεσία του παγκόσμιου ιστού έχει φέρει την επανάσταση στον χώρο των πληροφορικής και της επικοινωνίας όπως ξεπερνώντας κάθε προσδοκία, όπως επίσης και άλλο μέσο πχ τηλεόραση και ραδιόφωνο.

Η εφεύρεση του τηλέγραφου, του τηλεφώνου, του ραδιοφώνου, και του υπολογιστή ήταν τα πρώτα βήματα για να φτάσουμε στο σήμερα, όπου σε ελάχιστο χρόνο και με λίγα κλικ διαβάζουμε εφημερίδα, ακούμε ραδιόφωνο και πλέον βλέπουμε τηλεόραση. Τα τελευταία χρόνια η ωρίμασή του έχει επιτρέψει την χρήση του και ως εναλλακτικού μέσου σύστασης κοινοτήτων.

Το συγκριτικό πλεονέκτημα του Internet σε σχέση με τα παραδοσιακά μέσα είναι ότι έχει μια παγκόσμια κοινότητα χρηστών όπου η διάδοση της πληροφορίας πραγματοποιείται σε κλάσματα δευτερολέπτου, ανεξαρτήτως γεωγραφικής και σε μικρό χρονικό διάστημα η πληροφορία έχει μεταδοθεί παγκοσμίως.

Η ιστορία του Internet βασίζεται γύρω από τέσσερις πυλώνες. Ο πρώτος πυλώνας έχει να κάνει με την τεχνολογική εξέλιξη που άρχισε με την πρόωρη έρευνα για την εναλλακτική μετάδοση πληροφοριών και το ARPANET (και τις σχετικές με αυτό τεχνολογίες), όπου η τρέχουσα έρευνα συνεχίζει να επεκτείνει την υποδομή, να αυξήσει την απόδοση, και να φέρει τη λειτουργικότητα σε πιο υψηλό επίπεδο.

Υπάρχει ο πυλώνας διαχείρισης ολόκληρου του δικτύου, το οποίο είναι ένα τεράστιο θέμα προς συζήτηση αλλά είναι εκτός πλαισίου της πτυχιακής. Υπάρχει ο πυλώνας που έχει να κάνει με την συμμετοχή του κοινού, είτε έρχεται από τα sites κοινωνικής δικτύωσης (MySpace, face book, hi5, twitter κτλ), είτε έρχεται από τα blogs και forums.

Τέλος υπάρχει ο πυλώνας του e-commerce, όπου πληροφορίες και υλικά αγαθά πωλούνται, ανεξαρτήτως αποστάσεως, νομισματικής μονάδας, και χωρίς να χρειαστεί ο χρήστης να μετακινηθεί από τη θέση του.

Το Internet είναι σήμερα μια διαδεδομένη πηγή πληροφοριών. Η ιστορία του είναι σύνθετη και περιλαμβάνει πολλές πτυχές - τεχνολογικές, οργανωτικές, και κοινωνικές – ενώ η επιρροή του φθάνει όχι μόνο στην επικοινωνία μεταξύ υπολογιστών αλλά σε όλο το φάσμα των δραστηριοτήτων της σύγχρονης κοινωνίας (π.χ. ψυχαγωγία, εργασία, επικοινωνία, εμπόριο, πρόσβαση στην πληροφόρηση, ιατρική, αρχιτεκτονική κα).

Η ιστορία του Internet και του Παγκόσμιου Ιστού

1960 -1969

Η πρώτη καταγραμμένη περιγραφή των κοινωνικών αλληλεπιδράσεων που θα μπορούσαν να επιτραπούν μέσω της δικτύωσης ήταν μια σειρά υπομνημάτων που γράφτηκαν από τον J.C.R. Licklider του MIT τον Αύγουστο του 1962 που συζητούσαν την έννοια των «γαλαξιακών δικτύων του». Προέβλεψε ένα καθολικά συνδεδεμένο σύνολο υπολογιστών μέσω του οποίου το καθένα θα μπορούσε γρήγορα να έχει πρόσβαση σε πληροφορίες και λογισμικό από οποιαδήποτε γεωγραφική περιοχή. Ουσιαστικά, η έννοια ήταν παρόμοια με το Internet όπως είναι αυτό σήμερα.

Ο Licklider έκανε τα πρώτα βήματα της έρευνας τους σε υπολογιστές του DARPA (Defense Advanced Research Projects Agency), της αμερικάνικης υπηρεσίας υπεύθυνη για την ανάπτυξη στρατιωτικής τεχνολογίας, που άρχισε τον Οκτώβριο του 1962. Έτσι έπεισε τους διαδόχους του Ivan Sutherland, τον Bob Taylor, και τον ερευνητή του MIT Lawrence G. Roberts, για τη σπουδαιότητα των σκέψεων του.

Ο Leonard Kleinrock, εργαζόμενο στο MIT, δημοσίευσε την πρώτη μελέτη για τη θεωρία εναλλακτικής μετάδοσης πακέτου πληροφοριών τον Ιούλιο του 1961 και το πρώτο βιβλίο στο θέμα το 1964. Ο Kleinrock έπεισε τον Roberts για την θεωρητική δυνατότητα επικοινωνίας χρησιμοποιώντας τα πακέτα παρά κυκλώματα, η οποία ήταν ένα σημαντικό βήμα στην πορεία για την εξέλιξη των δικτύων υπολογιστών.

Το άλλο βασικό βήμα για την επίτευξη της δικτύωσης ήταν να μία πειραματική προσπάθεια. Το 1965 συνεργαζόμενος με τον Thomas Merrill, ο Roberts συνέδεσε τον υπολογιστή TX-2 της Μασαχουσέτης με τον Q-32 στη Καλιφόρνια με μια απλή τηλεφωνική σύνδεση δημιουργώντας έτσι το πρώτο (παρότι μικρό) δίκτυο που κατασκευάστηκε ποτέ. Το αποτέλεσμα αυτού του πειράματος ήταν η διαπίστωση ότι αφενός οι υπολογιστές θα μπορούσαν να λειτουργήσουν καλά μαζί, τρέχοντας προγράμματα και ανακτώντας πληροφορίες ανάλογα με τις ανάγκες από απομακρυσμένους υπολογιστές, αφετέρου ότι το ισχύον τηλεφωνικό σύστημα ήταν συνολικά ανεπαρκές να υποστηρίξει τις πραγματικές ανάγκες εργασίας. Πάντως η πεποίθηση του Kleinrock της ανάγκης για την εναλλακτική μετάδοση πακέτου πληροφοριών επιβεβαιώθηκε.

Στα τέλη του 1966 ο Roberts πήγε στο DARPA και δούλεψε σκληρά για να βάλει γρήγορα σε εφαρμογή το σχέδιό του για το "ARPANET", δημοσιεύοντας το 1967 τις σχετικές εργασίες. Στη διάσκεψη όπου παρουσίασε την εργασία του, υπήρξε επίσης μία ακόμα εργασία για τη μεταφορά πακέτων μέσω δικτύων από την Μεγάλη Βρετανία και τους Donald Davies και Roger Scantlebury του National Physics Laboratory (NPL).

Ο Scantlebury είπε στον Roberts για τη δουλειά του στο NPL καθώς επίσης και γι' αυτήν του Paul Baran και άλλων στον οργανισμό RAND. Η ομάδα RAND είχε ήδη συντάξει το 1964 μια αναφορά για τα δίκτυα εναλλακτικής μετάδοσης πακέτων και για την ασφαλή μετάδοση φωνής ανάμεσα σε στρατιωτικούς. Αυτό που συνέβη είναι ότι τόσο η εργασία στο MIT (1961-1967), στην RAND (1962-1965), αλλά και στο NPL (1964-1967) αναπτυσσόταν παράλληλα χωρίς οι ερευνητές να γνωρίζουν για τις άλλες εργασίες. Η λέξη "πακέτο" υιοθετήθηκε από την εργασία στο NPL και η προτεινόμενη ταχύτητα γραμμών που χρησιμοποιήθηκε στο σχέδιο ARPANET αναβαθμίστηκε από 2.4 kbps σε 50 kbps.

Εν τέλει τον Αύγουστο του 1968, μετά τον Roberts και το DARPA, καθορίστηκε η γενική δομή και οι προδιαγραφές για το ARPANET. Τότε μία προκήρυξη ανάθεσης έργου εκδόθηκε από το DARPA για την ανάπτυξη ενός από τα βασικά συστατικά, τους διακόπτες πακέτων αποκαλούμενους και αλλιώς ως Interface Message Processors (IMP's). Το έργο τελικά το ανέλαβε από μια ομάδα που διευθυνόταν από τον Frank Heart, την Bolt Beranek and Newman (BBN) τον Δεκέμβριο του 1968. Δεδομένου ότι η ομάδα BBN εργάστηκε στα

IMP's με τον Bob Kahn διαδραματίζοντας σημαντικότατο ρόλο στη γενική αρχιτεκτονική σχεδίαση του ARPANET, η τοπολογία και τα οικονομικά των δικτύων σχεδιάστηκαν και βελτιστοποιήθηκαν από τον Roberts συνεργαζόμενο με το Howard Frank και την ομάδα του στη Network Analysis Corporation, ενώ το σύστημα μέτρησης δικτύων προετοιμάστηκε από την ομάδα Kleinrock στο UCLA.

Λόγω του ότι ο Kleinrock ανέπτυξε πρόωρα την θεωρία εναλλακτικής μετάδοσης πακέτων πληροφοριών το πιο ικανό δίκτυο που θα μπορούσε να φιλοξενήσει τους πρώτους πειραματισμούς ήταν, το κέντρο μέτρησης δικτύων του πανεπιστημίου της Καλιφόρνια στο Los Angeles (UCLA), και έτσι έγινε ο πρώτος κόμβος του ARPANET.

Οι πρώτοι πειραματισμοί έγιναν τον Σεπτέμβριο του 1969 όταν η ομάδα BBN εγκατέστησε το πρώτο IMP στο UCLA και ο πρώτος host υπολογιστής συνδέθηκε. Το πρόγραμμα του Doug Engelbart για "την αύξηση της ανθρώπινης νοημοσύνης" στο ερευνητικό ίδρυμα του Στάνφορντ (SRI) παρείχε έναν δεύτερο κόμβο. Έναν μήνα αργότερα, όταν συνδέθηκε το SRI με το ARPANET, το πρώτο host-to-host μήνυμα εστάλη από το εργαστήριο Kleinrock στο SRI. Δύο ακόμα κόμβοι προστέθηκαν από το πανεπιστήμιο Santa Barbara και από το πανεπιστήμιο της Utah. Αυτοί οι τελευταίοι δύο κόμβοι ενσωμάτωσαν λογισμικό γραφικής απεικόνισης, με τους Glen Culler και Burton Fried στο UCSB να ψάχνουν τρόπο απεικόνισης των μαθηματικών εξισώσεων, στην οθόνη. Κατά συνέπεια, μέχρι το τέλος του 1969, τέσσερις host υπολογιστές συνδέθηκαν με το αρχικό ARPANET, και το Internet άρχισε σιγά - σιγά να ανθίζει.

1970 -1979

Κατά τη διάρκεια των επόμενων ετών προστέθηκαν και άλλοι υπολογιστές στο ARPANET, και η εργασία προχώρησε στην ολοκλήρωση ενός λειτουργικά πλήρους πρωτοκόλλου host-to-host αλλά και άλλου λογισμικού δικτύων. Τον Δεκέμβριο του 1970 η ομάδα εργασίας δικτύων (Network Working Group - NWG) που εργαζόνταν υπό τον S. Crocker τελείωσε το αρχικό host-to-host πρωτόκολλο για το ARPANET, αποκαλούμενο και ως πρωτόκολλο ελέγχου δικτύων (Network Control Protocol - NCP). Έτσι ενώ οι κόμβοι του ARPANET ολοκλήρωναν την εισαγωγή του NCP κατά τη διάρκεια της περιόδου 1971-1972, οι χρήστες θα μπορούσαν τελικά να αρχίσουν να αναπτύσσουν εφαρμογές.

Τον Οκτώβριο του 1972 ο Kahn οργάνωσε μια μεγάλη και πολύ επιτυχημένη επίδειξη του ARPANET στη στην International Computer Communication Conference - ICCO). Αυτή ήταν και η πρώτη δημόσια παρουσίαση αυτής της νέας τεχνολογίας στο κοινό. Ήταν επίσης το 1972 όταν η εντυπωσιακή εφαρμογή του ηλεκτρονικού ταχυδρομείου, παρουσιάστηκε για πρώτη φορά σε ευρύ κοινό.

Το Μάρτιο ο Ray Tomlinson έγραψε λογισμικό για αποστολή και λήψη ηλεκτρονικού ταχυδρομείου, παρακινούμενος από την ανάγκη των υπεύθυνων του ARPANET για την ανάπτυξη ενός εύκολου μηχανισμού για να μπορούν επικοινωνούν και να συντονίζονται. Τον Ιούλιο, ο Roberts επέκτεινε το λογισμικό του με δυνατότητες εμφάνισης λίστας, επιλεκτικής ανάγνωσης μηνυμάτων, αρχειοθέτησης, προώθησης, και απάντησης στα μηνύματα. Από τότε το ηλεκτρονικό ταχυδρομείο έγινε «must» εφαρμογή περισσότερο από μία δεκαετία.

Το ARPANET μετεξελίχθηκε στο Internet. Το Internet βασίστηκε στην ιδέα ότι θα υπήρχαν πολλαπλάσια ανεξάρτητα δίκτυα ενός μάλλον αυθαίρετου σχεδιασμού, αρχίζοντας με το ARPANET, αλλά σύντομα θα συμπεριλαμβάνονταν τα δορυφορικά δίκτυα, τα επίγεια δίκτυα ραδιοφωνικής μετάδοσης όπως και άλλα δίκτυα. Το Internet όπως το ξέρουμε τώρα ενσωματώνει την βασική ιδέα, της αφηρημένης ανοικτής αρχιτεκτονικής δικτύωσης υπολογιστών. Υπό αυτήν την προσέγγιση, η επιλογή οποιασδήποτε μεμονωμένης τεχνολογίας δικτύων δεν υπαγορεύθηκε από μια συγκεκριμένη αρχιτεκτονική

αλλά μάλλον θα μπορούσε να επιλεγεί ελεύθερα και να επικοινωνήσει με τα άλλα δίκτυα μέσω μιας meta-level αρχιτεκτονικής.

Μέχρι τότε υπήρχε μόνο μια γενική μέθοδος για τα δίκτυα. Αυτή ήταν η παραδοσιακή μέθοδος μετατροπής κυκλωμάτων όπου τα δίκτυα διασυνδέονταν στο επίπεδο κυκλωμάτων, περνώντας τα μεμονωμένα bits σύγχρονα κατά μήκος του καλωδίου που ένωνε τις δύο τοποθεσίες. Αξίζει να αναφέρουμε ότι ο Kleinrock είχε ισχυριστεί το 1961 ότι η εναλλακτική μετάδοση πακέτων ήταν μια αποδοτικότερη μέθοδος μετατροπής. Ενώ υπήρχαν και άλλοι τρόποι για να συνδεθούν τα διαφορετικά δίκτυα, απαιτούσαν ότι το ένα θα λειτουργούσε ως υποχείριο του άλλου, παρά ομότιμα.

Σε ένα δίκτυο ανοιχτής αρχιτεκτονικής, τα μεμονωμένα δίκτυα μπορούν να σχεδιαστούν χωριστά και να αναπτυχθούν και κάθε ένα μπορεί να έχει το δικό του μοναδικό περιβάλλον εργασίας. Κάθε δίκτυο μπορεί να σχεδιαστεί σύμφωνα με τις απαιτήσεις του περιβάλλοντος και των χρηστών του δικτύου. Δεν υπάρχει θεωρητικά κανένας περιορισμός στους τύπους δικτύων που μπορούν να περιληφθούν ή στη γεωγραφική τους θέση, αλλά στη πράξη όταν γίνεται ο σχεδιασμός, ακολουθούνται όλοι εκείνοι οι κανόνες που κάνουν το δίκτυο πραγματικά λειτουργικό.

Η ιδέα της δικτύωσης ανοικτής αρχιτεκτονικής εισήχθη αρχικά από τον Kahn αμέσως μετά από την άφιξη του DARPA το 1972. Αυτή η εργασία ήταν αρχικά μέρος του project για το ραδιόφωνο, αλλά στη συνέχεια έγινε ξεχωριστό project. Τότε, το πρόγραμμα ονομαζόταν «Internetting». Το κλειδί για την ανάπτυξη ραδιοφώνων συστημάτων επικοινωνίας με πακέτα, ήταν ένα αξιόπιστο end to end πρωτόκολλο που θα μπορούσε να διατηρήσει την αποτελεσματική επικοινωνία παρά το μπλοκάρισμα που παρουσιάζονταν και τις όποιες ραδιοφωνικές παρεμποδίσεις, ή να αντισταθεί στη απώλεια σήματος όπως αυτή προκαλούνταν όταν ο δέκτης βρισκόταν μέσα σε μια σήραγγα ή υπήρχαν εμπόδια στο σήμα που έστελνε ο πομπός.

Η υπόθεση όμως ήταν ότι το ARPANET δεν ήταν μεταβλητό εκείνη τη χρονική στιγμή. Το NCP στηρίχθηκε στο ARPANET για να παρέχει αξιοπιστία end to end. Εάν οποιοδήποτε πακέτο χανόταν το πρωτόκολλο (και υποθετικά και κάθε πρόγραμμα που το υποστήριζε) θα κόλλαγε. Σε αυτό το μοντέλο το NCP δεν διέθετε κανέναν έλεγχο σφάλματος από τον host, δεδομένου ότι το ARPANET επρόκειτο να είναι το μόνο δίκτυο και θα ήταν τόσο αξιόπιστο που κανένας έλεγχος σφάλματος δεν θα απαιτούταν εκ μέρους των hosts.

Κατά συνέπεια, ο Kahn αποφάσισε να αναπτύξει μια νέα έκδοση του πρωτοκόλλου που θα μπορούσε να ικανοποιήσει τις ανάγκες ενός περιβάλλοντος δικτύων ανοικτής αρχιτεκτονικής. Αυτό το πρωτόκολλο θα καλούνταν τελικά το Transmission Control Protocol/Internet Protocol (το γνωστό μας μέχρι σήμερα TCP/IP). Έτσι ενώ το NCP έτεινε να ενεργεί ως driver, το νέο πρωτόκολλο θα ήταν περισσότερο ένα πρωτόκολλο επικοινωνίας.

Τέσσερις βασικοί κανόνες ήταν κρίσιμοι για την αρχική σκέψη του Kahn:

1. Κάθε δίκτυο θα έπρεπε να είναι ανεξάρτητο από μόνο του και καμία εσωτερική αλλαγή δεν θα απαιτούσε τη σύνδεση με το Internet.
2. Εάν ένα πακέτο δεν έφτανε στον τελικό προορισμό, θα αναμεταδιδόταν σύντομα ξανά από την πηγή.
3. Μαύρα κιβώτια θα χρησιμοποιούνταν για να συνδέσουν τα δίκτυα, αυτά θα καλούνταν αργότερα ως πύλες και δρομολογητές. Δεν θα υπήρχε καμία πληροφορία που να διατηρείται από τις πύλες για τις μεμονωμένες ροές των πακέτων που περνούν μέσω τους, με αυτόν τον τρόπο τα κρατούσαν απλά και απέφευγαν τις περίπλοκες προσαρμογές και την αποκατάσταση από τους διάφορους τρόπους αποτυχίας.
4. Δεν θα υπήρχε κανένας καθολικός έλεγχος.

Άλλα βασικά ζητήματα που έπρεπε να αντιμετωπιστούν ήταν:

1. Αλγόριθμοι για να αποτρέπουν τα χαμένα πακέτα και να δίνουν την δυνατότητα να αναμεταδοθούν επιτυχώς από την πηγή.
2. Να επιτρέψει την σύνδεση host to host έτσι ώστε τα πολλαπλάσια πακέτα να μπορούν να είναι καθοδόν από την πηγή στον προορισμό, κατά την κρίση των συμμετεχόντων hosts, εάν τα ενδιάμεσα δίκτυα το επιτρέπουν.
3. Λειτουργίες πυλών για να επιτρέπουν να διαβιβάζονται τα πακέτα κατάλληλα. Αυτό περιελάμβανε την ερμηνεία των IP επικεφαλίδων για τη δρομολόγηση, χειρισμός του περιβάλλοντος εργασίας, να υπάρχει δυνατότητα να σπάνε τα πακέτα σε μικρότερα κομμάτια εάν είναι απαραίτητο, κλπ.
4. Η ανάγκη για έλεγχο αλλά και επανασυναρμολόγηση των πακέτων από τα κομμάτια τους, ακόμα και ανίχνευση των αντιγράφων αν υπάρχουν.
5. Η ανάγκη για παγκόσμια επικοινωνία.
6. Τεχνικές για host to host έλεγχο ροής.
7. Διασύνδεση με τα διάφορα λειτουργικά συστήματα.

Υπήρξαν επίσης και άλλες ανησυχίες, όπως η αποδοτικότητα της εφαρμογής, αλλά αυτές ήταν δευτερεύουσες έννοιες για την αρχή του σχεδίου.

Ο Kahn άρχισε να δουλεύει προσανατολισμένα στην επικοινωνία και στο σύνολο των αρχών τους λειτουργικά συστήματα ενώ ήταν στο BBN και τεκμηρίωσε μάλιστα μερικές από τις πρόωρες σκέψεις του σε ένα εσωτερικό υπόμνημα του BBN που τιτλοφορήθηκε "Αρχές Επικοινωνιών για τα Λειτουργικά Συστήματα". Σε αυτό το σημείο συνειδητοποίησε ότι θα ήταν απαραίτητο να μαθευτούν οι λεπτομέρειες της εκτέλεσης κάθε λειτουργικού συστήματος για να έχει μια πιθανότητα να ενσωματώσει οποιαδήποτε νέα πρωτόκολλα με έναν αποδοτικό τρόπο.

Κατά συνέπεια, την άνοιξη του 1973, ζήτησε από τον Vint Cerf (που ήταν τότε στο Στάνφορντ) να δουλέψει μαζί του στη λεπτομερή ανάπτυξη του πρωτοκόλλου. Ο Cerf είχε ενασχοληθεί στενά με το αρχικό σχέδιο NCP και την ανάπτυξη του και είχε ήδη τη γνώση για τη διασύνδεση με τα υπάρχοντα λειτουργικά συστήματα. Έτσι οπλισμένοι με την αρχιτεκτονική προσέγγιση του Kahn στις επικοινωνίες και με την εμπειρία του Cerf στο NCP, αποφάσισαν να συνεργαστούν για να δώσουν τις λεπτομέρειες αυτό που αργότερα έγινε γνωστό ως TCP/IP.

Η ανταλλαγή πληροφοριών ανάμεσα τους ήταν ιδιαίτερα παραγωγική και η πρώτη γραπτή έκδοση της αναδυόμενης προσέγγισης διανεμήθηκε σε μια ειδική συνεδρίαση της διεθνούς ομάδας εργασίας δικτύων (International Network Working Group - INWG) που ήταν οργανωμένη στο πλαίσιο συνδιάσκεψης στο πανεπιστήμιο του Sussex τον Σεπτέμβριο του 1973. Ο Cerf ήταν καλεσμένος για να προεδρεύσει της ομάδας και χρησιμοποίησε την ευκαιρία για να πραγματοποιήσει μια συνεδρίαση των μελών INWG που αντιπροσωπεύθηκαν στη διάσκεψη του Sussex.

Μερικές βασικές προσεγγίσεις προέκυψαν από αυτήν την συνεργασία μεταξύ Kahn και Cerf:

- Η επικοινωνία μεταξύ δύο διαδικασιών θα αποτελούνταν λογικά από πολλά bytes (τα αποκαλούμενα octets). Η θέση οποιουδήποτε octet στο ρεύμα θα χρησιμοποιούνταν για να το προσδιορίσει.
- Ο έλεγχος ροής θα γινόταν χρησιμοποιώντας acknowledgments (acks). Ο προορισμός θα μπορούσε να επιλέξει πότε να αναγνωριστεί κάθε ack και θα λειτουργούσε συσσωρευτικά για όλα τα πακέτα που παραλαμβάνονταν σε εκείνο το σημείο.
- Αφέθηκε ανοικτό το πως ακριβώς ο πομπός και ο δέκτης θα συμφωνούσαν σχετικά με τις παραμέτρους των «παραθύρων» που θα χρησιμοποιούνταν.
- Παρόλο που το Ethernet ήταν υπό ανάπτυξη στο Xerox PARC εκείνη την περίοδο, ο πολλαπλασιασμός των LANs δεν προβλέφθηκε σωστά, πολύ λιγότερο δε για τα PCs και τους τερματικούς σταθμούς. Το αρχικό μοντέλο ήταν εθνικά δίκτυα διαφόρων επιπέδων όπως το ARPANET του οποίου μόνο ένας σχετικά μικρός αριθμός αναμενόταν να υπάρξει. Κατά

συνέπεια μια διεύθυνση 32 bit IP χρησιμοποιήθηκε της οποίας τα πρώτα 8 bit δήλωναν το δίκτυο και τα υπόλοιπα 24 bit υποδείκνυαν τον host του δικτύου. Αυτή η υπόθεση, ότι δηλαδή 256 δίκτυα θα ήταν ικανοποιητικά για το εγγύς μέλλον, χρειάστηκε σαφώς αναθεώρηση όταν άρχισαν τα LANs να εμφανίζονται προς το τέλος της δεκαετίας του '70.

Το αρχικό έγγραφο των Cerf και Kahn για το διαδίκτυο περιέγραφε ένα πρωτόκολλο, αποκαλούμενο TCP, το οποίο παρείχε όλη τη μεταφορά και διαβίβαση των υπηρεσιών στο Internet. Ο Kahn σκόπευε το πρωτόκολλο TCP να υποστηρίξει μια σειρά από υπηρεσίες μεταφοράς πληροφορίας, από τη συνολικά αξιόπιστα τοποθετημένη διαδοχική παράδοση των πακέτων (εικονικό μοντέλο κυκλωμάτων) σε μια υπηρεσία στην οποία η εφαρμογή, έκανε την άμεση χρήση της υπηρεσίας δικτύων, η οποία προσπαθούσε να προβλέψει τα περιστασιακά χαμένα, αλλοιωμένα ή ξανασταλμένα πακέτα.

Εντούτοις, η αρχική προσπάθεια να εφαρμοστεί το TCP οδήγησε σε μια έκδοση που λειτουργούσε μόνο στα εικονικά κυκλώματα. Αυτό το μοντέλο λειτούργησε καλά για τη μεταφορά αρχείων και τις απομακρυσμένες εφαρμογές, αλλά μερικές από τις πρώτες εργασίες για τις προηγμένες εφαρμογές δικτύων, ειδικότερα τα πακέτα φωνής στη δεκαετία του '70, κατέστησαν σαφές ότι σε μερικές περιπτώσεις οι απώλειες πακέτων δεν πρέπει να διορθωθούν από το TCP, αλλά πρέπει να αφεθούν στην εφαρμογή για να εξεταστούν.

Αυτό οδήγησε σε μια αναδιοργάνωση του αρχικού TCP σε δύο πρωτόκολλα, την απλή IP που επέτρεπε μόνο την εξέταση και τη διαβίβαση των μεμονωμένων πακέτων, και το χωριστό TCP, το οποίο ενδιαφερόταν για τα χαρακτηριστικά γνωρίσματα υπηρεσιών όπως ο έλεγχος και η αποκατάσταση ροής από τα χαμένα πακέτα. Για εκείνες τις εφαρμογές που δεν ήθελαν τις υπηρεσίες του TCP, μια εναλλακτική λύση αποκαλούμενη πρωτόκολλο διαγραμμάτων δεδομένων χρηστών (User Datagram Protocol - UDP) προστέθηκε προκειμένου να παρασχεθεί η άμεση πρόσβαση στη βασική υπηρεσία του IP.

Ένα σημαντικό κίνητρο ακόμη και για το ARPANET αλλά και για το Internet ήταν το μοίρασμα των πόρων. Εντούτοις, ενώ η μεταφορά αρχείων και το Telnet ήταν πολύ σημαντικές εφαρμογές, το ηλεκτρονικό ταχυδρομείο είχε πιθανώς τη σημαντικότερη επίδραση απ' όλες τις καινοτομίες εκείνης της εποχής.

Υπήρξαν και άλλες εφαρμογές που προτάθηκαν τις πρώτες ημέρες του Internet, συμπεριλαμβανομένης της βασισμένης σε πακέτα μετάδοσης φωνής (ο πρόδρομος του Voice Over IP), των διάφορων μοντέλων μοιράσματος των αρχείων και του δίσκου, αλλά και πρώιμα προγράμματα «σκουλήκια» που εμφάνισαν την έννοια των agents, και φυσικά των ιών. Μια βασική έννοια του Internet είναι ότι δεν σχεδιάστηκε για μόνο μια εφαρμογή, αλλά ως γενική υποδομή στην οποία οι νέες εφαρμογές θα μπορούσαν να συλληφθούν, όπως εμφανίζεται αργότερα ο Παγκόσμιος Ιστός.

Το DARPA υπέγραψε τρεις συμβάσεις με το Στάνφορντ (Cerf), το BBN (Ray Tomlinson) και το UCL (Peter Kirstein) για να εφαρμοστεί το TCP/IP (λεγόταν απλά TCP στην εργασία των Cerf/Kahn αλλά περιείχε και τα δύο συστατικά). Η ομάδα του Στάνφορντ, που οδηγήθηκε από τον Cerf, παρήγαγε τη λεπτομερή προδιαγραφή και περίπου μέσα σε ένα έτος υπήρξαν τρεις ανεξάρτητες εφαρμογές του TCP που θα μπορούσαν να επικοινωνήσουν. Αυτό ήταν η αρχή του μακροπρόθεσμου πειραματισμού και της ανάπτυξης για να εξελίξει και να ωριμάσει τις έννοιες και την τεχνολογία του Internet. Αρχίζοντας με τα πρώτα τρία δίκτυα (ARPANET, Packet Radio και Packet Satellite) και τις αρχικές ερευνητικές κοινότητές τους, το πειραματικό περιβάλλον έτεινε να ενσωματώσει ουσιαστικά κάθε μορφή του δικτύου και μιας ευρείας βάσης ερευνητικής και αναπτυξιακής κοινότητας.

Οι πρώτες εφαρμογές του TCP έγιναν για τα μεγάλα συστήματα όπως το Tenex και το TOPS 20. Όταν οι υπολογιστές γραφείου εμφανίστηκαν αρχικά, θεωρήθηκε από μερικούς ότι το TCP ήταν πάρα πολύ μεγάλο και σύνθετο για να τρέξει σε έναν προσωπικό υπολογιστή. Ο David Clark και η ερευνητική ομάδα του στο MIT κατάφεραν να δείξουν ότι

μια συμπαγής και απλή εφαρμογή του TCP ήταν δυνατή. Παρήγαγαν μια εφαρμογή, πρώτα για το Xerox Alto (ο πρόωρος προσωπικός υπολογιστής που αναπτύσσεται σε Xerox PARC) και έπειτα για το PC της IBM. Εκείνη η εφαρμογή ήταν πλήρως συνεργάσιμη με τα άλλα TCPs, αλλά προσαρμόστηκε στους στόχους ακολουθίας και απόδοσης εφαρμογής του προσωπικού υπολογιστή, και έδειξε ότι οι τερματικοί σταθμοί, καθώς επίσης και τα μεγάλα time-sharing συστήματα, θα μπορούσαν να είναι ένα μέρος του Internet. Το 1976, ο Kleinrock δημοσίευσε το πρώτο βιβλίο για το ARPANET. Περιελάμβανε μια έμφαση στην πολυπλοκότητα των πρωτοκόλλων και των παγίδων που εισάγουν συχνά. Αυτό το βιβλίο είχε σημαντικότερη επίδραση στη διάδοση της γνώσης των δικτύων εναλλακτικής μετάδοσης πακέτου πληροφοριών σε μια πολύ ευρεία κοινότητα.

Η διαδεδομένη ανάπτυξη των lans, των υπολογιστών αλλά και των τερματικών σταθμών στη δεκαετία του '80 επέτρεψε στο Internet να ακμάσει. Η τεχνολογία Ethernet, που αναπτύχθηκε από τον Bob Metcalfe σε Xerox PARC το 1973, είναι τώρα πιθανώς η κυρίαρχη τεχνολογία δικτύωσης στο Internet και όπου αλλού μπορούμε να βρούμε δίκτυο. Αυτή η εξάπλωση από την κατοχή μερικών δικτύων με έναν μικρό αριθμό hosts (το αρχικό ARPANET μοντέλο) στην ύπαρξη πολλών δικτύων έχει οδηγήσει σε διάφορες νέες έννοιες και αλλαγές στην αρχική τεχνολογία. Κατ' αρχάς, οδήγησε στον καθορισμό τριών κλάσεων δικτύων (Α, Β και Γ) για να προσαρμόσει τη σειρά των δικτύων. Η κατηγορία Α αντιπροσώπευσε τα μεγάλα εθνικά δίκτυα κλίμακας (μικρός αριθμός δικτύων με τους μεγάλους αριθμούς hosts). Η κατηγορία Β αντιπροσώπευσε τα περιφερειακά δίκτυα κλίμακας και η κλάση Γ αντιπροσώπευσε τα δίκτυα τοπικής περιοχής (μεγάλος αριθμός δικτύων με σχετικά λίγους hosts).

Μια σημαντική μετατόπιση εμφανίστηκε ως αποτέλεσμα της αύξησης του Internet και των σχετικών διοικητικών ζητημάτων του. Για να διευκολύνουν τους χρήστες να χρησιμοποιήσουν το Internet, στους hosts ανατέθηκαν ονόματα, έτσι ώστε δεν ήταν απαραίτητο να αναφερθούν οι αριθμητικές διευθύνσεις. Αρχικά, υπήρξε ένας αρκετά περιορισμένος αριθμός hosts, έτσι ήταν εφικτό να διατηρηθεί ένας ενιαίος πίνακας όλων των hosts και των σχετικών ονομάτων και των διευθύνσεών τους. Η μετατόπιση στην ύπαρξη ενός μεγάλου αριθμού ανεξάρτητα διοικούμενων δικτύων (π.χ. LANs) σήμαινε ότι η κατοχή ενός ενιαίου πίνακα των hosts δεν ήταν πλέον εφικτή, και το σύστημα Domain Name System (DNS) ανακαλύφθηκε από τον Paul Mockapetris του USC/ISI. Το Dns επέτρεψε έναν εξελικτικό διανεμημένο μηχανισμό για τα ονόματα των hosts (π.χ. www.acm.org) σε μια διεύθυνση Διαδικτύου.

