ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΧΝΟΛΟΓΙΚΟ ΙΔΡΥΜΑ ΘΕΣΣΑΛΟΝΙΚΗΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ - ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Εφαρμογή Διαχείρισης Βιβλιογραφικών Αναφορών

Του Φοιτητή:

Κωνσταντίνου Θεοδώρου

Αρ. Μητρώου: 03/2178

Επιβλέπων Καθηγητής:

Φώτης Κόκκορας

Θεσσαλονίκη 2009

Πρόλογος

Η παρούσα πτυχιακή εργασία ασχολείται με το πρόβλημα της εύρεσης και διαχείρισης βιβλιογραφικών αναφορών. Παρόλο που στις μέρες μας υπάρχουν εξειδικευμένα συστήματα και μηχανές αναζήτησης βιβλιογραφικών αναφορών όπως τα Google Scholar και Citeseer^{x}, η έλλειψη ενός κοινού προτύπου κωδικοποίησης βιβλιογραφικών αναφορών, η ελλιπής ακρίβεια των παραπάνω συστημάτων καθώς και εγγενή προβλήματα του τομέα, κάνουν την όλη διαδικασία προβληματική, χρονοβόρα και κοπιαστική.

Για το λόγο αυτό αναπτύχθηκε μία διαδικτυακή εφαρμογή που αξιοποιώντας τις υπάρχουσες υποδομές αναζήτησης βοηθά στην καλύτερη διαχείριση των αποτελεσμάτων. Ειδικότερα, αναπτύχθηκε μια web εφαρμογή που χρησιμοποιώντας τεχνικές εξαγωγής περιεχομένου από τον παγκόσμιο ιστό (web content extraction) συγκεντρώνει σε δομημένη μορφή αποτελέσματα αναζήτησης που επιστρέφει η υπηρεσία Google Scholar και παρέχει κατάλληλα εργαλεία για την διαχείριση των αποτελεσμάτων. Το σύστημα που αναπτύχθηκε βοηθά έναν ακαδημαϊκό ή ερευνητή να έχει καλή εικόνα της εξέλιξης των αναφορών που κάνουν τρίτοι σε δικές του εργασίες και σταδιακά να αφιερώνει όλο και λιγότερο χρόνο για να διατηρεί ενημερωμένη την προσωπική του λίστα αναφορών.

Η παρούσα πτυχιακή εργασία εκπονήθηκε στο πλαίσιο του σχετικού μαθήματος του Τμήματος Πληροφορικής, της σχολής Τεχνολογικών Εφαρμογών του Αλεξάνδρειου Τεχνολογικού Ιδρύματος Θεσσαλονίκης, υπό την επίβλεψη του κ. Φώτη Κόκκορα (Επιστημονικός Συνεργάτης του ΤΕΙ).

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή μου για την καθοδήγησή του, τόσο κατά τη διάρκεια της ανάπτυξης της εφαρμογής όσο και κατά τη συγγραφή της πτυχιακής. Επίσης θα ήθελα να ευχαριστήσω τον Mathijs Soeters, επιβλέποντά μου στη Πρακτική μου Άσκηση και προγραμματιστή της Alexion Software στο Groningen της Ολλανδίας, για τις χρήσιμες συμβουλές του σε θέματα web προγραμματισμού. Το αντικείμενο ενασχόλησής μου, κατά τη

διάρκεια παραμονής μου στην Alexion Software, αποτέλεσε και αφορμή για την ανάπτυξη της παρούσας εφαρμογής σε διαδικτυακή μορφή.

Κωνσταντίνος Θεοδώρου

20/11/2009

Περίληψη

Οι βιβλιογραφικές αναφορές (citations) χρησιμοποιούνται εκτενώς από τους ακαδημαϊκούς και τους ερευνητές ως μέτρο της απήχησης που έχουν στην υπόλοιπη επιστημονική κοινότητα οι ερευνητικές/επιστημονικές τους εργασίες.

Στη παρούσα πτυχιακή εργασία γίνεται καταγραφή των διαθέσιμων τρόπων που έχει κάποιος για να εντοπίζει βιβλιογραφικές αναφορές και των προβλημάτων που αντιμετωπίζει. Ειδικότερα γίνεται αναφορά σε εξειδικευμένες μηχανές αναζήτησης όπως οι Citeseer, Google Scholar, κτλ καθώς και σε προβλήματα όπως τα matching citation, mixed citation και split citation.

Με δεδομένη την εκτενή χρήση του παγκόσμιου ιστού για τον εντοπισμό αναφορών, τον υψηλό φόρτο εργασίας που απαιτεί αυτή η προσέγγιση και ελλείψει ενός μοναδικού τρόπου κωδικοποίησης της σχετικής με μια αναφορά πληροφορίας, η παρούσα πτυχιακή προτείνει και υλοποιεί μια διαδικτυακή εφαρμογή εντοπισμού και διαχείρισης βιβλιογραφικών αναφορών.

Η εφαρμογή αυτή, με τη βοήθεια ενός συστήματος εξαγωγής περιεχομένου από τον παγκόσμιο ιστό (ΔΕiΧΤο), συγκεντρώνει σε δομημένη μορφή (XML) τα αποτελέσματα της αναζήτησης μέσω Google Scholar, τα προεπεξεργάζεται με διάφορους τρόπους ώστε να μειώσει το φόρτο του χρήστη και τα καταχωρεί σε βάση δεδομένων, παρέχοντας εύχρηστη γραφική διεπαφή για την περαιτέρω διαχείρισή τους. Η εφαρμογή είναι πλήρως διαδικτυακή και αναπτύχθηκε σε γλώσσα PHP με χρήση mySQL.

Η εργασία περιλαμβάνει και συγκριτική μελέτη περίπτωσης που αναδεικνύει τα πλεονεκτήματα της όλης προσέγγισης.

Abstract

The bibliographic references (citations) are used extensively by academics and researchers in order to determine how popular their work is among the scientific community.

This thesis presents the approaches used by someone who is interested in finding bibliographic references and the difficulties that he or she may encounter. More specifically, various specialized search engines such as Citeseer and Google Scholar are presented and problems such as the matching citation problem, the mixed citation problem and the split citation problem are mentioned and explained.

Given the extensive use of the WWW as a means to indentify references, the high workload this approach requires and the absence of a unique encoding scheme for citations, this thesis proposes and implements a web application to support the management of scientific citations.

The application developed uses a web content extraction system (DEiXTo), to aggregate search results from Google Scholar in a structured way (XML), filters them in various ways to reduce the user effort and inserts them in a database. Then it provides a user friendly graphical user interface for further citation management. The application is web based and was developed with PHP and mySQL.

This thesis also includes a comparative case study which proves the benefits of the approach used.

Περιεχόμενα

1 Εισαγωγή

Με την εξέλιξη της πληροφορικής, η αναζήτηση πληροφορίας έχει εξελιχθεί σε τέτοιο βαθμό που αποτελεί κομμάτι της καθημερινότητας μας. Εκατομμύρια άνθρωποι ανά το κόσμο, χρησιμοποιούν τις μηχανές αναζήτησης, για ερευνητικούς, εργασιακούς, ψυχαγωγικούς, ενημερωτικούς και άλλους λόγους.

Η όσο το δυνατόν καλύτερη αναζήτηση παίζει σημαντικό ρόλο όχι μόνο για την επιστημονική κοινότητα, που προσπαθεί συνεχώς να την εξελίσσει, αλλά και για τις επιχειρήσεις και τα τμήματα marketing που βλέπουν τις μηχανές αναζήτησης σαν σύμμαχο στις πωλήσεις τους. Από αυτή την εξέλιξη της αναζήτησης δε θα μπορούσε να λείπει φυσικά, η ερευνητική κοινότητα.

Μελετητές και ερευνητές απ' όλο το κόσμο χρησιμοποιούν καθημερινά τις μηχανές αναζήτησης για να εντοπίζουν τι τελευταίες εξελίξεις στο τομέα της τεχνολογίας ή για να συλλέξουν πληροφορίες, οποιουδήποτε περιεχομένου προκειμένου να τις χρησιμοποιήσουν στις εργασίες τους. Η εξέλιξη των μεθόδων αναζήτησης και η ευκολία δημοσίευσης στον παγκόσμιο ιστό είχε σαν αποτέλεσμα να αυξηθούν και οι βιβλιογραφικές αναφορές.

Οι βιβλιογραφικές αναφορές, ανέκαθεν έπαιζαν σημαντικό ρόλο για τους αναγνώστες, γιατί του επιτρέπει να μελετούν εκτενέστερα διάφορες πληροφορίες που τους ενδιαφέρουν, αλλά και για τους συγγραφείς γιατί όσο περισσότερες αναφορές γίνονται στα έργα τους τόσο πιο μεγάλη θεωρείται η επιρροή τους στην παγκόσμια επιστημονική κοινότητα. Για το λόγο αυτό προσπάθησαν, με διάφορες εμπειρικές μεθόδους αρχικά και με εξειδικευμένες στις βιβλιογραφικές αναφορές μηχανές αναζήτησης όπως το DBLP και το Google Scholar στη συνέχεια, να συλλέξουν αυτές τις πληροφορίες.

Παρόλα τα συστήματα που αναπτύχθηκαν για την αναζήτηση βιβλιογραφικών αναφορών, η αναζήτηση χρόνο με το χρόνο γίνεται όλο και πιο δύσκολη, καθώς η βιβλιογραφία αυξάνει όλο και περισσότερο, με αποτέλεσμα να απαιτεί αρκετό χρόνο και προσωπική δουλειά από τους συγγραφείς. Σε περίπτωση μάλιστα που ένας συγγραφέας έχει πολυετές έργο και έχει δημοσιεύσει δεκάδες ή

και εκατοντάδες εργασίες, οι βιβλιογραφικές αναφορές στις εργασίες του μπορεί να φτάνουν τις μερικές χιλιάδες.

Αυτόν ακριβώς το φόρτο έρχεται να απαλείψει η εφαρμογή που αναπτύχθηκε για τους σκοπούς της παρούσας πτυχιακής εργασίας. Πρόκειται για μία διαδικτυακή εφαρμογή που συλλέγει τα αποτελέσματα από την αναζήτηση του Google Scholar, με τη βοήθεια του DEiXTo, ενός εργαλείου εξαγωγής δεδομένων από ιστοσελίδες. Στη συνέχεια, τα αποθηκεύει σε μία βάση δεδομένων και δίνει τη δυνατότητα στο χρήστη να τα διαχειριστεί με κατάλληλο τρόπο ούτως ώστε σταδιακά να του κάνει ευκολότερη την διαδικασία τήρησης ενός ενημερωμένου αρχείου αναφορών. Επιπλέον, δίνει τη δυνατότητα στο χρήστη να εισάγει όλες τις πληροφορίες από νέες δημοσιεύσεις του και βιβλιογραφικές αναφορές και να παρουσιάζει όλες τις πληροφορίες σε σελίδες αναφορών.

Το υπόλοιπο κείμενο της πτυχιακής διαμορφώνεται ως εξής: Στο Κεφάλαιο 2 περιγράφονται οι λόγοι που καθιστούν σημαντικές τις βιβλιογραφικές αναφορές και περιγράφεται μια τυπική μεθοδολογία αναζήτησης και καταγραφής βιβλιογραφικών αναφορών με τη χρήση μιας απλής μηχανής αναζήτησης όπως είναι το Google. Επιπρόσθετα, παρουσιάζονται οι δυνατότητες μερικών από τις πιο γνωστές εξειδικευμένες μηχανές αναζήτησης βιβλιογραφικών αναφορών. Τέλος, αναλύονται τα βασικότερα προβλήματα που παρουσιάζονται στην αναζήτηση βιβλιογραφικών αναφορών (Citation Matching Problem, Mixed Citation Problem, Split Citation Problem) και αναλύονται κάποιες λύσεις που έχουν προταθεί για αυτά τα προβλήματα.

Στο Κεφάλαιο 3 εξηγούνται οι λόγοι που οδήγησαν στην παρούσα εφαρμογή και περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν (γλώσσες προγραμματισμού, εργαλεία συλλογής δεδομένων) καθώς η μέριμνα που λήφθηκε για τη σωστή λειτουργία της εφαρμογής σε διάφορους web browsers.

Στο Κεφάλαιο 4 γίνεται μια περιεκτική περιγραφή της εφαρμογής που αναπτύχθηκε. Αρχικά περιγράφεται η λειτουργία της εφαρμογής και ο τρόπος χρήσης της, ενώ στη συνέχεια παρουσιάζεται η αρχιτεκτονική της εφαρμογής και η βάση δεδομένων που την υποστηρίζει, ενώ αναλύονται και τα σημαντικότερα δομικά της στοιχεία, σε επίπεδο κώδικα.