Η αύξηση του μεγέθους του Internet προκάλεσε επίσης αλλαγές στους δρομολογητές. Αρχικά, υπήρχε ένας ενιαίος αλγόριθμος που εφαρμοζόταν ομοιόμορφα από όλους τους δρομολογητές. Δεδομένου ότι ο αριθμός δικτύων στο Internet έφτασε σε πολύ μεγάλο αριθμό, το αρχικό σχέδιο δεν θα μπορούσε να επεκταθεί ανάλογα με τις ανάγκες, έτσι αντικαταστάθηκε από ένα ιεραρχικό μοντέλο δρομολόγησης, με ένα εσωτερικό πρωτόκολλο πυλών (Interior Gateway Protocol - IGP) που χρησιμοποιήθηκε σε κάθε περιοχή του Internet, και ένα εξωτερικό πρωτόκολλο πυλών (Exterior Gateway Protocol - EGP) που χρησιμοποιήθηκε για να συνδέσει τις περιοχές.

Αυτό το σχέδιο επέτρεψε στις διαφορετικές περιοχές να χρησιμοποιήσουν ένα διαφορετικό IGP, έτσι ώστε οι διαφορετικές απαιτήσεις για το κόστος, το γρήγορο ανασχηματισμό, την ευρωστία και το μέγεθος του δικτύου να μπορούν να προσαρμοστούν εύκολα. Όχι μόνο ο αλγόριθμος δρομολόγησης, αλλά και το μέγεθος των πινάκων διευθύνσεων, έφερνε τους δρομολογητές στα όρια τους. Νέες προσεγγίσεις για τη συνάρθρωση διευθύνσεων έχουν εισαχθεί πρόσφατα για να ελέγχουν το μέγεθος των πινάκων δρομολογητών.

Δεδομένου ότι το Internet εξελίχθηκε, μια από τις σημαντικότερες προκλήσεις ήταν πώς να προσαρμόσει τις αλλαγές στο λογισμικό, ιδιαίτερα στο host λογισμικό. Το DARPA έδωσε στο πανεπιστήμιο του Berkley να ερευνήσει τις τροποποιήσεις στο λειτουργικό

σύστημα Unix, συμπεριλαμβανομένης της ενσωμάτωσης του TCP/IP που αναπτύχθηκε σε BBN. Αν και το Berkley ξανάγραψε αργότερα τον κώδικα BBN για να αρμόζει αποτελεσματικότερα στο Unix, η ενσωμάτωση του TCP/IP στις BSD Unix εκδόσεις αποδείχθηκε ένα κρίσιμο στοιχείο στη μετάδοση των πρωτοκόλλων στην ερευνητική κοινότητα.

Ένα μεγάλο μέρος της ερευνητικής κοινότητας άρχισε να χρησιμοποιεί το BSD Unix για το καθημερινό υπολογιστικό περιβάλλον τους. Κοιτάζοντας ξανά προς τα πίσω, η στρατηγική στα πρωτόκολλα του Internet σε ένα λειτουργικό σύστημα για την ερευνητική κοινότητα ήταν ένα από τα βασικά στοιχεία για την επιτυχή διαδεδομένη υιοθέτηση του Διαδικτύου.

Μια από τις πιο ενδιαφέρουσες προκλήσεις ήταν η μετάβαση του ARPANET host πρωτοκόλλου από NCP σε TCP/IP από τις 1 Ιανουαρίου 1983. Αυτή η μετάβαση, που απαιτούσε από όλους τους hosts να μετατραπούν ταυτόχρονα ή να αφεθούν σε μια επικοινωνία μέσω άλλων ειδικών μηχανισμών. Αυτή η μετάβαση προγραμματίστηκε προσεκτικά εντός της κοινότητας κατά τη διάρκεια αρκετών ετών προτού να πραγματοποιηθεί και να πάει τελικά εκπληκτικά ομαλά.

Το TCP/IP υιοθετήθηκε ως πρότυπο τρία έτη νωρίτερα το 1980. Αυτό έδωσε τη δυνατότητα στο υπουργείο άμυνας των ΗΠΑ να αρχίσει τη αποστολή πληροφοριών στη βάση τεχνολογίας των DARPA και το Internet, οδηγούμενοι άμεσα στον ενδεχόμενο χωρισμό σε στρατιωτικές και μη στρατιωτικές κοινότητες.

1980 -1989

Μέχρι το 1983, το ARPANET χρησιμοποιούταν από έναν σημαντικό αριθμό της αμυντικής R&D αλλά και άλλων οργανώσεων του στρατού. Η μετάβαση του ARPANET από NCP σε TCP/IP του επέτρεψε στο να χωριστεί σε ένα MILNET για τις στρατιωτικές λειτουργίες και ανάγκες και του ARPANET που ασχολούταν με την έρευνα. Κατά συνέπεια, μέχρι το 1985, το Internet είχε ήδη καλά καθιερωθεί ως τεχνολογία που υποστηρίζει μια ευρεία κοινότητα ερευνητών και υπεύθυνων για ανάπτυξη, και άρχισε να χρησιμοποιείται από άλλες κοινότητες για τις καθημερινές λειτουργίες.

Η χρησιμότητα της δικτύωσης των υπολογιστών - ειδικά του ηλεκτρονικού ταχυδρομείου - που καταδείχθηκε από το DARPA και το τμήμα άμυνας στο ARPANET δεν χάθηκε σε άλλες κοινότητες, κάτι που είχε ως αποτέλεσμα μέχρι τα μέσα της δεκαετίας του '70 δίκτυα υπολογιστών να εμφανίζονται συνέχεια αρκεί να υπήρχε η ανάλογη χρηματοδότηση.

Με εξαίρεση το BITNET και το USENET, αυτά τα πρόωρα δίκτυα (συμπεριλαμβανομένου και του ARPANET) ήταν κατασκευασμένα επί τούτου – δηλαδή προορίστηκαν, και κατά ένα μεγάλο μέρος περιορίστηκαν, στις κλειστές κοινότητες των ερευνητών. Επιπλέον, εναλλακτικές τεχνολογίες υπήρχαν στο εμπόριο, συμπεριλαμβανομένου του XNS από την Xerox, το DECnet, και το SNA από την IBM. Έμνε μόνο η βρετανική JANET (1984) και το NSFNET για να αναγγελθεί ρητά η πρόθεσή τους να εξυπηρετήσουν την πανεπιστημιακή κοινότητα. Ο όρος για ένα Αμερικανικό πανεπιστήμιο, για να λάβει τη χρηματοδότηση NSF για μια σύνδεση με το Διαδίκτυο ήταν η σύνδεση να δοθεί στη διάθεση όλων των καταρτισμένων χρηστών στην πανεπιστημιούπολη.

Το 1985, ο Dennis Jennings αποφάσισε να περάσει ένα έτος στο NSF. Συνεργάστηκε με την κοινότητα για να βοηθήσει το NSF να λάβει μια κρίσιμη απόφαση - ότι το TCP/ IP θα ήταν υποχρεωτικό για το πρόγραμμα NSFNET. Όταν ο Steve Wolff ανέλαβε το πρόγραμμα NSFNET το 1986, αναγνώρισε την ανάγκη για μια υποδομή δικτύωσης ευρεία περιοχής έτσι ώστε να υποστηρίξει τη γενική ακαδημαϊκή και ερευνητική κοινότητα, μαζί με την ανάγκη να αναπτυχθεί μια στρατηγική για τέτοια υποδομή σε μια ανεξάρτητη βάση από την

αμερικάνικη ομοσπονδιακή χρηματοδότηση. Πολιτικές και οι στρατηγικές υιοθετήθηκαν για να επιτευχθεί αυτός ο σκοπός.

Το NSF επίσης αποφάσισε να υποστηρίξει την οργανωτική υποδομή του Internet από το DARPA, που τακτοποιήθηκε ιεραρχικά κάτω από τον (τότε) πίνακα δραστηριοτήτων Internet (IAB). Η δημόσια δήλωση αυτής της επιλογής ήταν το κοινό συγγραφικό έργο από τη στρατιωτική δύναμη εφαρμοσμένης μηχανικής και αρχιτεκτονικής Internet του IAB και από την τεχνική συμβουλευτική ομάδα δικτύων NSF του RFC 985 («απαιτήσεις για τις πύλες Internet»), η οποία εξασφάλισε τυπική διαλειτουργικότητα των κομματιών DARPA και NSF του Διαδικτύου.

Εκτός από την επιλογή του TCP/IP για το πρόγραμμα NSFNET, οι ομοσπονδιακές αντιπροσωπείες έλαβαν και εφάρμοσαν διάφορες άλλες πολιτικές αποφάσεις που διαμόρφωσαν το Internet σήμερα. Οι ομοσπονδιακές αντιπροσωπείες μοιράστηκαν το κόστος της κοινής υποδομής, όπως τα υπερωκεάνια κυκλώματα. Για να συντονίσει αυτήν την διανομή, διαμορφώθηκε το Ομοσπονδιακό Συμβούλιο Δικτύωσης (Federation Network Comitee). Το FNC συνεργάστηκε επίσης και με άλλους διεθνείς οργανισμούς, όπως η RARE στην Ευρώπη, μέσω της συντονιστικής επιτροπής της διηπειρωτικής ερευνητικής δικτύωσης, το CCIRN (Coordinating Committee on Intercontinental Research Networking), για να συντονίσει την υποστήριξη του Internet και της ερευνητικής κοινότητας παγκοσμίως.

Αυτή η διανομή και η συνεργασία μεταξύ των αντιπροσωπειών στα ζητήματα του Internet είχε μια μακροχρόνια ιστορία. Μια πρωτοφανής συμφωνία του 1981 μεταξύ του Farber, ενεργώντας για το CSNET και το NSF, και του Kahn του DARPA, επέτρεψαν στο CSNET να μοιραστεί κυκλοφορία με το ARPANET σε μια στατιστική βάση.

Στη συνέχεια, με παρόμοιο τρόπο, το NSF ενθάρρυνε τα περιφερειακά (αρχικά ακαδημαϊκά) δίκτυά του NSFNET να παροτρύνουν τους εμπορικούς, μη ακαδημαϊκούς πελάτες, να επεκτείνουν τις εγκαταστάσεις τους για να τα εξυπηρετήσουν, και να εκμεταλλευτούν τις προκύπτουσες οικονομικές κλίμακες με τις χαμηλότερες δαπάνες συνδρομής.

Στη βάση του το NSFNET επέβαλε μια "αποδεκτή πολιτική χρήσης" (AUP - Acceptable Use Policy) που απαγόρευσε τη χρήση για λόγους "όχι υπέρ της έρευνας και της εκπαίδευσης. "Το προβλέψιμο (και προορισμένο) αποτέλεσμα της ενθάρρυνσης της εμπορικής κυκλοφορίας δικτύων στο τοπικό και περιφερειακό επίπεδο, αρνούμενο ταυτοχρόνως την πρόσβασή σε εθνική κλίμακα, ήταν για να υποκινηθεί η εμφάνιση ή/και η αύξηση" των ιδιωτικών " – ανταγωνιστικών δικτύων όπως το PSI, UUNET, το ANS CO+RE, και άλλα πιο πρόσφατα.

Το 1988, μια εθνική επιτροπή του ερευνητικού Συμβουλίου, που προέδρευε ο Kleinrock και είχε τους Kahn και Clark ως μέλη, συνέταξε μια έκθεση που ανατέθηκε από το NSF με τον τίτλο «προς ένα εθνικό ερευνητικό δίκτυο». Το 1994, μια εθνική έκθεση του ερευνητικού Συμβουλίου, που προέδρευε για ακόμα μια φορά ο Kleinrock (και με τους Kahn και Clark ως μέλη επίσης), τιτλοφορημένη «πραγματοποιώντας το μέλλον των πληροφοριών: Το Διαδίκτυο και το μέλλον».

Αυτή η έκθεση, που ανατέθηκε από το NSF, ήταν το έγγραφο στο οποίο ένα σχεδιάγραμμα για την εξέλιξη της λεωφόρου πληροφοριών αρθρώθηκε με αποτέλεσμα να έχει διαρκείς επιπτώσεις στον τρόπο σκέψης για την εξέλιξή του Internet. Ασχολήθηκε δε με τα κρίσιμα ζητήματα των δικαιωμάτων πνευματικής ιδιοκτησίας, της ηθικής, της τιμολόγησης, της εκπαίδευσης, της αρχιτεκτονικής και του κανονισμού για το Internet.

Το βάρος της χρηματοδότησης του NSFNET ήταν \$200 εκατομμύρια από το 1986 ως το 1995. Το 1990 όταν το ίδιο το ARPANET τέθηκε τελικά εκτός λειτουργίας, το TCP/IP είχε αντικαταστήσει ή είχε περιθωριοποιήσει τα περισσότερα άλλα πρωτόκολλα δικτύων. Κλειδί για την ταχεία ανάπτυξη του Διαδικτύου ήταν η ελεύθερη και ανοικτή πρόσβαση στα βασικά έγγραφα, ειδικά αυτά που περιέγραφαν τις προδιαγραφές των πρωτοκόλλων.

Οι αρχές του ARPANET και του Διαδικτύου στην πανεπιστημιακή ερευνητική κοινότητα προώθησαν την ακαδημαϊκή παράδοση ανοικτής δημοσίευσης των ιδεών και των αποτελεσμάτων. Εντούτοις, ο κανονικός κύκλος της παραδοσιακής ακαδημαϊκής δημοσίευσης ήταν πάρα πολύ επίσημος και πάρα πολύ αργός για την γρήγορη ανταλλαγή ουσιαστικών ιδεών στη δημιουργία των δικτύων.

Το 1969 ένα βασικό βήμα λήφθηκε από το S. Crocker (τότε ήταν στο UCLA) στην καθιέρωση των σημειώσεων του αιτήματος για τη σειρά σχολίων (Request for Comments - RFC). Αυτά τα υπομνήματα προορίστηκαν να είναι ένας άτυπος γρήγορος τρόπος διανομής να μοιραστούν οι ιδέες με άλλους ερευνητές δικτύων. Πρώτα το RFC τυπώθηκε σε χαρτί και διανεμήθηκε μέσω του κανονικού ταχυδρομείου. Καθώς το πρωτόκολλο μεταφοράς αρχείων (File Transfer Protocol -FTP) μπήκε στη χρήση, τα RFCs προετοιμάστηκαν ως απευθείας σύνδεση αρχεία και προσεγγίστηκαν μέσω του FTP. Τώρα, φυσικά, τα RFCs προσεγγίζονται εύκολα μέσω του World Wide Web επί των δωδεκάδων των τόπων σε όλο τον κόσμο.

Το SRI, στο ρόλο του ως κέντρο πληροφόρησης δικτύων, διατήρησε τους σε απευθείας σύνδεση καταλόγους αρχείων. Η επίδραση του RFCs ήταν να δημιουργηθεί ένα θετικό σύστημα ανατροφοδότησης πληροφοριών, με τις ιδέες ή τις προτάσεις που υποβλήθηκαν σε ένα RFC. Όταν κάποια συναίνεση (ή τουλάχιστον ένα συνεπές σύνολο ιδεών) ενωνόταν ένα έγγραφο προδιαγραφών θα καταρτιζόταν. Μια τέτοια προδιαγραφή θα χρησιμοποιούταν έπειτα ως βάση για τις εφαρμογές από τις διάφορες ερευνητικές ομάδες.

Το Διαδίκτυο είναι τόσο μια συλλογή κοινοτήτων όσο και μια συλλογή τεχνολογιών, και η επιτυχία της αποδίδεται κατά ένα μεγάλο μέρος και στην ικανοποίηση των αναγκών καθώς επίσης και στη χρήση της κοινότητας με έναν αποτελεσματικό τρόπο να ωθηθεί η υποδομή προς τα εμπρός. Αυτό το κοινοτικό πνεύμα έχει μια μακροχρόνια ιστορία αρχίζοντας με το πρόωρο ARPANET.

Οι πρώτοι ARPANET ερευνητές εργάστηκαν σαν μια μικρή κοινότητα. Επιπλέον, τα δορυφορικά δίκτυα, τα ραδιοφωνικά δίκτυα και διάφορα άλλα ερευνητικά προγράμματα πληροφορικής του DARPA ήταν συνεργάσιμες δραστηριότητες που χρησιμοποίησαν ευρέως τους διαθέσιμους μηχανισμούς που υπήρξαν για να συντονίσουν τις προσπάθειές τους, αρχίζοντας από το ηλεκτρονικό ταχυδρομείο και με την προσθήκη της διανομής αρχείων, της εξ' αποστάσεως πρόσβασης, και να καταλήξουν στον παγκόσμιο ιστό ή αλλιώς World Wide Web. Κάθε ένα από αυτά τα projects διαμόρφωσε μια ομάδα εργασίας, αρχίζοντας από την ARPANET ομάδα εργασίας δικτύων. Λόγω του μοναδικού ρόλου που το ARPANET διαδραμάτισε ως υποδομή που υποστηρίζει τα διάφορα ερευνητικά προγράμματα, δεδομένου και του ότι το Internet άρχισε να εξελίσσεται, η Network Working Group εξελίχθηκε σε Internet Working Group.

Προς τα τέλη της δεκαετίας του '70, η αναγνώριση ότι η αύξηση του Internet συνοδεύθηκε από μια αύξηση του μεγέθους της ενδιαφερόμενης ερευνητικής κοινότητας και επομένως μια αυξανόμενη ανάγκη για τους μηχανισμούς συντονισμού. Ο Vint Cerf, κατόπιν διευθυντής του προγράμματος Internet σε DARPA, διαμόρφωσε διάφορους οργανισμούς συντονισμού - έναν διεθνές συμβούλιο συνεργασίας (International Cooperation Board - ICB), που προοδεύοταν από τον Peter Kirstein στο UCL, για να συντονίσει τις δραστηριότητες με μερικές συνεργαζόμενες ευρωπαϊκές χώρες που ασχολήθηκαν στη στα δορυφορικά δίκτυα, μια ερευνητική ομάδα Internet που ήταν μια συμπεριλαμβανούσα ομάδα η οποία παρείχε ένα περιβάλλον για τη γενική ανταλλαγή των πληροφοριών.

Το 1983, όταν ανέλαβε ο Barry Leiner τη διαχείριση του ερευνητικού προγράμματος του Internet στο DARPA, αυτός και ο Clark αναγνώρισαν ότι η συνεχιζόμενη αύξηση της κοινότητας του Διαδικτύου απαιτούσε μια αναδιάρθρωση των μηχανισμών συντονισμού. Ο ICCB διαλύθηκε και σε ισχύ τέθηκε μια δομή με το όνομα Task Force, στρεφόμενη σε έναν

ιδιαίτερο τομέα της τεχνολογίας (π.χ. δρομολογητές, πρωτόκολλα end to end, κλπ.). Ο Internet Activities Board (IAB) διαμορφώθηκε από τις Task Forces.

Μετά από κάποιες μεταβολές των μελών στο IAB, ο Phill Gross έγινε διευθυντής μιας αναζωογονημένης ομάδας εργασίας εφαρμοσμένης μηχανικής Internet (Internet Engineering Task Force - IETF). Όπως είδαμε παραπάνω, μέχρι το 1985 υπήρξε μια τεράστια αύξηση της πλευράς πρακτικότερης/εφαρμοσμένης μηχανικής του Internet. Αυτή η αύξηση οδήγησε σε μια έκρηξη συμμετοχής στις συνεδριάσεις IETF, και ο Gross αναγκάστηκε να δημιουργήσει μια υποδομή στο IETF υπό μορφή ομάδων εργασίας. Αυτή η αύξηση συμπληρώθηκε από μια σημαντική επέκταση στην κοινότητα. Δεν ήταν πλέον το DARPA ο μόνος σημαντικός φορέας στη χρηματοδότηση του Internet.

Εκτός από το NSFNET και τις διάφορες άλλες χρηματοδοτημένες δραστηριότητες των ΗΠΑ αλλά και από τον υπόλοιπο κόσμο, το ενδιαφέρον για τον εμπορικό τομέα άρχισε να αυξάνεται. Επίσης το 1985, και ο Kahn και ο Leiner άφησαν το DARPA και υπήρξε μια σημαντική μείωση στη δραστηριότητα που αφορά το Internet. Κατά συνέπεια, το IAB αφέθηκε χωρίς έναν αρχικό χορηγό και άρχισε να αποκτά όλο και περισσότερο το μανδύα της ηγεσίας. Η αύξηση συνεχίστηκε, με συνέπεια ακόμη και την περαιτέρω υποδομή και μέσα στο IAB αλλά και στο IETF. Το IETF συνδύασε τις ομάδες εργασίας στις περιοχές, και υπέδειξε τους Area Directors. Μια ομάδα Internet Engineering Steering Group (IESG) διαμορφώθηκε από τους Area Directors. Το IAB αναγνώρισε την αυξανόμενη σημασία του IETF, και αναδόμησε τη διαδικασία προτύπων για να αναγνωρίσει ρητά το IESG ως σημαντικότερο σώμα αναθεώρησης για τα πρότυπα.

1990 – σήμερα

Η αύξηση του εμπορικού τομέα που παρουσιάστηκε επέφερε και αύξηση στην ανησυχία σχετικά με την ίδια τη διαδικασία προτύπων. Ξεκινώντας στις αρχές της δεκαετίας του '80 και συνεχιζόμενο μέχρι σήμερα, το Internet αυξήθηκε πέρα από τις αρχικές ερευνητικές ρίζες του για να περιλάβει και μια ευρεία κοινότητα χρηστών και αύξησε την εμπορική δραστηριότητα. Αυξανόμενη προσοχή δόθηκε για να καταστήσει τη διαδικασία αυτή ανοικτή και δίκαιη. Αυτό συνδέθηκε με μια αναγνωρισμένη ανάγκη για την ενίσχυση της Κοινότητας του Internet που οδηγήθηκε τελικά στο σχηματισμό της Internet Society το 1991, υπό την αιγίδα της εταιρίας Kahn για τις εθνικές ερευνητικές πρωτοβουλίες (Corporation for National Research Initiatives - CNRI) και την ηγεσία του Cerf, που τότε ήταν στο CNRI.

Το 1992, ακόμα μια αναδιοργάνωση πραγματοποιήθηκε. Το 1992, το Internet Activities Board αναδιοργανώθηκε και μετονομάστηκε σε Internet Architecture Board που λειτουργεί υπό την αιγίδα της Internet Society. Μια πιο «όμοια» σχέση καθορίστηκε μεταξύ του νέων IAB και του IESG, με το IETF και το IESG να παίρνουν μια μεγαλύτερη ευθύνη για την έγκριση των προτύπων. Τελικά, μια συνεταιριστική και αμοιβαία ενθαρρυντική σχέση διαμορφώθηκε μεταξύ του IAB, του IETF, και της Internet Society, με την με την τελευταία να αποκτά ως στόχο την παροχή των υπηρεσιών και άλλων μέτρων που θα διευκόλυναν τις εργασίες του IETF.

Η πρόσφατη ανάπτυξη και επέκταση του παγκόσμιου ιστού έφερε και μια νέα κοινότητα, καθώς πολλοί από τους ανθρώπους που εργάζονται για τον παγκόσμιο ιστό δεν έχουν σκεφτεί τους εαυτούς τους ως ερευνητές και υπεύθυνους για την ανάπτυξη δικτύων. Μια νέα οργάνωση συντονισμού διαμορφώθηκε, η κοινοπραξία World Wide Web (World Wide Web Consortium - W3C). Αρχικά ηγήθηκε ο Tim Berners-Lee (ο εφευρέτης του παγκόσμιου ιστού) και τον Al Vezza, το W3C έχει πάρει την ευθύνη για τα διάφορα πρωτόκολλα και τα πρότυπα που συνδέονται με τον παγκόσμιο ιστό.

Από τότε μέχρι σήμερα η συνέχεια είναι λίγο πολύ γνωστή. Το Google έχει γίνει η απόλυτη μηχανή αναζήτησης, από IP v4 περνάμε σιγά σιγά σε IP v6, έχουμε τα blogs, τα

forums, τα site κοινωνικής δικτύωσης (myspace, facebook, hi5, twitter κτλ), και πλέον μιλάμε για cloud. Έχουμε adsl2+, δορυφορικές συνδέσεις, triple play και πλέον το internet είναι τόσο σημαντικό όσο τα κλειδιά του σπιτιού μας.

Κεφάλαιο 2^ο

Λογισμικό Ανοιχτού Κώδικα

Εισαγωγή

Για τον επιχειρηματία, το ανοιχτό λογισμικό, πρόκειται απλώς για λογισμικό που διατίθεται δωρεάν και με όλο του τον κώδικα. (Το λογισμικό "γράφεται" με μια γλώσσα προγραμματισμού και στη συνέχεια αυτός ο "πηγαίος κώδικας", μετατρέπεται σε εκτελέσιμο πρόγραμμα.)

Αντίθετα, στο εμπορικό λογισμικό ο χρήστης όχι μόνο πληρώνει για να λάβει το πρόγραμμα που θα "τρέχει" στον Η/Υ του, αλλά δεν έχει στη διάθεσή του και τον πηγαίο κώδικα. Δεν μπορεί λοιπόν να αλλάξει το πρόγραμμα ο ίδιος και πρέπει να ζητά (και να πληρώνει) όποια τροποποίηση επιθυμεί μόνο στον αρχικό κατασκευαστή.

Η παραπάνω ερμηνεία είναι απλοϊκή. Ιδεολογικά υπάρχουν σημαντικές διαφορές ανάμεσα στο λογισμικό ανοικτού κώδικα (Open Source) και το ελεύθερο λογισμικό (Free Software), καθώς το πρώτο υποστηρίζει ότι ο κώδικας πρέπει να είναι διαθέσιμος σε όλους διότι αυτό κάνει τα προγράμματα καλύτερα, ενώ το δεύτερο ότι ο κώδικας πρέπει να είναι διαθέσιμος σε όλους διότι μόνο έτσι η ελευθερία του χρήστη δεν περιορίζεται από τις αποφάσεις του κατασκευαστή. Το ότι τα προγράμματα γίνονται καλύτερα λοιπόν αποτελεί ένα ευχάριστο αλλά δευτερεύον χαρακτηριστικό. Οι οπαδοί του Free Software θα το χρησιμοποιούσαν ακόμη κι αν το εμπορικό λογισμικό ήταν καλύτερης ποιότητας από το δικό τους.

Πλεονεκτήματα λογισμικού ανοιχτού κώδικα

- **Αξιοπιστία** - Πολλές έρευνες (Zdnet, Bloor Research, Syscontrol AG, Netcraft κ.λ.π.) έχουν δείξει την ανωτερότητα εφαρμογών όπως το λειτουργικό σύστημα GNU/Linux και ο Apache web server. Για παράδειγμα, σε δοκιμή του Zdnet διάρκειας 10 μηνών τα Windows NT "κράσαραν" κατά μέσο όρο μια φορά κάθε 6 εβδομάδες και απαιτούσαν 30 λεπτά για την επισκευή τους ενώ το GNU/Linux ποτέ! Αν αυτό φαίνεται "παράλογο" (το δωρεάν λειτουργεί πιο αξιόπιστα από το πληρωμένο) θυμηθείτε ότι οι εφαρμογές Open Source είναι αποτέλεσμα εθελοντικής εργασίας. Δεν υπάρχει λοιπόν πίεση από το Marketing και τις πωλήσεις να παρουσιάσουμε κάτι ακόμη κι αν δεν είναι έτοιμο ή καλά δοκιμασμένο, ενώ όλοι μπορούν να δουν, να σχολιάσουν και να διορθώσουν τη δουλειά των άλλων.
- **Αποδοτικότητα** - Μεγάλος αριθμός συγκριτικών δοκιμών έχει αποδείξει την ταχύτητα και αποδοτικότητα πολλών Open Source εφαρμογών βασισμένων στο GNU/Linux (π.χ. PC Magazine, Sys Admin magazine, SPEC Consortium, IBM, Fastcenter, Ziff Davis, Mindcraft κ.λ.π.).
- **Επεκτασιμότητα** - Οι περισσότερες επιχειρηματικές εφαρμογές επιθυμούν χαμηλό κόστος εκκίνησης με εύκολη και γρήγορη αναβάθμιση, αν αποδειχθεί ότι η υπηρεσία που δημιουργήθηκε είναι δημοφιλής. Λόγω του ελεύθερου κώδικά τους οι εφαρμογές Open Source μπορούν εύκολα να τοποθετηθούν σε πολλά

διαφορετικά είδη (πλατφόρμες) υπολογιστών, αλλά και να λειτουργήσουν "εν παραλλήλω" σε μεγάλα συστήματα υψηλών επιδόσεων.

- **Ασφάλεια δεδομένων** - Όλοι οι κρυπτογράφοι γνωρίζουν πως όποιο σύστημα ελέγχεται από πολλούς ανθρώπους είναι ασφαλέστερο από εκείνο που διαχειρίζονται μόνο λίγοι (όσο "καταρτισμένοι ή ευφυείς" κι αν είναι αυτοί). Χάρη στον ανοικτό τους κώδικα οι εφαρμογές Open Source "ελέγχονται" από χιλιάδες ανθρώπους και τα όποια προβλήματά τους ανακαλύπτονται πολύ γρήγορα. Για παράδειγμα, όπως έδειξε έρευνα της Attrition.org το 59% των defaced (τους άλλαξαν τη home page) sites στο Internet έτρεχαν Windows, ενώ μόνο το 21% GNU/Linux, πράγμα που θα ήταν φυσιολογικό μόνο αν στο δίκτυο υπήρχαν 3 φορές περισσότερα συστήματα Windows απ' ό,τι GNU/Linux (κάτι που φυσικά δεν συμβαίνει). Άλλη έρευνα (Security Portal) έδειξε πως η ταχύτητα διευθέτησης όσων προβλημάτων εμφανίζονται στο GNU/Linux είναι πολύ μεγαλύτερη απ' ό,τι στα Windows.
- **Προστασία από ιούς** - Υπάρχουν περίπου 60.000 ιοί για Windows και περίπου 40 για GNU/Linux.
- **Χαμηλότερο κόστος λειτουργίας (Total Cost of Ownership)** - Το Open Source λογισμικό (λειτουργικά συστήματα, βάσεις δεδομένων κ.λ.π.) παρέχεται δωρεάν και για απεριόριστο αριθμό χρηστών. Επίσης, η επιχείρηση μπορεί να αξιοποιήσει παλαιότερο εξοπλισμό (αφού είναι ελεύθερη να κάνει τροποποιήσεις στον κώδικα), μειώνοντας έτσι τα έξοδά της. Χαρακτηριστική εδώ είναι η περίπτωση του Amazon.com. Ελάχιστοι γνωρίζουν ότι το γνωστό βιβλιοπωλείο κατάφερε να καταστεί κερδοφόρο επειδή στράφηκε στο Open Source λογισμικό, μειώνοντας θεαματικά τα έξοδά του. Επίσης, πολλές χρηματιστηριακές επιχειρήσεις χρησιμοποιούν Linux γεγονός που κρατούν κρυφό, θεωρώντας το ως ένα σημαντικό ανταγωνιστικό πλεονέκτημα!
- **Ευκολότερες και φθηνότερες επεκτάσεις** - Επειδή ο χρήστης διαθέτει πρόσβαση στον κώδικα του λογισμικού που χρησιμοποιεί μπορεί να ζητήσει επεκτάσεις ή αλλαγές από οποιονδήποτε. Ο κατασκευαστής δηλαδή δεν διαθέτει μονοπωλιακή δύναμη επάνω του και δεν μπορεί να τον "εκβιάσει" με υποχρεωτικές αναβαθμίσεις ή υπέρογκες χρεώσεις για μικρές εργασίες. Αν ο χρήστης λοιπόν δεν είναι ικανοποιημένος από τη συνεργασία ενός προμηθευτή μπορεί να απευθυνθεί σε κάποιον άλλον, χωρίς να απαιτηθεί η αντικατάσταση των εφαρμογών που χρησιμοποιεί.
- **Διαρκής και εκτενής υποστήριξη** - Οι κατασκευαστές εμπορικού λογισμικού διορθώνουν μόνο τα πιο ενοχλητικά από τα προβλήματα των εφαρμογών τους και υποχρεώνουν τους πελάτες τους να αγοράζουν νέες αναβαθμίσεις ακόμη κι αν δεν τις χρειάζονται, σταματώντας την υποστήριξη των παλαιότερων εκδόσεων του προγράμματος. Στις εφαρμογές Open Source όμως η επιχείρηση μπορεί να διορθώσει η ίδια ότι την "πονάει" στην εφαρμογή και δεν υποχρεώνεται να αναβαθμίσει τίποτε αν δεν το επιθυμεί.

Μειονεκτήματα λογισμικού ανοιχτού κώδικα

- **Μερίδιο αγοράς** - Αν και ο Apache αποτελεί τον δημοφιλέστερο web server στο Internet και το GNU\Linux διαθέτει ένα μεγάλο ποσοστό της αγοράς των Internet Servers, στην Ελλάδα υπάρχουν ακόμη πολύ λίγοι τεχνικοί με εμπειρία σε παρόμοια συστήματα. Αναμφίβολα πάντως οι γνώσεις τους είναι πολύ μεγαλύτερες από εκείνες του μέσου windows administrator ο οποίος συνήθως μπορεί να εκτελέσει μόνο τις πολύ βασικές λειτουργίες ενός συστήματος και αγνοεί τις πιο προχωρημένες δυνατότητες και εφαρμογές του. (Είναι πιο εύκολο να μάθεις τις βασικές λειτουργίες ενός windows συστήματος, αλλά δυσκολεύεσαι κατόπιν πολύ περισσότερο απ' ό,τι στο GNU\Linux για να κατανοήσεις και να υλοποιήσεις κάτι προχωρημένο.)
- **Ενδοεταιρικός καταλογισμός ευθυνών** - Όπως λέει ένα γνωστό αμερικάνικο ρητό "Κανείς δεν απολύθηκε επειδή αγόρασε IBM". Ένα προϊόν της Microsoft ή της Oracle μπορεί να κοστίζει πολύ περισσότερα χρήματα και να προσφέρει λιγότερα απ' όσα μια εφαρμογή Open Source, αλλά όποιο πρόβλημα και αν παρουσιαστεί, το στέλεχος που εισηγήθηκε την αγορά του μπορεί πάντοτε να ισχυριστεί πως "έκανα μια επώνυμη επιλογή ενός μεγάλου ονόματος για να έχουμε το καλύτερο". Αν όμως πάει στραβά το παραμικρό σε μια εφαρμογή Open Source (και στην πληροφορική πάντα κάτι θα λειτουργήσει στραβά) η μομφή "γιατί εμπιστευθήκαμε κάτι φτηνιάρικο;" δύσκολα μπορεί να απαντηθεί με επιτυχία, ειδικά αν ο ερωτών είναι ο άσχετος από υπολογιστές προϊστάμενος όπως συμβαίνει συνήθως. Τα πράγματα θα ήταν καλύτερα αν υπήρχαν και στη χώρα μας εταιρείες εξειδικευμένες στην υποστήριξη open source εφαρμογών, αλλά απ' όσο γνωρίζω αυτό δεν συμβαίνει.