Στο Κεφάλαιο 5 γίνεται μία σύγκριση της εύρεσης βιβλιογραφικών αναφορών με τη χρήση μόνο του Google Scholar σε αντιπαράθεση με το συνδυασμό

του Google Scholar και της εφαρμογής που αναπτύχθηκε. Η σύγκριση γίνεται σε ένα δείγμα 10 αποτελεσμάτων και αναφέρονται οι χρόνοι που χρειάζεται ο χρήστης για να αναγνωρίσει και να διαχειριστεί τις βιβλιογραφικές αναφορές.

Το Κεφάλαιο 6 συνοψίζει την πτυχιακή καταγράφοντας τα συμπεράσματα που προέκυψαν και κάνοντας προτάσεις για πιθανές βελτιώσεις και επεκτάσεις της εφαρμογής.

2 Βιβλιογραφικές Αναφορές

Ο ρόλος των βιβλιογραφικών αναφορών στην επιστημονική κοινότητα είναι πολύ σημαντικός. Η συλλογή τους από τους διάφορους συγγραφείς κρίνεται απαραίτητη, καθώς κατ' αυτό τον τρόπο μπορούν να έχουν μια εικόνα για τον αντίκτυπο των εργασιών τους, στο κλάδο τους. Για το σκοπό αυτό αναπτύχθηκαν διάφορα μέσα όπως, εξειδικευμένες μηχανές αναζήτησης και ψηφιακές βιβλιοθήκες. Παρόλα αυτά τα προβλήματα αναζήτησης βιβλιογραφικών αναφορών δεν έπαψαν να εμφανίζονται. Βέβαια, διάφοροι μέθοδοι έχουν αναπτυχθεί για την επίλυση των προβλημάτων αυτών για τη βελτίωση της διαδικασίας αυτής.

2.1 Βασικές Έννοιες

Όταν μιλάμε για βιβλιογραφικές αναφορές, αναφερόμαστε στη χρησιμοποίηση κατά λέξη κειμένου ή αρκετές φορές τροποποιημένου κειμένου, από τη βιβλιογραφία ή τις πηγές. Η βιβλιογραφία μπορεί να αναφέρεται σε δημοσιευμένο υλικό, βιβλία, άρθρα, πρακτικά συνεδρίων έως και ιστοσελίδες. Η ύπαρξή τους εξυπηρετεί κυρίως τη παραπομπή του αναγνώστη στις διάφορες πηγές, ούτως ώστε σε περίπτωση που αυτός ενδιαφέρεται για περαιτέρω πληροφορίες πάνω σε συγκεκριμένο θέμα, να οδηγείται μέσω αυτών στο αυθεντικό κείμενο [6].

Η συλλογή των αναφορών, ωφελεί κυρίως την επιστημονική κοινότητα. Κατά αυτό τον τρόπο μπορεί ο συγγραφέας να διαπιστώσει το μέγεθος της επιρροής του στα υπόλοιπα μέλη της επιστημονικής κοινότητας. Κατά αναλογία λοιπόν, όσες περισσότερες αναφορές γίνονται σε συγγραφικό υλικό ενός συγγραφέα τόσο πιο αναγνωρισμένος θεωρείται στο κλάδο του. Η επιρροή αυτή ονομάζεται ως h-index και πρόκειται για μία αριθμητική τιμή. Το h-index δεν μετρά μόνο την αναγνώριση ενός επιστήμονα από την παγκόσμια επιστημονική κοινότητα, αλλά κατατάσσει επίσης, σχολές ή και πανεπιστήμια. Ο σωστός υπολογισμός του είναι μια δύσκολη διαδικασία γιατί απαιτεί πληροφορία που είναι δύσκολο να συλλεχθεί. Τελευταία, αυτό μπορεί να γίνει ευκολότερα με τη βοήθεια μηχανών αναζήτησης όπως το Google Scholar και βάσεων δεδομένων όπως το Scopus.

Παρόλα αυτά η διαδικασία υπολογισμού δε μπορεί να στηριχθεί μόνο στα αποτελέσματα μιας μηχανής αναζήτησης, για το λόγο ότι μπορεί να δίνει μεγαλύτερη βαρύτητα σε κάποιο συγκεκριμένο είδος δημοσίευσης, όπως σε άρθρα ή σε έγγραφα συνεδρίων. Απαιτείται λοιπόν, αρκετή δουλειά από τον ενδιαφερόμενο, για να συλλέξει και να ξεκαθαρίσει τις πληροφορίες και τελικά να καταφέρει να υπολογίσει, το προσωπικό του h-index. Ακόμη όμως και με αυτή τη διαδικασία δεν αντιμετωπίζεται πλήρως το πρόβλημα καθώς με την πάροδο του χρόνου οι αναφορές πληθαίνουν, τα αποτελέσματα του Google Scholar γίνονται περισσότερα με αποτέλεσμα να απαιτείται συνεχής ενημέρωση [4].

Κατά το παρελθόν, η αναζήτηση και η διαχείριση των βιβλιογραφικών αναφορών μόνο εύκολη υπόθεση δεν ήταν. Χωρίς τη βοήθεια των μηχανών αναζήτησης και του internet ένα τέτοιο εγχείρημα φάνταζε από δύσκολο ως αδύνατο. Αν και στην εποχή μας και με τη ραγδαία εξέλιξη της τεχνολογίας, οι μηχανές αναζήτησης και οι υπηρεσίες του internet έχουν φτάσει σε τέτοιο βαθμό που να καθιστούν δυνατή αυτή τη διαδικασία, νέα προβλήματα παρουσιάζονται και κάνουν τη συλλογή των βιβλιογραφικών αναφορών δύσκολη και κυρίως χρονοβόρα.

2.2 Απλές Μηχανές Αναζήτησης

Ο ποιο εύκολος τρόπος για να βρει κάποιος μια πληροφορία οποιουδήποτε περιεχομένου, είναι να χρησιμοποιήσει μια μηχανή αναζήτησης, όπως το δημοφιλές Google. Αυτός θα ήταν και ο τρόπος για έναν ακαδημαϊκό να εντοπίσει βιβλιογραφικές αναφορές πάνω σε δημοσιεύσεις του. Αν προσπαθήσουμε να αναλύσουμε τα βήματα που θα ακολουθούσε κάποιος συγγραφέας για να εντοπίσει τις βιβλιογραφικές αναφορές που έχουν γίνει στις εργασίες του, θα δούμε για πόσο δύσκολη πρόκειται.

Αρχικά ο συγγραφέας κάνει αναζήτηση με κλειδί αναζήτησης το επώνυμο του. Το πιο πιθανό είναι στη πρώτη σελίδα να εμφανιστούν κάποιες εργασίες του, ίσως η προσωπική του ιστοσελίδα, κάποιες ιστοσελίδες όπου υπάρχει το ονοματεπώνυμο του ακόμη και δημοσιεύσεις άλλου συγγραφέα σε περίπτωση συνωνυμίας. Μία καλύτερη λύση είναι να φτιάξει μία λίστα με όλες του τις δημοσιευμένες εργασίες και να ξαναπροσπαθήσει προσθέτοντας στο όνομά του, μέρος από το τίτλο της κάθε εργασίας ή και το επώνυμο ενός συνεργάτη του, αν

υπάρχει συνεργάτης. Ένα ποσοστό αυτών των αποτελεσμάτων θα πρόκειται για βιβλιογραφικές αναφορές σε δημοσιεύσεις του, αλλά ένα μεγάλο ποσοστό θα πρόκειται για μη χρήσιμη πληροφορία. Σε περίπτωση μη σχετικών με εργασίες, αποτελεσμάτων ο συγγραφέας θα μπορεί να τις αναγνωρίσει και να τις προσπεράσει εύκολα. Ακόμη και σε περίπτωση που η μη χρήσιμη πληροφορία αναφέρετε σε κάποια εργασία του ο συγγραφέας θα μπορεί να τη ξεδιαλύνει χωρίς μεγάλη δυσκολία, μόνο που θα χρειαστεί να σπαταλήσει χρόνο για να ανοίξει τους συνδέσμους και να διαπιστώσει το περιεχόμενο της πληροφορίας.

Αφού ο συγγραφέας συλλέξει, για κάθε εργασία του, τις βιβλιογραφικές αναφορές που θα βρει, θα πρέπει να τις καταγράψει ή να τις αποθηκεύσει σε κάποια βάση δεδομένων ή σε κάποια μορφή εγγράφου. Σε κάθε περίπτωση, αν ο συγγραφέας προσπαθήσει να ελέγξει τυχόν αναφορές στις δημοσιεύσεις του, θα χρειαστεί να σπαταλήσει, στην ιδανική περίπτωση, τον ίδιο χρόνο που σπατάλησε και στην αναζήτηση της πληροφορίας. Σε μη ιδανική περίπτωση ο συγγραφέας θα χρειαστεί να σπαταλήσει επιπλέον χρόνο για να αποκλείσει αναφορές που έχει ήδη καταγράψει. Η εργασία του εντοπισμού αναφορών που έχει ήδη καταγράψει, θα γίνεται όλο και πιο συχνή σε κάθε νέα αναζήτηση του συγγραφέα. Είναι επίσης πιθανό να υπάρξει κάποια σύγχυση με τα αποτελέσματα καθώς είναι σχεδόν αδύνατο, ύστερα από κάποιο χρονικό διάστημα, ο συγγραφέας να θυμάται αν ένα αποτέλεσμα πρόκειται για αναφορά ή μη, ή εάν το έχει αποκλείσει κατά το παρελθόν ή όχι. Αυτός ο έλεγχος επιβαρύνει ακόμη περισσότερο αυτή τη διαδικασία, με το συγγραφέα να ανατρέχει συνεχώς σε προηγούμενες σημειώσεις του για να διαπιστώσει την ύπαρξη των αναφορών ή τις αλλαγές σε αυτές.

Συνοψίζοντας αυτή τη μέθοδο αναζήτησης, ο συγγραφέας θα αναζητήσει αναφορές χρησιμοποιώντας κατάλληλα κλειδιά. Στη συνέχεια θα αποκλείσει όλα τα αποτελέσματα που από το σύνδεσμο τους ή από τη γενική περιγραφή τους, που εμφανίζεται στη σελίδα της μηχανής αναζήτησης, μπορεί εύκολα να κρίνει ότι δεν πρόκειται για σχετική πληροφορία. Τέλος θα ελέγξει τους συνδέσμους πιθανών βιβλιογραφικών αναφορών (όσων αποτελεσμάτων απομένουν), θα αποκλείσει τα μη σχετικά αποτελέσματα και θα καταγράψει όλες τις πραγματικές βιβλιογραφικές αναφορές.

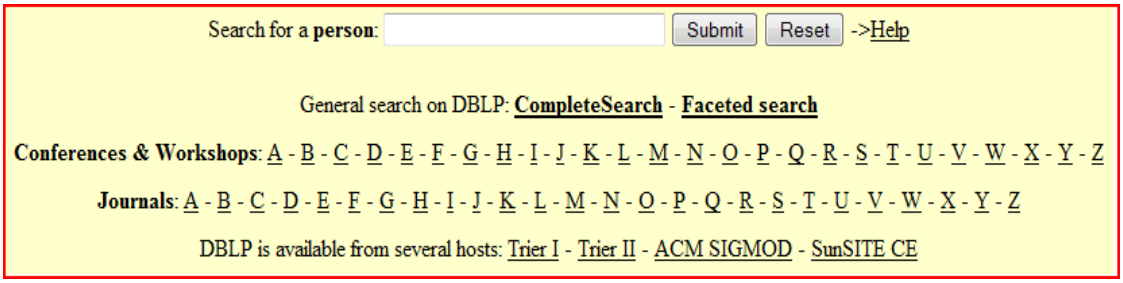
2.3 Εξειδικευμένες Μηχανές Αναζήτησης

Η αναζήτηση βιβλιογραφικών αναφορών, έστω και με τη βοήθεια μηχανών αναζήτησης είναι μία αρκετά δύσκολη διαδικασία. Για το λόγο αυτό αναπτύχθηκαν κάποιες εξειδικευμένες μηχανές αναζήτησης που επικεντρώνονται στις βιβλιογραφικές αναφορές και μειώνουν κατά πολύ το βαθμό δυσκολίας εύρεσης τους. Σε αυτή τη παράγραφο θα γίνει μία παρουσίαση αυτών των μηχανών αναζήτησης και των δυνατοτήτων τους.

2.3.1 DBLP

Το DBLP (Digital Bibliography & Library Project, http://dblp.uni-trier.de/) δημιουργήθηκε τη δεκαετία του '80 από το πανεπιστήμιο Trier της Γερμανίας και ξεκίνησε αρχικά σαν μία βάση δεδομένων και μία ιστοσελίδα όπου αποθηκεύονταν βιβλιογραφίες σχετικές με το λογικό προγραμματισμό. Εκείνη τη περίοδο DBLP σήμαινε (DataBase systems & Logic Programming). Βέβαια με τα χρόνια η λίστα με τις βιβλιογραφίες έγινε «βιβλιοθήκη» και σήμερα διαθέτει πάνω από 1,2 εκατομμύρια άρθρα πάνω στην επιστήμη των υπολογιστών.

Κατά την αναζήτηση ό χρήστης μπορεί να εισάγει το επώνυμο, του προς αναζήτηση συγγραφέα ή να περιηγηθεί μέσα από τις ταξινομημένες σε αλφαβητική σειρά λίστες, ανάλογα με το πρώτο γράμμα του τίτλου της εργασίας, βλ. εικόνα 2-1.



Εικόνα 2-1 μηχανή αναζήτησης του DBLP [14]

Μετά από την εισαγωγή του ονόματος ενός συγγραφέα, παρατίθενται οι εργασίες του, κατά φθίνουσα χρονολογική σειρά. Σε κάθε εργασία παρουσιάζονται τα εξής:

- Η λίστα με τα ονόματα των συγγραφέων, σε μορφή συνδέσμων ούτως ώστε εάν ο χρήστης το επιθυμεί να οδηγηθεί στην αντίστοιχη λίστα κά-

ποιου άλλου συγγραφέα ανοίγοντας το σύνδεσμο του συγκεκριμένου συγγραφέα.

- Ένας κωδικός – ονομασία μαζί με τη χρονολογία του εγγράφου, σε μορφή συνδέσμου, που έχει αναφερθεί στη συγκεκριμένη εργασία και οι σελίδες στις οποίες έγινε η αναφορά στην εργασία.
- Μία λίστα από συνδέσμους στην αριστερή πλευρά της εγγραφής που οδηγούν στις εξής ιστοσελίδες:
    - Σε μία ιστοσελίδα όπου κάποιος μπορεί να βρει την εργασία ή το βιβλίο που αποτελεί τη δημοσίευση του συγγραφέα και συνήθως δίνεται και η δυνατότητα να την/το αγοράσει.
    - Στο http://www.pubzone.org , ένα διαδικτυακό forum που αφορά επιστημονικές εκδόσεις.
    - Στη σελίδα αποτελεσμάτων αναζήτησης του CiteSeer, μιας μηχανής αναζήτησης βιβλιογραφικών αναφορών που περιγράφεται στη συνέχεια.
    - Στη σελίδα αποτελεσμάτων του Google Scholar, μιας μηχανής αναζήτησης βιβλιογραφικών αναφορών που επίσης περιγράφεται στη συνέχεια.
    - Σε μια σελίδα όπου εμφανίζει την εγγραφή σε μορφή BibTeX, ένός προτύπου για την ομαδοποίηση βιβλιογραφικών αναφορών.
    - Και τέλος σε μία σελίδα όπου παρουσιάζεται η εγγραφή στην XML μορφή της.

Εικόνα 2-2 Εμφάνιση αποτελεσμάτων DBLP [14]

Επιπρόσθετα, στο τέλος της σελίδας των αποτελεσμάτων παρουσιάζεται μια λίστα με τα ονόματα των υπολοίπων συγγραφέων και δίπλα σε κάθε όνομα τον αριθμό της εγγραφής, που αντιστοιχεί στην εργασία, που έχουν συμμετάσχει.

1	Walid G. Aref	[3]
2	Nick Bassiliades (N. Bassiliades)	[4] [5] [6]
3	Ahmed K. Elmagarmid	[3]
4	Elias N. Houstis	[3]
5	Haitao Jiang	[3]
6	Efstratios Kontopoulos	[6]
7	Efstratia Lampridou	[7]
8	Konstantinos Ntonas	[7]
9	Demetrios G. Sampson	[2]
10	Ioannis P. Vlahavas	[1] [2] [3] [4] [5] [6] [7]
11	Dimitris Vrakas	[4] [6]

Εικόνα 2-3 Λίστα υπολοίπων συγγραφέων [14]

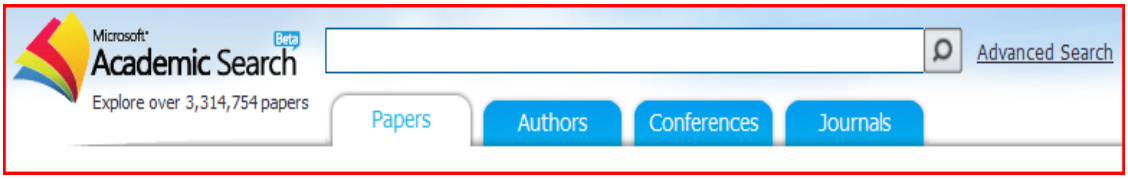
Γενικά η DBLP διαθέτει ένα αρκετά δύσχρηστο περιβάλλον εργασίας, εμφανίζοντας τα δεδομένα με πεπαλαιωμένο τρόπο, χρησιμοποιώντας απλούς πίνακες HTML και λίστες, με σχεδόν καθόλου γραφικά. Παρόλα αυτά δε παύει να είναι μια εξαιρετική βάση δεδομένων που αυξάνει συνεχώς τον αριθμό των αποθηκευμένων πληροφοριών της. [13], [14]

2.3.2 Microsoft Academic Search

Το Microsoft Academic Search ή Libra (http://academic.research.microsoft.com/), όπως ήταν παλιότερα γνωστό, ήταν η προσπάθεια της Microsoft να εισχωρήσει και σε αυτό το τομέα της πληροφορικής. Αναπτύχθηκε από την ερευνητική ομάδα της Microsoft στην Ασία και η βάση δεδομένων της βασιζόταν σε πληροφορίες σχετικά με έγγραφα συνεδρίων και σε δημοσιεύσεις που έχουν γίνει σε επιστημονικά περιοδικά. Το 2008 η βάση δεδομένων της μηχανής αναζήτησης σταμάτησε να ανανεώνεται και συγχωνεύτηκε με την απλή αναζήτηση της Microsoft, κάνοντας μέχρι τότε αναζήτηση σε πάνω από 3 εκατομμύρια έγγραφα.

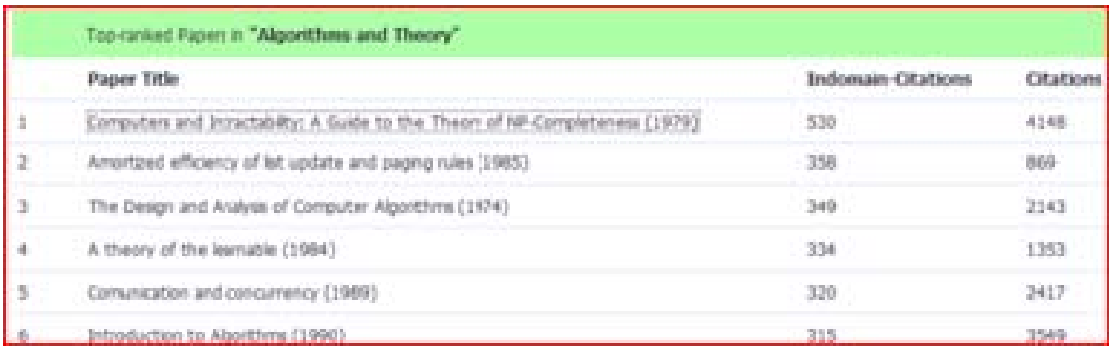
Το σύστημα χειριζόταν τέσσερα διαφορετικά πεδία σαν αντικείμενα, έγγραφα, συγγραφείς, συνέδρια και περιοδικά, συλλέγοντας πληροφορίες για κάθε

αντικείμενο προσπαθούσε να τα συσχετίσει ούτως ώστε να προσφέρει καλύτερα αποτελέσματα στις αναζητήσεις.



Εικόνα 2-4 Αναζήτηση με το Microsoft Academic Search [16]

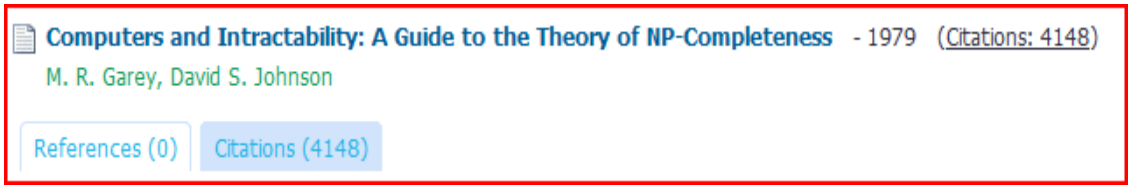
Ένα πολύ καλό στοιχείο της μηχανής αναζήτησης ήταν ότι διέθετε μία λίστα με τα πιο υψηλά καταταγμένα έγγραφα, συγγραφείς, συνέδρια και περιοδικά, για πολλά διαφορετικά πεδία της επιστήμης της πληροφορικής όπως, θεωρία των αλγορίθμων, βιοπληροφορική, λειτουργικά συστήματα, πολυμέσα, μηχανική λογισμικού και άλλα πολλά. Πατώντας πάνω σε έναν από τους συνδέσμους των πεδίων, εμφάνιζε τη λίστα με τα αποτελέσματα.


Top-ranked Papers in "Algorithms and Theory"

	Paper Title	Indomain-Citations	Citations
1	Computers and Intractability: A Guide to the Theory of NP-Completeness (1979)	530	4148
2	Amortized efficiency of list update and paging rules (1985)	358	869
3	The Design and Analysis of Computer Algorithms (1974)	349	2143
4	A theory of the learnable (1984)	334	1353
5	Comunication and concurrency (1989)	320	3417
6	Introduction to Algorithms (1990)	315	3569

Εικόνα 2-5 Λίστα αποτελεσμάτων για το πεδίο "Θεωρία αλγορίθμων" [16]

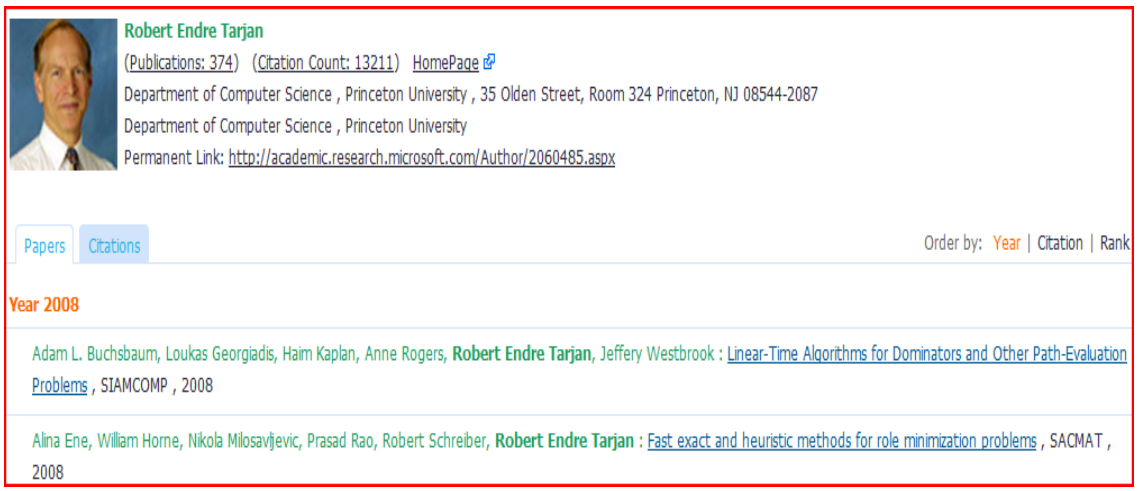
Πατώντας εκ νέου πάνω στο τίτλο ενός αποτελέσματος, εμφανιζόταν ο τίτλος του εγγράφου συνοδευόμενο από τη χρονολογία τους συγγραφείς και τις βιβλιογραφικές αναφορές που έγιναν σε αυτό.



Εικόνα 2-6 Βιβλιογραφικές αναφορές στο συγκεκριμένο έγγραφο [16]

Ανοίγοντας την καρτέλα “Citations” εμφανίζεται μία λίστα με τις αναφορές που έχουν γίνει στο συγκεκριμένο έγγραφο και πληροφορίες σχετικά με αυτές.