Κεφάλαιο 3^ο

Εφαρμογές ΛΑΚ

Εισαγωγή

Στις πρώιμες ημέρες του Internet η δημιουργία ενός site ήταν απλή υπόθεση. Συνήθως το έφτιαχνε και το συντηρούσε ένας άνθρωπος (ο Webmaster) μέσα από μια διαδικασία δημιουργίας ενός μόνο τύπου περιεχομένου : την στατική σελίδα HTML. Η τοποθεσία και η δημιουργία του site ήταν απλή υπόθεση : κάθε site είχε μια δενδροειδή μορφή όπου η στατική σελίδα βρισκόταν «θαμμένη» σε βάθος τριών, τεσσάρων ή και πέντε επιπέδων και ο μόνος τρόπος να την επισκεφτεί ο χρήστης του Internet ήταν μέσω links στην αρχική σελίδα ή σε άλλες σελίδες. Έτσι κι αλλιώς ένα site ήταν κάτι καινούργιο τότε και αρκούσε να έχει κάποιος ένα ώστε να είναι μπροστά από τον ανταγωνισμό.

Τα πράγματα βέβαια δεν είναι πια τόσο απλά. Ένα οποιοδήποτε site μεσαίου μεγέθους ή μεγαλύτερο χρειάζεται έναν σημαντικό αριθμό εργαζομένων για να το ενημερώνει (πολλοί από τους οποίους δεν έχουν τεχνικές γνώσεις). Το περιεχόμενο έχει πλέον πολλές πηγές αλλά και πολλές “κατευθύνσεις”. Επίσης, ο αριθμός των επισκεπτών έχει αυξηθεί εκθετικά το οποίο συνεπάγεται την ανάγκη ύπαρξης σημαντικής υποδομής σε λογισμικό, hardware και bandwidth για την εξυπηρέτηση των μεγάλων αριθμών χρηστών.

Για την αντιμετώπιση αυτών των αναγκών δημιουργήθηκαν αρχικά οι δυναμικές ιστοσελίδες, σελίδες δηλαδή που δημιουργούνται αυτόματα από προγράμματα που εκτελούνται στον server. Πολλές φορές χρησιμοποιούνται και βάσεις δεδομένων στις οποίες είναι αποθηκευμένο το περιεχόμενο των σελίδων και τα προγράμματα (σε γλώσσες όπως Perl, ASP.NET, PHP, JSP κλπ) αναλαμβάνουν την λήψη των πληροφοριών και την διαμόρφωση της HTML, δυναμικά, κατά τον χρόνο που το ζητάει ο επισκέπτης.

Ένα σύστημα διαχείρισης περιεχομένου όμως είναι κάτι πολύ διαφορετικό από απλώς ένα δυναμικό site. Τα CMS έχουν ένα μεγάλο αριθμό δυνατοτήτων και έχουν την δυνατότητα διαχείρισης ιδιαίτερα πολύπλοκων και μεγάλων site. Η αγορά των συστημάτων διαχείρισης περιεχομένου είναι μια νέα αναπτυσσόμενη αγορά και το μέγεθός της από 3.5 δις δολάρια το 2001 έφτασε τα 7 δις. δολάρια το 2006.

Τι είναι το Σύστημα Διαχείρισης Περιεχομένου (CMS)

Το Σύστημα Διαχείρισης Περιεχομένου, είναι ένα πρόγραμμα ειδικά σχεδιασμένο για τη διαχείριση sites. Δημιουργείται και εγκαθίσταται από τους σχεδιαστές ιστοσελίδων, αλλά προορίζεται για χρήση από τελικούς χρήστες. Αρχικά, προσφέρει έναν εύκολο, εύχρηστο τρόπο ενημέρωσης περιεχομένου. Αυτό συνήθως γίνεται με τη χρήση ενός συστήματος πλοήγησης (browser). Ο χρήστης απλά εισάγει το νέο κείμενο και το αποθηκεύει. Η ιστοσελίδα ενημερώνεται αμέσως! Το ίδιο απλό είναι να προστεθούν νέες σελίδες, να διαγραφούν παλαιές, ή να αναδιαμορφωθεί μια ιστοσελίδα ώστε να συμβαδίζει με νέες απαιτήσεις ή προδιαγραφές.

Το Σύστημα Διαχείρισης Περιεχομένου αυτοματοποιεί διάφορες διαδικασίες όπως η διατήρηση της εμφάνισης των σελίδων σε όλο το site καθώς και η δημιουργία των σχετικών μενού, συνδέσμων κλπ. Επίσης η ύπαρξη και αρκετών άλλων εργαλείων διαχείρισης, επιτρέπει την εστίαση στις λέξεις και όχι στην τεχνολογία.

Τι σημαίνει περιεχόμενο (content)

Υπάρχουν διάφοροι ορισμοί για το τον όρο περιεχόμενο. Μια απλή περιγραφή είναι ότι περιεχόμενο είναι η (ψηφιακή) πληροφορία η οποία έχει τεθεί σε χρήση. Η πληροφορία τίθεται σε χρήση όταν μορφοποιείται και δημοσιεύεται για έναν συγκεκριμένο σκοπό. Συνήθως το περιεχόμενο δεν είναι μια απλή μονάδα πληροφορίας αλλά ένας συνδυασμός βασικών μονάδων (κείμενο, εικόνες, διαγράμματα, media κλπ) που έχουν οργανωθεί σε ένα ολοκληρωμένο σύνολο. Π.χ. βιβλία (που περιέχουν κεφάλαια και παραγράφους), περιοδικά (που περιέχουν άρθρα, διαφημίσεις, εικόνες και αγγελίες) – και τώρα πια και websites που περιέχουν άρθρα, διαφημίσεις, forums, καταλόγους κλπ.

Το CMS σε βάθος

Το Content Management είναι ουσιαστικά η διαχείριση του περιεχομένου (όπως αυτό ορίστηκε προηγουμένως) με την χρήση κανόνων, διαδικασιών και / ή προδιαγεγραμμένων ροών εργασίας (workflows), με τέτοιο τρόπο ώστε οι διαχειριστές, προγραμματιστές και συγγραφείς περιεχομένου να δημιουργούν, τροποποιούν, διαχειρίζονται και να εκδίδουν όλο το περιεχόμενο μιας ή περισσότερων ιστοσελίδων υπακούοντας πάντα σε ένα συγκεκριμένο πλαίσιο απαιτήσεων. Ένα τέτοιο σύστημα βοηθάει στην οργάνωση και αυτοματοποίηση της συλλογής διαχείρισης και έκδοσης του περιεχομένου. Χρειάζεται όταν:

Υπάρχει μεγάλος όγκος πληροφορίας ώστε να μην είναι πρακτική η χειρωνακτική επεξεργασία. Η πληροφορία τροποποιείται πολύ γρήγορα ώστε να μην είναι πρακτική η χειρωνακτική επεξεργασία. Από μια σειρά πληροφορίας πρέπει να δημιουργηθούν πολλαπλές μορφές περιεχομένου. Ο σχεδιασμός της εμφάνισης της έκδοσης (π.χ. της ιστοσελίδας) θα πρέπει να διαχωριστεί από το περιεχόμενο έτσι ώστε αλλαγές στην δομή της παρουσίασης να μην πρέπει να γίνουν σε κάθε σελίδα του site.

Content domain

Μια βασική έννοια στην συζήτηση του content management είναι το content domain. Το content domain είναι το εύρος της πληροφορίας η οποία θα καταγραφεί, συνδυασθεί και εκδοθεί. Το content domain συσχετίζεται άμεσα με τους στόχους του CMS και κάθε ένα τέτοιο σύστημα θα πρέπει να έχει ένα καλά ορισμένο content domain. Συνήθως το content domain είναι μια φράση δύο- τριών γραμμών που θα πρέπει να δίνει με σαφήνεια τι θα συμπεριλαμβάνεται στο περιεχόμενο του site και τι όχι.

Μορφή (Format) παρουσίασης

Το θέμα της μορφής της παρουσίασης μπορεί να είναι αρκετά πολύπλοκο. Αν ο στόχος είναι απλώς η δημιουργία ενός site τότε τα πράγματα είναι απλά - χρησιμοποιείται η HTML. Αν όμως υπάρχουν απαιτήσεις παρουσίασης της ίδιας πληροφορίας με διαφορετική μορφή (π.χ. εκτός από την κλασική ιστοσελίδα να υπάρχει και εναλλακτική μορφή της ίδιας σελίδας για χρήστες κινητού, χρήστες palm καθώς και για εκτύπωση) τότε

το CMS θα πρέπει να είναι πιο πολύπλοκο και θα πρέπει να υποστηρίζει τον διαχωρισμό μεταξύ περιεχομένου και τρόπου παρουσίασης του ώστε να είναι δυνατή η δημιουργία εναλλακτικών μορφών παρουσίασης χωρίς να χρειάζεται να ξανά δημιουργηθεί το περιεχόμενο.

Λειτουργίες ενός CMS

Τα CMS διαφοροποιούνται μεταξύ τους σε αρκετά σημεία, επειδή όμως όλα έχουν κοινό στόχο θα πρέπει οπωσδήποτε να υποστηρίζουν κάποιες βασικές λειτουργίες. Έτσι, διακρίνονται κάποια υπο-συστήματα τα οποία είναι βασικά και θα πρέπει να τα διαθέτει οποιοδήποτε σοβαρό CMS. Αυτά είναι:

- Σύστημα σύνταξης (authoring)
- Σύστημα διαχείρισης (management)
- Σύστημα αυτοματοποίησης κύκλου εργασιών (workflow automation)
- Σύστημα έκδοσης

Σύστημα σύνταξης (authoring)

Αναλυτικά όσον αφορά το authoring θα μπορούσαμε να πούμε τα εξής: Περιλαμβάνει όλα τα εργαλεία που απαιτούνται για την δημιουργία, τροποποίηση και διαγραφή περιεχομένου στο site. Παραδοσιακά αυτή η εργασία ήταν χρονοβόρα, κοπιαστική και επιρρεπής σε λάθη. Τα σύγχρονα συστήματα όμως, επιτρέπουν στους συγγραφείς περιεχομένου (content authors) να δημιουργούν και αποθηκεύουν το περιεχόμενο σε μια κεντρική βάση δεδομένων μαζί με άλλες πληροφορίες όπως συγγραφέας, διορθωτής, ημερομηνίες έκδοσης.

Οι πιο χρήσιμες υλοποιήσεις είναι αυτές που επιτρέπουν την χρήση απλής διεπαφής, βασισμένης σε φόρμες που ανοίγουν από έναν κοινό browser. Η προσέγγιση αυτή επιτρέπει την πρόσβαση στο σύστημα από οπουδήποτε (διαμέσων του Internet) δίνοντας έτσι την δυνατότητα ύπαρξης πραγματικά κατανεμημένων ομάδων εργασίας.

Ένα άλλο τμήμα του συστήματος authoring είναι αυτό που αναλαμβάνει την συλλογή πληροφοριών από διάφορες πηγές, την μετατροπή τους σε άλλα format και την αποθήκευσή τους στην βάση δεδομένων. Τέτοια εργαλεία θα πρέπει να είναι ως επί το πλείστον αυτοματοποιημένα και να απαιτούν ελάχιστη ανθρώπινη επέμβαση, κυρίως για λόγους συντήρησης. Ένα τέτοιο σύστημα, π.χ. για ένα site οικονομικού περιεχομένου θα ήταν ένα πρόγραμμα το οποίο έπαιρνε αυτόματα κάθε ημέρα τις τιμές κλεισίματος των μετοχών και ενημέρωνε την βάση δεδομένων.

Σύστημα διαχείρισης (management)

Το σύστημα διαχείρισης περιλαμβάνει τα εργαλεία που επιτρέπουν την οργάνωση του site σε έναν οποιοδήποτε αριθμό ενότητων (sections). Μια ενότητα περιεχομένου είναι μια περιοχή του site που συνήθως εμφανίζει περιεχόμενο ενός συγκεκριμένου τύπου (π.χ. δελτία τύπου, άρθρα, προδιαγραφές προϊόντων, κλπ). Ένα προχωρημένο CMS επιτρέπει στους διαχειριστές την δημιουργία ενότητων και την ανάθεση της διαχείρισης του σε άλλους χρήστες.

Ακόμα, το σύστημα διαχείρισης θα πρέπει να ενημερώνει αυτόματα τις περιοχές πλοήγησης στο site (navigation toolbars, χάρτη site κλπ). Επίσης το σύστημα διαχείρισης θα πρέπει να δίνει την δυνατότητα της εύκολης διασύνδεσης δυναμικών τμημάτων με συγκεκριμένες ενότητες (π.χ. τα άρθρα μπορεί να είναι συνδεδεμένα με ένα online poll,

κάτι που δεν θα ισχύει για τα δελτία τύπου). Τέλος, το σύστημα θα πρέπει να διευκολύνει τους διαχειριστές στην δημιουργία και συντήρηση τέτοιων σχέσεων.

Σύστημα αυτοματοποίησης κύκλου εργασιών (workflow automation)

Συνήθως οι οργανισμοί έχουν κανόνες για το ποιος μπορεί να δημιουργεί, τροποποιεί και εγκρίνει ότι το περιεχόμενο είναι έτοιμο για δημοσίευση. Τα πράγματα γίνονται πιο πολύπλοκα αν αναλογιστεί κανείς ότι για την δημιουργία περιεχομένου για το Internet συνήθως πρέπει να συνεργαστούν άτομα με διαφορετικές ειδικότητες – άλλοι γράφουν το κείμενο, άλλοι δημιουργούν διαγράμματα, άλλοι διαλέγουν τα γραφικά κλπ. Ένα CMS θα πρέπει λοιπόν να διαθέτει ένα σύστημα αυτοματοποίησης αυτών των εργασιών. Το σύστημα αυτό λέγεται σύστημα αυτοματοποίησης ροών εργασίας (workflow automation system) και θα πρέπει να επιτρέπει:

- Τον διαχωρισμό των χρηστών βάση ρόλων . Οι ρόλοι σε έναν εκδοτικό οργανισμό είναι λίγο πολύ σταθεροί : υπάρχουν οι συντάκτες (authors), οι διορθωτές (editors), οι σχεδιαστές κ.α. Συνήθως το σύστημα διαθέτει έναν προκαθορισμένο αριθμό ρόλων ενώ δίνει την δυνατότητα για την δημιουργία νέων.
- Την περιγραφή των διαδικασιών παραγωγής περιεχομένου μέσω μιας σειράς βημάτων και απαιτούμενων εγκρίσεων που θα πρέπει να εκτελέσουν οι ρόλοι. Ο διαχειριστής θα πρέπει να μπορεί να δημιουργήσει / τροποποιήσει εύκολα αυτές τις διαδικασίες και να δώσει δικαιώματα πρόσβασης ανά κατηγορία χρήστη. Επίσης θα πρέπει να υπάρχει η δυνατότητα δημιουργίας διαφορετικού κύκλου εργασίας ανάλογα με τον τύπο περιεχομένου που δημιουργείται (π.χ. ένα δελτίο τύπου θα απαιτεί διαφορετική διαδικασία από το τεχνικό φυλλάδιο ενός προϊόντος)
- Την δυνατότητα εκκίνησης διαδικασιών και ανάθεσης εργασιών (όπως “σύνταξε”, “διόρθωσε”, “ανέβασε”) από τους managers σε ρόλους ή και σε απλούς χρήστες, καθώς και την δυνατότητα λήψης αναφορών (status reports) από τους χρήστες
- Την ενημέρωση των χρηστών για νέα assignments είτε με e-mail είτε απευθείας στον χώρο εργασίας τους.
- Την αυτοματοποίηση ανάθεσης ορισμένων εργασιών (π.χ. όταν ένας συντάκτης αναφέρει ότι έχει ολοκληρώσει την συγγραφή του κειμένου αυτό να προωθείται αυτόματα στον διορθωτή χωρίς να χρειάζεται άλλη ανθρώπινη επέμβαση).
- Το σύστημα διαχείρισης ροών εργασίας σε συνεργασία με το σύστημα διαχείρισης θα πρέπει να επιτρέπει την παρακολούθηση της συντακτικής ιστορίας ενός αντικειμένου καθώς και την δυνατότητα ανίρεσης ορισμένων αλλαγών από εξουσιοδοτημένους ρόλους. Αυτή η δυνατότητα λέγεται Version Control.
- Τέλος, και ίσως το πιο σημαντικό, θα πρέπει το σύστημα να είναι ευέλικτο και εύκολα παραμετροποιήσιμο. Στα πιο προχωρημένα συστήματα δίνονται γραφικά εργαλεία σχεδίασης των κύκλων εργασίας (και των φορμών που συσχετίζονται με αυτά) και το σύστημα δημιουργεί αυτόματα τα προγράμματα που χρειάζονται.

Σύστημα έκδοσης (publication system)

Το σύστημα έκδοσης (publication system) είναι τα εργαλεία και προγράμματα τα οποία λαμβάνουν τις πληροφορίες από την βάση δεδομένων, μορφοποιούν το περιεχόμενο και το εμφανίζουν στο Web αλλά και σε άλλα μέσα. Για να μπορεί να αντεπεξέλθει σε μεγάλο εύρος απαιτήσεων, το σύστημα έκδοσης θα πρέπει να υποστηρίζει:

- Τον διαχωρισμό παρουσίασης και περιεχομένου. Αυτό επιτυγχάνεται μέσω συστημάτων templates και δίνει την δυνατότητα της έκδοσης του ίδιου

περιεχομένου σε διαφορετικές ιστοσελίδες, σε διαφορετικά site ή ακόμα και σε διαφορετικά μέσα. Κλασικό παράδειγμα είναι η δυνατότητα που δίνουν πολλά site στον επισκέπτη να εμφανίσει την σελίδα σε printer friendly format. Αυτό είναι παράδειγμα εφαρμογής συστήματος templates όπου το ίδιο περιεχόμενο εμφανίζεται με δύο διαφορετικές μορφές : μία για απεικόνιση στον browser και μία για εκτύπωση. Με αυτό τον τρόπο υπάρχει η δυνατότητα εμφάνισης του περιεχομένου σε άλλες μορφές όπως WML για εμφάνιση σε κινητά WAP, απλό κείμενο για αποστολή με e-mail, κάποιο XML format για αυτόματη ενημέρωση άλλων συστημάτων κ.α.

- Προεπισκόπηση (preview) των σελίδων, ακόμα και ολόκληρων ενοτήτων του site προτού αυτές εμφανιστούν στο web.
- Για πιο περίπλοκες ανάγκες θα πρέπει να υπάρχει πρόσβαση σε μια πλήρη γλώσσα προγραμματισμού. Η γλώσσα προγραμματισμού θα πρέπει να είναι μια από τις ευρέως γνωστές γιατί αλλιώς θα είναι δύσκολο να βρεθούν μηχανικοί που την γνωρίζουν. Το σύστημα θα πρέπει μέσω καλά τεκμηριωμένων APIs (Application Programming Interfaces) να επιτρέπει προγραμματιστική πρόσβαση στην βάση δεδομένων και στα αντικείμενα ελέγχου του CMS. Οι συνηθέστερες επιλογές είναι η Java, JSP, ASP, PHP, Perl κλπ.
- Δυναμική επίλυση αλληλεξαρτήσεων (dependency resolution). Όταν το περιεχόμενο προστίθεται στην βάση δεδομένων δεν είναι δυνατόν να γνωρίζει ο συντάκτης πως και σε ποιες σελίδες θα χρησιμοποιηθεί, είτε άμεσα είτε έμμεσα μέσω ενός link. Το σύστημα έκδοσης θα πρέπει να ελέγχει και να δημιουργεί αυτόματα τα κατάλληλα link όταν δημιουργείται η σελίδα. Επίσης θα πρέπει να εμφανίζει μηνύματα λάθους αν επιχειρείται να δημιουργηθεί σύνδεσμος με περιεχόμενο που δεν έχει εκδοθεί ακόμα.

Άλλες λειτουργίες και δυνατότητες

Επιπρόσθετα των παραπάνω ένα CMS μπορεί να υποστηρίζει και επιπρόσθετες λειτουργίες οι οποίες μπορεί από ορισμένους site να κρίνονται απαραίτητες ενώ σε άλλα να είναι αδιάφορες. Τέτοιες είναι :

- i. Personalization
- ii. Διαχείριση Metadata
- iii. Επεκτασιμότητα (Scalability)

Personalization

Ο όρος personalization είναι πολύ γενικός και υπάρχουν διάφορα επίπεδα. Γενικά σημαίνει ότι το site αναγνωρίζει ποιος είναι ο επισκέπτης και του προσαρμόζει, βάσει κανόνων που ορίζει ο διαχειριστής του site και ο επισκέπτης, το τι και πως θα το βλέπει. Υπάρχουν πολλοί τρόποι υλοποίησης του personalization. Μια ενδεικτική λίστα είναι:

- **Nominal:** Το site γνωρίζει το όνομα του χρήστη καθώς και λίγα ακόμα δεδομένα, π.χ. την τελευταία φορά που συνδέθηκε και την τελευταία σελίδα που είδε.
- **Group/demographics:** Το site ζητάει από τον χρήστη να απαντήσεις σε ορισμένες ερωτήσεις (π.χ. ηλικία, φύλο, επάγγελμα, εισόδημα, ενδιαφέροντα κλπ) και βάσει αυτών των απαντήσεων επιλέγει το περιεχόμενο αλλά ενδεχομένως και ποιες διαφημίσεις θα ενδιαφέρουν τον χρήστη.
- **Συνδρομητική:** Ο χρήστης επιλέγει την εγγραφή του σε συνδρομητικές mailing lists/newsletters και λαμβάνει περιοδικά, μέσω e-mail το περιεχόμενο του site.

my.site personalization. Το site δίνει την δυνατότητα επιλογής κατηγοριών περιεχομένου που ο χρήστης πιστεύει ότι τον ενδιαφέρουν. Π.χ. Το ειδησεογραφικό my.yahoo που επιτρέπει σε έναν χρήστη να διαλέξει ποιες κατηγορίες ειδήσεων θα βλέπει στην σελίδα του.

- **Full content personalization:** Το site λειτουργεί σαν knowledge base για τον κάθε χρήστη. Χρησιμοποιώντας δεδομένα που εισάγονται από τον χρήστη, η knowledge base φιλτράρετε και μόνο οι περιοχές ενδιαφέροντος εμφανίζονται.

Διαχείριση Metadata

Κάθε αξιόλογο CMS θα πρέπει να επιτρέπει και να διευκολύνει την διαχείριση των metadata. Η έννοια metadata σημαίνει «πληροφορίες για την πληροφορία». Για παράδειγμα, τα metadata για ένα άρθρο είναι η ημερομηνία έκδοσης, ο συγγραφέας, η ενότητα του site στην οποία ανήκει, λέξεις-κλειδιά, το κοινό που απευθύνεται κ.α.

Υπάρχουν πολλές ανάγκες διαχείρισης και συντήρησης τέτοιων δεδομένων, η σημαντικότερη είναι ότι διευκολύνουν την αναζήτηση. Τα σύγχρονα CMS παρέχουν τέτοιες δυνατότητες είτε από τους συντάκτες είτε από τους διαχειριστές είτε από ειδικούς χρήστες. Ορισμένα πιο προχωρημένα συστήματα επιτρέπουν και την αυτόματη ή ημιαυτόματη δημιουργία metadata. Χρησιμοποιούνται διάφορες τεχνικές, απλές (π.χ. το CMS εισάγει σαν ημέρα συγγραφής την ημέρα του συστήματος) αλλά και πιο πολύπλοκες (π.χ. από την ανάλυση του περιεχομένου είναι δυνατόν να προκύψουν πληροφορίες όπως ο σκοπός, outline, κοινό στόχος κλπ).

Επεκτασιμότητα (Scalability)

Με τον όρο επεκτασιμότητα εννοούμε την δυνατότητα ενός συστήματος να μπορεί να αντεπεξέλθει σε μεγάλη αύξηση της ζήτησης χωρίς μεγάλες αλλαγές και χωρίς μεγάλες επενδύσεις σε χρήμα, χρόνο και ανθρώπινο δυναμικό. Σχεδόν όλα τα site ξεκινούν με μικρή κίνηση, ορισμένα όμως λόγω της ποιότητας της πετυχαίνουν να έχουν χιλιάδες, ακόμα και εκατομμύρια επισκέπτες.

Η εξυπηρέτηση τόσο μεγάλου όγκου κίνησης δεν είναι κάτι εύκολο. Και βέβαια το πιο άσχημο για ένα site είναι να φτάσει σε ένα πολύ καλό επίπεδο αλλά πλέον το σύστημα που αρχικά διάλεξε να μην το εξυπηρετεί. Θα πρέπει να αλλάξει σύστημα, αλλά η εμπειρία έχει δείξει ότι τέτοιες αλλαγές συνήθως είναι ιδιαίτερα επίπονες, χρονοβόρες και πολυέξοδες.

Η επεκτασιμότητα γενικά δεν είναι απλή υπόθεση. Δεν υπάρχει ένα συγκεκριμένο εξάρτημα που είναι το κρίσιμο σημείο. Σε άλλες περιπτώσεις π.χ. μπορεί να είναι ο database server και σε άλλες ο application server. Γενικά, η επεκτασιμότητα είναι κυρίως θέμα αρχιτεκτονικής.

Λοιπές Τεχνολογίες

PHP

Η PHP είναι μία γλώσσα server side scripting language, σχεδιασμένη αποκλειστικά για ανάπτυξη διαδικτυακών εφαρμογών. Ωστόσο τα τελευταία χρόνια, ιδιαίτερα από την έκδοση 5.0 και μετά με τη προσθήκη κλάσεων και namespaces, προσπαθούν να κάνουν τη γλώσσα να έρθει πιο κοντά σε γλώσσες όπως είναι η Java, η C#, ωστόσο υστερεί σε πολλά ακόμη. Μέσα σε μια HTML σελίδα μπορούμε να ενσωματώσουμε PHP κώδικα, που θα εκτελείται κάθε φορά που θα επισκεπτόμαστε τη σελίδα. Ο PHP κώδικας εκτελείται στο

server και μεταφράζεται σε HTML κώδικα, τον οποίο εν τέλει θα δει ο χρήστης στον browser του.

Η ιστορία της PHP ξεκινά από το 1995, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε, χρησιμοποιώντας τη Perl, ένα απλό script με όνομα `php.cgi`, για προσωπική χρήση. Το script αυτό είχε ως σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το βιογραφικό του σημείωμα online. Αργότερα το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter.

Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στη C και περισσότερα από 50.000 websites τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP.

Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Τα περισσότερα websites επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5. Τον Απρίλιο του 2007 ο αριθμός των websites που χρησιμοποιούσαν την `php` έφτανε και 20.917.850 domains παγκοσμίως.

Η PHP είναι προϊόν ανοιχτού κώδικα (Open Source), που σημαίνει ότι οποιοσδήποτε έχει πρόσβαση στον source code και μπορεί να τον χρησιμοποιήσει, να τον αλλάξει και να τον διανέμει χωρίς χρέωση. Η PHP αρχικά ήταν το ακρωνύμιο του Personal Home Page, αλλά άλλαξε σύμφωνα με τη σύμβαση GNU, και τώρα είναι ακρωνύμιο του PHP Hypertext Preprocessor. Η τρέχουσα βασική έκδοση της PHP είναι η 5.2.9. Αυτή η έκδοση έχει υποστεί πλήρη ανανέωση στη Zend μηχανή της και μερικές μεγάλες βελτιώσεις στη γλώσσα.

Η ενσωμάτωση `php` κώδικα σε ένα `html` αρχείο επέκτασης δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση `.php`, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται PHP κώδικα. Ο Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με Linux αλλά και Windows, υποστηρίζει εξ ορισμού επεξεργασία κώδικα PHP.

Για περισσότερες πληροφορίες <http://www.php.net/>.

MySQL

Η MySQL είναι ένα πολύ δυνατό, γρήγορο RDBMS. Μία mysql βάση σας επιτρέπει να αποθηκεύετε, να αναζητάτε, να ταξινομείτε και να ανακαλείτε τα δεδομένα σας αποτελεσματικά. Ο MySQL server ελέγχει την πρόσβαση στα δεδομένα σας για να διασφαλίσει ότι πολλοί χρήστες θα μπορούν να δουλεύουν ταυτόχρονα, για να παρέχει γρήγορη πρόσβαση και για να διασφαλίζει ότι μόνο οι πιστοποιημένοι χρήστες θα έχουν πρόσβαση. Συνεπώς η MySQL είναι ένας multi thread server πολλαπλών χρηστών. Χρησιμοποιεί την SQL (Structured Query Language), όπως και τα περισσότερα συστήματα βάσεων δεδομένων.

Η MySQL είναι διαθέσιμη από το 1996 αλλά η ιστορία της ξεκινά από 1979. Είναι παγκοσμίως το πιο δημοφιλές open source RDBMS. Η MySQL είναι τώρα διαθέσιμη με άδειες δύο ειδών. Μπορείτε να τη χρησιμοποιήσετε δωρεάν με άδεια ανοιχτού κώδικα (open source license), εφόσον συμφωνήσετε με τους όρους αυτής της άδειας. Αν

Θέλετε να διανείμετε μια εφαρμογή που δεν είναι ανοιχτού κώδικα που να περιλαμβάνει την Mysql, μπορείτε να αγοράσετε μια εμπορική άδεια αντίστοιχα.

Για περισσότερες πληροφορίες <http://www.mysql.com/>

Apache Server

Ο Apache HTTP γνωστός και απλά ως Apache είναι ένας web server. Όποτε επισκέπτεστε έναν website ο browser σας επικοινωνεί με έναν HTTP server. Ο Apache είναι ένας από τους δημοφιλέστερους, εν μέρει γιατί λειτουργεί πάνω σε διαφορετικά λειτουργικά συστήματα χωρίς κανένα πρόβλημα. Διατηρείται τώρα από μια open source κοινότητα με επιτήρηση από το Apache Software Foundation.

Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, φτιαγμένη από τον Robert McCool, κυκλοφόρησε το 1993. Θεωρείται ότι έπαιξε σημαντικό ρόλο στην αρχική επέκταση του ιστού. Ήταν η πρώτη βιώσιμη εναλλακτική επιλογή που παρουσιάστηκε απέναντι στον HTTP server της Netscape και από τότε έχει εξελιχθεί τόσο πολύ έτσι ώστε να ανταγωνίζεται άλλους unix servers σε λειτουργικότητα και απόδοση.

Από το 1996 ήταν από τους πιο δημοφιλείς όμως από τον Μάρτιο του 2006 έχει μειωθεί το ποσοστό της εγκατάστασής του κυρίως από τον Microsoft Internet Information Services (IIS) και τη πλατφόρμα .NET . Τον Οκτώβριο του 2007 το μερίδιο του ήταν 47.73% από όλους τα websites.

Ο Apache είναι open source και αυτός. Εκτός του κόστους, είναι ιδιαίτερα διαμορφώσιμος και έχει μία τεράστια γκάμα προηγμένων χαρακτηριστικών, όπου και αυτός είναι και ο κύριος λόγος για τον οποίο οι περισσότεροι web hosts τον χρησιμοποιούν.

Για την υλοποίηση της πτυχιακής μας άσκησης χρησιμοποιήσαμε μία integrated πλατφόρμα, την XAMMP (έκδοση 1.70) for Windows η οποία μπορεί να βρεθεί στην διεύθυνση <http://www.apachefriends.org/en/xampp-windows.html>. Τα χαρακτηριστικά της πλατφόρμας είναι τα εξής:

- Apache HTTPD 2.2.11 + Openssl 0.9.8i
- MySQL 5.1.30
- PHP 5.2.8
- phpMyAdmin 3.1.1
- FileZilla FTP Server 0.9.29
- Mercury Mail Transport System 4.52

Για περισσότερες πληροφορίες <http://www.apachefriends.org/en/xampp.html>

Aptana IDE

Η εταιρεία Aptana, βρίσκεται στο San Mateo, της California. Ιδρύθηκε από τον Paul Colton το 2005, και επικεντρώνεται στη ανάπτυξη εργαλείων και προϊόντων για προγραμματιστές που χρησιμοποιούν scripting γλώσσες (JavaScript, Ruby, PHP και Python) για ανάπτυξη διαδικτυακών εφαρμογών. Τα κύρια προϊόντα της εταιρείας Aptana είναι το Aptana Studio, το Aptana Cloud και το Aptana Jaxer.

Το Aptana Studio είναι ένα open source IDE το οποίο βασίζεται στο Eclipse και είναι κατάλληλο για ανάπτυξη rhp εφαρμογών. Μπορεί να εγκατασταθεί σε Windows, Linux αλλά και σε Mac. Περιλαμβάνει υποστήριξη για Javascript, XHTML, DOM, CSS, όπως επίσης έχει intellisense και auto complete. Επιπλέον έχει ενσωματωμένα τα εξής χαρακτηριστικά: έναν Javascript Server, τον Jaxer, Ajax βιβλιοθήκες, όπως επίσης και βιβλιοθήκες για

RadRails, Python, Adobe AIR, iPhone. Τέλος η λειτουργικότητα του Aptana, δηλαδή όσα προανέφερα υπάρχει και ως stand alone plug-in για το Eclipse.

Για περισσότερες πληροφορίες <http://www.aptana.com>

Μέρος 2^ο – Σχεδίαση και ανάπτυξη

- Κεφ. 4^ο – Ανάλυση του component
- Κεφ. 5^ο – Υλοποίηση του component
- Κεφ. 6^ο – Έλεγχος – Εικονικό Σενάριο
- Κεφ. 7^ο – Packaging και Distribution
- Κεφ. 8^ο – Συμπεράσματα

Κεφάλαιο 4^ο

Ανάλυση του component

Εισαγωγή

Οι βασικοί στόχοι που οριοθετήθηκαν συνοψίζονται ως εξής: Πρώτος στόχος, είναι η δυνατότητα του καθηγητή να δημιουργεί ερωτηματολόγια μέσα από φιλικό και αυτοματοποιημένο περιβάλλον, που θα του προσφέρει όλη την απαραίτητη λειτουργικότητα. Επίσης μετά τη δημιουργία του ερωτηματολογίου, το ερωτηματολόγιο, με κάποιον τρόπο θα έπρεπε να ταξινομηθεί βάση σχολής και τμήματος, όπως επίσης να μπορεί να ανασυρθεί και να συμπληρωθεί.

Για αυτό το λόγο προτιμήθηκαν δύο διαδικτυακές πλατφόρμες, το LimeSurvey που θεωρείται από τα καλύτερα Open Source Survey Tools, αλλά και το Joomla, το καλύτερο Open Source CMS που κυριαρχεί σήμερα. Να τονίσουμε πως το LimeSurvey θεωρείται CMS, αλλά εξειδικευμένου περιεχομένου, όπου και η διεπαφή του είναι αναλόγως εξειδικευμένη.

Έτσι, όσον αφορά την δημιουργία και αποθήκευση του ερωτηματολογίου θα χρησιμοποιήσουμε το LimeSurvey. Όσον αφορά τη διαχείριση των ερωτηματολογίων αυτό θα γίνει με το Joomla και συγκεκριμένα με ένα custom component που αναπτύχθηκε στα πλαίσια της πτυχιακής. Το component δεν θα διαχειρίζεται τα ίδια τα ερωτηματολόγια, αλλά το url του ερωτηματολογίου που παράγει το LimeSurvey. Λεπτομερειακά θα το δούμε παρακάτω.

Λίγα λόγια για το Joomla

Το Joomla έχει πολλές δυνατότητες και ταυτόχρονα είναι εξαιρετικά ευέλικτο και φιλικό, είναι εφαρμογή με την οποία μπορεί κάποιος να δημοσιεύσει στο διαδίκτυο μια προσωπική ιστοσελίδα, αλλά και έναν ολόκληρο εταιρικό δικτυακό site. Είναι προσαρμόσιμο σε διάφορα περιβάλλοντα επιχειρηματικής κλίμακας όπως τα intranets μεγάλων επιχειρήσεων ή οργανισμών. Οι δυνατότητες επέκτασής του είναι πρακτικά απεριόριστες. Εγκαθίσταται σε τον web server και ο χρήστες έχουν πρόσβαση στο περιβάλλον διαχείρισης μέσω ενός browser.

Από τη στιγμή που κάποιος είναι administrator, μπορεί να προσθέσει οποιοδήποτε περιεχόμενο, δηλαδή κείμενο, εικόνα, βίντεο, flash και έτσι να δημιουργήσει τις ιστοσελίδες του. Μπορεί να χρησιμοποιήσει το Joomla μόνος του και υπάρχουν διάφοροι τρόποι να ξεκινήσει. Εάν κάποιος γνωρίζει τη χρήση ενός επεξεργαστή κειμένου, βρίσκεστε σε καλό δρόμο ώστε να μπορέσει να προχωρήσει. Σε γενικές γραμμές, εξαρτάται από το τι θέλει να κάνει με το Joomla.

Οι δυνατότητες χρήσης του Joomla περιορίζονται μόνον από τη δημιουργικότητα του χρήστη. Κάποιοι χρησιμοποιούν το Joomla για διασκέδαση, για τη κατασκευή μιας προσωπικής ή οικογενειακής ιστοσελίδας. Ενσωματώνοντας διάφορα, δωρεάν διαθέσιμα, πρόσθετα, δηλαδή components, modules και plugins για τα οποία θα μιλήσουμε παρακάτω αναλυτικά, διευρύνονται οι δυνατότητες και οι λειτουργίες του, κάνοντάς το μια εφαρμογή

αξιόπιστη για την ανάπτυξη σοβαρών εταιρικών sites. Επίσης χρησιμοποιεί μια ισχυρή templating engine που δίνει τη δυνατότητα να χρησιμοποιήσει ο καθένας το δικό του, εξατομικευμένο, template. Για το σκοπό αυτό μπορεί είτε να δημιουργήσει το δικό του template, είτε να το κατεβάσει.

Τέλος, το Joomla έχει πολλά χαρακτηριστικά γνωρίσματα που είναι δύσκολο να αναφερθούμε σε όλα. Επιγραμματικά, αναφέρουμε τις πιο δημοφιλείς δυνατότητες:

- Μπορεί κάποιος να προσθέσει περιεχόμενο στο site του, του από οποιονδήποτε υπολογιστή διαθέτει σύνδεση στο διαδίκτυο. Πληκτρολογεί το κείμενο, ανεβάζει τις φωτογραφίες και τα δημοσιεύει.
- Μπορεί κάποιος να χρησιμοποιήσει το Joomla για να συνεργαστεί με όσους μπορούν να συμβάλλουν στην επεξεργασία του περιεχομένου. Δίνει τη δυνατότητα σε συνεργάτες του να δημοσιεύσουν περιεχόμενο, στο πλαίσιο ασφαλώς των αρμοδιοτήτων τους. Είναι εύκολο. Μπορούν να το κάνουν από κάθε υπολογιστή με σύνδεση στο διαδίκτυο.
- Μπορεί να δημοσιεύσει απεριόριστες σελίδες, χωρίς να περιορίζεται από το Joomla. Χωρίς κανένα πρόβλημα, μπορεί να κάνει αναζητήσεις περιοχομένου και να τις αρχειοθετήσει.
- Η εφαρμογή υποστηρίζει τα διαφημιστικά banners, οπότε δίνει την δυνατότητα να προωθήσει κάποιος τα δικά του προϊόντα και υπηρεσίες, ή να τα χρησιμοποιήσει ως διαφημιστικό μέσο για τρίτους.
- Επίσης μπορούν να προστεθούν forum, photo galleries, βιβλιοθήκες αρχείων, βιβλία επισκεπτών και φόρμες επικοινωνίας κτλ.

Component διαχείρισης των ερωτηματολογίων

Μέρος της επεκτασιμότητας του Joomla είναι και τα components. Τα components είναι ουσιαστικά εφαρμογές, όπου στο backend υπάρχει όλη η διαχείριση, ενώ στο frontend εμφανίζονται τα αποτελέσματα. Στο backend αλλά και στο frontend τα components «πιάνουν» το κυρίως μέρος της σελίδας. Το Joomla είναι έτσι φτιαγμένο, ώστε μία σελίδα, να μπορεί να δείχνει μόνο ένα component. Κατά συνέπεια, η βασική λειτουργικότητα διαχείρισης του περιεχομένου είναι και αυτό ένα component.

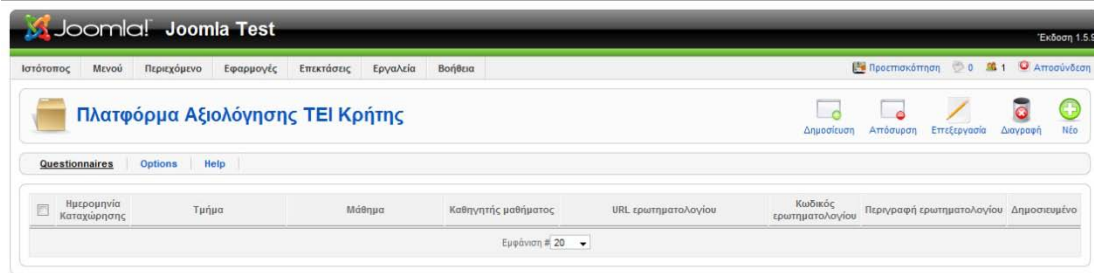
Έτσι και εμείς για τη πτυχιακή δημιουργήσαμε ένα component όπου στο backend υπάρχει η διεπαφή χωρίζεται σε 3 μέρη:

1. Διεπαφή διαχείρισης των urls των ερωτηματολογίων.
2. Διεπαφή των ρυθμίσεων του component
3. Διεπαφή της βοήθειας.

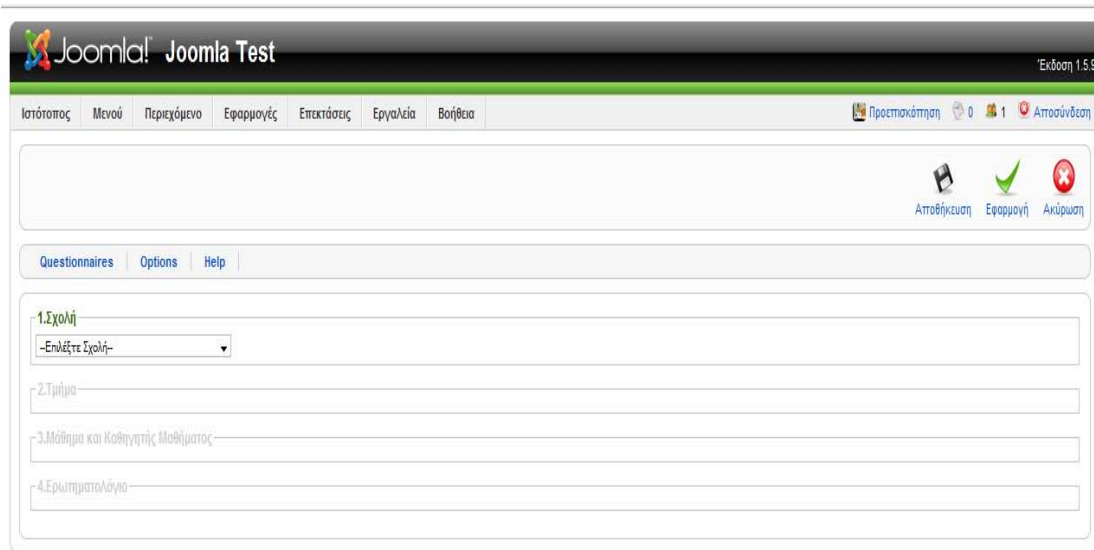
Στο frontend της εφαρμογής τα urls είναι ταξινομημένα, βάση σχολών και τμημάτων της εκάστοτε σχολής. Επίσης υπάρχει ειδική επιλογή που δίνει στους καθηγητές τη φόρμα να υποβάλουν το url του ερωτηματολογίου και κάποιες πληροφορίες επιπλέον.

Διεπαφή Διαχείρισης των Ερωτηματολογίων

Όπως βλέπουμε και στη φωτογραφία, η βασική διεπαφή είναι άδεια μιας και δεν έχει κατατεθεί κανένα ερωτηματολόγιο. Επίσης βλέπουμε πως εκτός από το url του ερωτηματολογίου στη βάση, υπάρχουν και άλλες πληροφορίες τις οποίες ανασύρουμε. Αυτές οι πληροφορίες είναι η «Ημερομηνία Καταχώρησης» βάση της οποίας γίνεται και η ταξινόμηση, το «Τμήμα», το «Μάθημα», το «Url» εννοείται, ο «Κωδικός» του ερωτηματολογίου, ένα πεδίο «Περιγραφής», και τέλος αν το ερωτηματολόγιο είναι δημοσιευμένο, δηλαδή διαθέσιμο στο τελικό χρήστη.



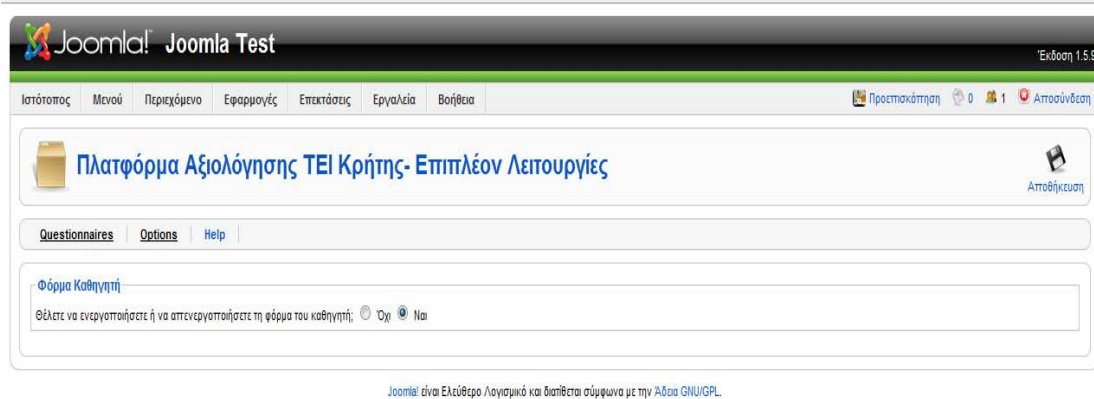
Για να εισάγουμε ένα νέα καταχώρηση πατάμε το κουμπί «Νέο» που βρίσκεται άκρα δεξιά στην οθόνη μας. Με το που το πατήσουμε εμφανίζεται μία νέα διεπαφή, σε μορφή φόρμας συμπλήρωσης από το χρήστη. Η εικόνα παρακάτω δείχνει την διεπαφή



Όπως βλέπουμε και από την εικόνα, τα δεδομένα τα οποία μπορούμε να εισάγουμε είναι τα όσα περιγράψαμε παραπάνω τα οποία εμφανίζονται με το url του ερωτηματολογίου. Αναλυτικά πως εισάγουμε ένα ερωτηματολόγιο θα το δούμε παρακάτω.

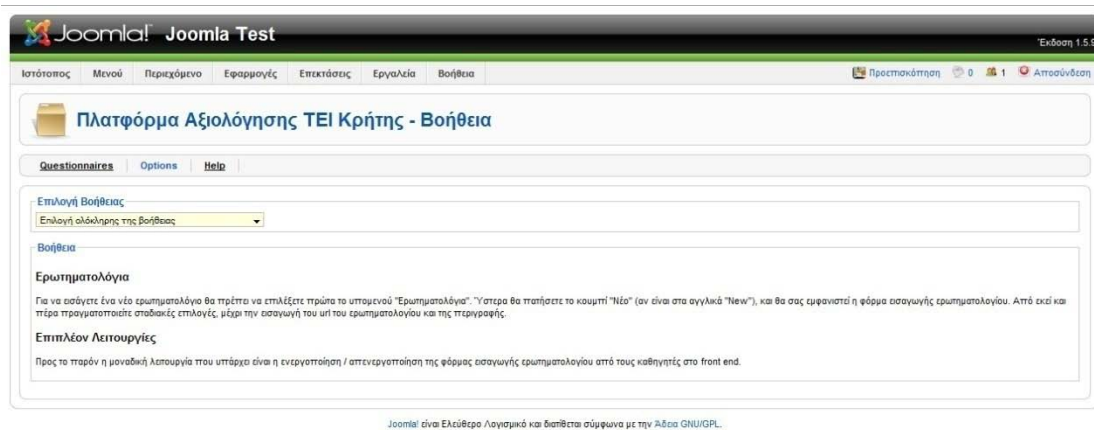
Διεπαφή των ρυθμίσεων του component.

Πατώντας την επιλογή «Options» εμφανίζεται η διεπαφή των ρυθμίσεων. Προς το παρόν η μόνη ρύθμιση που έχουμε είναι η ενεργοποίηση / απενεργοποίηση της φόρμας του καθηγητή στο frontend. Η φόρμα λειτουργεί το ίδιο όπως και η φόρμα που χρησιμοποιεί ο administrator για την εισαγωγή ερωτηματολογίων.



Διεπαφή της βοήθειας

Τέλος πατώντας το help, έχουμε τη διεπαφή της βοήθειας. Η διεπαφή είναι απλή και εύχρηστη με ένα combobox που σου επιτρέπει να επιλέξεις της βοήθεια της επιλογής σου. Οι επιλογές για τη βοήθεια είναι οι εξής: Ολόκληρη η βοήθεια, η βοήθεια για τα ερωτηματολόγια και η βοήθεια για τις ρυθμίσεις.

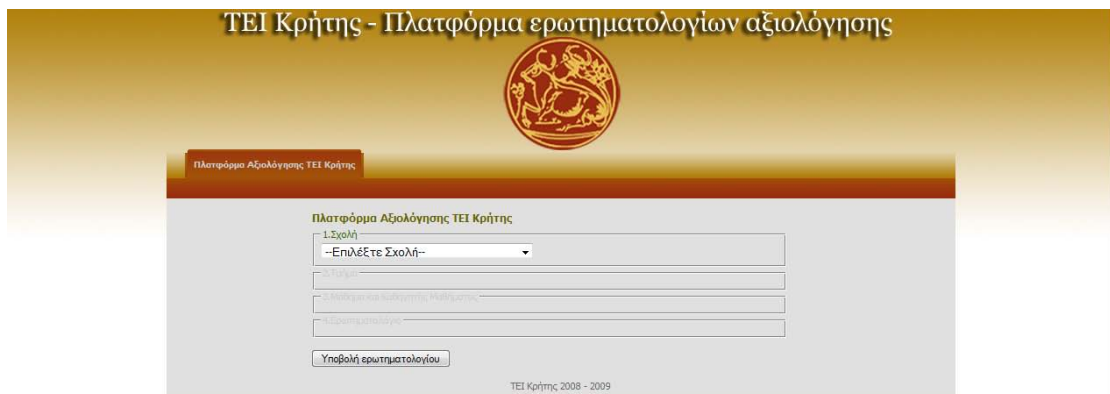


Front end διεπαφή

Η front end διεπαφή είναι ως προς τη χρήση. Είναι ταξινομημένη σε tabs, όπου κάθε tab είναι κάθε σχολή, και όταν ο φοιτητής επιλέξει τη σχολή θα του εμφανιστούν τα τμήματα που ανήκουν στη σχολή. Μετά θα μπορεί να επιλέγει το τμήμα και θα του εμφανίζονται όλα τα ερωτηματολόγια, επιλέγοντας εκείνο που θέλει. Αναλυτικά θα το δούμε παρακάτω στην εκτέλεση του εικονικού σεναρίου.



Όπως βλέπετε, κάτω από το πλαίσιο υπάρχει μία γραμμή που λέει «Αν είστε καθηγητής πατήστε **εδώ**». Πατώντας στο **εδώ** μεταφέρεται ο καθηγητής στη φόρμα εισαγωγής του ερωτηματολογίου. Η φόρμα λειτουργεί με τον ίδιο τρόπο όπως η φόρμα του administrator για την εισαγωγή των ερωτηματολογίων.



Κεφάλαιο 5^ο

Υλοποίηση Component

Εισαγωγή

Για να αναπτύξει κάποιος ένα component, και όχι μόνο, για το Joomla θα πρέπει να έχει γνώσεις του Joomla Framework. Το Joomla Framework ουσιαστικά είναι ένα σύνολο από κλάσεις, και includes που ικανοποιούν την λογική που έχουν θέσει οι προγραμματιστές του CMS. Το documentation του Framework, αλλά και το wiki δεν είναι ακόμη πολύ ώριμα μιας και έχουν ελλείψεις στο να σου εξηγήσουν, μεθόδους, ορίσματα, όπως επίσης υπάρχει και μεγάλη έλλειψη οργανωμένων παραδειγμάτων.

Ανάπτυξη του component

Δημιουργία των φακέλων στις κατάλληλες θέσεις

Ξεκινώντας την ανάπτυξη ενός component η διαδικασία είναι διαφορετική από αυτή που ξέρουμε από τις desktop εφαρμογές, και στη συγκεκριμένη περίπτωση πολλές διαδικασίες είναι πρότυπες και δε μπορείς να ξεφύγεις από αυτές. Έτσι ξεκινώντας να φτιάχνεις ένα component το πρώτο πράγμα που πρέπει να κάνεις είναι να δημιουργήσεις έναν φάκελο όπου θα έχει το πρόθεμα **com_**. Είναι απαραίτητο το πρόθεμα για να καταλάβει το Joomla πως πρόκειται για φάκελο που περιέχει component. Τον φάκελο τον αυτόν τον φτιάχνουμε δύο φορές. Ο ένας θα μπει στο φάκελο components για το front end, ενώ ο άλλος μέσα στο φάκελο components που βρίσκεται μέσα στο φάκελο administrator. Ο φάκελος ο δικός μας θα έχει το όνομα **com_surveys**.

Όταν το component φορτωθεί από το front end, το Joomla θα ψάξει για ένα php αρχείο με μοναδικό όνομα ίδιο με το όνομα του φακέλου. Έτσι στη περίπτωση μας έχουμε ένα φάκελο με το όνομα **com_surveys**, και μέσα του ένα αρχείο με το όνομα **surveys.php**. Όμοια αν τρέξουμε το component από το back end το Joomla θα κοιτάξει για αρχείο με πρόθεμα **admin**. ακολουθημένο από το όνομα του φακέλου, δηλαδή **admin.surveys.php**. Τα δύο αυτά αρχεία θα βρίσκονται στους φακέλους του component για το front end και το back end αντίστοιχα.

Όλα τα request στο front-end «περνάνε» από το index.php το οποίο βρίσκεται στο ριζικό κατάλογο. Για να φορτωθεί ένα component δίπλα στο index.php, «κολλάμε» ένα query string, όπου έχουμε ένα key-value pair, με key **option** και value **com_όνομα component**. Π.χ. στη περίπτωση μας, αν είμαστε στο front end και θέλουμε επισκεφτούμε το component στο address bar θα γράψουμε **http://localhost/joomla/index.php?option=com_surveys**. Εάν γράψουμε το προηγούμενο url στο browser μας, τότε ναι με το Joomla θα μας εκεί, αλλά θα συναντήσουμε ένα άδειο component. Αντίστοιχα το ίδιο συμβαίνει και στο backend. Εάν γράψουμε **http://localhost/joomla/administrator/index.php?option=com_surveys** θα δούμε το component αλλά και πάλι θα είναι άδειο. Σιγά σιγά θα «γεμίσουμε» το αρχείο με κώδικα,

όπως επίσης θα δημιουργήσουμε και άλλα αρχεία απαραίτητα για την ορθή του λειτουργία. Προς το παρόν ας δούμε τι πρέπει να κάνουμε για να καταχωρήσουμε το component στη βάση.

Εισαγωγή δεδομένων στη βάση

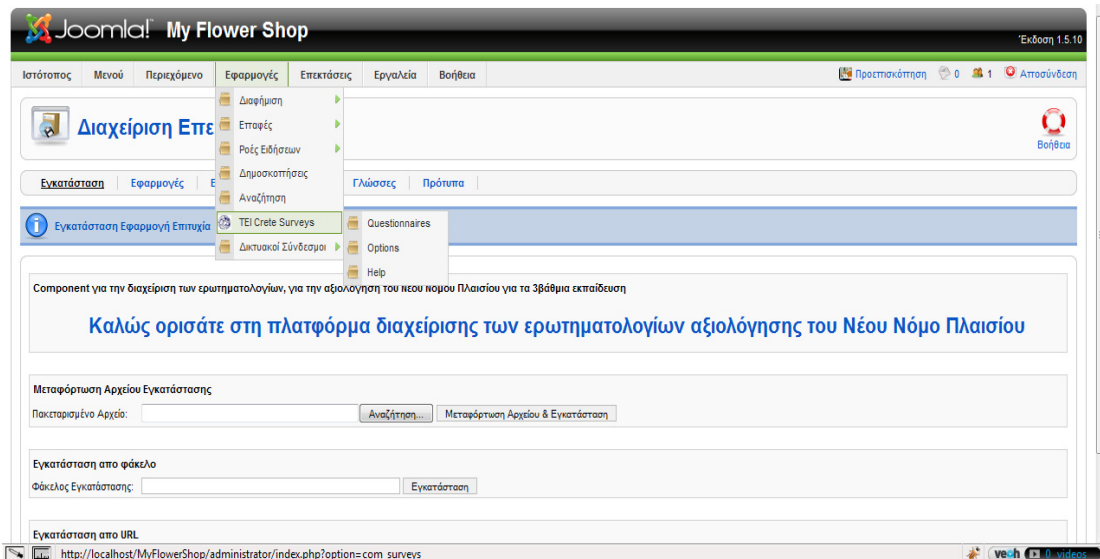
Για να είναι το component διαθέσιμο από το μενού του Joomla πρώτα πρέπει να το καταχωρήσουμε στη βάση με το χέρι. Σαφέστατα εννοείται πως όταν φτάσει στη τελική του μορφή θα υπάρχει sql ερώτημα που θα το καταχωρεί στη βάση και δε χρειαστεί να γίνεται «με το χέρι». Προς το παρόν θα το κάνουμε έτσι καθαρά για λόγους ανάπτυξης.

Ανοίγουμε το phradmin και πηγαίνουμε στον πίνακα components. Εκεί εισάγουμε τα παρακάτω στοιχεία, στα αντίστοιχα πεδία:

Πεδίο	Τύπος	Έλεγχος	Κενό	Τιμή
id	int(11)			
name	varchar(50)			TEI Crete Surveys
link	varchar(255)			option=com_surveys
menuid	int(11) unsigned			0
parent	int(11) unsigned			0
admin_menu_link	varchar(255)			option=com_surveys
admin_menu_alt	varchar(255)			TEI Crete Surveys
option	varchar(50)			com_surveys
ordering	int(11)			0
admin_menu_img	varchar(255)			components/com_surveys/images/tei_icon.jpg
iscore	tinyint(4)			0
params	text			
enabled	tinyint(4)			1

- Το πεδίο **id** το αφήνουμε έτσι όπως είναι, κενό. Είναι auto increment και η mysqι θα βάλει από μόνη της τη σωστή τιμή.
- Στο **name** θα βάλουμε το όνομα του component. Στη περίπτωση μας **TEI Crete Surveys**. Να τονίσω εδώ πως αν εδώ χρησιμοποιήσουμε ελληνικούς χαρακτήρες θα τους πάρει κανονικά, αλλά όταν έρθει ώρα να το εγκαταστήσουμε κανονικά από την κονσόλα εγκατάστασης του Joomla τότε θα πρέπει να χρησιμοποιήσουμε μόνο χαρακτήρες ascii μιας και ο parser του Joomla δεν «βλέπει» ελληνικά, όσον αφορά τα ονόματα των μενού. Για άλλα στοιχεία πχ το όνομα του δημιουργού εκεί βλέπει τα ελληνικά κανονικά
- Στο **link** γράφουμε το key value pair για το front end
- Στο **admin_menu_link** γράφουμε πάλι το ίδιο, αλλά αυτή τη φορά για το back end.
- Στο **admin_menu_alt** μπαίνει το κείμενο που θα εμφανίζεται στο tooltip
- Στο **option** μπαίνει η τιμή com_surveys. Αυτό γίνεται για το hard coded link.
- Στο **admin_menu_img** μπαίνει η εικόνα που θα είναι δίπλα από το τίτλο του component στο μενού.
- Τα υπόλοιπα πεδία τα αφήνουμε κενά.

Έτσι μετά την εισαγωγή των δεδομένων στη βάση, αν πάμε στο Joomla και επιλέξουμε την επιλογή «Εφαρμογές», θα δούμε πλέον και το component μας.



Επίσης το component θα έχει τους δικούς του πίνακες στη βάση δεδομένων του Joomla. Θα πρέπει, και πάλι για λόγους ανάπτυξης και μόνο, να φτιάξουμε αυτούς τους πίνακες με το χέρι, αλλά θα πρέπει να γράψουμε και rhr κώδικα, σε συγκεκριμένη πρότυπη μορφή, όπου αυτά τα αρχεία θα χρησιμοποιηθούν για την εισαγωγή δεδομένων στους πίνακες. Αρχικά θα φτιάξουμε τους πίνακες που θα χρησιμοποιεί το component. Οι πίνακες έχουν ως εξής:

1. Ο πίνακας **survey_categories** περιέχει μία δενδρική δομή όπου περιέχει, το ίδρυμα του ΤΕΙ, τις σχολές, και τα τμήματα.

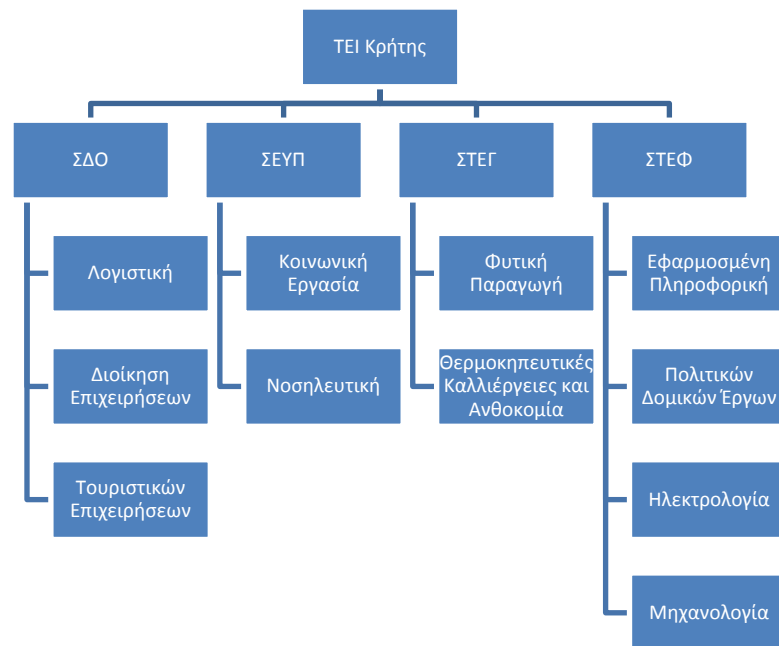
	category_id	category_name	father_category
<input type="checkbox"/>	1	ΑΤΕΙ ΚΡΗΤΗΣ	0
<input type="checkbox"/>	2	Σχολή Διοίκησης και Οικονομίας	1
<input type="checkbox"/>	3	Σχολή Επαγγελματών Υγείας και Πρόνοιας	1
<input type="checkbox"/>	4	Σχολή Τεχνολογίας Γεωπονίας	1
<input type="checkbox"/>	5	Σχολή Τεχνολογικών Εφαρμογών	1
<input type="checkbox"/>	6	Τμήμα Διοίκησης Επιχειρήσεων	2
<input type="checkbox"/>	7	Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων	5
<input type="checkbox"/>	8	Τμήμα Ηλεκτρολογίας	5
<input type="checkbox"/>	9	Τμήμα Θερμοκηπιακών Καλλιεργειών και Ανθοκομίας	4
<input type="checkbox"/>	10	Τμήμα Κοινωνικής Εργασίας	3
<input type="checkbox"/>	11	Τμήμα Λογιστικής	2
<input type="checkbox"/>	12	Τμήμα Μηχανολογίας	5
<input type="checkbox"/>	13	Τμήμα Νοσηλευτικής	3
<input type="checkbox"/>	14	Τμήμα Πολιτικών Δομικών Έργων	5
<input type="checkbox"/>	15	Τμήμα Τουριστικών Επιχειρήσεων	2
<input type="checkbox"/>	16	Τμήμα Φυτικής Παραγωγής	4

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση: Εγγραφές αρχίζοντας από την εγγραφή

σε οριζόντια μορφή με επανάληψη επικεφαλίδων ανά κελιά

Εάν μετατρέψουμε τις εγγραφές σε δέντρο θα πάρουμε το παρακάτω σχήμα.



2. Ο πίνακας **surveys_help** περιέχει το κείμενο της βοήθειας που αναφέραμε ψηλότερα στην περιγραφή του component
3. Ο πίνακας **surveys_lessons** εμπεριέχει όλα τα μαθήματα, από όλα τα τμήματα που βρίσκονται στο Ηράκλειο. Εκτός από το id και το όνομα του μαθήματος, υπάρχουν και άλλα δύο πεδία. Το ένα είναι το εξάμηνο που ανήκουν, και το άλλο το τμήμα που ανήκουν.
4. Ο πίνακας **surveys_more** περιέχει τις ρυθμίσεις που προς το παρόν είναι μία και μόνο, η ενεργοποίηση – απενεργοποίηση της front end φόρμας.
5. Τέλος είναι ο πίνακας **surveys_questionnaire** είναι ο πίνακας που θα αποθηκεύεται το url του ερωτηματολογίου και οι υπόλοιπες σχετικές πληροφορίες.

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
<input type="checkbox"/> questionnaire_id	int(11)			Όχι	Κανένα	auto_increment
<input type="checkbox"/> questionnaire_lesson	int(11)			Όχι	Κανένα	
<input type="checkbox"/> questionnaire_lesson_teacher	varchar(100)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> questionnaire_description	text	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> questionnaire_url	varchar(150)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> questionnaire_dept_that_belongs	int(11)			Όχι	Κανένα	
<input type="checkbox"/> questionnaire_password	varchar(20)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> questionnaire_timestamp	timestamp		on update CURRENT_TIMESTAMP	Όχι	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP
<input type="checkbox"/> published	tinyint(1)		UNSIGNED	Όχι	0	

Συγγραφή κώδικα για την επικοινωνία με τη βάση

Θα μπορούσαμε να γράψουμε επιμέρους συναρτήσεις για τη διαχείριση των sql ερωτημάτων. Ωστόσο, καλύτερο είναι να της αποφύγουμε. Ευτυχώς, το Joomla Framework έχει ήδη τη κλάση JTable, η οποία παρέχει λειτουργίες για τη δημιουργία, την ανάγνωση, την ενημέρωση και τη διαγραφή εγγραφών από έναν πίνακα στη βάση δεδομένων. Για να επωφεληθούμε από το JTable, πρέπει να γράψουμε κώδικα, όπου θα δημιουργήσουμε μία κλάση η οποία θα κληρονομεί από τη JTable. Μέσα στο administrator/components/com_reviews, δημιουργήστε ένα φάκελο με το όνομα tables. Σε αυτό το φάκελο, δημιουργούμε τα εξής τέσσερα αρχεία:

1. Το fad.php (= faculties and departments), το οποίο θα έχει τη κλάση για τον πίνακα survey_categories.
2. Το lessons.php, το οποίο θα έχει τη κλάση για το πίνακα survey_lessons
3. Το more.php, το οποίο θα έχει τη κλάση για το πίνακα survey_more.
4. Το questionnaire.php, το οποίο θα έχει τη κλάση για το πίνακα survey_questionnaire.