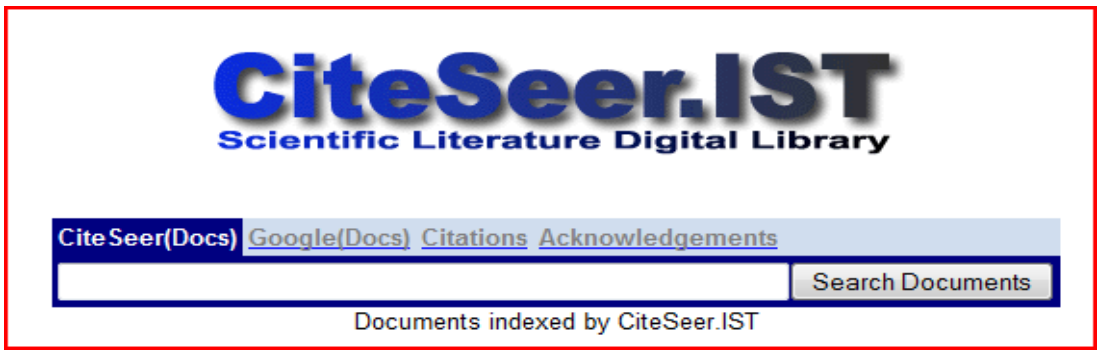
Σε περίπτωση που κάναμε την ίδια διαδικασία για κάποιο άλλο αντικείμενο, π.χ. το συγγραφέα, θα είχαμε διαφορετικά αποτελέσματα καθώς η λίστα μας αυτή τη φορά θα βασιζόταν στον κορυφαίο συγγραφέα σε αυτό το πεδίο. Πατώντας εν συνεχεία στο όνομα του συγγραφέα για περεταίρω πληροφορίες θα μας εμφανιζόταν μία λίστα με προσωπικές πληροφορίες για τον ίδιο το συγγραφέα όπως, τον αριθμό των εργασιών του, τον αριθμό των αναφορών σε αυτές, τη προσωπική του ιστοσελίδα και το τμήμα του πανεπιστημίου στο οποίο εδρεύει. Κάτω από αυτές τις πληροφορίες θα εμφανίζονταν με τη σειρά τους οι εργασίες του συγγραφέα, ταξινομημένες κατά φθίνουσα χρονολογική σειρά [16], [17].



Εικόνα 2-7 Ο κορυφαίος συγγραφέας στο πεδίο "Θεωρία αλγορίθμων", σύμφωνα με τη Microsoft Academic Search [16]

2.3.3 CiteSeer^x

Το citeseer^x (http://citeseerx.ist.psu.edu/) είναι μια μηχανή αναζήτησης και ψηφιακή βιβλιοθήκη για την αναζήτηση ακαδημαϊκών εγγράφων, κυρίως για θέματα πληροφορικής.



Εικόνα 2-8 Η μηχανή αναζήτησης του citeseer [18]

Είναι η εξέλιξη της μηχανής αναζήτησης βιβλιογραφικών αναφορών citeseer (http://citeseer.ist.psu.edu/) που δημιουργήθηκε το 1997 από τους Steve Lawrence, Lee Giles και Kurt Bollacker.

Το citeseer, που είναι ο προκάτοχος του citeseer^{x}, το οποίο έφτασε στα όρια του το 2005, όταν δεικτοδοτούσε σε 750.000 διαφορετικά έγγραφα και οι καθημερινές αιτήσεις στον server άγγιζαν τις 1.500.000. Για τις ανάγκες της ερευνητικής κοινότητας αποφασίστηκε ότι θα έπρεπε να σχεδιαστεί μία νέα αρχιτεκτονική του συστήματος και κατά αυτόν τον τρόπο προέκυψε το citeseer^{x}.

Εικόνα 2-9 Η μηχανή αναζήτησης του citeseer^{x} [19]

Το citeseer^{x} παρέχει δεδομένα για αλγορίθμους, μεταδεδομένα, υπηρεσίες, τεχνικές και λογισμικό που μπορεί να χρησιμοποιηθεί για την προώθηση άλλων ηλεκτρονικών βιβλιοθηκών. Έχει αναπτύξει καινούριες μεθόδους και αλγορίθμους για τη δεικτοδότηση Postscript και PDF άρθρων στο διαδίκτυο και διαθέτει επίσης μία πληθώρα από λειτουργίες, οι οποίες είναι :

- Αυτόνομη δεικτοδότηση βιβλιογραφικών αναφορών (Autonomus Citation Indexing, ACI).

  Χρησιμοποιεί το ACI για να δημιουργήσει δείκτες βιβλιογραφικών αναφορών που μπορούν να χρησιμοποιηθούν στα έγγραφα, για την καλύτερη αναζήτηση και αξιολόγησή τους.

- Στατιστικά για τις βιβλιογραφικές αναφορές

  Κρατά στατιστικές μετρήσεις για όλα τα έγγραφα που βρίσκονται στη βάση δεδομένων του.

- Σύνδεση βιβλιογραφικών αναφορών

  Έχει αυτοματοποιήσει τη περιήγηση σε βιβλιογραφικές αναφορές μέσω συνδέσμων που βρίσκονται σε κάθε έγγραφο

- Εντοπισμός και ευαισθησία

  Εντοπίζει και ειδοποιεί για νέες αναφορές σε έγγραφα

- Περιεχόμενα βιβλιογραφικών αναφορών

  Δίνει τη δυνατότητα στους ερευνητές να δουν γρήγορα και εύκολα τα περιεχόμενα ενός εγγράφου.

- Σχετικά έγγραφα

  Χρησιμοποιεί μεθόδους που βασίζονται σε λέξεις του εγγράφου για να εντοπίσει σχετικά έγγραφα και ενημερώνει τη βάση δεδομένων του.

- Δεικτοδότηση κειμένου

  Δεικτοδοτεί ολόκληρο το κείμενο από τα άρθρα και με αυτό τον τρόπο παρέχει μια πλήρης αναζήτηση.

- Ανανέωση

  Ανανεώνει τη βάση δεδομένων του με τις εκχωρήσεις των χρηστών του και με διάφορες ανιχνεύσεις.

- Ισχυρή αναζήτηση

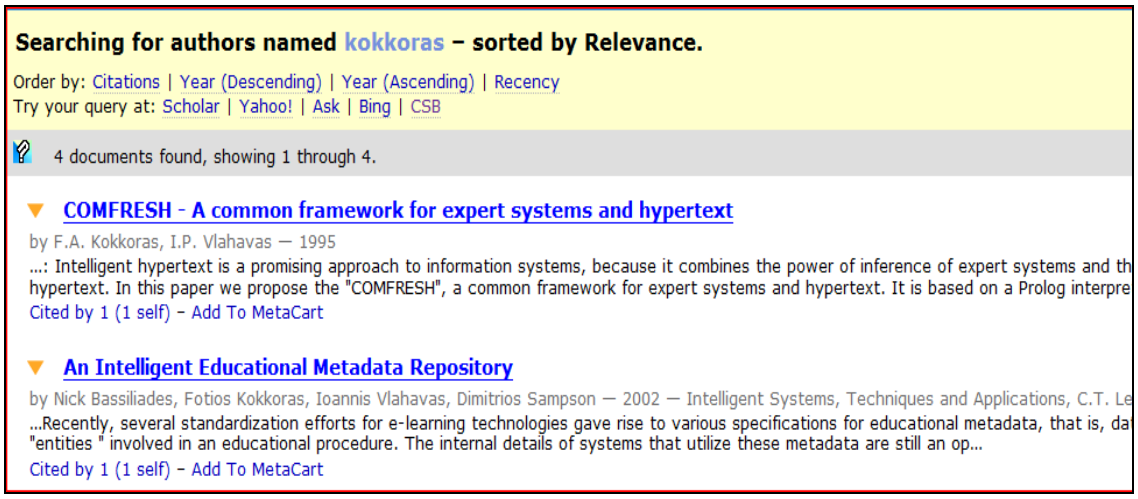
  Χρησιμοποιεί σύνθετα ερωτήματα πάνω στο περιεχόμενο για να κάνει την αναζήτηση του πιο αποτελεσματική και επιτρέπει τη χρήση των αρχικών του συγγραφέα για μια πιο ευέλικτη αναζήτηση.

- Συγκομιδή άρθρων

  Συλλέγει αυτόματα άρθρα από το διαδίκτυο.

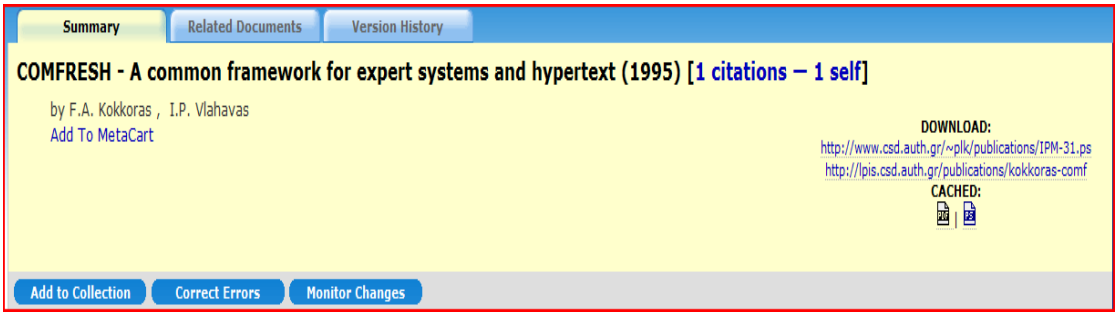
- Μεταδεδομένα άρθρων

  Εξάγει και συλλέγει τα μεταδεδομένα από όλα τα δεικτοδοτημένα άρθρα.



Εικόνα 2-10 Αποτελέσματα αναζήτησης του citeseer, με βάση το επώνυμο του συγγραφέα [19].

Στο citeseer ο χρήστης μπορεί να δημιουργήσει ένα προσωπικό λογαριασμό ούτως ώστε να αποθηκεύει πληροφορίες από τις αναζητήσεις του. Αν κάνουμε μια αναζήτηση στο citeseer με το επώνυμο του συγγραφέα (εικόνα 2-10), μας επιστρέφεται μία λίστα με εργασίες του συγγραφέα αρχικά χωρίς ταξινόμηση. Μας δίνεται βέβαια η δυνατότητα για μετέπειτα ταξινόμηση κατά, βιβλιογραφικές αναφορές, κατά φθίνουσα η αύξουσα χρονολογική σειρά ή κατά συχνότητα εμφάνισης. Επίσης δίνεται η δυνατότητα να δοκιμάσει ο χρήστης το ίδιο ερώτημα και σε άλλες μηχανές αναζήτησης όπως το Google Scholar, το yahoo, το Bing κ.α. [19]

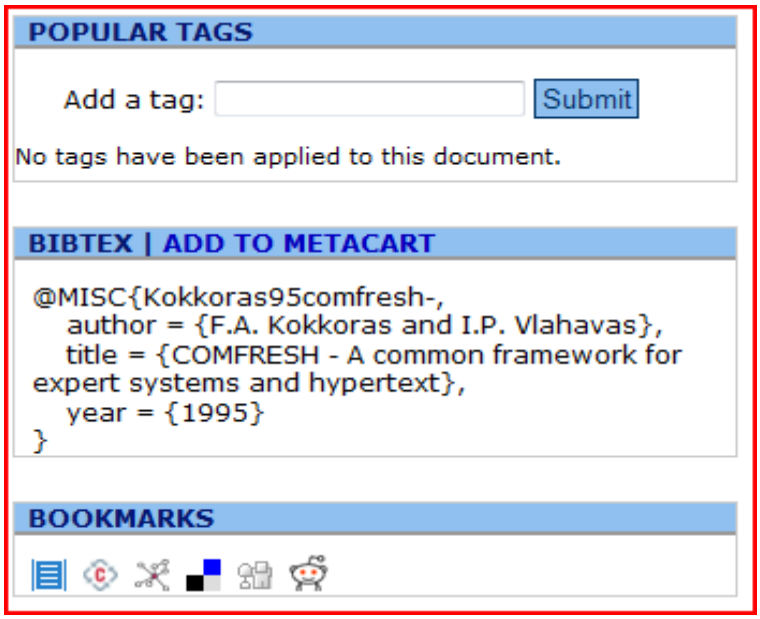


Εικόνα 2-11 Οι βασικές λειτουργίες πάνω σε ένα έγγραφο, όπως το παρουσιάζει το citeseer^{x} [19]

Πατώντας στο τίτλο μιας εργασία και ανοίγοντας το σύνδεσμο της, μας παρουσιάζεται μία σελίδα με τα ονόματα των συγγραφέων της εργασίας, μια γενική περιγραφή της και τις αναφορές που έχουν γίνει σε αυτή (εικόνα 2-11). Υπάρχει επίσης η δυνατότητα παρακολούθησης για τυχόν αλλαγές στην εργασία, διόρ-

θωσης λαθών, όπως και προσθήκη σε μια συλλογή από εργασίες. Στο πάνω μέρος της σελίδας υπάρχουν δύο επιπλέον καρτέλες μία για την παρουσίαση σχετικών εγγράφων και μία για τη παρουσίαση του ιστορικού της έκδοσης.