Ο πηγαίος κώδικας των αρχείων δίδεται παρακάτω.

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * fad.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class TableFaD extends JTable
{
    /**
     * Μεταβλητή category_id
     */
    var $category_id = null;

    /**
     * Μεταβλητή category_name
     */
    var $category_name = null;

    /**
     * Μεταβλητή father_category
     */
    var $father_category = null;

    function __construct(&$db)
    {
        parent::__construct('#__survey_categories','category_id', $db);
    }
}
?>
```

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * lessons.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class TableLessons extends JTable
{
    /**
     * Μεταβλητή lesson_id
     */

    var $lesson_id = null;

    /**
     * Μεταβλητή lesson_name
     */

    var $lesson_name = null;

    /**
     * Μεταβλητή lesson_dept
     */

    var $lesson_dept = null;

    /**
     * Μεταβλητή lesson_semester
     */

    var $lesson_semester = null;

    function __construct(&$db)
    {
        parent::__construct('#__survey_lessons','lesson_id', $db);
    }
}
?>
```

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * more.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class TableMore extends JTable
{
    /**
     * Μεταβλητή more_id
     */

    var $more_id = null;

    /**
     * Μεταβλητή IsTeacherFrontEndFormAvailable
     */

    var $IsTeacherFrontEndFormAvailable = null;

    function __construct(&$db)
    {
        parent::__construct('#__survey_more','more_id', $db);
    }
}
```

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * questionnaire.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class TableQuestionnaire extends JTable
{
    /**
     * Μεταβλητή questionnaire_id
     */

    var $questionnaire_id = null;

    /**
     * Μεταβλητή questionnaire_lesson
     */

    var $questionnaire_lesson = null;

    /**
     * Μεταβλητή questionnaire_lesson_teacher
     */

    var $questionnaire_lesson_teacher = null;

    /**
     * Μεταβλητή questionnaire_description
     */

    var $questionnaire_description = null;

    /**
     * Μεταβλητή questionnaire_url
     */

    var $questionnaire_url = null;

    /**
     * Μεταβλητή questionnaire_dept_that_belongs
     */

    var $questionnaire_dept_that_belongs = null;
}
```

```

/**
 * Μεταβλητή questionnaire_password
 */

var $questionnaire_password = null;

/**
 * Μεταβλητή $published
 */

var $published = null;

function __construct(&$db)
{
    parent::__construct('#__survey_questionnaire','questionnaire_id', $db);
}
?>
?>

```

Όταν κληρονομούμε τη κλάση JTable, προσθέτουμε τις μεταβλητές κλάσης με τα ονόματα των στηλών που υπάρχουν στους πίνακες, αλλιώς τα δεδομένα δε θα περνάνε στη βάση. Αφού το κάνουμε αυτό τις θέτουμε όλες null. Επίσης, κάνουμε override η συνάρτηση του constructor. Η δικιά μας __construct () θα λάβει σαν όρισμα ένα database type αντικείμενο και θα καλέσει τον βασικό constructor, όπου ο βασικό constructor παίρνει σαν ορίσματα το όνομα του πίνακα, το πρωτεύον κλειδί του, και το database type object. Όλες οι κλάσεις που φτιάξαμε κληρονομούν τις συναρτήσεις bind(), store(), load() και delete(). Οι τέσσερις αυτές συναρτήσεις μας επιτρέπουν να διαχειριζόμαστε τους πίνακες της βάσης χωρίς να χρειαστεί να γράψουμε ούτε ένα sql ερώτημα.

Δομές φακέλων

Ήρθε η ώρα να δούμε τη δομή που θα έχουν τα php αρχεία για το component, όπως επίσης τι κάνει το καθένα από αυτά. Όταν τελειώσουμε το component και πρέπει να εγκατασταθεί θα εγκατασταθούν και άλλα αρχεία, που προς το παρόν δε μας χρειάζονται.

Αγαπημένες συνδέσεις	Όνομα	Ημ/νία τροποποι...	Τύπος	Μέγεθος	Ετικέτες
Εγγραφα	images	20/4/2009 20:08	Φάκελος αρχείων		
Εικόνες	modules	20/4/2009 20:08	Φάκελος αρχείων		
Μουσική	scripts	20/4/2009 20:08	Φάκελος αρχείων		
Τροποποιήθηκαν π...	styles	20/4/2009 20:08	Φάκελος αρχείων		
Αναζητήσεις	tables	20/4/2009 20:08	Φάκελος αρχείων		
Καυόχρηστος	admin.surveys.html.php	20/4/2009 20:08	Αρχείο PHP	14 KB	
Projects	admin.surveys.php	20/4/2009 20:08	Αρχείο PHP	10 KB	
	toolbar.surveys.html.php	20/4/2009 20:08	Αρχείο PHP	2 KB	
	toolbar.surveys.php	20/4/2009 20:08	Αρχείο PHP	1 KB	

Μέσα στο φάκελο com_surveys έχουμε μερικούς φακέλους και αρχεία. Στο images έχουμε, όπως φαίνεται από το όνομα, εικόνες. Είναι η εικόνα που είναι στο μενού. Στο φάκελο

modules έχει αρχεία με κώδικα επαναχρησιμοποιήσιμο που χρησιμοποιούν κάποια από τα υπόλοιπα αρχεία. Προσοχή το όνομα modules δεν έχει καμία σχέση με το modules του Joomla. Η ονομασία του φακέλου είναι κλεμμένη από τα modules της VB, όπου ήταν αρχεία με επαναχρησιμοποιήσιμο κώδικα. Ο φάκελος scripts περιέχει javascript αρχεία, ενώ ο φάκελος styles περιέχει css αρχεία. Τέλος ο φάκελος tables περιέχει τις κλάσεις για τους πίνακες που περιγράψαμε αναλυτικά πιο πάνω. Τα υπόλοιπα τέσσερα αρχεία, δηλαδή τα admin.surveys.html.php, admin.surveys.php, toolbars.surveys.html.php και toolbars.surveys.php είναι τα τέσσερα πιο σημαντικά αρχεία του component, όσον αφορά το back end πάντα. Ωστόσο θα πρέπει πρώτα να εξηγήσουμε κάποιες συγκεκριμένες λειτουργικότητες του framework.

Σχεδίαση διεπαφών

Το Joomla, όταν σχεδιάζουμε διεπαφές, μας δίνει τη δυνατότητα να χωρίζουμε το business logic από το view. Από επιτυχάνετε με το να βάζουμε το business logic σε ένα αρχείο της μορφής admin.nameofcomponent.php, ενώ το view στο admin.nameofcomponent.html.php. Το ίδιο ισχύει και όταν φτιάχνουμε toolbars. Δεν υπάρχει καμία διαφορά όσον αφορά το διαχωρισμό του business logic από το view, απλά εδώ χρησιμοποιούμε «έτοιμο» κώδικα του framework για να φτιάξουμε το toolbar.

Θα συνεχίσουμε την γρήγορη δημιουργία του component, και αφού καταφέραμε να φτιάξουμε τη δομή των φακέλων αλλά και τις κλάσεις για τη χρήση δεδομένων από τους πίνακες, ας ξεκινήσουμε σιγά σιγά να φτιάχνουμε τις διεπαφές ξεκινώντας από το toolbar.

Για να δημιουργήσουμε ένα toolbar αρχικά μέσα στο φάκελο com_surveys που είναι στο back end φτιάχνουμε ένα νέο αρχείο με το όνομα toolbar.surveys.html.php. Αφού το ανοίξουμε γράφουμε μέσα τον εξής κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * toolbar.surveys.html.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class TOOLBAR_surveys
{
    function _NEW()
    {
        JToolBarHelper::save();
        JToolBarHelper::apply();
        JToolBarHelper::cancel();
    }

    function _DEFAULT()
    {
        JToolBarHelper::title(JText::_('Πλατφόρμα Αξιολόγησης ΤΕΙ Κρήτης'));

        JToolBarHelper::publishList();
        JToolBarHelper::unpublishList();
        JToolBarHelper::editList();
    }
}
```

```
        JToolBarHelper::deleteList();
        JToolBarHelper::addNew();
    }
}

class TOOLBAR_surveys_more
{
    function _DEFAULT()
    {
        JToolBarHelper::title(JText::_('Πλατφόρμα Αξιολόγησης ΤΕΙ Κρήτης-
Επιπλέον Λειτουργίες'));
        JToolBarHelper::save('saveMore'); //το saveMore είναι το task
    }
}

class TOOLBAR_surveys_help
{
    function _DEFAULT()
    {
        JToolBarHelper::title(JText::_('Πλατφόρμα Αξιολόγησης ΤΕΙ Κρήτης -
Βοήθεια'));
    }
}
?>
```

Μετά θα δημιουργήσουμε το toolbar-surveys.php αρχείο όπου μέσα θα βάλουμε τον εξής κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 * toolbar-surveys.php
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require(JApplicationHelper::getPath('toolbar_html'));

switch($task)
{
    //Ερωτηματολόγια
    case 'edit':
    case 'add':
        TOOLBAR_surveys::_NEW();
        break;

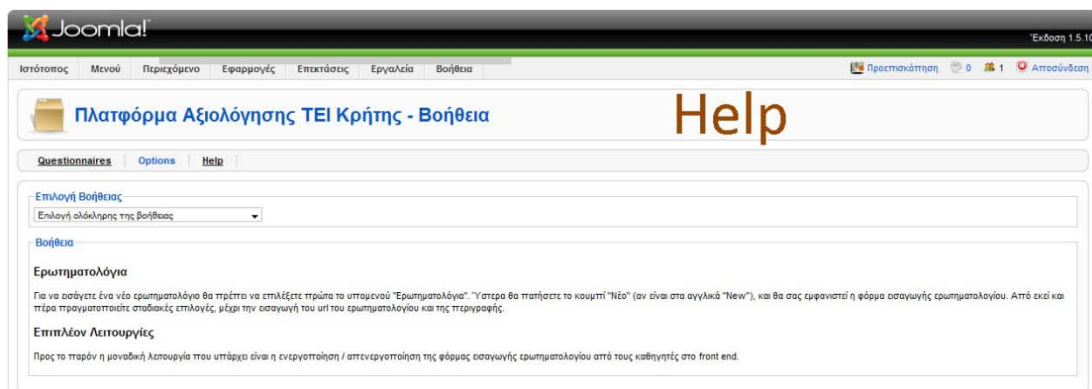
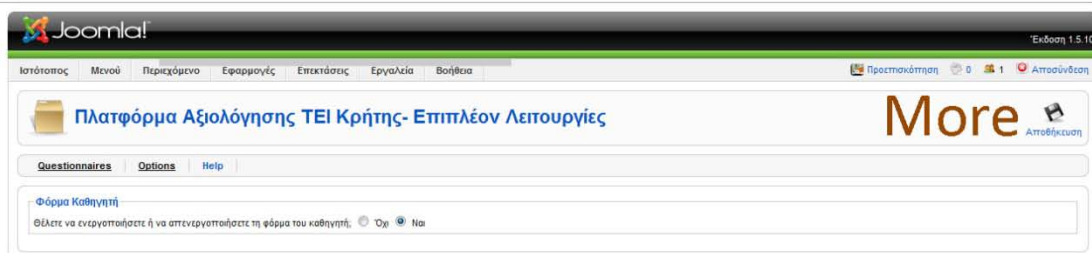
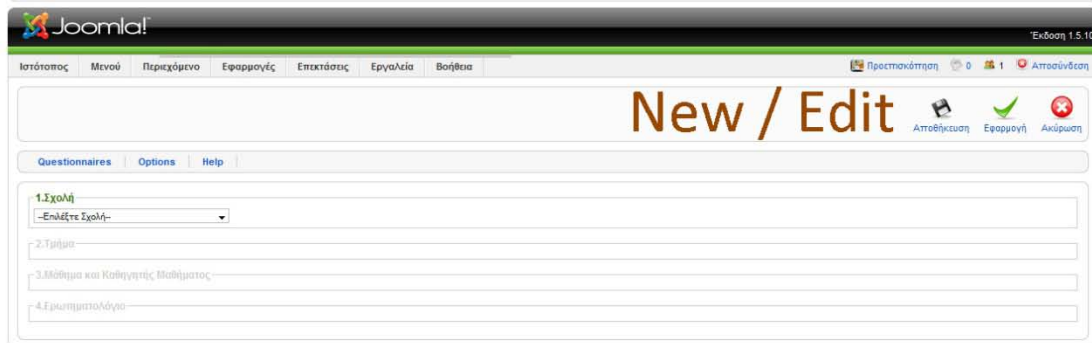
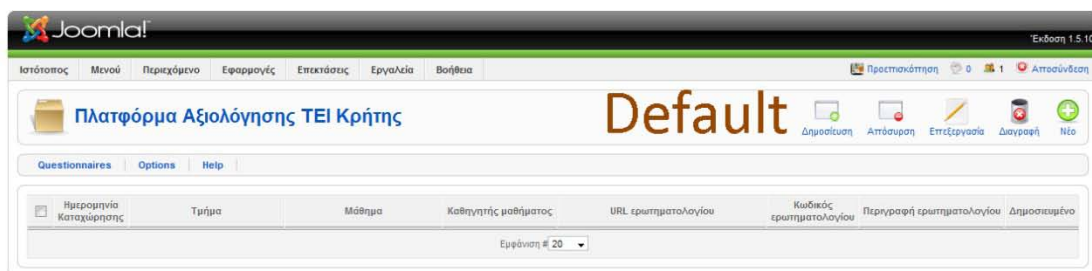
    //Επιπλέον Λειτουργίες
```

```
case 'more':
TOOLBAR_surveys_more::_DEFAULT();
break;

//Βοήθεια
case 'help':
TOOLBAR_surveys_help::_DEFAULT();
break;

//Default
default:
TOOLBAR_surveys::_DEFAULT();
break;
}
?>
```


Αντίστοιχα η διεπαφή ανάλογα, με την επιλογή που έχουμε κάνει εμφανίζει και το ανάλογο toolbar.



Τώρα θα δούμε πως θα φτιάξουμε την πρώτη μας διεπαφή. Θα είναι διεπαφή όπου θα βλέπουμε όλα τα ερωτηματολόγια. Ανοίγουμε το αρχείο admin.surveys.php και γράφουμε τον παρακάτω κώδικα:

```
admin.surveys.php
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require_once(JApplicationHelper::getPath('admin_html'));
JTable::addIncludePath(JPATH_COMPONENT.DS.'tables');

switch($task)
{
    //Default

    default:
        showQuestionnaires($option);
        break;
}

/**
 * Εμφανίζουμε τη λίστα με τα ερωτηματολόγια
 */

function showQuestionnaires($option)
{
    global $mainframe;

    //Pagination
    $limit = JRequest::getVar('limit', $mainframe->getConfig('list_limit'));
    $limitstart = JRequest::getVar('limitstart',0);
    $db =& JFactory::getDBO();
    $query = "SELECT count(*) FROM #__survey_questionnaire";
    $db->setQuery($query);
    $total = $db->loadResult();

    //Questionnaires

    $query = "SELECT category_name, lesson_name, questionnaire_id,
questionnaire_lesson_teacher, questionnaire_url, questionnaire_description,
questionnaire_password, questionnaire_timestamp, published
FROM #__survey_categories,#__survey_lessons, #__survey_questionnaire
WHERE category_id = questionnaire_dept_that_belongs AND lesson_id =
questionnaire_lesson
ORDER BY questionnaire_timestamp DESC";

    $db->setQuery($query, $limitstart, $limit);
    $rows = $db->loadObjectList();

    if ($db->getErrorNum())
```

```
{
    echo $db->stderr();
    return false;
}

jimport('joomla.html.pagination');
$pageNav = new JPagination($total, $limitstart, $limit);

HTML_survey::showQuestionnaires($option, $rows, $pageNav);
}

?>
```

Τώρα ανοίγουμε το αρχείο `admin.surveys.html.php` και γράφουμε τον εξής κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 *
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class HTML_survey
{
    function showQuestionnaires($option, &$rows, &$pageNav)
    {
?>
        <form action="index.php" method="post" name="adminForm">
            <table class="adminlist">
                <thead>
                    <tr>
                        <th width="20">
                            <input type="checkbox" name="toggle"
value="" onclick="checkAll(<?php echo count($rows);?>);"/>
                        </th>
                        <th class="title">Ημερομηνία Καταχώρησης</th>
                        <th class="title" width="15%">Τμήμα</th>
                        <th class="title" width="15%">Μάθημα</th>
                        <th class="title" width="11%">Καθηγητής
μαθήματος</th>
                        <th class="title" width="20%">URL
ερωτηματολογίου</th>
                        <th class="title" width="8%">Κωδικός
ερωτηματολογίου</th>
                        <th class="title" nowrap="nowrap"
width="10%">Περιγραφή ερωτηματολογίου</th>
                        <th class="title">Δημοσιευμένο</th>
                    </tr>
```

```

</thead>

<?php

    jimport('joomla.filter.output');
    $k=0;
    for($i=0, $n= count($rows); $i < $n; $i++)
    {
        $row =& $rows[$i];
        $checked = JHTML::_('grid.id', $i, $row->questionnaire_id);
        $published = JHTML::_('grid.published', $row, $i);

        $link =
JFilterOutput::ampReplace('index.php?option=!. $option."&task=edit&cid[]=".$row-
>questionnaire_id);
        ?>

        <tr class="<?php echo "row$k"; ?>">
            <td>
                <?php echo $checked; ?>
            </td>

            <!-- Ημερομηνία καταχώρησης -->
            <td>
                <?php
                    $part1 = explode("-", $row-
>questionnaire_timestamp);
                    $pieces = explode(" ", $part1[2]);
                    echo $pieces[0]."/". $part1[1]."/". $part1[0]." -
". $pieces[1];
                ?>
            </td>

            <!-- Τμήμα -->
            <td>
                <?php
                    echo $row->category_name;
                ?>
            </td>

            <!-- Μάθημα -->
            <td>
                <?php
                    echo '<a href="'. $link.'">!. $row->lesson_name."</a>";
                ?>
            </td>

            <td>
                <?php echo $row->questionnaire_lesson_teacher; ?>
            </td>

```

```

<!--Url ερωτηματολογίου-->
<td>
    <?php echo $row->questionnaire_url; ?>
</td>

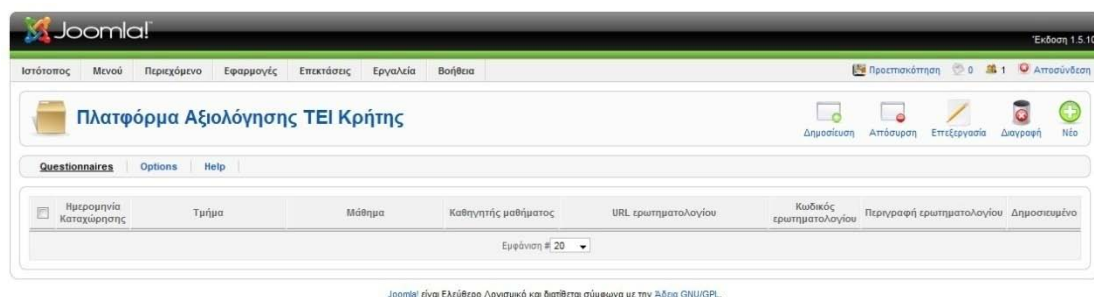
<!--Κωδικός ερωτηματολογίου-->
<td>
    <?php echo $row->questionnaire_password; ?>
</td>

<!--Περιγραφή ερωτηματολογίου-->
<td>
    <?php echo $row->questionnaire_description; ?>
</td>

<td align="center">
    <?php echo $published; ?>
</td>
</tr>
<?php
$х = 1 - $х;
}
?>
<tfoot>
    <td colspan="9">
        <?php echo $pageNav->getListFooter(); ?></td>
</tfoot>
</table>
<input type="hidden" name="option" value="<?php echo $option;?>" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="checked" value="0" />
</form>
<?php
}}?>

```

Το τελικό αποτέλεσμα μας είναι η εικόνα παρακάτω.



Τώρα θα φτιάξουμε την διεπαφή για την εισαγωγή και επεξεργασία δεδομένων. Ανοίγουμε πάλι το admin.surveys.php και μέσα στο switch εισάγουμε, μέσα τον παρακάτω κώδικα, που έχει χρώμα καφέ και είναι bold:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require_once(JApplicationHelper::getPath('admin_html'));
JTable::addIncludePath(JPATH_COMPONENT.DS.'tables');

switch($task)
{
    //Ερωτηματολόγια

    case 'edit':
    case 'add':
        addQuestionnaire($option, $task);
        break;

    //Default
    default:
        showQuestionnaires($option);
        break;
}

/**
 * Η συνάρτηση για το προσθήκη των ερωτηματολογίων
 */

function addQuestionnaire($option, $task)
{
    if ($task == 'edit')
    {
        if(JRequest::getVar('faculty'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');

            HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
            $errorMessages, $task);
        }
        elseif(JRequest::getVar('dept'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');
```

```

        $task = 'add';
        HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
    }
    else
    {
        require_once('modules/edit.php');
        require_once('modules/facultyEdit.php');
        require_once('modules/deptEdit.php');
        require_once('modules/lessonsEdit.php');
        HTML_survey::addQuestionnaire($questionnaire,
$comboboxes, $option, $errorMessages,$task);
    }
}
else
{
    require_once('modules/add.php');
    require_once('modules/facultyAdd.php');
    require_once('modules/deptAdd.php');
    require_once('modules/lessonsAdd.php');
    HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
}
}

/**
 * Εμφανίζουμε τη λίστα με τα ερωτηματολόγια
 */

function showQuestionnaires($option)
{
    global $mainframe;

    //Pagination
    $limit = JRequest::getVar('limit', $mainframe->getCfg('list_limit'));
    $limitstart = JRequest::getVar('limitstart',0);
    $db =& JFactory::getDBO();
    $query = "SELECT count(*) FROM #__survey_questionnaire";
    $db->setQuery($query);
    $total = $db->loadResult();

    //Questionnaires
    $query = "SELECT category_name, lesson_name, questionnaire_id,
questionnaire_lesson_teacher, questionnaire_url, questionnaire_description,
questionnaire_password, questionnaire_timestamp, published
FROM #__survey_categories,#__survey_lessons, #__survey_questionnaire

```

```
WHERE category_id = questionnaire_dept_that_belongs AND lesson_id =
questionnaire_lesson
ORDER BY questionnaire_timestamp DESC";

$db->setQuery($query, $limitstart, $limit);
$rows = $db->loadObjectList();

if ($db->getErrorNum())
{
    echo $db->stderr();
    return false;
}

jimport('joomla.html.pagination');
$pageNav = new JPagination($total, $limitstart, $limit);

HTML_survey::showQuestionnaires($option, $rows, $pageNav);
}

?>
```

Ενώ μέσα στο αρχείο admin.surveys.html.php εισάγουμε τον παρακάτω κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 *
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class HTML_survey
{

function addQuestionnaire($questionnaire, $comboboxes, $option, $errorMessages,
$task)
{
    if($task=='add')
    {
        JHTML::script('add.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');

        JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_
surveys/styles/');
    }
    else
    {
        JHTML::stylesheet('edit.admin.surveys.html.css','administrator/components/com_
surveys/styles/');
    }
}
}
```



```

        JHTML::script('edit.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');
    }

?>
<form action="index.php" method="post" name="adminForm" id="adminForm">
    <?php

        if(!empty($errorMessages))
        {
            echo '<fieldset>';
            echo '<legend id="errorsLegend">Λάθη</legend>';
            foreach($errorMessages as $msg)
            {
                echo $msg."<br/>";
            }
            echo '</fieldset>';
        }

    ?>

    <!--Αρχή για το combobox "Faculties" -->
    <fieldset class="adminForm" id="facultiesFieldSet">
        <legend id="facultiesLegend">1.Σχολή</legend>
        <?php
            echo $comboboxes['faculties'];
        ?>
        <label id="noSaveFaculties"></label>
    </fieldset>
    <!--Τέλος για το combobox "Faculties" -->

    <!-- Αρχή για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Σχολή -->
    <fieldset class="adminForm" id="deptsFieldSet">
        <legend id="deptsLegend">2.Τμήμα</legend>

        <?php
            echo $comboboxes['depts'];
        ?>
        <label id="noSaveDepts"></label>
    </fieldset>
    <!--Τέλος για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει επιλεχθεί
Σχολή -->

    <!--Αρχή για το combobox "Lessons" που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Τμήμα -->
    <fieldset class="adminForm" id="lessonsFieldSet">
        <legend id="lessonsLegend">3.Μάθημα και Καθηγητής
Μαθήματος</legend>
        <table summary="" id="lessonsTable">
            <tr>

```

```

        <td>
            <?php
            echo $comboboxes['lessons'];
            ?>
            <label id="noSaveLessons"></label>
        </td>
        <td>Όνομα καθηγητή μαθήματος
            <input type="text" class="text_area" size="100"
name="questionnaire_lesson_teacher" id="questionnaire_lesson_teacher" value="<?php
echo $questionnaire->questionnaire_lesson_teacher; ?>" />
            <label id="noSaveLessonTeacher"></label>
        </td>
    </tr>
</table>
</fieldset>
<!--Τέλος για το combobox Lessons θα εμφανίζεται μόνο αν έχει επιλεχθεί Τμήμα
-->

<!-- Αρχή για τα πεδία του ερωτηματολογίου θα εμφανίζεται μόνο αν έχει
επιλεχθεί Μάθημα -->
<fieldset class="adminForm" id="questionnaireFieldSet">
    <legend id="questionnaireLegend">4.Ερωτηματολόγιο</legend>

    <table summary="" id="questionnaireTable">
        <tr>
            <td class="title">URL ερωτηματολογίου</td>
            <td>
                <input type="text" class="text_area" size="100"
name="questionnaire_url" id="questionnaire_url" value="<?php echo $questionnaire-
>questionnaire_url; ?>" />
                <label id="noSaveUrl"></label>
                <input type="hidden" id="regExHidden"
value="((http):((//))+survey.teicrete.gr/index.php)+([?])+ (sid=)+([0-9])+([&])+ (lang=el))"
/>
            </td>
            <td class="title"><label id="wrongURL"></label></td>
        </tr>
        <tr>
            <td class="title">Περιγραφή <br/><i>(δεν είναι
υποχρεωτική η συμπλήρωση της)</i></td>
            <td>
                <textarea rows="10" cols="57"
id="questionnaire_description" name="questionnaire_description">
                <?php echo $questionnaire-
>questionnaire_description; ?>
                </textarea>
            </td>
            <td></td>
        </tr>
    </table>
</fieldset>

```

```

                <td>Δημοσιευμένο<?php echo
$comboboxes['published'];?></td>
            </tr>
        </table>
    </fieldset>
    <!--Τέλος για τα πεδία του ερωτηματολογίου που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Μάθημα -->
    <input type="hidden" name="questionnaire_id" id="questionnaire_id"
value="<?php echo $questionnaire->questionnaire_id; ?>"/>
    <input type="hidden" name="option" value="<?php echo $option;?>"/>
    <input type="hidden" name="task" value=""/>
</form>
<script language="javascript" type="text/javascript">
    function submitbutton(pressbutton)
    {
        var form = document.adminForm;

        if (pressbutton == 'cancel') {
            submitform( pressbutton );
            return;
        }

        // Validation φόρμας πριν την αποθήκευση ή την εφαρμογή

        if (document.getElementById("faculties").value == "")
        {
            document.getElementById("faculties").style.backgroundColor =
"#ffcccc";
            document.getElementById("noSaveFaculties").style.color =
"#ff0000";
            document.getElementById("noSaveFaculties").innerHTML = "Δεν
έχετε επιλέξει σχολή!";
        } else if
(document.getElementById("questionnaire_dept_that_belongs").value == ""){

            document.getElementById("questionnaire_dept_that_belongs").style.background
Color = "#ffcccc";
            document.getElementById("noSaveDepts").style.color = "#ff0000";
            document.getElementById("noSaveDepts").innerHTML = "Δεν
έχετε επιλέξει τμήμα!";
        } else if
((document.getElementById("questionnaire_lesson").selectedIndex == 0) ||
(document.getElementById("questionnaire_lesson_teacher").value == "")){

            if
(document.getElementById("questionnaire_lesson").selectedIndex != 0)
            {
                //Για το όνομα του καθηγητή

```

```
        document.getElementById("questionnaire_lesson_teacher").style.backgroundColor = "#ffcccc";
        document.getElementById("noSaveLessonTeacher").style.color = "#ff0000";
        document.getElementById("noSaveLessonTeacher").innerHTML = "Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"
    }
    else
if(document.getElementById("questionnaire_lesson_teacher").value != "")
    {
        //Για το μάθημα

        document.getElementById("questionnaire_lesson").style.backgroundColor = "#ffcccc";
        document.getElementById("noSaveLessons").style.color = "#ff0000";
        document.getElementById("noSaveLessons").innerHTML = "Δεν έχετε επιλέξει μάθημα!"
    }
    else
    {
        //Για το μάθημα

        document.getElementById("questionnaire_lesson").style.backgroundColor = "#ffcccc";
        document.getElementById("noSaveLessons").style.color = "#ff0000";
        document.getElementById("noSaveLessons").innerHTML = "Δεν έχετε επιλέξει μάθημα!";

        //Για το όνομα του καθηγητή

        document.getElementById("questionnaire_lesson_teacher").style.backgroundColor = "#ffcccc";
        document.getElementById("noSaveLessonTeacher").style.color = "#ff0000";
        document.getElementById("noSaveLessonTeacher").innerHTML = "Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!";
    }

    } else if
(document.getElementById("questionnaire_lesson_teacher").value == ""){

        document.getElementById("questionnaire_lesson_teacher").style.backgroundColor = "#ffcccc";
        document.getElementById("noSaveLessonTeacher").style.color = "#ff0000";
```

```

        document.getElementById("noSaveLessonTeacher").innerHTML =
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"

        } else if (document.getElementById("questionnaire_url").value == ""){

        document.getElementById("questionnaire_url").style.backgroundColor =
"#ffcccc";

        document.getElementById("noSaveUrl").style.color = "#ff0000";
        document.getElementById("noSaveUrl").innerHTML = "Δεν έχετε
εισάγει url ερωτηματολογίου!";
        } else if (document.getElementById("questionnaire_url").value != "") {

        var regEx = new
RegExp(document.getElementById("regExHidden").value);

        if(!document.getElementById("questionnaire_url").value.match(regEx)) {

        setTimeout("document.getElementById('questionnaire_url').focus()", -1000000);

        setTimeout("document.getElementById('questionnaire_url').select()",-1000000);

        document.getElementById("questionnaire_url").style.backgroundColor = "#ffffff";
        document.getElementById("noSaveUrl").innerHTML = "";
        document.getElementById("wrongURL").style.color = "#ff0000";
        document.getElementById("wrongURL").innerHTML = "Το url που
καταθέσατε δεν είναι url του limesurvey";
        return false;
        }else {submitform(pressbutton);}

        } else {
        submitform( pressbutton );
        }

        }

</script>
<?php
}

function showQuestionnaires($option, &$rows, &$pageNav)
{
?>

        <form action="index.php" method="post" name="adminForm">
        <table class="adminlist">
        <thead>
        <tr>
        <th width="20">
        <input type="checkbox" name="toggle"
value="" onclick="checkAll(<?php echo count($rows);?>);"/>
        </th>
        <th class="title">Ημερομηνία Καταχώρησης</th>

```

```

<th class="title" width="15%">Τμήμα</th>
<th class="title" width="15%">Μάθημα</th>
<th class="title" width="11%">Καθηγητής
μαθήματος</th>
<th class="title" width="20%">URL
ερωτηματολογίου</th>
<th class="title" width="8%">Κωδικός
ερωτηματολογίου</th>
<th class="title" nowrap="nowrap"
width="10%">Περιγραφή ερωτηματολογίου</th>
<th class="title">Δημοσιευμένο</th>
<tr/>
</thead>

<?php
    jimport('joomla.filter.output');
    $k =0;
    for($i=0, $n= count($rows); $i < $n; $i++)
    {
        $row =& $rows[$i];
        $checked = JHTML::_('grid.id', $i, $row->questionnaire_id);
        $published = JHTML::_('grid.published', $row, $i);

        $link =
JFilterOutput::ampReplace('index.php?option='.$option."&task=edit&cid[]=".$row-
>questionnaire_id);
        ?>

<tr class="<?php echo "row$k"; ?>">
    <td>
        <?php echo $checked; ?>
    </td>

    <!-- Ημερομηνία καταχώρησης -->
    <td>
        <?php
            $part1 = explode("-", $row-
>questionnaire_timestamp);
            $pieces = explode(" ", $part1[2]);
            echo $pieces[0]."/".$part1[1]."/".$part1[0]." -
".$pieces[1];
        ?>
    </td>

    <!-- Τμήμα -->
    <td>
        <?php
            echo $row->category_name;
        ?>
    </td>

```

```

        <!--Μάθημα-->
        <td>
            <?php
            echo '<a href="'. $link. "'>!. $row->lesson_name.</a>";
            ?>
        </td>

        <td>
            <?php echo $row->questionnaire_lesson_teacher; ?>
        </td>

        <!--Url ερωτηματολογίου-->
        <td>
            <?php echo $row->questionnaire_url; ?>
        </td>

        <!--Κωδικός ερωτηματολογίου-->
        <td>
            <?php echo $row->questionnaire_password; ?>
        </td>

        <!--Περιγραφή ερωτηματολογίου-->
        <td>
            <?php echo $row->questionnaire_description; ?>
        </td>