Ακόμη στο δεξιό μέρος της σελίδας (εικόνα 2-12) υπάρχει ένα πεδίο εισαγωγής λέξεων σήμανσης, ένα πεδίο όπου παρουσιάζεται η BIBTEX μορφή του εγγράφου, ένα πεδίο για τοποθέτηση της σελίδας σε διάφορα κοινωνικά δίκτυα (reddit, dig, delicious κ.α) και σε μερικές περιπτώσεις, όπου τα δεδομένα που έχουν συλλεχθεί είναι ικανοποιητικά, παρουσιάζεται μια γραφική αναπαράσταση για τις βιβλιογραφικές αναφορές ανά χρονιά δημοσίευσής τους. [19], [20]



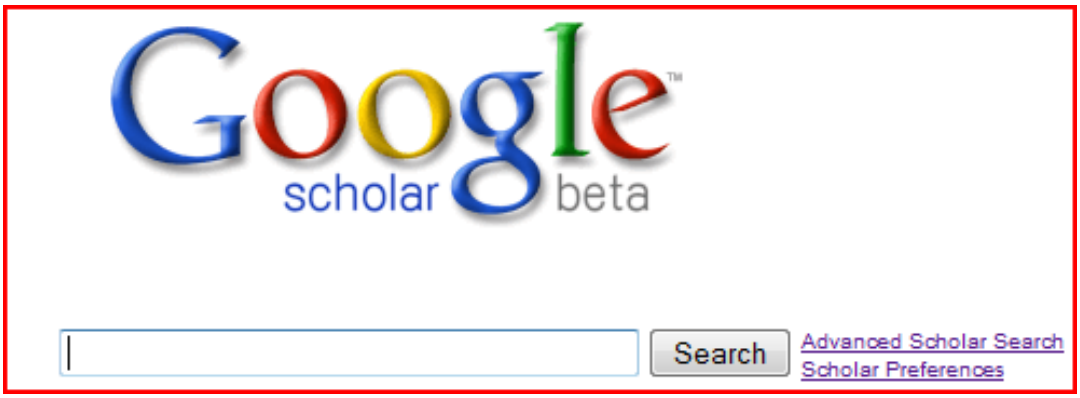
Εικόνα 2-12 Η δεξιά στήλη των αποτελεσμάτων του citeseer[x] [19]

2.3.4 Google Scholar

Το Google Scholar (εικόνα 2-13) κυκλοφόρησε σε δοκιμαστική έκδοση το 2004 και γρήγορα έγινε ένα από τα πιο δημοφιλή συστήματα του είδους του. Οι λόγοι για την προτίμηση των χρηστών ήταν η απλότητα της χρήσης του και η δομή του συστήματος που είναι παρόμοια με την απλή μηχανή αναζήτησης της Google, που όλοι λίγο πολύ έχουμε χρησιμοποιήσει.

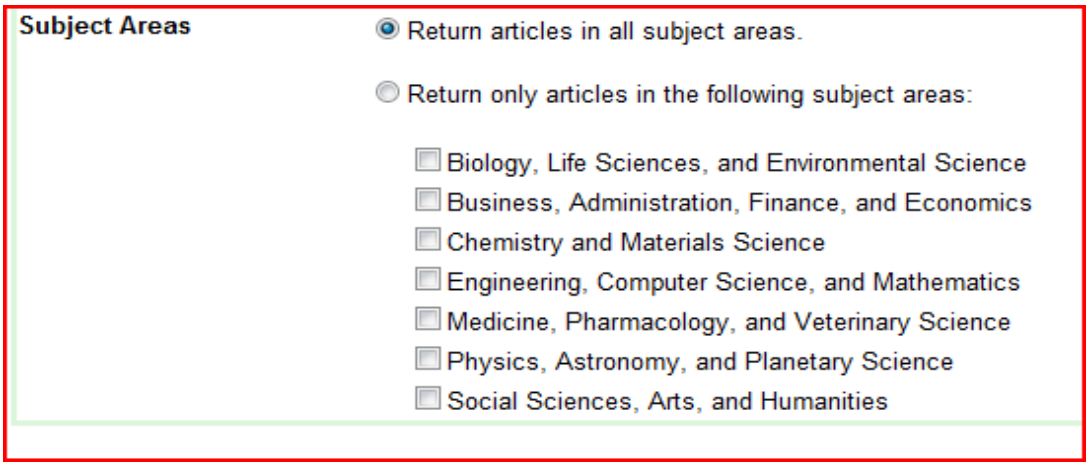
Ένα ακόμη θετικό στοιχείο του Google Scholar είναι, ότι δεν επικεντρώνεται μόνο πάνω στο επιστημονικό πεδίο της πληροφορικής, όπως τα συστήματα που έχουν ήδη περιγραφεί, αλλά προσπαθεί να συλλέξει πληροφορίες για όλα τα επιστημονικά πεδία. Βέβαια η αναζήτηση σε όλα τα επιστημονικά πεδία με μία η και περισσότερες λέξεις κλειδιά δεν δίνει πάντα σωστά αποτελέσματα. Το

Google Scholar διαθέτει ένα τρόπο φιλτραρίσματος των αποτελεσμάτων της αναζήτησης χρησιμοποιώντας τη προχωρημένη αναζήτηση ή επιλέγοντας συγκεκριμένες προτιμήσεις πάνω στη αναζήτηση.



Εικόνα 2-13 Η μηχανή αναζήτησης του Google Scholar [21]

Στη προχωρημένη αναζήτηση, εκτός από λέξεις και φράσεις κλειδιά που μπορεί να εισάγει ο χρήστης, μπορεί επίσης να εισάγει όνομα συγγραφέα, έκδοση, και να επιλέξει τα έγγραφα που έχουν εκδοθεί σε συγκεκριμένες ημερομηνίες. Το σημαντικότερο όμως είναι ότι δίνεται η δυνατότητα να επιλέξει ένα η περισσότερα επιστημονικά πεδία, για να μειώσει το εύρος της αναζήτησης, βλ. εικόνα 2-14.



Εικόνα 2-14 Τα επιστημονικά πεδία του Google Scholar [21]

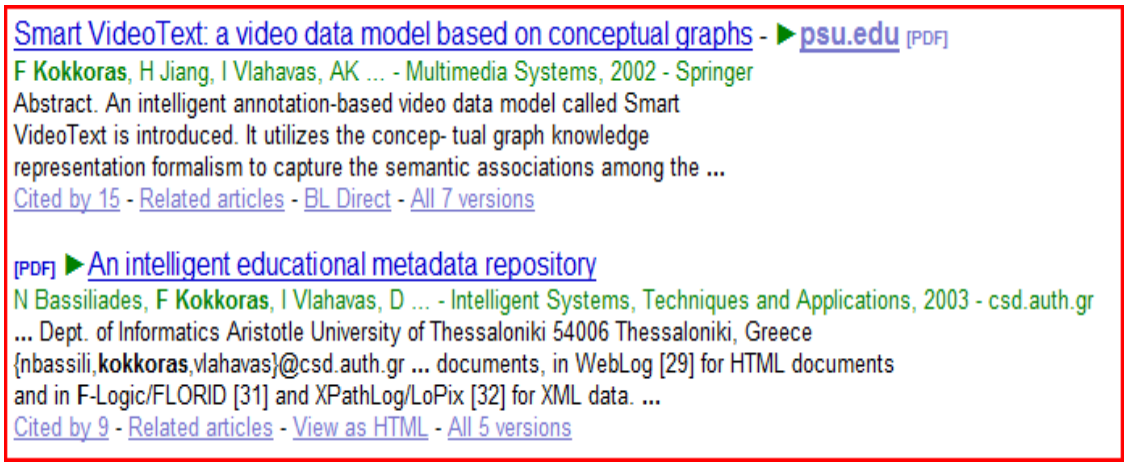
Στις προτιμήσεις ο χρήστης μπορεί να επιλέξει αρχικά σε τι γλώσσα επιθυμεί να εμφανίζεται το Google Scholar, επιλέγοντας μία από τις πολλές μεταφράσεις που υπάρχουν. Έχει επίσης, τη δυνατότητα να επιλέξει μία ή περισσότερες γλώσσες για την επιστροφή αποτελεσμάτων σε αυτή/ες, βλ εικόνα 2-15.

Εικόνα 2-15 Η επιλογή γλώσσας για την εμφάνιση των αποτελεσμάτων [21]

Ακόμη η αναζήτηση μπορεί να περιοριστεί σε συγκεκριμένες βιβλιοθήκες, π.χ. τη βιβλιοθήκη του πανεπιστημίου Harvard, έχοντας όμως κάποιους περιορισμούς, όπως ότι ο χρήστης πρέπει να χρησιμοποιεί κάποιο υπολογιστή που είναι συνδεδεμένος στο δίκτυο του πανεπιστημίου ή ακόμη και να είναι συνδεδεμένος στη βιβλιοθήκη με το προσωπικό του λογαριασμό. Τέλος ο χρήστης μπορεί να επιλέξει την εμφάνιση ενός συνδέσμου που θα παρέχει το αποτέλεσμα της αναζήτησης σε κάποια συγκεκριμένη μορφή, όπως BIBTEX, για την εισαγωγή των αποτελεσμάτων σε διαχειριστές βιβλιογραφικών αναφορών.

Η εμφάνιση των αποτελεσμάτων του Google Scholar, θυμίζει την εμφάνιση των αποτελεσμάτων της απλής αναζήτησης με το Google. Και με τα δύο συστήματα η Google προσπαθεί να παρουσιάσει όσο το δυνατόν περισσότερη πληροφορία χρησιμοποιώντας το μικρότερο δυνατό χώρο. Επίσης Χρησιμοποιεί διαφορετικά χρώματα και μεγέθη γραμματοσειράς σε διαφορετικές πληροφορίες για να δώσει την ανάλογη έμφαση κάθε φορά.

Όπως φαίνεται στην εικόνα 2-16, αρχικά εμφανίζεται ο τίτλος του εγγράφου. Σε αρκετές περιπτώσεις εμφανίζεται και το είδος της δημοσίευσης, για παράδειγμα στην εικόνα 2-16, το είδος της δημοσίευσης και των δύο αποτελεσμάτων είναι PDF, με ο προσδιοριστικό βέβαια να εμφανίζεται σε διαφορετικά σημεία για το καθένα. Ακόμη όταν στο δεξιό μέρος του τίτλου υπάρχει ένας σύνδεσμος σε κάποια επιστημονική κοινότητα, βλ. πρώτο αποτέλεσμα εικόνας 2-16, υπάρχει η δυνατότητα κατεβάσματος του εγγράφου, στον υπολογιστή του χρήστη, σε μορφή PDF συνήθως. Μπορεί να υπάρχουν και μερικοί ακόμη σύνδεσμοι που να δηλώνουν σε ποια βιβλιοθήκη βρέθηκε μια ηλεκτρονική έκδοση του εγγράφου και αν ανήκει σε μια ομάδα από παρόμοια έγγραφα.

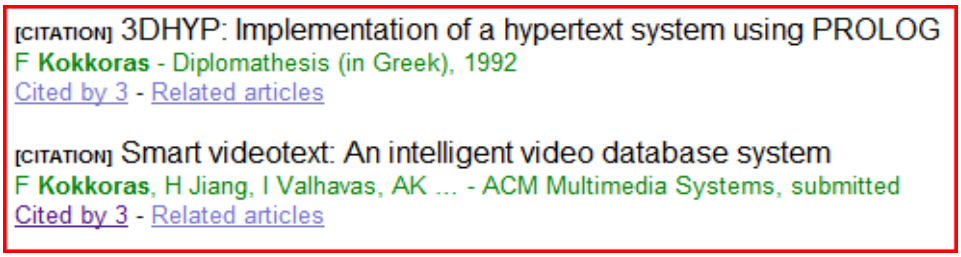


Εικόνα 2-16 Εμφάνιση των αποτελεσμάτων στο Google Scholar,
σε αναζήτηση με βάση το επώνυμο του συγγραφέα. [21]

Κάτω από το τίτλο και με πράσινα γράμματα, εμφανίζονται πληροφορίες όπως, τα ονόματα του/των συγγραφέα/ων, τη δημοσίευση και την ημερομηνία έκδοσης του εγγράφου καθώς και ο εκδοτικός οίκος. Στη συνέχεια εμφανίζονται μερικές γραμμές με κάποιο γενικό κείμενο (abstract) πάνω στη δημοσίευση και τέλος μια σειρά από συνδέσμους. Πρώτος σύνδεσμος είναι το “cited by”, το οποίο οδηγεί σε μια νέα λίστα με δημοσιεύσεις που έχουν αναφερθεί στην συγκεκριμένη εργασία.

Οι δημοσιεύσεις αυτές αποτελούν τις βιβλιογραφικές αναφορές του εγγράφου και παρουσιάζονται με τον ίδιο τρόπο, όπως και τα υπόλοιπα αποτελέσματα του Google Scholar. Δίπλα βρίσκεται ο σύνδεσμος “Related Articles” ο οποίος οδηγεί σε παρόμοια με το συγκεκριμένο, έγγραφα. Οι υπόλοιποι σύνδεσμοι μπορεί να διαφέρουν από αποτέλεσμα σε αποτέλεσμα. Μπορεί να υπάρχουν σύνδεσμοι που να προωθούν τη περαιτέρω αναζήτηση στο διαδίκτυο για περισσότερες πληροφορίες μέσο της μηχανής αναζήτησης του Google, σύνδεσμοι που οδηγούν σε βιβλιοθήκες όπου υπάρχει ένα φυσικό αντίγραφο της δημοσίευσης ή ακόμη και σύνδεσμοι που παρουσιάζουν τη δημοσίευση σε HTML μορφή. Σε μερικά αποτελέσματα εμφανίζεται επίσης, ο σύνδεσμος “BL Direct”. Αυτός ο σύνδεσμος οδηγεί στη Βρετανική βιβλιοθήκη (British Library) και δίνει τη δυνατότητα στους χρήστες να αγοράσουν ένα αντίγραφο της δημοσίευσης αυτής, με τη Google να δηλώνει ότι δεν παίρνει κάποια αποζημίωση για την υπηρεσία της αυτή.

Σε μια αναζήτηση με το επώνυμο του συγγραφέα μπορεί να μας επιστραφούν αποτελέσματα με τις εργασίες του συγγραφέα, αλλά και αποτελέσματα που είναι βιβλιογραφικές αναφορές σε εργασίες του συγκεκριμένου συγγραφέα μιας και είναι και το ζητούμενο της αναζήτησης.


[CITATION] 3DHYP: Implementation of a hypertext system using PROLOG
F **Kokkoras** - Diplomathesis (in Greek), 1992
Cited by 3 - Related articles

[CITATION] Smart videotext: An intelligent video database system
F **Kokkoras**, H Jiang, I Valhavas, AK ... - ACM Multimedia Systems, submitted
Cited by 3 - Related articles

Εικόνα 2-17 Αποτελέσματα που το Google Scholar εμφανίζει σαν βιβλιογραφικές αναφορές [21]

Σε αυτή τη περίπτωση ο τίτλος εμφανίζεται με μαύρο χρώμα γραμματοσειράς και φέρει μπροστά την ένδειξη “citation”. Από κάτω εμφανίζονται τα ονόματα των συγγραφέων, το είδος της δημοσίευσης και η χρονολογία με πράσινα γράμματα και τέλος 2 σύνδεσμοι, ένας με το όνομα “cited by” και ακολουθούμενος από τον αριθμό των εγγράφων που έχουν αναφερθεί σε αυτή την εργασία και έναν ακόμη με σχετικά άρθρα. Από τα αποτελέσματα αυτά, παραλείπονται αρκετές πληροφορίες, όπως π.χ. το abstract κείμενο, βλ. εικόνα 2-17.

2.4 Προβλήματα σε Βιβλιογραφικές Αναφορές

Είτε ένας συγγραφέας ακολουθήσει τον εμπειρικό τρόπο αναζήτησης, είτε χρησιμοποιήσει ένα από τα συστήματα που αναφέρθηκαν προηγουμένως, το σίγουρο είναι ότι διάφορα προβλήματα θα συνεχίζουν να εμφανίζονται. Τα πιο σημαντικά προβλήματα είναι το Citation Matching, το Mixed Citation και το Split Citation. Για την αντιμετώπιση αυτών των συχνών προβλημάτων έχουν προταθεί διάφοροι αλγόριθμοι και μέθοδοι, οι σημαντικότεροι των οποίων παρουσιάζονται στη συνέχεια.

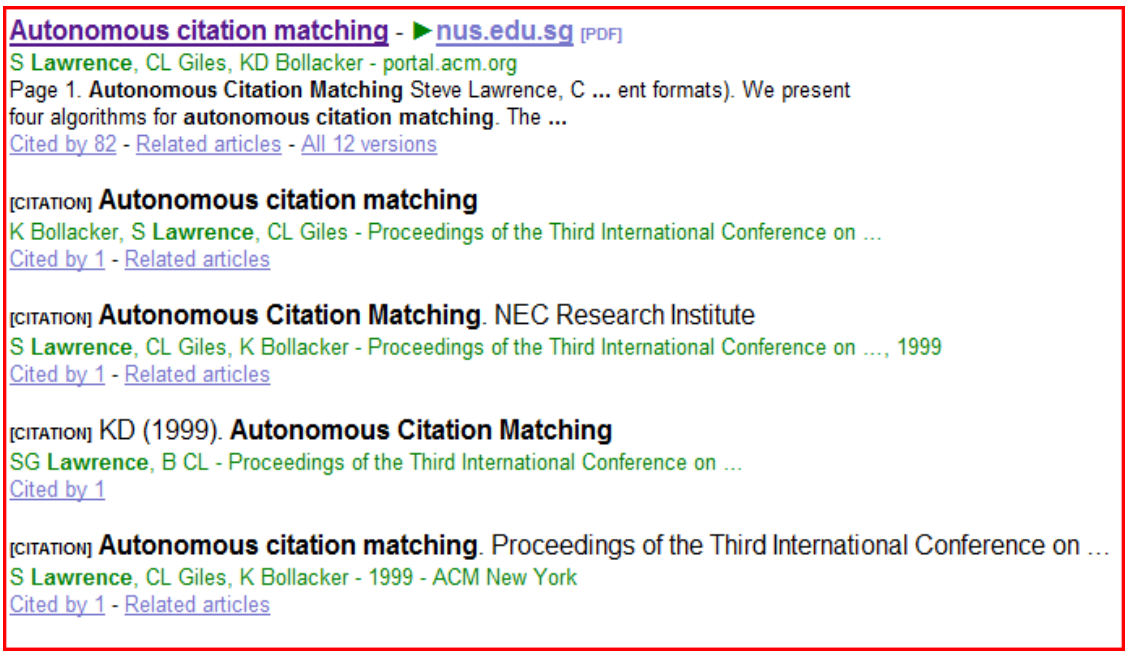
2.4.1 Citation Matching Problem

Σύμφωνα με το πρόβλημα αυτό, ένα αποτέλεσμα εμφανίζεται δύο η περισσότερες φορές στη λίστα των αποτελεσμάτων. Αυτό συμβαίνει για το λόγο ότι, οι συγγραφείς δεν εφαρμόζουν ένα κοινό πρότυπο στη συγγραφή της βιβλιογρα-

φίας. Πολλές φορές ακόμη και ο ίδιος ο συγγραφέας, χρησιμοποιεί διαφορετικό τρόπο γραφής της βιβλιογραφίας σε διαφορετικές εργασίες του. Η σειρά με την οποία γράφονται ο τίτλος, η χρονολογία δημοσίευσης, οι συγγραφείς κτλ., διαφέρει σημαντικά και προκαλεί μια σύγχυση στις μηχανές αναζήτησης που χρειάζονται ένα κοινό πρότυπο για να αυτοματοποιήσουν τη διαδικασία εύρεσης αποτελεσμάτων. [3]

Από τη στιγμή βέβαια, που το πρόβλημα έγινε ορατό, γίνεται μια προσπάθεια από τις μηχανές αναζήτησης να το μειώσουν όσο γίνεται περισσότερο συγχωνεύοντας τα αποτελέσματα σε ένα ή ακόμη και να το εξαλείψουν. Ακόμη βέβαια δεν έχουν καταφέρει κάτι τέτοιο και η εμφάνιση τέτοιων προβλημάτων δεν είναι σπάνιο φαινόμενο.

Σε μια αναζήτηση με τη βοήθεια του Google Scholar, με λέξεις κλειδιά “Autonomous citation matching”, εμφανίστηκαν τα παρακάτω αποτελέσματα όπως φαίνονται στην εικόνα 2-18.



Εικόνα 2-18 Citation Matching Problem στο Google Scholar
με κλειδί αναζήτησης “Autonomous citation matching” [21]

Στα παραπάνω αποτελέσματα, αναφορά της ίδιας δημοσίευσης παρουσιάζεται 5 φορές. Συνήθως το Google Scholar ομαδοποιεί τις βιβλιογραφικές αναφορές που γίνονται στην ίδια δημοσίευση και δίνει τη δυνατότητα στο χρήστη να τις δει πατώντας το σύνδεσμο “cited by”. Σε αυτή τη περίπτωση δε έγινε αυτή η

ομαδοποίηση, κάτι που οφείλεται στη διαφορετική δομή που έχουν τα αποτελέσματα. Με μια πρώτη ματιά, ο άνθρωπος μπορεί να ξεχωρίσει ότι πρόκειται για αναφορές στο ίδιο έγγραφο, αλλά κοιτάζοντας καλύτερα τα αποτελέσματα μπορούμε να εντοπίσουμε τη διαφορετική δομή τους. Αυτό είναι και ο λόγος που σύγχυσε και τη μηχανή αναζήτησης και δεν μπόρεσε να τα ομαδοποιήσει.

Όπως φαίνεται και στην εικόνα 2-18, στο πρώτο αποτέλεσμα έχουν ομαδοποιηθεί 82 αναφορές σε αυτή τη δημοσίευση. Στο δεύτερο και στο τέταρτο αποτέλεσμα, τα ονόματα των συγγραφέων είναι γραμμένα με διαφορετική σειρά ενώ το τέταρτο φέρει επιπλέον πληροφορίες στον τίτλο του, όπως η χρονολογία δημοσίευσής του. Στο τρίτο και στο πέμπτο αποτέλεσμα παρόλο που τα ονόματα των συγγραφέων είναι γραμμένα με τη σωστή σειρά και με το ίδιο τρόπο, φέρουν στο τίτλο τους επιπλέον πληροφορίες. Στο τρίτο αναφέρεται το ερευνητικό κέντρο, ενώ στο πέμπτο ότι πρόκειται για τα πρακτικά ενός συνεδρίου.

Αντιμετώπιση του Citation Matching Problem

Πολλά από τα citation προβλήματα μπορούν να λυθούν με τη χρησιμοποίηση «Παγκόσμιων αναγνωριστικών» (global IDs). Όποιες και να είναι οι διαφορές δύο βιβλιογραφικών αναφορών, στη περίπτωση που φέρουν ένα global ID θα θεωρούνται σαν όμοιες αναφορές. Δύο από τα πιο δημοφιλή global IDs είναι τα ISBNs (International Standard Book Numbers) και τα DOIs (Digital Object Identifiers). [2]

Ωστόσο, παρόλα τα πλεονεκτήματα που μπορεί να έχει ένα global ID, οι συγγραφείς τα έχουν εν μέρει συμπεριλάβει και οι χρήστες στην πλειοψηφία τους τα αγνοούν, κυρίως όταν πρόκειται για αναφορές στο διαδίκτυο. Πάντως, ακόμη κι αν όλοι οι συγγραφείς συμπεριλάμβαναν τα global IDs σε όλες τους τις εργασίες, ακόμη θα υπήρχε το πρόβλημα των ήδη δημοσιευμένων εργασιών. Η ενημέρωση αυτών θα κόστιζε αρκετά και σε χρόνο αλλά και σε χρήματα. Αν ακόμη θεωρήσουμε ότι γινόταν αυτή η διαδικασία, της ενημέρωσης όλων των δημοσιεύσεων με κάποιο παγκόσμιο αναγνωριστικό, τότε θα αντιμετωπίζαμε ένα νέο πρόβλημα, του τύπου του global ID. Θα έπρεπε ή να συμφωνηθεί η χρησιμοποίηση μόνος ενός global ID, π.χ. το ISBN ή το DOI, ή ακόμη και να έβρισκαν μια αντιστοιχία στα δύο η και σε περισσότερα global IDs. Σε κάθε περίπτωση για να γίνει απολύτως χρήσιμη η λύση των global IDs θα πρέπει να δοθεί μία λύση στα άλλα δύο προβλήματα που προκύπτουν. [2]

Μία άλλη λύση που χρησιμοποιείται είναι οι αλγόριθμοι μέτρησης αποστάσεων, όπως του Levenshtein, του Jaro ή η Cosine. Αυτοί οι αλγόριθμοι μπορούν να μετρήσουν τις διαφορές δύο βιβλιογραφικών αναφορών B_α και B_β και ανάλογα με ένα κατώτατο όριο που έχει τεθεί στην απόσταση δύο αναφορών, να αποφασίσουν αν πρόκειται για όμοιες ή διαφορετικές αναφορές. [2]