        <td align="center">
            <?php echo $published; ?>
        </td>
    </tr>
<?php
$k = 1 - $k;
}
?>
<tfoot>
    <td colspan="9">
        <?php echo $pageNav->getListFooter(); ?></td>
</tfoot>
</table>
<input type="hidden" name="option" value="<?php echo $option;?>"/>
<input type="hidden" name="task" value=""/>
<input type="hidden" name="checkboxchecked" value="0"/>
</form>
<?php
}}?>

```

Το αποτέλεσμα του κώδικα είναι η παρακάτω διεπαφή:



Τώρα θα φτιάξουμε την διεπαφή για τις ρυθμίσεις (Options). Για να γίνει αυτό ανοίγουμε πάλι το αρχείο `admin.surveys.php` και εισάγουμε τον παρακάτω κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require_once(JApplicationHelper::getPath('admin_html'));
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

switch($task)
{
    //Ερωτηματολόγια

    case 'edit':
    case 'add':
        addQuestionnaire($option, $task);
        break;

    //Επιπλέον Λειτουργίες

    case 'more':
        showMore($option);
        break;

    //Default

    default:
        showQuestionnaires($option);
        break;
}
```



```
/**
 * Η συνάρτηση για το προσθήκη των ερωτηματολογίων
 */

function addQuestionnaire($option, $task)
{
    if ($task == 'edit')
    {
        if(JRequest::getVar('faculty'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');

            HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
        }
        elseif(JRequest::getVar('dept'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');
            $task = 'add';
            HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
        }
        else
        {
            require_once('modules/edit.php');
            require_once('modules/facultyEdit.php');
            require_once('modules/deptEdit.php');
            require_once('modules/lessonsEdit.php');
            HTML_survey::addQuestionnaire($questionnaire,
$comboboxes, $option, $errorMessages,$task);
        }
    }
    else
    {
        require_once('modules/add.php');
        require_once('modules/facultyAdd.php');
        require_once('modules/deptAdd.php');
        require_once('modules/lessonsAdd.php');
        HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
    }
}
```

```
/**
 * Επιπλέον Λειτουργίες
 */

function showMore($option)
{
    $more =& JTable::getInstance('More', 'Table');
    $more->load(1);

    $radioSelection = JHTML::_('select.booleanlist', 'IsTeacherFrontEndFormAvailable',
'class="inputbox"', $more->IsTeacherFrontEndFormAvailable);

    HTML_survey::showMore($option,$more, $radioSelection);
}

function saveMore($option)
{
    global $mainframe;

    $more =& JTable::getInstance('More', 'Table');
    $more->load(1);

    //Εάν έχει κάνει αλλαγές για τη φόρμα του καθηγητή
    if($more->more_id)
    {
        if(!$more->bind(JRequest::get('post')))
        {
            echo '<script>alert('.$more->getError().)';
            echo '</script>';
            exit();
        }

        if(!$more->store())
        {
            echo '<script>alert('.$more->getError().)';
            echo '</script>';
            exit();
        }

        $mainframe->redirect('index.php?option='.$option.'&task=more','Οι
αλλαγές αποθηκεύτηκαν');
    }
}
}
```

```
/**
 * Εμφανίζουμε τη λίστα με τα ερωτηματολόγια
 */

function showQuestionnaires($option)
{
    global $mainframe;

    //Pagination
    $limit = JRequest::getVar('limit', $mainframe->getConfig('list_limit'));
    $limitstart = JRequest::getVar('limitstart',0);
    $db =& JFactory::getDBO();
    $query = "SELECT count(*) FROM #__survey_questionnaire";
    $db->setQuery($query);
    $total = $db->loadResult();

    //Questionnaires
    $query = "SELECT category_name, lesson_name, questionnaire_id,
questionnaire_lesson_teacher, questionnaire_url, questionnaire_description,
questionnaire_password, questionnaire_timestamp, published
FROM #__survey_categories,#__survey_lessons, #__survey_questionnaire
WHERE category_id = questionnaire_dept_that_belongs AND lesson_id =
questionnaire_lesson
ORDER BY questionnaire_timestamp DESC";

    $db->setQuery($query, $limitstart, $limit);
    $rows = $db->loadObjectList();

    if ($db->getErrorNum())
    {
        echo $db->stderr();
        return false;
    }

    jimport('joomla.html.pagination');
    $pageNav = new JPagination($total, $limitstart, $limit);
    HTML_survey::showQuestionnaires($option, $rows, $pageNav);
}
```

Ενώ στο αρχείο admin.surveys.html.php βάζουμε τον εξής κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 *
 */
```

```
defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class HTML_survey
{

function addQuestionnaire($questionnaire, $comboboxes, $option, $errorMessages, $task)
    {

        if($task=='add')
        {
            JHTML::script('add.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');

            JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_surveys/styles/');
        }
        else
        {

            JHTML::stylesheet('edit.admin.surveys.html.css','administrator/components/com_surveys/styles/');
            JHTML::script('edit.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');
        }

    }

?>
<form action="index.php" method="post" name="adminForm" id="adminForm">
    <?php

        if(!empty($errorMessages))
        {
            echo '<fieldset>';
            echo '<legend id="errorsLegend">Λάθη</legend>';
            foreach($errorMessages as $msg)
            {
                echo $msg."<br/>";
            }
            echo '</fieldset>';
        }

    ?>

    <!--Αρχή για το combobox "Faculties" -->
    <fieldset class="adminForm" id="facultiesFieldSet">
        <legend id="facultiesLegend">1.Σχολή</legend>
        <?php
            echo $comboboxes['faculties'];
        ?>
        <label id="noSaveFaculties"></label>
    </fieldset>
```

```

<!--Τέλος για το combobox "Faculties" -->

    <!-- Αρχή για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Σχολή -->
    <fieldset class="adminForm" id="deptsFieldSet">
        <legend id="deptsLegend">2.Τμήμα</legend>

        <?php
        echo $comboboxes['depts'];
        ?>
        <label id="noSaveDepts"></label>
    </fieldset>
    <!--Τέλος για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει επιλεχθεί
Σχολή -->

    <!--Αρχή για το combobox "Lessons" που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Τμήμα -->
    <fieldset class="adminForm" id="lessonsFieldSet">
        <legend id="lessonsLegend">3.Μάθημα και Καθηγητής
Μαθήματος</legend>
        <table summary="" id="lessonsTable">
            <tr>
                <td>
                    <?php
                    echo $comboboxes['lessons'];
                    ?>
                    <label id="noSaveLessons"></label>
                </td>
                <td>Όνομα καθηγητή μαθήματος
                <input type="text" class="text_area" size="100"
name="questionnaire_lesson_teacher" id="questionnaire_lesson_teacher" value="<?php
echo $questionnaire->questionnaire_lesson_teacher; ?>"/>
                <label id="noSaveLessonTeacher"></label>
                </td>
            </tr>
        </table>
    </fieldset>
    <!--Τέλος για το combobox Lessons θα εμφανίζεται μόνο αν έχει επιλεχθεί Τμήμα --
>

    <!-- Αρχή για τα πεδία του ερωτηματολογίου θα εμφανίζεται μόνο αν έχει επιλεχθεί
Μάθημα -->
    <fieldset class="adminForm" id="questionnaireFieldSet">
        <legend id="questionnaireLegend">4.Ερωτηματολόγιο</legend>

        <table summary="" id="questionnaireTable">
            <tr>
                <td class="title">URL ερωτηματολογίου</td>
                <td>

```

```

        <input type="text" class="text_area" size="100"
name="questionnaire_url" id="questionnaire_url" value="<?php echo $questionnaire-
>questionnaire_url; ?>" />
        <label id="noSaveUrl"></label>
        <input type="hidden" id="regExHidden"
value="((http):((/))+survey.teicrete.gr/index.php)+([?])+(sid=)+([0-9])+([&])+(lang=el)" />
        </td>
        <td class="title"><label id="wrongURL"></label></td>
    </tr>
    <tr>
        <td class="title">Περιγραφή <br/><i>(δεν είναι
υποχρεωτική η συμπλήρωση της)</i></td>
        <td>
            <textarea rows="10" cols="57"
id="questionnaire_description" name="questionnaire_description">
                <?php echo $questionnaire-
>questionnaire_description; ?>
            </textarea>
        </td>
    </tr>
    <tr>
        <td></td>
        <td></td>
    </tr>
    <tr>
        <td></td>
        <td>Δημοσιευμένο<?php echo
$comboboxes['published'];?></td>
    </tr>
</table>
</fieldset>
<!--Τέλος για τα πεδία του ερωτηματολογίου που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Μάθημα -->
    <input type="hidden" name="questionnaire_id" id="questionnaire_id"
value="<?php echo $questionnaire->questionnaire_id; ?>"/>
    <input type="hidden" name="option" value="<?php echo $option;?>"/>
    <input type="hidden" name="task" value=""/>
</form>
<script language="javascript" type="text/javascript">
    function submitbutton(pressbutton)
    {
        var form = document.adminForm;

        if (pressbutton == 'cancel') {
            submitform( pressbutton );
            return;
        }

        // Validation φόρμας πριν την αποθήκευση ή την εφαρμογή

        if (document.getElementById("faculties").value == "")
        {
            document.getElementById("faculties").style.backgroundColor =
"#ffcccc";

```

```
document.getElementById("noSaveFaculties").style.color =
"#ff0000";
document.getElementById("noSaveFaculties").innerHTML = "Δεν
έχετε επιλέξει σχολή!";

} else if
(document.getElementById("questionnaire_dept_that_belongs").value == ""){

document.getElementById("questionnaire_dept_that_belongs").style.backgroundCo
lor = "#ffcccc";
document.getElementById("noSaveDepts").style.color = "#ff0000";
document.getElementById("noSaveDepts").innerHTML = "Δεν έχετε
επιλέξει τμήμα!";

} else if ((document.getElementById("questionnaire_lesson").selectedIndex
== 0) || (document.getElementById("questionnaire_lesson_teacher").value == "")){

if (document.getElementById("questionnaire_lesson").selectedIndex
!= 0)
{
//Για το όνομα του καθηγητή

document.getElementById("questionnaire_lesson_teacher").style.backgroundColor
= "#ffcccc";
document.getElementById("noSaveLessonTeacher").style.color =
"#ff0000";
document.getElementById("noSaveLessonTeacher").innerHTML =
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"
}
else
if(document.getElementById("questionnaire_lesson_teacher").value != "")
{
//Για το μάθημα

document.getElementById("questionnaire_lesson").style.backgroundColor =
"#ffcccc";
document.getElementById("noSaveLessons").style.color =
"#ff0000";
document.getElementById("noSaveLessons").innerHTML = "Δεν
έχετε επιλέξει μάθημα!"
}
else
{
//Για το μάθημα

document.getElementById("questionnaire_lesson").style.backgroundColor =
"#ffcccc";
document.getElementById("noSaveLessons").style.color =
"#ff0000";
```

```
        document.getElementById("noSaveLessons").innerHTML = "Δεν  
έχετε επιλέξει μάθημα!";  
  
        //Για το όνομα του καθηγητή  
  
        document.getElementById("questionnaire_lesson_teacher").style.backgroundColor  
= "#ffcccc";  
        document.getElementById("noSaveLessonTeacher").style.color =  
"#ff0000";  
        document.getElementById("noSaveLessonTeacher").innerHTML =  
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!";  
    }  
  
    } else if (document.getElementById("questionnaire_lesson_teacher").value  
== ""){  
  
        document.getElementById("questionnaire_lesson_teacher").style.backgroundColor  
= "#ffcccc";  
        document.getElementById("noSaveLessonTeacher").style.color =  
"#ff0000";  
        document.getElementById("noSaveLessonTeacher").innerHTML =  
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"  
  
    } else if (document.getElementById("questionnaire_url").value == ""){  
  
        document.getElementById("questionnaire_url").style.backgroundColor = "#ffcccc";  
        document.getElementById("noSaveUrl").style.color = "#ff0000";  
        document.getElementById("noSaveUrl").innerHTML = "Δεν έχετε  
εισάγει url ερωτηματολογίου!";  
    } else if (document.getElementById("questionnaire_url").value != "") {  
  
        var regEx = new  
RegExp(document.getElementById("regExHidden").value);  
  
        if(!document.getElementById("questionnaire_url").value.match(regEx)) {  
  
            setTimeout("document.getElementById('questionnaire_url').focus()", -1000000);  
  
            setTimeout("document.getElementById('questionnaire_url').select()",-1000000);  
  
            document.getElementById("questionnaire_url").style.backgroundColor = "#ffffff";  
            document.getElementById("noSaveUrl").innerHTML = "";  
            document.getElementById("wrongURL").style.color = "#ff0000";  
            document.getElementById("wrongURL").innerHTML = "To url που  
καταθέσατε δεν είναι url του limesurvey";  
            return false;  
        }else {submitform(pressbutton);}  
  
    } else {
```



```

        submitform( pressbutton );
    }

}

</script>
<?php
}

function showQuestionnaires($option, &$rows, &$pageNav)
{
?>
    <form action="index.php" method="post" name="adminForm">
        <table class="adminlist">
            <thead>
                <tr>
                    <th width="20">
                        <input type="checkbox" name="toggle"
value="" onclick="checkAll(<?php echo count($rows);?>);"/>
                    </th>
                    <th class="title">Ημερομηνία Καταχώρησης</th>
                    <th class="title" width="15%">Τμήμα</th>
                    <th class="title" width="15%">Μάθημα</th>
                    <th class="title" width="11%">Καθηγητής
μαθήματος</th>
                    <th class="title" width="20%">URL
ερωτηματολογίου</th>
                    <th class="title" width="8%">Κωδικός
ερωτηματολογίου</th>
                    <th class="title" nowrap="nowrap"
width="10%">Περιγραφή ερωτηματολογίου</th>
                    <th class="title">Δημοσιευμένο</th>
                </tr>
            </thead>

            <?php

                jimport('joomla.filter.output');
                $k =0;
                for($i=0, $n= count($rows); $i < $n; $i++ )
                {
                    $row =& $rows[$i];
                    $checked = JHTML::_('grid.id', $i, $row->questionnaire_id);
                    $published = JHTML::_('grid.published', $row, $i);

                    $link =
JFilterOutput::ampReplace('index.php?option='.$option."&task=edit&cid[]=". $row-
>questionnaire_id);
                ?>

```

```

<tr class="<?php echo "row$k"; ?>">
    <td>
        <?php echo $checked; ?>
    </td>

    <!-- Ημερομηνία καταχώρησης -->
    <td>
        <?php
            $part1 = explode("-", $row-
>questionnaire_timestamp);
            $pieces = explode(" ", $part1[2]);
            echo $pieces[0]."/".$part1[1]."/".$part1[0]." -
".$pieces[1];
        ?>
    </td>

    <!-- Τμήμα -->
    <td>
        <?php
            echo $row->category_name;
        ?>
    </td>

    <!-- Μάθημα -->
    <td>
        <?php
            echo '<a href="'. $link. "'>'. $row->lesson_name. '</a>';
        ?>
    </td>

    <td>
        <?php echo $row->questionnaire_lesson_teacher; ?>
    </td>

    <!-- Url ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_url; ?>
    </td>

    <!-- Κωδικός ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_password; ?>
    </td>

    <!-- Περιγραφή ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_description; ?>
    </td>

    <td align="center">
        <?php echo $published; ?>

```

```

        </td>
    </tr>
<?php
$k = 1 - $k;
}
?>
<tfoot>
    <td colspan="9">
        <?php echo $pageNav->getListFooter(); ?></td>
</tfoot>
</table>
<input type="hidden" name="option" value="<?php echo $option;?>" />
<input type="hidden" name="task" value="" />
<input type="hidden" name="checkboxchecked" value="0" />
</form>
<?php
}

function showMore($option,$more, $radioSelection)
{
    JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_
surveys/styles/');
    ?>

    <form action="index.php" method="post" name="adminForm" id="adminForm">

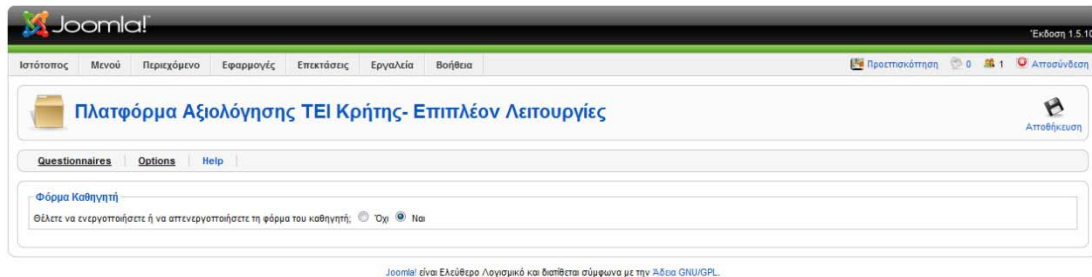
        <fieldset>
            <legend>Φόρμα Καθηγητή</legend>
            Θέλετε να ενεργοποιήσετε ή να απενεργοποιήσετε τη φόρμα του
καθηγητή;
                <?php echo $radioSelection; ?>
            </fieldset>

            <input type="hidden" name="more_id" id="more_id" value="<?php echo
$more->more_id; ?>" />
            <input type="hidden" name="option" value="<?php echo $option;?>" />
            <input type="hidden" name="task" value="more" />
        </form>

<?php
}

```

Η διεπαφή που δημιουργείται είναι η εξής:



Τώρα θα φτιάξουμε τη διεπαφή που θα εμφανίζει τη βοήθεια. Ανοίγουμε για ακόμη μια φορά το admin.surveys.php και εισάγουμε τον παρακάτω κώδικα:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require_once(JApplicationHelper::getPath('admin_html'));
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

switch($task)
{
    //Ερωτηματολόγια

    case 'edit':
    case 'add':
        addQuestionnaire($option, $task);
        break;

    //Επιπλέον Λειτουργίες

    case 'more':
        showMore($option);
        break;

    //Βοήθεια

    case 'help':
        showHelp($option);
        break;

    //Default

    default:
        showQuestionnaires($option);
        break;
}
```

```
/**
 * Η συνάρτηση για το προσθήκη των ερωτηματολογίων
 */

function addQuestionnaire($option, $task)
{

if ($task == 'edit')
    {
        if(JRequest::getVar('faculty'))
            {
                require_once('modules/add.php');
                require_once('modules/facultyAdd.php');
                require_once('modules/deptAdd.php');
                require_once('modules/lessonsAdd.php');

                HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
                $errorMessages, $task);
            }
        elseif(JRequest::getVar('dept'))
            {
                require_once('modules/add.php');
                require_once('modules/facultyAdd.php');
                require_once('modules/deptAdd.php');
                require_once('modules/lessonsAdd.php');
                $task = 'add';
                HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
                $errorMessages, $task);
            }
        else
            {
                require_once('modules/edit.php');
                require_once('modules/facultyEdit.php');
                require_once('modules/deptEdit.php');
                require_once('modules/lessonsEdit.php');
                HTML_survey::addQuestionnaire($questionnaire,
                $comboboxes, $option, $errorMessages,$task);
            }
    }

else
    {

        require_once('modules/add.php');
        require_once('modules/facultyAdd.php');
        require_once('modules/deptAdd.php');
        require_once('modules/lessonsAdd.php');
        HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
        $errorMessages, $task);
    }
}
```

```
}  
  
    }  
  
/**  
 * Εμφανίζουμε τη λίστα με τα ερωτηματολόγια  
 */  
  
function showQuestionnaires($option)  
{  
    global $mainframe;  
  
    //Pagination  
    $limit = JRequest::getVar('limit', $mainframe->getConfig('list_limit'));  
    $limitstart = JRequest::getVar('limitstart',0);  
    $db =& JFactory::getDBO();  
    $query = "SELECT count(*) FROM #__survey_questionnaire";  
    $db->setQuery($query);  
    $total = $db->loadResult();  
  
    //Questionnaires  
    $query = "SELECT category_name, lesson_name, questionnaire_id,  
questionnaire_lesson_teacher, questionnaire_url, questionnaire_description,  
questionnaire_password, questionnaire_timestamp, published  
FROM #__survey_categories,#__survey_lessons, #__survey_questionnaire  
WHERE category_id = questionnaire_dept_that_belongs AND lesson_id =  
questionnaire_lesson  
ORDER BY questionnaire_timestamp DESC";  
  
    $db->setQuery($query, $limitstart, $limit);  
    $rows = $db->loadObjectList();  
  
    if ($db->getErrorNum())  
    {  
        echo $db->stderr();  
        return false;  
    }  
  
    jimport('joomla.html.pagination');  
    $pageNav = new JPagination($total, $limitstart, $limit);  
    HTML_survey::showQuestionnaires($option, $rows, $pageNav);  
}  
  
/**  
 * Επιπλέον Λειτουργίες  
 */  
  
function showMore($option)  
{
```

```

    $more =& JTable::getInstance('More', 'Table');
    $more->load(1);

    $radioSelection = JHTML::_('select.booleanlist', 'IsTeacherFrontEndFormAvailable',
'class="inputbox"', $more->IsTeacherFrontEndFormAvailable);

    HTML_survey::showMore($option,$more, $radioSelection);
}
}

/**
 * Βοήθεια
 */

function showHelp($option)
{
    $db =& JFactory::getDBO();

    if (JRequest::getVar('help_id')==0)
    {
        $query = "SELECT Help FROM #__survey_help";
    }
    else
    {
        $query = "SELECT Help FROM #__survey_help WHERE
Help_id=".JRequest::getVar('help_id');
    }

    $db->setQuery($query);

    if(!$db->query())
    {
        $error = "Παρουσιάστηκε σφάλμα, κατά την έλευση του κειμένου
της βοήθειας!<br/>". $db->getErrorMessage;
    }
    else
    {
        $help = $db->loadObjectList();
    }

    HTML_survey::showHelp($option, $error, $help);
}
?>

```

Ενώ στο αρχείο admin.surveys.html.php βάζουμε τον παρακάτω κώδικα.

```

<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ

```

```
* Κωνσταντίνος Γρεβενίτης AM - 1123
* Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
*
*/

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');

class HTML_survey
{

function addQuestionnaire($questionnaire, $comboboxes, $option, $errorMessages, $task)
    {

        if($task=='add')
        {
            JHTML::script('add.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');

            JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_surveys/styles/');
        }
        else
        {

            JHTML::stylesheet('edit.admin.surveys.html.css','administrator/components/com_surveys/styles/');
            JHTML::script('edit.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');
        }

    }

?>
<form action="index.php" method="post" name="adminForm" id="adminForm">
    <?php

        if(!empty($errorMessages))
        {
            echo '<fieldset>';
            echo '<legend id="errorsLegend">Λάθη</legend>';
            foreach($errorMessages as $msg)
            {
                echo $msg."<br/>";
            }
            echo '</fieldset>';
        }

    ?>

    <!--Αρχή για το combobox "Faculties" -->
    <fieldset class="adminForm" id="facultiesFieldSet">
        <legend id="facultiesLegend">1.Σχολή</legend>
```



```

        <?php
        echo $comboboxes['faculties'];
        ?>
        <label id="noSaveFaculties"></label>
    </fieldset>
    <!--Τέλος για το combobox "Faculties" -->

        <!-- Αρχή για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει
    επιλεγθεί Σχολή -->
        <fieldset class="adminForm" id="deptsFieldSet">
    <legend id="deptsLegend">2.Τμήμα</legend>

        <?php
        echo $comboboxes['depts'];
        ?>
        <label id="noSaveDepts"></label>
    </fieldset>
    <!--Τέλος για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει επιλεγθεί
    Σχολή -->

        <!--Αρχή για το combobox "Lessons" που θα εμφανίζεται μόνο αν έχει
    επιλεγθεί Τμήμα -->
        <fieldset class="adminForm" id="lessonsFieldSet">
        <legend id="lessonsLegend">3.Μάθημα και Καθηγητής
    Μαθήματος</legend>
        <table summary="" id="lessonsTable">
            <tr>
                <td>
                    <?php
                    echo $comboboxes['lessons'];
                    ?>
                    <label id="noSaveLessons"></label>
                </td>
                <td>Όνομα καθηγητή μαθήματος
                    <input type="text" class="text_area" size="100"
    name="questionnaire_lesson_teacher" id="questionnaire_lesson_teacher" value="<?php
    echo $questionnaire->questionnaire_lesson_teacher; ?>" />
                    <label id="noSaveLessonTeacher"></label>
                </td>
            </tr>
        </table>
    </fieldset>
    <!--Τέλος για το combobox Lessons θα εμφανίζεται μόνο αν έχει επιλεγθεί Τμήμα --
    >

    <!-- Αρχή για τα πεδία του ερωτηματολογίου θα εμφανίζεται μόνο αν έχει επιλεγθεί
    Μάθημα -->
    <fieldset class="adminForm" id="questionnaireFieldSet">
        <legend id="questionnaireLegend">4.Ερωτηματολόγιο</legend>

        <table summary="" id="questionnaireTable">

```

```

        <tr>
            <td class="title">URL ερωτηματολογίου</td>
            <td>
                <input type="text" class="text_area" size="100"
name="questionnaire_url" id="questionnaire_url" value="<?php echo $questionnaire-
>questionnaire_url; ?>" />
                <label id="noSaveUrl"></label>
                <input type="hidden" id="regExHidden"
value="((http):(//))+survey.teicrete.gr/index.php+([?])+(sid=)+([0-9])+([&])+(lang=el))" />
                </td>
        <td class="title"><label id="wrongURL"></label></td>
    </tr>
    <tr>
        <td class="title">Περιγραφή <br/><i>(δεν είναι
υποχρεωτική η συμπλήρωση της)</i></td>
        <td>
            <textarea rows="10" cols="57"
id="questionnaire_description" name="questionnaire_description">
                <?php echo $questionnaire-
>questionnaire_description; ?>
            </textarea>
        </td>
    </tr>
    <tr>
        <td>Δημοσιευμένο<?php echo
$comboboxes['published'];?></td>
    </tr>
</table>
</fieldset>
<!-- Τέλος για τα πεδία του ερωτηματολογίου που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Μάθημα -->
    <input type="hidden" name="questionnaire_id" id="questionnaire_id"
value="<?php echo $questionnaire->questionnaire_id; ?>" />
    <input type="hidden" name="option" value="<?php echo $option;?>" />
    <input type="hidden" name="task" value="" />
</form>
<script language="javascript" type="text/javascript">
    function submitbutton(pressbutton)
    {
        var form = document.adminForm;

        if (pressbutton == 'cancel') {
            submitform( pressbutton );
            return;
        }

        // Validation φόρμας πριν την αποθήκευση ή την εφαρμογή

        if (document.getElementById("faculties").value == "")
    {

```

```
document.getElementById("faculties").style.backgroundColor =
"#ffcccc";
document.getElementById("noSaveFaculties").style.color =
"#ff0000";
document.getElementById("noSaveFaculties").innerHTML = "Δεν
έχετε επιλέξει σχολή!";

} else if
(document.getElementById("questionnaire_dept_that_belongs").value == ""){

document.getElementById("questionnaire_dept_that_belongs").style.backgroundCo
lor = "#ffcccc";
document.getElementById("noSaveDepts").style.color = "#ff0000";
document.getElementById("noSaveDepts").innerHTML = "Δεν έχετε
επιλέξει τμήμα!";

} else if ((document.getElementById("questionnaire_lesson").selectedIndex
== 0) || (document.getElementById("questionnaire_lesson_teacher").value == "")){

if (document.getElementById("questionnaire_lesson").selectedIndex
!= 0)
{
//Για το όνομα του καθηγητή

document.getElementById("questionnaire_lesson_teacher").style.backgroundColor
= "#ffcccc";
document.getElementById("noSaveLessonTeacher").style.color =
"#ff0000";
document.getElementById("noSaveLessonTeacher").innerHTML =
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"
}
else
if(document.getElementById("questionnaire_lesson_teacher").value != "")
{
//Για το μάθημα

document.getElementById("questionnaire_lesson").style.backgroundColor =
"#ffcccc";
document.getElementById("noSaveLessons").style.color =
"#ff0000";
document.getElementById("noSaveLessons").innerHTML = "Δεν
έχετε επιλέξει μάθημα!"
}
else
{
//Για το μάθημα

document.getElementById("questionnaire_lesson").style.backgroundColor =
"#ffcccc";
```

```
document.getElementById("noSaveLessons").style.color =
"#ff0000";
document.getElementById("noSaveLessons").innerHTML = "Δεν
έχετε επιλέξει μάθημα!";

//Για το όνομα του καθηγητή

document.getElementById("questionnaire_lesson_teacher").style.backgroundColor
= "#ffcccc";
document.getElementById("noSaveLessonTeacher").style.color =
"#ff0000";
document.getElementById("noSaveLessonTeacher").innerHTML = "Δεν έχετε εισάγει όνομα
καθηγητή μαθήματος!";
}

} else if (document.getElementById("questionnaire_lesson_teacher").value
== ""){

document.getElementById("questionnaire_lesson_teacher").style.backgroundColor
= "#ffcccc";
document.getElementById("noSaveLessonTeacher").style.color =
"#ff0000";
document.getElementById("noSaveLessonTeacher").innerHTML =
"Δεν έχετε εισάγει όνομα καθηγητή μαθήματος!"

} else if (document.getElementById("questionnaire_url").value == ""){

document.getElementById("questionnaire_url").style.backgroundColor = "#ffcccc";
document.getElementById("noSaveUrl").style.color = "#ff0000";
document.getElementById("noSaveUrl").innerHTML = "Δεν έχετε
εισάγει url ερωτηματολογίου!";
} else if (document.getElementById("questionnaire_url").value != "") {

var regEx = new
RegExp(document.getElementById("regExHidden").value);

if(!document.getElementById("questionnaire_url").value.match(regEx)) {

setTimeout("document.getElementById('questionnaire_url').focus()", -1000000);

setTimeout("document.getElementById('questionnaire_url').select()", -1000000);

document.getElementById("questionnaire_url").style.backgroundColor = "#ffffff";
document.getElementById("noSaveUrl").innerHTML = "";
document.getElementById("wrongURL").style.color = "#ff0000";
document.getElementById("wrongURL").innerHTML = "Το url που
καταθέσατε δεν είναι url του limesurvey";
return false;
} else {submitform(pressbutton);}
```

```

        } else {
            submitform( pressbutton );
        }
    }

</script>
<?php
}

function showQuestionnaires($option, &$rows, &$pageNav)
{
?>
    <form action="index.php" method="post" name="adminForm">
        <table class="adminlist">
            <thead>
                <tr>
                    <th width="20">
                        <input type="checkbox" name="toggle"
value="" onclick="checkAll(<?php echo count($rows);?>);"/>
                    </th>
                    <th class="title">Ημερομηνία Καταχώρησης</th>
                    <th class="title" width="15%">Τμήμα</th>
                    <th class="title" width="15%">Μάθημα</th>
                    <th class="title" width="11%">Καθηγητής
μαθήματος</th>
                    <th class="title" width="20%">URL
ερωτηματολογίου</th>
                    <th class="title" width="8%">Κωδικός
ερωτηματολογίου</th>
                    <th class="title" nowrap="nowrap"
width="10%">Περιγραφή ερωτηματολογίου</th>
                    <th class="title">Δημοσιευμένο</th>
                <tr/>
            </thead>

            <?php
                jimport('joomla.filter.output');
                $k=0;
                for($i=0, $n= count($rows); $i < $n; $i++)
                {
                    $row =& $rows[$i];
                    $checked = JHTML::_('grid.id', $i, $row->questionnaire_id);
                    $published = JHTML::_('grid.published', $row, $i);

                    $link =
JFilterOutput::ampReplace('index.php?option='.$option."&task=edit&cid[]=$row->questionnaire_id);
                ?>

```

```

<tr class="<?php echo "row$k"; ?>">
    <td>
        <?php echo $checked; ?>
    </td>

    <!-- Ημερομηνία καταχώρησης -->
    <td>
        <?php
            $part1 = explode("-", $row-
>questionnaire_timestamp);
$pieces = explode(" ", $part1[2]);
            echo $pieces[0]."/".$part1[1]."/".$part1[0]." -
".$pieces[1];
        ?>
    </td>

    <!-- Τμήμα -->
    <td>
        <?php
            echo $row->category_name;
        ?>
    </td>

    <!-- Μάθημα -->
    <td>
        <?php
            echo '<a href="'. $link. "'>'. $row->lesson_name. "</a>";
        ?>
    </td>

    <td>
        <?php echo $row->questionnaire_lesson_teacher; ?>
    </td>

    <!-- Url ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_url; ?>
    </td>

    <!-- Κωδικός ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_password; ?>
    </td>

    <!-- Περιγραφή ερωτηματολογίου -->
    <td>
        <?php echo $row->questionnaire_description; ?>
    </td>

    <td align="center">

```

```

                <?php echo $published; ?>
            </td>
        </tr>
    </table>
    <?php
    $k = 1 - $k;
    }
    ?>
    <tfoot>
        <td colspan="9">
            <?php echo $pageNav->getListFooter(); ?></td>
        </tfoot>
    </table>
    <input type="hidden" name="option" value="<?php echo $option;?>" />
    <input type="hidden" name="task" value="" />
    <input type="hidden" name="checkboxchecked" value="0" />
    </form>
    <?php
    }

function showMore($option,$more, $radioSelection)
{
    JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_surveys/styles/');
    ?>

    <form action="index.php" method="post" name="adminForm" id="adminForm">

        <fieldset>
            <legend>Φόρμα Καθηγητή</legend>
            Θέλετε να ενεργοποιήσετε ή να απενεργοποιήσετε τη φόρμα του
καθηγητή;
                <?php echo $radioSelection; ?>
            </fieldset>

            <input type="hidden" name="more_id" id="more_id" value="<?php echo
    $more->more_id; ?>" />
            <input type="hidden" name="option" value="<?php echo $option;?>" />
            <input type="hidden" name="task" value="more" />
        </form>

    <?php
    }

function showHelp($option, $error, $help)
{
    JHTML::stylesheet('add.admin.surveys.html.css','administrator/components/com_surveys/styles/');
    JHTML::script('help.admin.surveys.html.js',
'administrator/components/com_surveys/scripts/');
    ?>