2.4.2 Mixed Citation Problem

Το mixed citation πρόβλημα αναφέρεται στη περίπτωση όπου τα ονόματα δύο ή και περισσοτέρων συγγραφέων είναι τα ίδια, οπότε μπορεί να υπάρξει κάποια σύγχυση από τις μηχανές αναζήτησης βιβλιογραφικών αναφορών και να εμφανίσουν λανθασμένες αναφορές για τις δημοσιεύσεις ενός προσώπου, ενώ ανήκουν σε κάποιο άλλο συνονόματο του. Όπως φαίνεται και στην εικόνα 2-19, σε μια αναζήτηση με το όνομα “Dongwon Lee”, στο έτος 2004 υπήρχαν δύο αποτελέσματα τα οποία αναφέρονταν στο ίδιο όνομα συγγραφέα, αλλά σε διαφορετικό πρόσωπο. Τον Dongwon Lee του κλάδου της πληροφορικής και του Downgwon Lee του κλάδου των επιχειρήσεων. [1]

List of publications from the DBLP Bibliography Server - FAQ

Coauthor Index - Ask others: ACM DL - ACM Guide - CiteSeer - CSB - Google

		2004
27		Alberto H. F. Laender, Dongwon Lee, Marc Ronthaler: Sixth ACM CIKM International Workshop on Web Information and Data Management (WIDM 2004), Washington, DC, USA, November 12-13, 2004 ACM 2004
26	EE	Bo Luo, Dongwon Lee, Wang-Chien Lee, Peng Liu: QFilter: fine-grained run-time XML access control via NFA-based query rewriting. CIKM 2004: 543-552
25	EE	Dongwon Lee, Divesh Srivastava: Counting Relaxed Twig Matches in a Tree. DASFAA 2004: 88-99
24	EE	Yoojin Hong, Byung-Won On, Dongwon Lee: System Support for Name Authority Control Problem in Digital Libraries: OpenDBLP Approach. ECDL 2004: 134-144
23	EE	Robert J. Kauffman, Dongwon Lee: Should We Expect Less Price Rigidity in the Digital Economy? HICSS 2004
22	EE	Byung-Won On, Dongwon Lee: PaSE: Locating Online Copy of Scientific Documents Effectively. ICADL 2004: 408-418
21		Robert J. Kauffman, Dongwon Lee: Price Rigidity on the Internet: New Evidence from the Online Bookselling Industry. ICIS 2004: 843-848

Εικόνα 2-19 Το mixed citation problem, σε αναζήτηση στο DBLP και κλειδί αναζήτησης το όνομα “Dongwon Lee” [1]

Όπως ήδη αναφέρθηκε, οι συγγραφείς χρησιμοποιούν τις βιβλιογραφικές αναφορές που έχουν γίνει στις δημοσιεύσεις του για να μετρήσουν την επιρροή τους στο σύνολο της επιστημονικής κοινότητας. Σε μια περίπτωση mixed citation προβλήματος, θα γίνεται λάθος εκτίμηση του βαθμού επιρροής και από

τους δύο συγγραφείς. Ο ένας θα υπερεκτιμήσει το έργο του ενώ ο άλλος θα το υποτιμήσει.

Αντιμετώπιση του Mixed Citation Problem

Η δυσκολία σε αυτή τη περίπτωση είναι ότι δε μπορεί να χρησιμοποιηθεί κάποιος από τους αλγορίθμους μέτρησης αποστάσεων, μιας και πρόκειται για δύο ίδια ονόματα. Ωστόσο, μία πρόταση για την αντιμετώπιση αυτού του προβλήματος είναι να χρησιμοποιηθούν οι υπόλοιπες, συνοδευτικές του ονόματος του συγγραφέα, πληροφορίες της αναφοράς. Στις πληροφορίες αυτές ανήκουν, τα ονόματα των συνεργατών του και λέξεις κλειδιά που συναντώνται συχνά σε τίτλους δημοσιεύσεών του. [1]

Με αυτές τις πληροφορίες προτάθηκε ένας αλγόριθμος για τη λύση του προβλήματος. Σε αυτό τον αλγόριθμο, έχουμε μία βιβλιογραφική αναφορά β_i που περιέχει μία λίστα με τα ονόματα των συνεργατών συγγραφέων $\Sigma\{\sigma_1, \ldots, \sigma_n\}$ και μία λίστα με τις λέξεις κλειδιά από τους τίτλους $T\{\tau_1, \ldots, \tau_n\}$. Αν αφαιρέσουμε τον σ_i από τη β_i μπορέσουμε να με τη βοήθεια μιας μεθόδου που χρησιμοποιεί τα συσχετιζόμενα στοιχεία που έχουμε συλλέξει, να μαντέψουμε το όνομα του συγγραφέα αυτού, τότε θα έχουμε τη λύση στο πρόβλημά μας.

Υποθέτοντας λοιπόν ότι έχουμε αυτή τη μέθοδο f, την εφαρμόζουμε για κάθε βιβλιογραφική αναφορά που ανήκει στη λίστα με τις βιβλιογραφικές αναφορές, που έχουμε λάβει σαν αποτέλεσμα από μία αναζήτηση, ως εξής:

Για κάθε αναφορά β_i

- Αφαιρούμε το όνομα σ_1 (το αυθεντικό όνομα) από τη λίστα Σ (συνεργατών συγγραφέων) της β_i
- Χρησιμοποιούμε τη μέθοδο f για να μαντέψουμε το σ_2
- Αν το $\sigma_1 \neq \sigma_2$ τότε το β_i είναι λάθος αναφορά και αφαιρείται από τη λίστα των βιβλιογραφικών αναφορών

Αυτός ο έλεγχος επαναλαμβάνεται για κάθε βιβλιογραφική αναφορά στη λίστα και στο τέλος μένουν μόνο οι σωστές. Έτσι με τη εφαρμογή αυτής της μεθόδου μπορούμε να λύσουμε το συγκεκριμένο πρόβλημα. [1]

2.4.3 Split Citation Problem

Στη περίπτωση του split citation προβλήματος, συμβαίνει ακριβώς το αντίθετο απ' ότι στο mixed citation πρόβλημα. Έχουμε split citation όταν, οι αναφορές σε δημοσιεύσεις ενός συγγραφέα εμφανίζονται ξεχωριστά σα να ήταν αναφορές κάποιου άλλου.

Αυτό συμβαίνει γιατί το όνομα του συγγραφέα μπορεί να είναι ελαφρός παραλλαγμένο, από βιβλιογραφία σε βιβλιογραφία. Για παράδειγμα ας υποθέσουμε ότι, έχουμε τον συγγραφέα "John Doe", ο οποίος ας υποθέσουμε ότι έχει δημοσιεύσει 100 άρθρα. Σε μια ψηφιακή βιβλιοθήκη έχει εκχωρηθεί με δύο διαφορετικά ονόματα, σαν "John Doe" και σαν "J D Doe" με το καθένα να έχει 50 βιβλιογραφικές αναφορές σε δημοσιεύσεις του. Σε μια αναζήτηση στο όνομα "John Doe", θα παρουσιαστεί στο χρήστη, ένας συγγραφέας με 50 βιβλιογραφικές αναφορές σε έργα του. Κατά αυτό τον τρόπο η επιρροή του συγγραφέα μειώνεται αμέσως στο μισό, με το έργο του να υποτιμείται σημαντικά, για το λόγο ότι λανθασμένα ο "J D Doe", του κλέβει ένα μέρος από τη φήμη του. Τέτοιες περιπτώσεις διφορούμενων ονομάτων υπάρχουν σε αρκετές από τις ψηφιακές βιβλιοθήκες. [1]

Αντιμετώπιση του Split Citation problem

Μία απλή λύση για την επίλυση του προβλήματος split citation θα ήταν να αντιμετωπίσουμε κάθε όνομα συγγραφέα σαν μία συμβολοσειρά και να κάνουμε ένα έλεγχο, με κάποιον από τους αλγορίθμους μέτρησης της απόστασης, για όλα τα πιθανά ζεύγη συμβολοσειρών. Ωστόσο ένας τόσο απλός έλεγχος δε θα ήταν απόλυτα αποτελεσματικός. Σε μια περίπτωση όπου δύο συγγραφείς είχαν παρόμοια ονόματα η σύγχυση του αλγορίθμου θα ήταν σχεδόν σίγουρη, υποδεικνύοντάς τα ως όμοια [1]. Μερικοί από τους αλγορίθμους μέτρησης αποστάσεων σε συμβολοσειρές είναι:

- Η απόσταση Levenshtein.

  Η μέθοδος αυτή, πήρε το όνομα της από το Ρώσο Vladimir Levenshtein που την υπολόγισε το 1965. Μετρά το σύνολο των αλλαγών που χρειάζονται για να μετατραπεί η μία συμβολοσειρά στην άλλη. Η υλοποίηση του αλγορίθμου περιλαμβάνει ένα πίνακα (n + 1) × (m + 1), όπου n και m είναι τα μήκη των δύο συμβολοσειρών, ενώ η πολυπλοκότητα του αλγορίθμου είναι O(nm). [7], [8]

- Η απόσταση Jaro-Winkler

  Πρόκειται για μια επέκταση της απόστασης Jaro που ανέπτυξε ο Winkler το 1999. Όσο ψηλότερα βρίσκεται η Jaro-Winkler απόσταση, τόσο πιο κοντά βρίσκονται οι δύο συμβολοσειρές. Ταιριάζει περισσότερο σε μικρές συμβολοσειρές, όπως ονόματα και είναι πολυπλοκότητας O(nm). [9]

- Η ομοιότητα συνημίτονων (cosine similarity)

  Αντιμετωπίζει τις συμβολοσειρές ως διανύσματα και με τη βοήθεια του Ευκλείδειου κανόνα συνημίτονου, μετρά την απόσταση των δύο συμβολοσειρών, βάση της γωνίας που σχηματίζουν. [10], [11]

- TF-IDF βαρύτητα (Term Frequency – Inverse Document Frequency)

  Αυτή η μέθοδος χρησιμοποιεί στατιστικές μετρήσεις για να διαπιστώσει πόσο σημαντική είναι μια λέξη σε ένα κείμενο. Όσο πιο συχνά εμφανίζεται μια λέξη στο κείμενο, τόσο μεγαλύτερης βαρύτητας είναι. Πιο αναλυτικά η βαρύτητα υπολογίζεται αναλόγως με τη συχνότητα που εμφανίζεται μία λέξη στο σύνολο των λέξεων ενός κειμένου και το κείμενο με τη σειρά του στο σύνολο των εγγράφων.

  Αν λ είναι μία λέξη το σύνολο των λέξεων σε ένα έγγραφο είναι $\Sigma_{\lambda,}$ με Σ_{ε} να είναι το σύνολο των εγγράφων και ε τον αριθμό των εγγράφων, από το συνολικό αριθμό, που εμφανίζεται η λέξη λ, τότε η βαρύτητα Β υπολογίζεται ως εξής [12], [13]:

$$B = (\lambda/\Sigma_{\lambda}) \times \ln(\Sigma_{\varepsilon}/\varepsilon)$$

Μία καλύτερη λύση θα ήταν, αντί να ελέγχουμε μόνο τα δύο ονόματα με της μεθόδους μέτρησης αποστάσεων, να χρησιμοποιούμε και τις συνοδευτικές πληροφορίες των ονομάτων, όπως τα ονόματα των συνεργατών συγγραφέων και το τίτλο της δημοσίευσης. Για παράδειγμα αντί να συγκρίνουμε δύο ονόματα, θα μπορούσαμε να δούμε τι κοινά συνοδευτικά στοιχεία έχουν αυτά τα δύο ονόματα. Για τη σωστή λειτουργία αυτής της μεθόδου υπάρχει μία προϋπόθεση, να συγκρίνονται τα ίδια συνοδευτικά όταν αυτά υπάρχουν, δηλαδή να ελέγχεται αν υπάρχει συνεργάτης συγγραφέας και έπειτα να συγκρίνεται με τον αντίστοιχο, εφόσον υπάρχει, της άλλης εγγραφής.[1]

3 Υποκίνηση και Τεχνολογίες

Για τις ανάγκες της παρούσας πτυχιακής εργασίας αναπτύχθηκε μια διαδικτυακή εφαρμογή για την διαχείριση των βιβλιογραφικών αναφορών. Σε αυτή την ενότητα γίνεται μια περιγραφή των αναγκών που οδήγησαν σε αυτή την εφαρμογή καθώς και των εργαλείων (γλωσσών προγραμματισμού, κτλ) που χρησιμοποιήθηκαν για την ανάπτυξή της και τον έλεγχο καλής λειτουργίας.

3.1 Τι Οδήγησε στην Εφαρμογή

Οι ακαδημαϊκοί και οι ερευνητές συνηθίζουν να διατηρούν μία λίστα με τις εργασίες τους και τις βιβλιογραφικές αναφορές που έχουν γίνει σε αυτές από τρίτους. Η διαχείριση μιας διαδικασίας σαν αυτή απαιτεί αρκετή χειρονακτική εργασία και συχνή περιοδική ενασχόληση.