```

```

<form action="index.php" method="post" name="adminForm">
  <fieldset>
    <legend>
      Επιλογή Βοήθειας
    </legend>

    <select id="HelpSelect">
      <option value="0">Επιλογή ολόκληρης της βοήθειας</option>
      <option value="1">Επιλογή της βοήθειας για τα
ερωτηματολόγια</option>
      <option value="2">Επιλογή της βοήθειας για τις επιπλέον
λειτουργίες</option>
    </select>

  </fieldset>

  <?php
  if($error != null)
  {
    echo '<label id="error">'. $error. '</label>';
  }
  else
  {
    echo '<fieldset><legend>Βοήθεια</legend>'. $help[0]-
>Help.$help[1]->Help.'</fieldset>';
  }

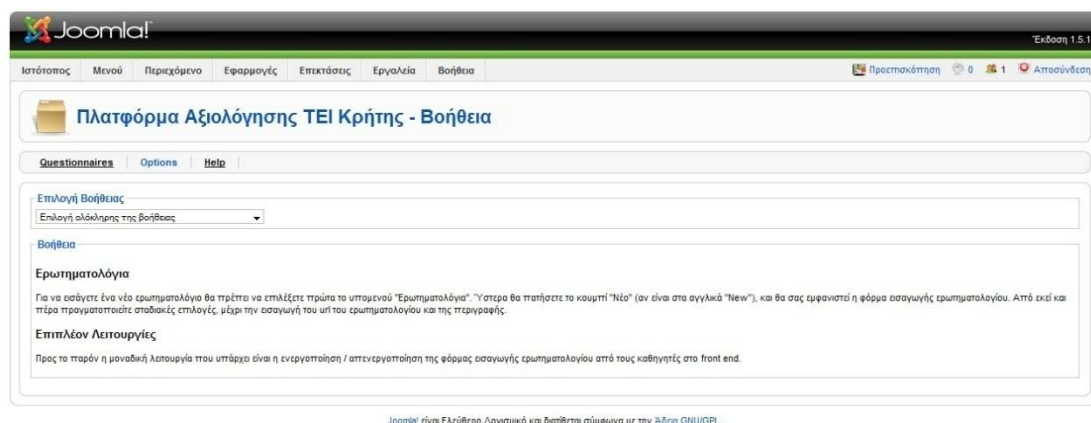
  ?>

</form>

<?php
}
}
?>

```

Η διεπαφή που δημιουργείται είναι η παρακάτω:



Τέλος ο κώδικας που μένει είναι για αποθήκευση και διαγραφή εγγραφών, όπως επίσης και για την αποθήκευση ρυθμίσεων. Αυτός ο κώδικας μπαίνει μέσα στο admin.surveys.php


```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 */

defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
require_once(JApplicationHelper::getPath('admin_html'));
JTable::addIncludePath(JPATH_COMPONENT_DS.'tables');

switch($task)
{
    //Ερωτηματολόγια

    case 'edit':
    case 'add':
        addQuestionnaire($option, $task);
        break;

    case 'apply':
    case 'save':
        saveQuestionnaire($option, $task);
        break;

    case 'remove':
        removeQuestionnaire($option);
        break;

    case 'publish':
    case 'unpublish':
        publish($option);
        break;

    //Επιπλέον Λειτουργίες
    case 'more':
        showMore($option);
        break;

    case 'saveMore':
        saveMore($option);
        break;

    //Βοήθεια
    case 'help':
        showHelp($option);
        break;

    //Default
    default:
        showQuestionnaires($option);
```

```
        break;
    }

    /**
     * Η συνάρτηση για το προσθήκη των ερωτηματολογίων
     */

function addQuestionnaire($option, $task)
{
    if ($task == 'edit')
    {
        if(JRequest::getVar('faculty'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');

            HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
            $errorMessages, $task);
        }
        elseif(JRequest::getVar('dept'))
        {
            require_once('modules/add.php');
            require_once('modules/facultyAdd.php');
            require_once('modules/deptAdd.php');
            require_once('modules/lessonsAdd.php');
            $task = 'add';
            HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
            $errorMessages, $task);
        }
        else
        {
            require_once('modules/edit.php');
            require_once('modules/facultyEdit.php');
            require_once('modules/deptEdit.php');
            require_once('modules/lessonsEdit.php');
            HTML_survey::addQuestionnaire($questionnaire,
            $comboboxes, $option, $errorMessages,$task);
        }
    }
    else
    {
        require_once('modules/add.php');
        require_once('modules/facultyAdd.php');
        require_once('modules/deptAdd.php');
        require_once('modules/lessonsAdd.php');
```

```
HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages, $task);
}

}

/**
 * Η συνάρτηση για την αποθήκευση των ερωτηματολογίων
 */

function saveQuestionnaire($option, $task)
{
    global $mainframe;

    $questionnaire =& JTable::getInstance('Questionnaire', 'Table');

    if($questionnaire->questionnaire_id)
    {

        /**
         * Ελέγχουμε για ύπαρξη λαθών και αν υπάρχουν τότε
         * βγαίνει javascript alert
         */
        if(!$questionnaire->bind(JRequest::get('post')))
        {
            echo '<script>';
            echo 'alert("'.$questionnaire->getError().'");';
            echo 'window.history.go(-1);';
            echo '</script>';
            exit();
        }

        if (!$questionnaire->store())
        {
            echo '<script>';
            echo 'alert("'.$questionnaire->getError().'");';
            echo 'window.history.go(-1);';
            echo '</script>';
            exit();
        }

        /**
         * Ελέγχουμε αν είναι save ή apply
         */

        switch ($task)
        {
            case 'apply':
```

```

        $msg = "Οι αλλαγές αποθηκεύτηκαν στο ερωτηματολόγιο";
        $link = 'index.php?option='.$option.'&task=edit&cid[]=$questionnaire-
>questionnaire_id;
        break;

        case 'save':
        default:
        $msg = "Το νέο ερωτηματολόγιο αποθηκεύτηκε";
        $link = 'index.php?option='.$option;
        break;
    }
    /**
    * Τέλος τα δεδομένα φεύγουν στη βάση
    */
    $mainframe->redirect($link,$msg);
}
else
{
    /**
    * Επιλέγεται ένα τυχαίος τετραψήφιος αριθμός για κωδικός για password
    * του ερωτηματολογίου
    *
    */

    $questionnaire->questionnaire_password = mt_rand(1000,9999);

    /**
    * Ελέγχουμε για ύπαρξη λαθών και αν υπάρχουν τότε
    * βγαίνει javascript alert
    */

    if(!$questionnaire->bind(JRequest::get('post')))
    {
        echo '<script>';
        echo 'alert("'" . $questionnaire->getError(). "');";
        echo 'window.history.go(-1);';
        echo '</script>';
        exit();
    }

    if (!$questionnaire->store())
    {
        echo '<script>';
        echo 'alert("'" . $questionnaire->getError(). "');";
        echo 'window.history.go(-1);';
        echo '</script>';
        exit();
    }

    /**
    * Ελέγχουμε αν είναι save ή apply

```

```

*/

switch ($task)
{
    case 'apply':
        $msg = "Οι αλλαγές αποθηκεύτηκαν στο ερωτηματολόγιο";
        $link = 'index.php?option='.$option.'&task=edit&cid[]='.$questionnaire->questionnaire_id;
        break;

    case 'save':
    default:
        $msg = "Το νέο ερωτηματολόγιο αποθηκεύτηκε";
        $link = 'index.php?option='.$option;
        break;
}

$mainframe->redirect($link,$msg);
}

}

/**
 * Εμφανίζουμε τη λίστα με τα ερωτηματολόγιο
 */

function showQuestionnaires($option)
{
    global $mainframe;

    //Pagination
    $limit = JRequest::getVar('limit', $mainframe->getConfig('list_limit'));
    $limitstart = JRequest::getVar('limitstart',0);
    $db =& JFactory::getDBO();
    $query = "SELECT count(*) FROM #__survey_questionnaire";
    $db->setQuery($query);
    $total = $db->loadResult();

    //Questionnaires
    $query = "SELECT category_name, lesson_name, questionnaire_id,
questionnaire_lesson_teacher, questionnaire_url, questionnaire_description,
questionnaire_password, questionnaire_timestamp, published
FROM #__survey_categories,#__survey_lessons, #__survey_questionnaire
WHERE category_id = questionnaire_dept_that_belongs AND lesson_id =
questionnaire_lesson
ORDER BY questionnaire_timestamp DESC";

    $db->setQuery($query, $limitstart, $limit);
    $rows = $db->loadObjectList();

    if ($db->getErrorNum())

```

```

    {
        echo $db->stderr();
        return false;
    }

jimport('joomla.html.pagination');
$pageNav = new JPagination($total, $limitstart, $limit);
HTML_survey::showQuestionnaires($option, $rows, $pageNav);
}

/**
 * Διαγραφή εγγραφής
 */

function removeQuestionnaire($option)
{
    global $mainframe;
    $cid = JRequest::getVar('cid', array(), '', 'array');
    $db =& JFactory::getDBO();

    if (count($cid))
    {
        $cids = implode(",", $cid);
        $query = "DELETE FROM #__survey_questionnaire WHERE
questionnaire_id IN ( $cids)";

        $db->setQuery($query);

        if(!$db->query())
        {
            echo '<script>';
            echo 'alert("'.$db->getErrorMsg().'");';
            echo 'window.history.go(-1);';
            echo '</script>';
        }
    }
    $mainframe->redirect('index.php?option='.$option);
}

function publish($option)
{
    global $mainframe;

    $questionnaire =& JTable::getInstance('Questionnaire', 'Table');
    $cid = JRequest::getVar('cid', array(), '', 'array');
    $db =& JFactory::getDBO();
    $query = "SELECT published FROM #__survey_questionnaire WHERE
questionnaire_id=$cid[0]";

    $db->setQuery($query);
}

```

```

        if(!$db->query())
        {
            echo '<script>';
            echo 'alert("'" . $db->getErrMsg(). "'");';
            echo 'window.history.go(-1);';
            echo '</script>';
        }

        $total = $db->loadResult();

        if ($total==0)
        {
            $total=1;

            $questionnaire->publish($cid, $total);
            $mainframe->redirect('index.php?option=' . $option);
        }

        if ($total==1)
        {
            $total=0;

            $questionnaire->publish($cid, $total);
            $mainframe->redirect('index.php?option=' . $option);
        }
    }

    /**
     * Επιπλέον Λειτουργίες
     */

    function showMore($option)
    {
        $more =& JTable::getInstance('More', 'Table');
        $more->load(1);

        $radioSelection = JHTML::_('select.booleanlist', 'IsTeacherFrontEndFormAvailable',
        'class="inputbox"', $more->IsTeacherFrontEndFormAvailable);

        HTML_survey::showMore($option,$more, $radioSelection);
    }

    function saveMore($option)
    {
        global $mainframe;

        $more =& JTable::getInstance('More', 'Table');
        $more->load(1);
    }

```

```

//Εάν έχει κάνει αλλαγές για τη φόρμα του καθηγητή
if($more->more_id)
{

    if(!$more->bind(JRequest::get('post')))

    {

        echo '<script>alert('.$more->getError().)';
        echo '</script>';
        exit();
    }

    if(!$more->store())

    {
        echo '<script>alert('.$more->getError().)';
        echo '</script>';
        exit();
    }

    $mainframe->redirect('index.php?option=.'.$option.'&task=more','Οι
αλλαγές αποθηκεύτηκαν');
    }
}

/**
 * Βοήθεια
 */

function showHelp($option)
{
    $db =& JFactory::getDBO();

    if (JRequest::getVar('help_id')==0)
    {
        $query = "SELECT Help FROM #__survey_help";
    }
    else
    {
        $query = "SELECT Help FROM #__survey_help WHERE
Help_id=".JRequest::getVar('help_id');
    }

    $db->setQuery($query);

    if(!$db->query())
    {
        $error = "Παρουσιάστηκε σφάλμα, κατά την έλευση του κειμένου
της βοήθειας!<br/>". $db->getErrorMsg;
    }
}

```



```
    }
else
    {
        $help = $db->loadObjectList();
    }

HTML_survey::showHelp($option, $error, $help);
} ?>
```

Ο κώδικας για το front end ακολουθεί την ίδια λογική, αλλά με τις απαραίτητες διαφοροποιήσεις είναι απαραίτητες. Τα ονόματα των αρχείων για το front end είναι surveys.php και surveys.html.php

```
<?php
/**
 * TEI Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 *
 */
defined('_JEXEC') or die('Δεν επιτρέπεται η πρόσβαση');
echo '<div class="componentheading">Πλατφόρμα Αξιολόγησης TEI Κρήτης</div>';
jimport('joomla.application.helper');
require_once( JApplicationHelper::getPath('html') );
JTable::addIncludePath(JPATH_ADMINISTRATOR.DS.'components'.DS.$option.DS.'tables');

switch($task)
{
    default:
        showPublishedQuestionnaires($option);
        break;

    case 'view':
        beforeLimeSurvey($option);
        break;

    case 'add':
        addQuestionnaire($option);
        break;

    case 'submit':
        submitQuestionnaire($option);
        break;

    case 'result':
        showResults($option);
        break;
}
```

```

function showPublishedQuestionnaires($option)
{
    $db =& JFactory::getDBO();

    // ΣΔΟ

    // Τμήμα Διοίκησης Επιχειρήσεων
    $query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 6 ORDER BY lesson_name,
questionnaire_lesson_teacher";
    $db->setQuery($query);
    $rows[0] = $db->loadObjectList();

    // Τμήμα Λογιστικής
    $query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 11 ORDER BY lesson_name,
questionnaire_lesson_teacher";
    $db->setQuery($query);
    $rows[1] = $db->loadObjectList();

    // Τμήμα Τουριστικών Επιχειρήσεων
    $query = "SELECT lesson_name, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 15 ORDER BY lesson_name,
questionnaire_lesson_teacher";
    $db->setQuery($query);
    $rows[2] = $db->loadObjectList();

    // ΣΕΥΠ

    // Τμήμα Κοινωνικής εργασίας
    $query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 10 ORDER BY lesson_name,
questionnaire_lesson_teacher";
    $db->setQuery($query);
    $rows[3] = $db->loadObjectList();

    // Τμήμα Νοσηλευτικής
    $query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 13 ORDER BY lesson_name,
questionnaire_lesson_teacher";
    $db->setQuery($query);
}

```

```
$rows[4] = $db->loadObjectList();

//ΣΤΕΓ

// Τμήμα Θερμοκηπιακών Καλλιεργειών και Ανθοκομίας
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 9 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[5] = $db->loadObjectList();

// Τμήμα Φυτικής Παραγωγής
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 16 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[6] = $db->loadObjectList();

//ΣΤΕΦ

//Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 7 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[7] = $db->loadObjectList();

// Τμήμα Ηλεκτρολογίας
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 8 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[8] = $db->loadObjectList();

// Τμήμα Μηχανολογίας
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 12 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[9] = $db->loadObjectList();
```

```

//      Τμήμα Πολιτικών Δομικών Έργων
$query = "SELECT lesson_name, questionnaire_id, questionnaire_lesson_teacher,
questionnaire_description, questionnaire_url FROM #__survey_lessons,
#__survey_questionnaire WHERE questionnaire_lesson = lesson_id AND published = 1 AND
questionnaire_dept_that_belongs = 14 ORDER BY lesson_name,
questionnaire_lesson_teacher";
$db->setQuery($query);
$rows[10] = $db->loadObjectList();

if($db->getErrorNum())
{
    echo $db->stderr();
    return false;
}

$teacherForm = null;

$query = "SELECT IsTeacherFrontEndFormAvailable FROM #__survey_more";
$db->setQuery($query);

$IsTeacherFrontEndFormAvailable = $db->loadResult();

if($IsTeacherFrontEndFormAvailable==1)
{
    $teacherForm = '<label><br/>Αν είστε καθηγητής και θέλετε να
υποβάλλεται ερωτηματολόγιο πατήστε <a
href=index.php?option='. $option .'&task=add&Itemid='.JRequest::getVar("Itemid").'>εδώ</a
></label>';
}
HTML_survey::showQuestionnaires($rows, $option, $teacherForm);
}

function addQuestionnaire($option)
{
    if (substr($ref, 0, 58) ==
http://'. $_SERVER["SERVER_NAME"]. $_SERVER['PHP_SELF']. "?option=com_surveys")

    {
        require_once('modules/add.php');
        require_once('modules/facultyAdd.php');
        require_once('modules/deptAdd.php');
        require_once('modules/lessonsAdd.php');
        HTML_survey::addQuestionnaire($questionnaire, $comboboxes, $option,
$errorMessages);
    }
    else
    {
        echo '<b>Δεν έχετε πρόσβαση στη συγκεκριμένη σελίδα</b>';
    }
}
}

```

```

function submitQuestionnaire($option)
{
    global $mainframe;
    $questionnaire =& JTable::getInstance('Questionnaire', 'Table');

    /**
     * Επιλέγεται ένα τυχαίος τετραψήφιος αριθμός για κωδικός για password
     * του ερωτηματολογίου
     *
     */

    $questionnaire->questionnaire_password = mt_rand(1000,9999);

    /**
     * Ελέγχουμε για ύπαρξη λαθών και αν υπάρχουν τότε
     * βγαίνει javascript alert
     */

    if(!$questionnaire->bind(JRequest::get('post')))
    {
        echo '<script>';
        echo 'alert("'.$questionnaire->getError().'");';
        echo 'window.history.go(-1);';
        echo '</script>';
        exit();
    }

    if (!$questionnaire->store())
    {
        echo '<script>';
        echo 'alert("'.$questionnaire->getError().'");';
        echo 'window.history.go(-1);';
        echo '</script>';
        exit();
    }

    $mainframe-
>redirect('index.php?option='.$option.'&task=result&Itemid='.JRequest::getVar("Itemid"));
}

function showResults($option)
{
    HTML_survey::showResults($option);
}

function beforeLimeSurvey($option)
{
    $db =& JFactory::getDBO();
}

```

```
$query = "SELECT questionnaire_password, questionnaire_url FROM
#_survey_questionnaire WHERE questionnaire_id = ".JRequest::getVar('id')." AND
published = 1 ";
$db->setQuery($query);
$row = $db->loadObjectList();
$password = $row[0]-> questionnaire_password ;
$url = $row[0]-> questionnaire_url;
HTML_survey::beforeLimeSurvey($option, $password,$url );
}
?>
```

Ενώ στο αρχείο surveys.html.php έχει ως εξής:

```
<?php
/**
 * ΤΕΙ Κρήτης - Σχολή Τεχνολογικών Εφαρμογών - Τμήμα ΕΠΠ
 * Κωνσταντίνος Γρεβενίτης AM - 1123
 * Πτυχιακή "Ανάπτυξη Συστήματος για online συμπλήρωση ερωτηματολογίων"
 *
 */

class HTML_survey
{

    function showQuestionnaires($rows, $option, $teacherForm)
    {

        JHTML::stylesheet('surveys.html.php.css','components/com_surveys/styles/');
        JHTML::script('surveys.html.php.js','components/com_surveys/scripts/');

    }

}

<div class="tabs" id="Sdo">ΣΔΟ</div>
<div class="tabs" id="Seyp">ΣΕΥΠ</div>
<div class="tabs" id="Steg">ΣΤΕΓ</div>
<div class="tabs" id="Stef">ΣΤΕΦ</div>

<div class="content" id="contentSdo">
    <div class="lessonsExpander" id="DioikisiEpixiriseon">
        Τμήμα Διοίκησης Επιχειρήσεων
    </div>

    <div class="lessons" id="lessonsDioikisiEpixiriseon">
        <table cellpadding="0" cellspacing="0">
            <?php
                foreach($rows[0] as $row)
                {
                    $link = 'index.php?option='.$option.'&id='.$row->
                    questionnaire_id.'&task=view';
                    echo '<tr>';
```

```

        echo '<td>';
        echo          '<a          href="'. $link. "'>' . $row-
>lesson_name.</a>'."&nbsp;";
        echo '</td>';
        echo '<td>';
        echo " - ". $row->questionnaire_lesson_teacher;
        echo '</td>';
        echo '</tr>';

        if ($row->questionnaire_description != null)
        {
            echo '<tr>';
            echo '<td colspan="2">';
            echo $row->questionnaire_description;
            echo '</td>';
            echo '</tr>';
        }
    }
?>
</table>
</div>

<div class="lessonsExpander" id="Logistiki">
    Τμήμα Λογιστικής
</div>

<div class="lessons" id="lessonsLogistiki">
    <table>
        <?php
        foreach($rows[1] as $row)
        {
            $link = 'index.php?option=' . $option . '&id=' . $row-
>questionnaire_id . '&task=view';
            echo '<tr>';
            echo '<td>';
            echo          '<a          href="'. $link. "'>' . $row-
>lesson_name.</a>';
            echo '</td>';
            echo '<td>';
            echo $row->questionnaire_lesson_teacher;
            echo '</td>';
            echo '</tr>';

            if ($row->questionnaire_description != null)
            {
                echo '<tr>';
                echo '<td colspan="2">';
                echo $row->questionnaire_description;
                echo '</td>';
                echo '</tr>';
            }
        }
    }
?>
</table>
</div>

```

```

    }
    }
    ?>
</table>
</div>

<div class="lessonsExpander" id="TouristikonEpixiriseon">
    Τμήμα Τουριστικών Επιχειρήσεων
</div>

<div class="lessons" id="lessonsTouristikonEpixiriseon">
    <table>
        <?php
            foreach($rows[2] as $row)
            {
                $link =
'index.php?option='.$option.'&id='.$row->questionnaire_id.'&task=view';
                echo '<tr>';
                echo '<td>';
                echo ' <a href="'. $link.'">'. $row-
>lesson_name.</a>';
                echo '</td>';
                echo '<td>';
                echo ' $row-
>questionnaire_lesson_teacher;
                echo '</td>';
                echo '</tr>';

                if ($row->questionnaire_description
!= null)
                {
                    echo '<tr>';
                    echo '<td colspan="2">';
                    echo ' $row-
>questionnaire_description;
                    echo '</td>';
                    echo '</tr>';
                }
            }
        }
    ?>
</table>
</div>

</div>

<div class="content" id="contentSeyp">
    <div class="lessonsExpander" id="KoinonikiErgasia">

```



```

        Τμήμα Κοινωνικής Εργασίας
    </div>

    <div class="lessons" id="lessonsKoinonikiErgasia">
        <table>
            <?php
                foreach($rows[3] as $row)
                {
                    $link = 'index.php?option='.$option.'&id='.$row->
                    >questionnaire_id.'&task=view';
                    echo '<tr>';
                    echo '<td>';
                    echo          '<a          href="'. $link.'">'. $row->
                    >lesson_name.'</a>';
                    echo '</td>';
                    echo '<td>';
                    echo $row->questionnaire_lesson_teacher;
                    echo '</td>';
                    echo '</tr>';

                    if ($row->questionnaire_description != null)
                    {
                        echo '<tr>';
                        echo '<td colspan="2">';
                        echo $row->questionnaire_description;
                        echo '</td>';
                        echo '</tr>';
                    }
                }
            ?>
        </table>
    </div>

    <div class="lessonsExpander" id="Nosileftiki">
        Τμήμα Νοσηλευτικής
    </div>

    <div class="lessons" id="lessonsNosileftiki">
        <table>
            <?php
                foreach($rows[4] as $row)
                {
                    $link = 'index.php?option='.$option.'&id='.$row->
                    >questionnaire_id.'&task=view';
                    echo '<tr>';
                    echo '<td>';
                    echo          '<a          href="'. $link.'">'. $row->
                    >lesson_name.'</a>';
                    echo '</td>';
                    echo '<td>';

```

```

        echo $row->questionnaire_lesson_teacher;
        echo '</td>';
        echo '</tr>';

        if ($row->questionnaire_description != null)
        {
            echo '<tr>';
            echo '<td colspan="2">';
            echo $row->questionnaire_description;
            echo '</td>';
            echo '</tr>';
        }
    }
?>
</table>
</div>
</div>
<div class="content" id="contentSteg">
    <div class="lessonsExpander" id="Theka">
        Τμήμα Θερμοκηπιακών Καλλιεργειών και Ανθοκομίας
    </div>
    <div class="lessons" id="lessonsTheka">
        <table>
            <?php
                foreach($rows[5] as $row)
                {
                    $link = 'index.php?option='.$option.'&id='.$row->
                    questionnaire_id.'&task=view';
                    echo '<tr>';
                    echo '<td>';
                    echo '<a href="'.$link.'">'.$row->
                    lesson_name.</a>';
                    echo '</td>';
                    echo '<td>';
                    echo $row->questionnaire_lesson_teacher;
                    echo '</td>';
                    echo '</tr>';

                    if ($row->questionnaire_description != null)
                    {
                        echo '<tr>';
                        echo '<td colspan="2">';
                        echo $row->questionnaire_description;
                        echo '</td>';
                        echo '</tr>';
                    }
                }
            </?php
        </table>
    </div>
</div>

```

```

        }
    ?>
</table>
</div>

<div class="lessonsExpander" id="FP">
    Τμήμα Φυτικής Παραγωγής
</div>

<div class="lessons" id="lessonsFP">
    <table>
        <?php
            foreach($rows[6] as $row)
            {
                $link = 'index.php?option='.$option.'&id='.$row->
questionnaire_id.'&task=view';
                echo '<tr>';
                echo '<td>';
                echo '<a href="'.$link.'">'.$row->
lesson_name.'</a>';
                echo '</td>';
                echo '<td>';
                echo $row->questionnaire_lesson_teacher;
                echo '</td>';
                echo '</tr>';

                if ($row->questionnaire_description != null)
                {
                    echo '<tr>';
                    echo '<td colspan="2">';
                    echo $row->questionnaire_description;
                    echo '</td>';
                    echo '</tr>';
                }
            }
        ?>
    </table>
</div>

<div class="content" id="contentStef">

    <div class="lessonsExpander" id="EPP">
        Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων
    </div>

    <div class="lessons" id="lessonsEPP">
        <table>
            <?php

```

```

                foreach($rows[7] as $row)
                {
                    $link = 'index.php?option='.$option.'&id='.$row->questionnaire_id.'&task=view';
                    echo '<tr>';
                    echo '<td>';
                    echo '<a href="'.$link.'">'. $row->lesson_name.'</a>';
                    echo '</td>';
                    echo '<td>';
                    echo $row->questionnaire_lesson_teacher;
                    echo '</td>';
                    echo '</tr>';

                    if ($row->questionnaire_description != null)
                    {
                        echo '<tr>';
                        echo '<td colspan="2">';
                        echo $row->questionnaire_description;
                        echo '</td>';
                        echo '</tr>';
                    }
                }
            ?>
        </table>
    </div>

    <div class="lessonsExpander" id="Hlektrologia">
        Τμήμα Ηλεκτρολογίας
    </div>

    <div class="lessons" id="lessonsHlektrologia">

        <table>
        <?php
            foreach($rows[8] as $row)
            {
                $link = 'index.php?option='.$option.'&id='.$row->questionnaire_id.'&task=view';
                echo '<tr>';
                echo '<td>';
                echo '<a href="'.$link.'">'. $row->lesson_name.'</a>';
                echo '</td>';
                echo '<td>';
                echo $row->questionnaire_lesson_teacher;
                echo '</td>';
                echo '</tr>';

                if ($row->questionnaire_description != null)

```

```

        {
            echo '<tr>';
            echo '<td colspan="2">';
            echo $row->questionnaire_description;
            echo '</td>';
            echo '</tr>';
        }
    }
?>
</table>
</div>

<div class="lessonsExpander" id="Mixanologia">
    Τμήμα Μηχανολογίας
</div>

<div class="lessons" id="lessonsMixanologia">
    <table>
        <?php
            foreach($rows[9] as $row)
            {
                $link = 'index.php?option='.$option.'&id='.$row->
questionnaire_id.'&task=view';
                echo '<tr>';
                echo '<td>';
                echo '<a href="'. $link.'">'. $row->
lesson_name.'</a>';
                echo '</td>';
                echo '<td>';
                echo $row->questionnaire_lesson_teacher;
                echo '</td>';
                echo '</tr>';

                if ($row->questionnaire_description != null)
                {
                    echo '<tr>';
                    echo '<td colspan="2">';
                    echo $row->questionnaire_description;
                    echo '</td>';
                    echo '</tr>';
                }
            }
        }
    ?>
</table>
</div>

<div class="lessonsExpander" id="Domikon">
    Τμήμα Πολιτικών Δομικών Έργων
</div>

```

```

        <div class="lessons" id="lessonsDomikon">
            <table>
                <?php
                    foreach($rows[10] as $row)
                    {
                        $link = 'index.php?option='.$option.'&id='.$row->
>questionnaire_id.'&task=view';
                        echo '<tr>';
                        echo '<td>';
                        echo '        <a                href=""'.$link.'">'. $row->
>lesson_name.'</a>';
                        echo '</td>';
                        echo '<td>';
                        echo $row->questionnaire_lesson_teacher;
                        echo '</td>';
                        echo '</tr>';

                        if ($row->questionnaire_description != null)
                        {
                            echo '<tr>';
                            echo '<td colspan="2">';
                            echo $row->questionnaire_description;
                            echo '</td>';
                            echo '</tr>';
                        }
                    }
                <?php
            </table>
        </div>
</div>

<?php echo $teacherForm;?>

    <?php
}

function addQuestionnaire($questionnaire, $comboboxes, $option, $errorMessages)
{
    JHTML::script('add.surveys.html.php.js', 'components/com_surveys/scripts/');
    JHTML::stylesheet('add.surveys.html.php.css', 'components/com_surveys/styles/');

    ?>
<form action="index.php" method="post" name="adminForm" id="adminForm">
    <?php

        if(!empty($errorMessages))
        {
            echo '<fieldset>';
            echo '<legend id="errorsLegend">Λάθη</legend>';

```

```

        foreach($errorMessagees as $msg)
        {
            echo $msg."<br/>";
        }
        echo '</fieldset>';
    }

    ?>
<!--Αρχή για το combobox "Faculties" -->
<fieldset class="adminForm" id="facultiesFieldSet">
    <legend id="facultiesLegend">1.Σχολή</legend>
    <?php
        echo $comboboxes['faculties'];
    ?>
    <label id="noSaveFaculties"></label>
</fieldset>
<!--Τέλος για το combobox "Faculties" -->

    <!-- Αρχή για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει
    επιλεχθεί Σχολή -->
    <fieldset class="adminForm" id="deptsFieldSet">
        <legend id="deptsLegend">2.Τμήμα</legend>

        <?php
            echo $comboboxes['depts'];
        ?>
        <label id="noSaveDepts"></label>
    </fieldset>
    <!--Τέλος για το combobox "Depts" που θα εμφανίζεται μόνο αν έχει επιλεχθεί
    Σχολή -->

    <!--Αρχή για το combobox "Lessons" που θα εμφανίζεται μόνο αν έχει
    επιλεχθεί Τμήμα -->
    <fieldset class="adminForm" id="lessonsFieldSet">
        <legend id="lessonsLegend">3.Μάθημα και Καθηγητής
        Μαθήματος</legend>
        <table summary="" id="lessonsTable">
            <tr>
                <td>
                    <?php
                        echo $comboboxes['lessons'];
                    ?>
                    <label id="noSaveLessons"></label>
                </td>
            </tr>
            <tr>
                <td>Όνομα καθηγητή μαθήματος:<br/>
                    <input type="text" maxlength="100" size="70"
                    name="questionnaire_lesson_teacher" id="questionnaire_lesson_teacher" value="<?php
                    echo $questionnaire->questionnaire_lesson_teacher; ?>"/>
                    <label id="noSaveLessonTeacher"></label>

```

```

        </td>
    </tr>
</table>
</fieldset>
<!--Τέλος για το combobox Lessons θα εμφανίζεται μόνο αν έχει επιλεχθεί Τμήμα -->
<!-- Αρχή για τα πεδία του ερωτηματολογίου θα εμφανίζεται μόνο αν έχει επιλεχθεί
Μάθημα -->
<fieldset class="adminForm" id="questionnaireFieldSet">
    <legend id="questionnaireLegend">4.Ερωτηματολόγιο</legend>

    <table summary="" id="questionnaireTable">
        <tr>
            <td class="title">URL ερωτηματολογίου</td>
            <td>
                <input type="text" maxlength="100" size="70"
name="questionnaire_url" id="questionnaire_url" value="<?php echo $questionnaire-
>questionnaire_url; ?>" />
                <label id="noSaveUrl"></label>
                <input type="hidden" id="regExHidden"
value="((http):(//))+<survey.teicrete.gr/index.php>+([?])+<(sid=)+<[0-9]>+<(&)+<(lang=el)>" />
                </td>
            <td class="title"><label id="wrongURL"></label></td>
        </tr>
        <tr>
            <td class="title">Περιγραφή <br/><i>(δεν είναι
υποχρεωτική η συμπλήρωση της)</i></td>
            <td>
                <textarea rows="10" cols="57"
id="questionnaire_description" name="questionnaire_description">
                <?php echo $questionnaire-
>questionnaire_description; ?>
                </textarea>
            </td>
        </tr>
    </table>
</fieldset>
<!--Τέλος για τα πεδία του ερωτηματολογίου που θα εμφανίζεται μόνο αν έχει
επιλεχθεί Μάθημα -->
<br/>
<input type="submit" id="button" name="button" value="Υποβολή
ερωτηματολογίου">

<input type="hidden" name="questionnaire_id" id="questionnaire_id"
value="<?php echo $questionnaire->questionnaire_id; ?>" />
<input type="hidden" name="option" value="<?php echo $option; ?>" />
<input type="hidden" name="task" value="submit" />

```



```
<input type="hidden" name="Itemid" value="<?php echo
JRequest::getVar('Itemid');?>"/>

</form>

<?php
}

function showResults($option)
{
    echo '<p>
                Το ερωτηματολόγιο σας αποθηκεύτηκε κανονικά. Για
οποιαδήποτε απορία...
                </p>';
}

function beforeLimeSurvey($option, $password,$url)
{
    if (JRequest::getVar('passwordText')== $password)
    {
        header('Location:'. $url);
    }
    ?>

    <form method="post" action="<?php echo
$_SERVER['PHP_SELF'].'?'.$_SERVER['QUERY_STRING']; ?>">
        <label>Εισάγετε τον κωδικό του ερωτηματολογίου</label><br/>
        <input type="text" name="passwordText" id="passwordText"
maxlength="4" size="4" />
        <input type="submit" value="Ερωτηματολόγιο">

    </form>

<?php
}
}
?>
```

Και η διεπαφή που παράγεται είναι η παρακάτω:



Κεφάλαιο 6^ο

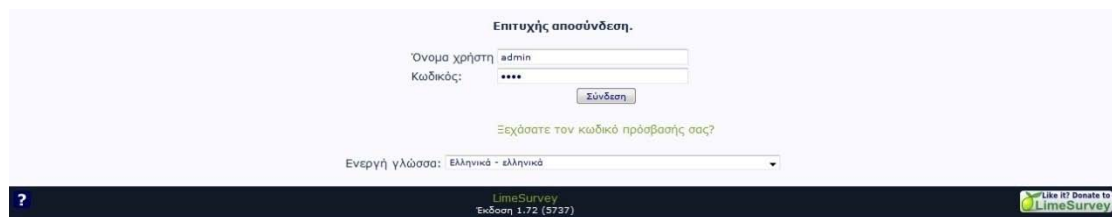
Έλεγχος – Εικονικό Σενάριο

Εισαγωγή

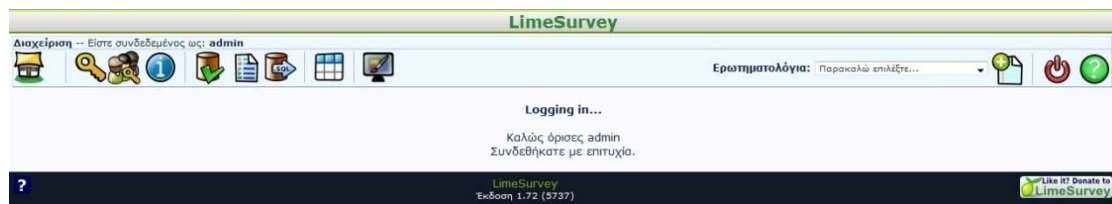
Αφού ολοκληρώσαμε την ανάπτυξη ήρθε η ώρα να κάνουμε τους απαραίτητους ελέγχους και δοκιμές, έτσι ώστε να επαληθεύσουμε την σωστή λειτουργία της εφαρμογής μας και αν ικανοποιεί τις προδιαγραφές μας.

LimeSurvey

Αρχικά κάνουμε login στο lime survey έτσι ώστε να έχουμε πρόσβαση στη πλατφόρμα.



Εάν μπούμε επιτυχώς στην εφαρμογή τότε ηοθόνη που θα συναντήσουμε είναι η παρακάτω:





Ύστερα πατάμε το εικονίδιο για να δημιουργήσουμε μία νέα έρευνα. Αφού το πατήσουμε έρχεται μπροστά μας η εξής διεπαφή:

The screenshot shows the LimeSurvey administration interface. At the top, there's a navigation bar with the LimeSurvey logo and a user profile 'admin'. Below that, a breadcrumb trail reads 'Δημιουργία ή Εισαγωγή Νέου Ερωτηματολογίου'. The main content area is divided into tabs: 'Γενικά', 'Παρουσίαση και Πλοήγηση', 'Έλεγχος δημοσίευσης και πρόσβασης', 'Έλεγχος ενημέρωσης και διαχείρισης δεδομένων', and 'Εισαγωγή Έρευνας'. The 'Γενικά' tab is active. It contains several form fields: 'Βασική γλώσσα' (English), 'Τίτλος' (υποχρεωτικό), 'Περιγραφή', and 'Καλωσόρισμα'. At the bottom, there are fields for 'Διαχειριστής', 'Email Διαχειριστή', 'Επιστρεφόμενα email', and 'Τηλέφωνο για Fax'. A 'Δημιουργία ερωτηματολογίου' button is at the bottom center.

Εδώ μπορούμε να εισάγουμε διάφορα στοιχεία για την έρευνα μας, τα οποία είναι ταξινομημένα – ομαδοποιημένα σε tabs και όχι ένα ολόκληρο κατεβατό. Αναλυτικά για το πώς εισάγεται μία έρευνα βρίσκεται στο εγχειρίδιο του LimeSurvey που υπάρχει στο mini cd. Έτσι λοιπόν αν καταφέρουμε να φτιάξουμε μία έρευνα θα πάρουμε την παρακάτω εικόνα:

The screenshot shows a survey preview page. The title is 'Δοκιμαστικό Ερωτηματολόγιο'. Below the title, it says 'Αυτό είναι ένα δοκιμαστικό ερωτηματολόγιο στα πλαίσια της πτυχιακής μου εργασίας.' There is a section titled 'Υπάρχει μία ερώτηση σε αυτή το ερωτηματολόγιο' with a sub-section 'Σημείωση σχετικά με το προσωπικό απόρρητο'. The text explains that the survey is for research purposes and that the user's responses are anonymous. At the bottom, there are two buttons: 'Επανεκκίνηση ημετελούς ερωτηματολογίου' and 'Επόμενη >>'. A link at the bottom right says '[Εξέρδος από το ερωτηματολόγιο και διαγραφή απαντήσεων]'.

Joomla

Τώρα αυτό που πρέπει να κάνουμε είναι να κρατήσουμε το url το ερωτηματολογίου. Δηλαδή το **http://localhost/limesurvey/index.php?sid=13169&lang=el**. Στην πραγματική εφαρμογή έχουμε θέσει ένα regular expression για την εισαγωγή του url έτσι ώστε να εισάγονται σωστά url στη βάση που λειτουργούν και όχι λάθος καταχωρήσεις. Ωστόσο για τη παρουσίαση της πτυχιακής θα θεωρήσουμε πως το url είναι το επιθυμητό. Τώρα αυτό που πρέπει να κάνουμε είναι να πάμε στο Joomla, στο front end και να εισάγουμε το url. Δηλαδή εδώ:



Υπενθυμίζω πως υπάρχει link στο τέλος που λέει «Αν είστε καθηγητής και θέλετε να υποβάλλεται ερωτηματολόγιο πατήστε εδώ». Αφού πατήσουμε το link θα δούμε τη φόρμα εισαγωγής του ερωτηματολογίου.

Εισάγαμε τα στοιχεία στη φόρμα και αν όλα είναι οκ και δεν υπάρχει πρόβλημα θα μπορούμε να εισάγουμε τα δεδομένα στη βάση. Εδώ μας βγήκε το λάθος, γιατί υπάρχουν τα regular expressions τα οποία περιμένουν συγκεκριμένο url που αντιστοιχεί στο όνομα του server του ΤΕΙ που έχει στηθεί το limesurvey. Στη περίπτωση μας όμως δε πειράζει μιας και για ακαδημαϊκή επίδειξη μπορούμε να το παραβλέψουμε.

Έστω ότι το ερωτηματολόγιο έχει κατατεθεί. Ο administrator όταν θα ανοίξει την πλατφόρμα θα αντιμετωπίσει αυτή την εικόνα:

Ημερομηνία Καταχώρισης	Τμήμα	Μάθημα	Καθηγητής μαθήματος	URL ερωτηματολογίου	Κωδικός ερωτηματολογίου	Περιγραφή ερωτηματολογίου	Δημοσιευμένο
22/04/2009 - 12:24:34	Τμήμα Νοσηλευτικής	Διατροφή (Θεωρία)	Κώστας Γρεβενίτης	http://localhost/limesurvey/index.php?sid=13169&el	2636		

Joomla! είναι Ελεύθερο Λογισμικό και διατίθεται σύμφωνα με την Άδεια GNU/GPL.

Βλέπουμε πως έχουμε το ερωτηματολόγιο μας με όλα τα στοιχεία, εκτός της περιγραφής. Παρατηρούμε πως δεν είναι δημοσιευμένο το ερωτηματολόγιο, άρα και μη διαθέσιμο προς το τελικό χρήστη, δηλαδή τον φοιτητή. Για να το ενεργοποιήσουμε πατάμε πάνω στο κόκκινο x και γίνεται πράσινο v, όπως στην εικόνα παρακάτω.

Ημερομηνία Καταχώρισης	Τμήμα	Μάθημα	Καθηγητής μαθήματος	URL ερωτηματολογίου	Κωδικός ερωτηματολογίου	Περιγραφή ερωτηματολογίου	Δημοσιευμένο
22/04/2009 - 12:29:57	Τμήμα Νοσηλευτικής	Διατροφή (Θεωρία)	Κώστας Γρεβενίτης	http://localhost/limesurvey/index.php?sid=13169&el	2636		

Joomla! είναι Ελεύθερο Λογισμικό και διατίθεται σύμφωνα με την Άδεια GNU/GPL.

Τώρα είναι διαθέσιμο προς το τελικό χρήστη το ερωτηματολόγιο. Αλλά έστω ότι θέλουμε να κάνουμε μία μικρή αλλαγή στο όνομα του καθηγητή. Επιλέγουμε το ερωτηματολόγιο και πατάμε «Επεξεργασία». Η οθόνη που αντιμετωπίζουμε είναι η παρακάτω:

1.Σχολή
Σχολή Επαγγελματιών Υγείας και Πρόνοιας

2.Τμήμα
Τμήμα Νοσηλευτικής

3.Μάθημα και Καθηγητής Μαθήματος
Διατροφή (Θεωρία) Όνομα καθηγητή μαθήματος **Κωνσταντίνος Παλαιολόγος**

4.Ερωτηματολόγιο
URL ερωτηματολογίου: http://localhost/limesurvey/index.php?sid=13169&lang=el
Περιγραφή (δεν είναι υποχρεωτική η συμπλήρωσή της):

Δημοσιευμένο Όχι Ναι

Αφού κάναμε την αλλαγή στο όνομα πατάμε «Αποθήκευση» και επιστρέφουμε στην αρχική οθόνη όπου πλέον μας δείχνει τα προηγούμενα στοιχεία αλλά με την αλλαγή που έχουμε κάνει:

Ημερομηνία Καταχώρησης	Τμήμα	Μάθημα	Καθηγητής μαθήματος	URL ερωτηματολογίου	Κωδικός ερωτηματολογίου	Περιγραφή ερωτηματολογίου	Δημοσιευμένο
22/04/2009 - 12:44:26	Τμήμα Νοσηλευτικής	Διατροφή (Θεωρία)	Κωνσταντίνος Παλαιολόγος	http://localhost/limesurvey/index.php?sid=13169 (=el)	2636		✓

Εμφάνιση # 20 ▾

Τώρα θα πάμε στο front end και θα ακολουθήσουμε τη διαδικασία που πρέπει να κάνει ο φοιτητής για να συμπληρώσει το ερωτηματολόγιο. Αρχικά πάμε στο front end. Εκεί επιλέγουμε το tab «ΣΕΥΠ», ύστερα το «Τμήμα Νοσηλευτικής», και ερχόμαστε αντιμέτωποι με τα στοιχεία για το ερωτηματολόγιο:



Εάν πατήσουμε επάνω του θα έρθουμε αντιμέτωποι με μία πλατφόρμα ελέγχου. Εκεί ο φοιτητής θα βάλει τον κωδικό που θα του δωθεί εκείνη την ώρα. Ο κωδικός παράγεται αυτόματα από την PHP κατά την υποβολή του ερωτηματολογίου. Εάν δώσει σωστά τον κωδικό θα τον μεταφέρει στο ερωτηματολόγιο, αν τον δώσει λάθος θα τον επιστρέψει στην ίδια σελίδα. Επίσης το textbox είναι ρυθμισμένο έτσι ώστε να δέχεται μόνο 4 ψηφία.



Έχουμε εισάγει τον κωδικό. Εάν πατήσουμε το ερωτηματολόγιο, και είναι όλα σωστά τότε μας στέλνει στο ερωτηματολόγιο.

Δοκιμαστικό Ερωτηματολόγιο
Αυτό είναι ένα δοκιμαστικό ερωτηματολόγιο στα πλαίσια της πτυχιακής μου εργασίας

Υπάρχει μία ερώτηση σε αυτή το ερωτηματολόγιο

Σημείωση σχετικά με το προσωπικό απόρρητο
Το ερωτηματολόγιο αυτό είναι ανώνυμο.
Κατά τη συμμετοχή σας στην έρευνα δεν καταγράφει κανένα στοιχείο που να σας προσδιορίζει, πέραν των στοιχείων που πιθανώς δώσατε ως απάντηση σε κάποια ερώτηση. Αν χρησιμοποιήσατε άδεια αναγνώρισης για να έχετε πρόσβαση στη συμπλήρωση του ερωτηματολογίου, σας ενημερώνουμε πως η μόνη πληροφορία που καταγράφουμε για κάθε άδεια είναι αν έχει χρησιμοποιηθεί ή όχι. Δηλαδή, η άδεια αναγνώρισης σας δεν καταγράφει μαζί με την απάντησή σας. Συνεπώς είναι αδύνατο να συσχετιστεί η απάντησή σας με την άδεια που χρησιμοποιήσατε και κατ'επέκταση με το όνομα σας.

[\[Έξοδος από το ερωτηματολόγιο και διαγραφή απαντήσεων\]](#)

Κεφάλαιο 7^ο

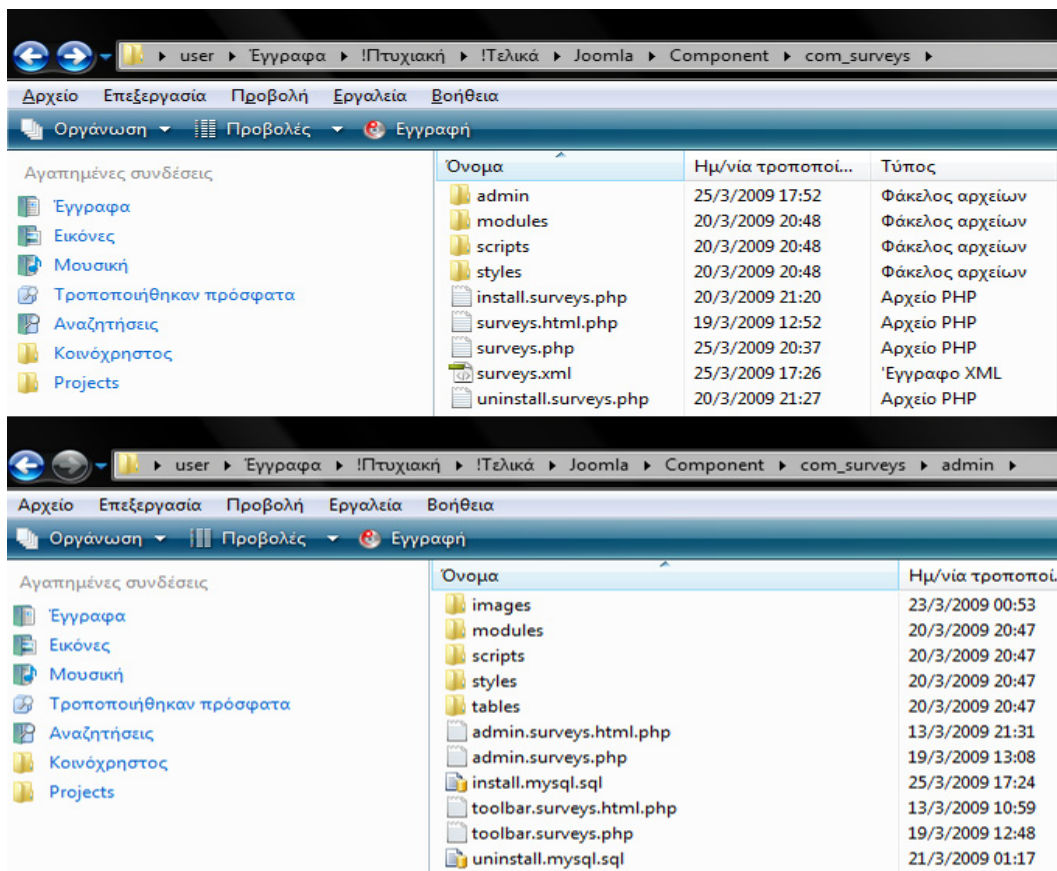
Packaging και Distribution

Εισαγωγή

Αφού ολοκληρώσαμε την ανάπτυξη της εφαρμογής και κάναμε τις απαραίτητες δοκιμές θα πρέπει να το κάνουμε ικανό να μπορεί να εγκατασταθεί στο Joomla. Όλα οι επεκτάσεις του Joomla (components, plug ins και modules), εγκαθίστανται στο Joomla μέσω συμπιεσμένων zip αρχείων. Ο Parser του Joomla «διαβάζει» τα περιεχόμενα του zip αρχείου και ξέρει τι πρέπει να κάνει. Η διαδικασία είναι αυτοποιημένη και ο χρήστης δε χρειάζεται να κάνει κάτι με το χέρι.

Packaging

Όταν πακετάρουμε ένα component πέρα από τα php χρειάζονται δύο sql αρχεία, ένα xml και δύο ειδικά αρχεία php, τα install.surveys.php και το uninstall.surveys.php.



Τα sql αρχεία χρειάζονται για την εγκατάσταση – απεγκατάσταση του component, αλλά και για την εγκατάσταση των πινάκων του component. Το xml αρχείο ουσιαστικά λειτουργεί ως

manifest file κατευθύνοντας τα αρχεία στο προορισμό τους, δηλαδή αν πρέπει να πάνε στο back end ή στο front end, όπως επίσης και άλλα στοιχεία πχ το όνομα του component που θα φαίνεται στη λίστα με αυτά και τα στοιχεία του δημιουργού. Το αρχείο πρέπει να έχει, σχεδόν πάντα το όνομα του com φακέλου που θα εγκατασταθεί το component μέσα στο joomla, είτε στο front end είτε στο back end, στη περίπτωση surveys. Ο κώδικας του xml αρχείου, με το όνομα surveys.xml είναι ο παρακάτω:

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

<install type="component" version="1.5.0">

<name>surveys</name>

<author>Κωνσταντίνος Γρεβενίτης AM 1123</author>
<creationDate>Νοέμβριος 2008 - Απρίλης 2009</creationDate>
<copyright>(C) 2008 - 2009</copyright>
<authorEmail></authorEmail>
<authorUrl></authorUrl>
<version>1.5.0</version>
<licence></licence>
<description>Component για την διαχείριση των ερωτηματολογίων, για την αξιολόγηση του
Νέου Νόμου Πλαισίου για τα 3βάθμια εκπαίδευση
</description>

<installfile>install.surveys.php</installfile>
<uninstallfile>uninstall.surveys.php</uninstallfile>

<install>
<sql>
<file driver="mysql" charset="utf8">install.mysql.sql</file>
</sql>
</install>

<uninstall>
<sql>
<file driver="mysql" charset="utf8">uninstall.mysql.sql</file>
</sql>
</uninstall>

<files>
<filename>surveys.php</filename>
<filename>surveys.html.php</filename>
<filename>modules/add.php</filename>
<filename>modules/deptAdd.php</filename>
<filename>modules/facultyAdd.php</filename>
<filename>modules/lessonsAdd.php</filename>
<filename>scripts/surveys.html.php.js</filename>
<filename>scripts/add.surveys.html.php.js</filename>
<filename>styles/add.surveys.html.php.css</filename>
<filename>styles/surveys.html.php.css</filename>
</files>
```

```
<administration>
<menu img="components/com_surveys/images/tei_icon.png">TEI Crete Surveys</menu>
<submenu>
<menu link="option=com_surveys">Questionnaires</menu>
<menu link="option=com_surveys&task=more">Options</menu>
<menu link="option=com_surveys&task=help">Help</menu>
</submenu>

<files folder="admin">
<filename>install.mysql.sql</filename>
<filename>uninstall.mysql.sql</filename>
<filename>admin.surveys.php</filename>
<filename>admin.surveys.html.php</filename>
<filename>toolbar.surveys.php</filename>
<filename>toolbar.surveys.html.php</filename>
<filename>images/tei_icon.png</filename>
<filename>images/tei_logo.png</filename>
<filename>modules/add.php</filename>
<filename>modules/edit.php</filename>
<filename>modules/facultyAdd.php</filename>
<filename>modules/facultyEdit.php</filename>
<filename>modules/deptAdd.php</filename>
<filename>modules/deptEdit.php</filename>
<filename>modules/lessonsAdd.php</filename>
<filename>modules/lessonsEdit.php</filename>
<filename>scripts/add.admin.surveys.html.js</filename>
<filename>scripts/edit.admin.surveys.html.js</filename>
<filename>scripts/help.admin.surveys.html.js</filename>
<filename>styles/add.admin.surveys.html.css</filename>
<filename>styles/edit.admin.surveys.html.css</filename>
<filename>tables/lessons.php</filename>
<filename>tables/questionnaire.php</filename>
<filename>tables/fad.php</filename>
<filename>tables/more.php</filename>
</files>

</administration>
</install>
```

Παρακάτω παραθέτουμε τον κώδικα από τα sql αρχεία με ονόματα install.mysql.sql και uninstall.mysql.sql, όπως πρέπει να είναι γραμμένος. Στο install αρχείο έχουμε βάλει και τη συνθήκη IF NOT EXISTS σε περίπτωση που έχει εγκατασταθεί παλαιότερα το component, οπότε έτσι δε θα χρειαστεί να εγκατασταθεί ξανά.

```
Install.mysql.sql

SET NAMES 'greek';

CREATE TABLE IF NOT EXISTS `#__survey_categories` (
  `category_id` int(11) NOT NULL AUTO_INCREMENT,
  `category_name` varchar(50) NOT NULL,
  `father_category` int(11) NOT NULL,
  PRIMARY KEY (`category_id`)
);

INSERT INTO `#__survey_categories` (`category_id`, `category_name`, `father_category`)
VALUES
(1, 'ΑΤΕΙ ΚΡΗΤΗΣ', 0),
(2, 'Σχολή Διοίκησης και Οικονομίας', 1),
(3, 'Σχολή Επαγγελματιών Υγείας και Πρόνοιας', 1),
(4, 'Σχολή Τεχνολογίας Γεωπονίας', 1),
(5, 'Σχολή Τεχνολογικών Εφαρμογών', 1),
(6, 'Τμήμα Διοίκησης Επιχειρήσεων', 2),
(7, 'Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων', 5),
(8, 'Τμήμα Ηλεκτρολογίας', 5),
(9, 'Τμήμα Θερμοκηπιακών Καλλιεργειών και Ανθοκομίας', 4),
(10, 'Τμήμα Κοινωνικής Εργασίας', 3),
(11, 'Τμήμα Λογιστικής', 2),
(12, 'Τμήμα Μηχανολογίας', 5),
(13, 'Τμήμα Νοσηλευτικής', 3),
(14, 'Τμήμα Πολιτικών Δομικών Έργων', 5),
(15, 'Τμήμα Τουριστικών Επιχειρήσεων', 2),
(16, 'Τμήμα Φυτικής Παραγωγής', 4);

CREATE TABLE IF NOT EXISTS `#__survey_lessons` (
  `lesson_id` int(11) NOT NULL AUTO_INCREMENT,
  `lesson_name` varchar(100) NOT NULL,
  `lesson_dept` int(11) NOT NULL,
  `lesson_semester` int(11) NOT NULL,
  PRIMARY KEY (`lesson_id`)
);

INSERT INTO `#__survey_lessons` (`lesson_id`, `lesson_name`, `lesson_dept`,
`lesson_semester`) VALUES
(1, 'Οικονομική Ι (Θεωρία)', 6, 1),
(2, 'Η/Υ Ι (Θεωρία)', 6, 1),
(3, 'Η/Υ Ι (Εργαστήριο)', 6, 1),
(4, 'Αρχές Διοίκησης (Θεωρία)', 6, 1),
(5, 'Οικονομικά Μαθηματικά (Θεωρία)', 6, 1),
....
(850, 'Ειδική Εντομολογία (Θεωρία)', 16, 7),
(851, 'Ειδική Εντομολογία (Εργαστήριο)', 16, 7),
(852, 'Μετασυλλ.Φυσ. καιΤεχν Οπωροκηπευτικών (Θεωρία)', 16, 7),
(853, 'Μετασυλλ.Φυσ.&Τεχν Οπωροκηπευτικών (Εργαστήριο)', 16, 7),
(854, 'Κοινή Αγροτική Πολιτική (Θεωρία)', 16, 7);
```

```
CREATE TABLE IF NOT EXISTS `#__survey_questionnaire` (  
  `questionnaire_id` int(11) NOT NULL AUTO_INCREMENT,  
  `questionnaire_lesson` int(11) NOT NULL,  
  `questionnaire_lesson_teacher` varchar(100) NOT NULL,  
  `questionnaire_description` text NOT NULL,  
  `questionnaire_url` varchar(150) NOT NULL,  
  `questionnaire_dept_that_belongs` int(11) NOT NULL,  
  `questionnaire_password` varchar(20) CHARACTER SET utf8 NOT NULL,  
  `questionnaire_timestamp` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON  
UPDATE CURRENT_TIMESTAMP,  
  `published` tinyint(1) unsigned NOT NULL DEFAULT '0',  
  PRIMARY KEY (`questionnaire_id`)  
);
```

```
CREATE TABLE IF NOT EXISTS `#__survey_more` (  
  `more_id` int(11) NOT NULL AUTO_INCREMENT,  
  `IsTeacherFrontEndFormAvailable` tinyint(4) NOT NULL,  
  PRIMARY KEY (`more_id`)  
);
```

```
INSERT INTO `#__survey_more` (`more_id`, `IsTeacherFrontEndFormAvailable`) VALUES  
(1, 1);
```

```
CREATE TABLE IF NOT EXISTS `#__survey_help` (  
  `Help_id` int(11) NOT NULL AUTO_INCREMENT,  
  `Help` text NOT NULL,  
  PRIMARY KEY (`Help_id`),  
  FULLTEXT KEY `Help` (`Help`)  
);
```

```
INSERT INTO `#__survey_help` (`Help_id`, `Help`) VALUES  
(1, '<h3>Ερωτηματολόγια</h3>\r\n<p>\r\nΓια να εισάγετε ένα νέο ερωτηματολόγιο θα  
πρέπει να επιλέξετε πρώτα το υπομενού "Ερωτηματολόγια". Ύστερα θα πατήσετε το κουμπί  
"Νέο" (αν είναι στα αγγλικά "New"), και θα σας εμφανιστεί η φόρμα εισαγωγής  
ερωτηματολογίου. Από εκεί και πέρα πραγματοποιείτε σταδιακές επιλογές, μέχρι την  
εισαγωγή του url του ερωτηματολογίου και της περιγραφής. \r\n</p>'),  
(2, '<h3>Επιπλέον Λειτουργίες</h3>\r\n<p>\r\nΠρος το παρόν η μοναδική λειτουργία που  
υπάρχει είναι η ενεργοποίηση / απενεργοποίηση της φόρμας εισαγωγής ερωτηματολογίου  
από τους καθηγητές στο front end.\r\n</p>');
```

Uninstall.mysql.sql

```
DROP TABLE #__survey_categories;  
DROP TABLE #__survey_lessons;  
DROP TABLE #__survey_questionnaire;
```

```
DROP TABLE #__survey_more;  
DROP TABLE #__survey_help;
```

Όσον αφορά τα `install.surveys.php`, `uninstall.surveys.php` τα αρχεία αυτά χρησιμοποιούνται για να εμφανίζουν μήνυμα μετά την επιτυχή εγκατάσταση ή απεγκατάσταση του component. Συγκεκριμένα όταν κάνουμε εγκατάσταση / απεγκατάσταση ένα component το Joomla εμφανίζει ένα τυποποιημένο μήνυμα χρησιμοποιώντας και την περιγραφή (<description>) που υπάρχει στο xml ως μέρος του μηνύματος. Ωστόσο μπορούμε να κάνουμε override αυτά τα μηνύματα και να χρησιμοποιήσουμε δικά μας κάνοντας χρήση αυτών των δύο php αρχείων. Ο πηγαίος κώδικας των δύο αυτών αρχείων είναι ο παρακάτω:

`install.surveys.php`

```
<?php  
defined('_JEXEC') or die('Απογορεύεται η πρόσβαση');  
  
function com_install()  
{  
    ?>  
    <div class="header">Καλώς ορισάτε στη πλατφόρμα διαχείρισης των  
ερωτηματολογίων αξιολόγησης του Νέου Νόμο  
    Πλαισίου</div>  
    <?php  
}  
?>
```

`uninstall.surveys.php`

```
<?php  
defined('_JEXEC') or die('Απαγορεύεται η πρόσβαση');  
  
function com_uninstall()  
{  
    ?>  
    <div class="header">Η εφαρμογή απεγκαταστάθηκε σωστά</div>  
  
<?php  
}  
?>
```

Joomla! θα φορτώσει το `install.reviews.php` στην εγκατάσταση και στην απεγκατάσταση το `reviews.php`. Η `com_install ()` συνάρτηση καλείται αφού η εγκατάσταση έχει ολοκληρωθεί, και σαν μήνυμα μπορούμε να γράψουμε ότι θέλουμε, από ένα μήνυμα καλωσορίσματος μέχρι ένα μικρό εγχειρίδιο. Ομοίως, η `com_uninstall ()` καλείται λίγο πριν το component απεγκατασταθεί πλήρως αλλά το μήνυμα θα εμφανιστεί αφού το component έχει απεγκατασταθεί πλήρως. Αν `com_install ()` ή `com_uninstall ()` επιστρέψουν `false`, η εγκατάσταση θα μπει σε rollback. Αυτό είναι χρήσιμο για την πρόληψη από εγκατάσταση components που δεν πληρούν τις ελάχιστες προϋποθέσεις.

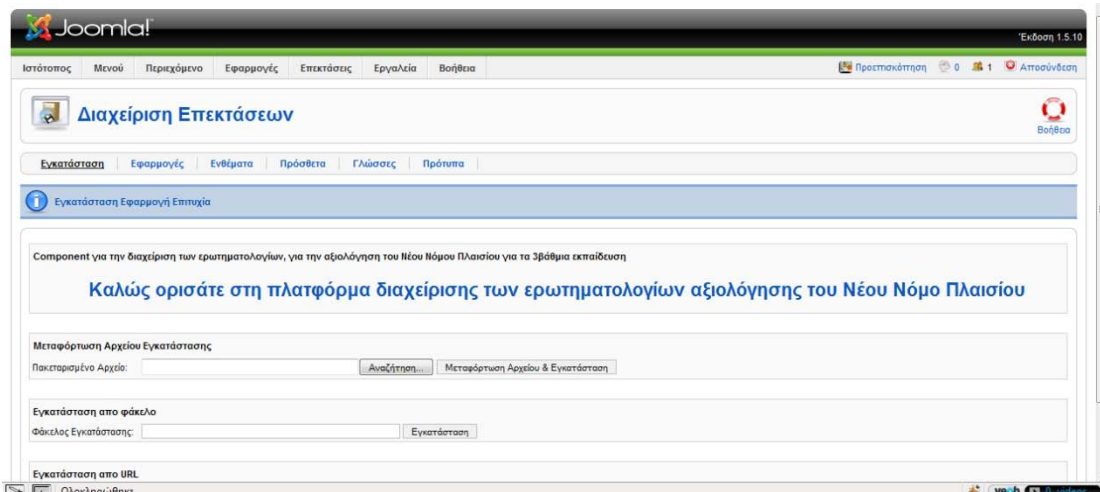
Distribution

Αφού ολοκληρώσουμε τις παραπάνω διαδικασίες συμπιέζουμε όλα τα αρχεία, με τη συγκεκριμένη δομή που παρουσιάσαμε παραπάνω, σε ένα zip αρχείο. Προσοχή υποχρεωτικά σε zip, γιατί rar δεν «διαβάζει» το Joomla.

Τέλος συμπιέζουμε όλα όσα φτιάξαμε σε ένα zip αρχείο, με όποιο όνομα θέλουμε. Για να το εγκαταστήσουμε πάμε «Επεκτάσεις» → «Εγκατάσταση / Απεγκατάσταση».



Μετά επιλέγουμε από το κουμπί «Αναζήτηση» το ζιπαρισμένο αρχείο και τέλος πατάμε «Μεταφόρτωση Αρχείου και Εγκατάσταση». Εάν εγκατασταθεί σωστά τότε θα έχουμε την παρακάτω εικόνα.



Κεφάλαιο 8^ο

Συμπεράσματα

Η ανάπτυξη μίας on line εφαρμογής η οποία την διαχείριση urls ηλεκτρονικών ερωτηματολογίων και την ανάρτηση αυτών, για συμπλήρωση, είναι γεγονός

Το component, ως μέρος του Joomla, θα «βρίσκεται» στο διαδίκτυο, όπου εκεί υπάρχει ένας διαχωρισμός σε front end και σε back end. Στο back end έχει πρόσβαση ο διαχειριστής του συστήματος, ενώ το front end χωρίζεται σε 2 μέρη, στο ένα είναι ταξινομημένα τα ερωτηματολόγια ανά σχολή, τμήμα και εξάμηνο προς ανάκτηση και συμπλήρωση από τους φοιτητές, ενώ στο δεύτερο θα υπάρχει μία φόρμα υποβολής του ερωτηματολογίου από τον εκάστοτε καθηγητή του μαθήματος που έχει φτιάξει το ερωτηματολόγιο.

Μετά την ολοκλήρωση της πτυχιακής τα συμπεράσματα στο οποία θα μπορούσε να φτάσει κάποιος είναι πολλά. Ξεκινώντας από την ίδια την ίδια την εφαρμογή θα μπορούσαμε να πούμε πως το καλύτερο είναι να υπάρχει μία ενιαία πλατφόρμα, ένα ενιαίο μεγάλο CMS, όπου μέσα από αυτό θα μπορούσε κάποιος να αναπτύξει και ερωτηματολόγια, αλλά και να τα χειριστεί αναλόγως. Ωστόσο για να γίνει αυτό στα πλαίσια μίας πτυχιακής, είναι αν όχι ακατόρθωτο, πάντως πάρα πολύ δύσκολο.

Έτσι προς το παρόν προτιμήθηκαν δύο έτοιμες πλατφόρμες, το Joomla και το Lime Surveys και αναπτύξαμε μία «επικοινωνία» μεταξύ τους. Βέβαια, σαν εμπειρία και γνώση είναι θετικό να ξέρεις να φτιάχνεις component για το Joomla. Θεωρείται προσόν στην αγορά εργασίας. Επιπλέον είναι «πόντοι» εμπειρίας για τις προγραμματιστικές δυνατότητες.

Τέλος, όσον αφορά το ίδιο το component έχει πάρα πολλά περιθώρια βελτίωσης.

Μερικά από αυτά είναι τα εξής:

1. Να γίνει το component σε MVC
2. Θα μπορούσε να προστεθεί Ajax
3. Ίσως κάποια αισθητική βελτίωση του front end, ουσιαστικά ένα νέο template για το Joomla

Βιβλιογραφία

- [1]. Learning Joomla 1.5 Extension Development – Joseph LeBlanc
- [2]. <http://en.wikipedia.org>
- [3]. <http://www.w3c.com>
- [4]. <http://www.limesurvey.org/>
- [5]. <http://www.joomla.org>

Ευρετήριο

A

Arpanet, σελ. 8
apache server, σελ 30
aptana ide, σελ 30
authoring, σελ 25

C

cms, σελ 23
component, σελ 34

D

darpa, σελ 10
distribution, σελ 128

I

Internet, σελ 8

J

Joomla, σελ 33

L

Limesurvey, σελ 33

M

management, σελ 25
metadata, σελ 28
mysql, σελ 29

P

packaging, σελ 122
personalization, σελ 27
php, σελ 28
publication system, σελ 26

S

scalability, σελ 28

W

workflow automation, σελ 26
world wide web, σελ 8