Με την ανάπτυξη του διαδικτύου και των νέων υπηρεσιών που αυτό παρέχει, η πλειοψηφία των εργασιών μπορεί να βρεθεί σε ηλεκτρονική μορφή σε διάφορες ψηφιακές βιβλιοθήκες. Η διαδικτυακή διαθεσιμότητα των εργασιών έδωσε μια ώθηση στη διαδικασία αναζήτησης βιβλιογραφικών αναφορών, κυρίως για εργασίες του τομέα των θετικών επιστημών, καθώς η όλη διαδικασία πλέον μπορούσε να γίνει με τη χρήση μηχανών αναζήτησης στο παγκόσμιο ιστό.

Αυτή η ψηφιοποίηση των δημοσιεύσεων δημιούργησε την ανάγκη για εξειδικευμένα συστήματα στην αναζήτηση βιβλιογραφικών αναφορών, όπως αυτά που αναφέρθηκαν στο $2^{ο}$ κεφάλαιο. Αυτές οι εξειδικευμένες μηχανές αναζήτησης βοήθησαν σημαντικά στον εντοπισμό όλο και περισσότερων βιβλιογραφικών αναφορών και στη διευκόλυνση κατά μεγάλο βαθμό της διαδικασίας αναζήτησης.

Παρόλο που η βοήθεια αυτών των μηχανών αναζήτησης ήταν σημαντική, η διαδικασία που χρειαζόταν κάποιος για τη διαχείριση και ενημέρωση της λίστας του ακόμη και για το ξεκαθάρισμα των αποτελεσμάτων και τον εντοπισμό των σωστών αναφορών, ήταν αρκετά χρονοβόρα. Λόγο της έλλειψης

ενός μοναδικού τρόπου κωδικοποίησης των βιβλιογραφικών αναφορών διάφορα προβλήματα δυσχέραιναν την όλη διαδικασία.

Για να εντοπίσει κάποιος βιβλιογραφικές αναφορές σε εργασίες του, έπρεπε να χρησιμοποιήσει γενικές και εξειδικευμένες μηχανές αναζήτησης, να ελέγξει ένα προς ένα όλα τα αποτελέσματα και πολλές φορές να περιηγηθεί μέσω συνδέσμων σε διάφορες σελίδες ούτως ώστε να εξακριβώσει αν μία βιβλιογραφική αναφορά ήταν έγκυρη ή όχι, ή ακόμη και για να συλλέξει πληροφορίες για αυτή.

Επίσης κάτι που πρέπει σημειωθεί είναι ότι οι αναφορές σε μια εργασία μπορεί να μένουν ως έχουν αλλά το πιο πιθανό είναι να αυξάνεται ο αριθμός τους. Αυτή η εξέλιξη απαιτεί έλεγχο των αποτελεσμάτων και επανάληψη της όλης διαδικασίας ανά τακτά χρονικά διαστήματα. Ακόμη και στις περιπτώσεις που έχουν καταγραφεί οι αναφορές σε μια εργασία, κατά το παρελθόν, θα πρέπει να ελεγχθούν τυχόν αλλαγές ούτως ώστε η λίστα του χρήστη να παραμένει ενημερωμένη.

Όλα τα παραπάνω δημιούργησαν την ανάγκη για ένα σύστημα που θα υποβοηθούσε τις εξειδικευμένες μηχανές αναζήτησης. Η ανάγκη αυτή οδήγησε στη παρούσα πτυχιακή εργασία και στην ανάπτυξη μιας εφαρμογής για τον εντοπισμό και τη καλύτερη διαχείριση βιβλιογραφικών αναφορών. Το ότι πρόκειται για μια διαδικτυακή εφαρμογή βοηθά το χρήστη στη εύκολη πρόσβαση του και τον έλεγχο της λίστα από οποιονδήποτε υπολογιστή και ανά πάσα στιγμή. Επίσης συντελεί στη πιο γρήγορη εξοικείωσή του, καθώς η γραφική της διεπαφή, θυμίζει εφαρμογή ηλεκτρονικού ταχυδρομείου.

Η βασική της λειτουργία είναι να υποστηρίζει το Google Scholar όσο το δυνατόν καλύτερα, για την εξάλειψη των προβλημάτων που αναφέρθηκαν. Αυτό το επιτυγχάνει με το να αποθηκεύει τα αποτελέσματα σε μια βάση δεδομένων, να αποκλείει τα μη σχετικά αποτελέσματα και τα αποτελέσματα τα οποία έχουν καταχωρηθεί σε προηγούμενη αναζήτηση και να ενημερώνει το χρήστη για τυχόν αλλαγές που έχουν γίνει σε εγγραφές, που είναι ήδη καταχωρημένες στη βάση δεδομένων. Ακόμη, ο χρήστης έχει τη δυνατότητα να επεξεργαστεί τα δεδομένα για την καλύτερη παρουσίασή τους χρησιμοποιώντας απλές διαδικτυακές φόρμες.

Τέλος, δημιουργήθηκαν αναφορές (reports) σε δομή που εξυπηρετεί τη συγγραφή ενός βιογραφικού ή την παραγωγή λίστας δημοσιεύσεων.

3.2 Εργαλεία και Τεχνολογίες

Στις παρακάτω παραγράφους θα γίνει περιγραφούν οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής

3.2.1 Η Γλώσσα Προγραμματισμού PHP

Η PHP είναι μία script γλώσσα προγραμματισμού που σχεδιάστηκε για τη δημιουργία δυναμικών ιστοσελίδων και διαδικτυακών εφαρμογών. Είναι μια server - side γλώσσα προγραμματισμού (εκτελείτε στο server και όχι στο browser σαν την απλή html) και συνήθως είναι συνοδευτική της html βοηθώντας σε λειτουργίες όπως διάβασμα και εγγραφή βάσεων δεδομένων, εγγραφή σε αρχεία, δημιουργία εικόνων, σύνδεση με απομακρυσμένους υπολογιστές κ.α.

Η PHP ξεκίνησε το 1995 από Rasmus Lerdorf, για προσωπική χρήση και μετά τη δεύτερή της έκδοση το 1997 (βασισμένη στη γλώσσα προγραμματισμού C), άρχισε να διαδίδεται και να χρησιμοποιείται σε όλο και περισσότερες ιστοσελίδες. Σύμφωνα με στατιστικές, οι περισσότερες ιστοσελίδες χρησιμοποιούν τις εκδόσεις 4 και 5 που κυκλοφόρησαν το 1998 και 2004 αντίστοιχα, ενώ ήδη έχουν δοθεί σε κυκλοφορία, δοκιμαστικές εκδόσεις της PHP 6. [35]

Από τη στιγμή που χρειάστηκε να αναπτυχθεί μια διαδικτυακή εφαρμογή, η PHP δε θα μπορούσε να είναι τίποτε άλλο παρά μόνο η πρώτη επιλογή. Τα αναρίθμητα tutorials που υπάρχουν στο διαδίκτυο αλλά και οι συμβουλές που μπορεί, ένας μαθητευόμενος, να πάρει από έμπειρους προγραμματιστές που προσφέρουν τις γνώσεις τους μέσο διαφόρων forum, e-mail lists, αλλά και της επίσημης ιστοσελίδας της PHP (http://www.php.net), κάνει την εκμάθηση της γλώσσας πιο γρήγορη και πιο εύκολη. Παρ' ότι πρόκειται για μία scripting γλώσσα, οι ομοιότητές της με τις C τύπου γλώσσες προγραμματισμού, όπως η C++ και η Java, βοηθά τους ήδη έμπειρους προγραμματιστές, σε αυτές τις γλώσσες, να κατανοήσουν γρήγορα το συντακτικό της.

Επίσης η PHP διαθέτει ένα άκρος κατατοπιστικό documentation, με πληθώρα από παραδείγματα και με τη δυνατότητα ανάγνωσής του, μεταφρασμένο σε αρκετές διαφορετικές γλώσσες. Πολλές και χρήσιμες είναι και οι μέθοδοι που έχουν γραφτεί για τη γλώσσα, για τις οποίες έχουν συγγραφεί και έχουν δημοσιευθεί αρκετές εναλλακτικές εκδόσεις, για όλες τις ανάγκες και δωρεάν προς όφελος της παγκόσμιας κοινότητας.

Εκτός από τις ευκολίες εκμάθησης στα πλεονεκτήματα της PHP μπορούν να προστεθούν το ότι είναι πολύ σταθερή σαν γλώσσα, αρκετά ασφαλής, γρήγορη στο να παράγει αποτελέσματα και φυσικά είναι δωρεάν.

Η σταθερότητα της γλώσσας προκύπτει από το γεγονός ότι είναι μια open source γλώσσα. Αυτό σημαίνει ότι όπως συμβαίνει και με οτιδήποτε το open source, ισχύει το «δούνε και λαβείν». Τα έμπειρα μέλη της open source κοινότητας συμβάλουν συνεχώς στη διόρθωση διαφόρων bugs και ακόμη όπως ήδη ανέφερα, προσφέρει μια συνεχή υποστήριξη μέσο blogs, forums, e-mail lists κλπ., ενώ βελτιώνουν συνεχώς το κώδικα ούτως ώστε να συμβαδίζει με τις σύγχρονες ανάγκες της αγοράς.

Ο όρος ασφαλής αναφέρεται στο ότι, η PHP παρέχει πολλαπλά επίπεδα ασφαλείας, τα οποία μπορεί να επεξεργαστεί κάποιος μέσα από το .ini αρχείο.

Η αναφορά της ως γρήγορη γλώσσα, σημαίνει ότι δε χρησιμοποιεί πολλές από τις πηγές του συστήματος και κατ' επέκταση δε καθυστερεί τις άλλες διαδικασίες που τρέχουν ταυτόχρονα.

Τέλος, η γλώσσα δεν απαιτεί την εγκατάσταση κάποιας άλλη εφαρμογής ή βοηθητικών στοιχείων, αλλά οι εφαρμογές εμφανίζονται κανονικά, με τη βοήθεια της HTML, σε ένα browser, όπως και μια απλή ιστοσελίδα. Επίσης συνεργάζεται άψογα με το MySQL, ένα επίσης διαδεδομένο open source σύστημα διαχείρισης βάσεων δεδομένων, ενώ διαθέτει μια πολύ καλή και εύκολη στη χρησιμοποίησή της βιβλιοθήκη για τη διαχείριση XML αρχείων, όπως και για τη διαχείριση γραφικών και κρυπτογράφησης.

Η PHP υποστηρίζεται από διάφορους servers όπως Apache, IIS, Roxen, THTTPD και AOLServer και επίσης τρέχει σαν cgi module. Επιπλέον, εκτός του MySQL, διάφορα συστήματα βάσεων δεδομένων είναι διαθέσιμα όπως, MS SQL, Oracle, Postgre SQL, ενώ στη περίπτωση που ένα σύστημα βάσε-

ων δεδομένων δεν υποστηρίζεται υπάρχει η δυνατότητα να χρησιμοποιηθεί η ODBC επιλογή. [23]

3.2.2 MySQL: Σύστημα Διαχείρισης ΒΔ

Το MySQL είναι ένα open source σύστημα διαχείρισης βάσεων δεδομένων που αρχικά αναπτύχθηκε από τους Michael Widenious και David Axmark το 1994. Ακολούθησαν αρκετές εκδόσεις μέχρι να φτάσουμε στην παρούσα, 5η έκδοση που κυκλοφόρησε το 2005, ενώ σε εξέλιξη βρίσκεται και η 6η έκδοση του MySQL.

Πρόκειται για ένα αυτοδύναμο ΣΔΒΔ που χρησιμοποιείται ευρέως σε διαδικτυακές εφαρμογές για τους εξής λόγους:

- Είναι ένα δωρεάν σύστημα, καθότι open source και μπορεί να χρησιμοποιηθεί από τον οποιοδήποτε για ιδιωτική αλλά και για εμπορική χρήση. (Διάφοροι οργανισμοί όπως η Google, το Facebook και το Youtube χρησιμοποιούν MySQL για να καλύψουν τις ανάγκες τους).
- Προσφέρει ένα πολυχρηστικό περιβάλλον, όπου ένας οι περισσότεροι χρήστες μπορούν να αλληλεπιδράσουν ταυτόχρονα με τις βάσεις δεδομένων τους.
- Διαθέτει τα περισσότερα χαρακτηριστικά που διαθέτουν και άλλα εμπορικά συστήματα όπως η Oracle και είναι εξαιρετικά εύκολο στην εγκατάσταση του.
- Είναι ένα γρήγορο σύστημα που μπορεί να υποστηρίξει περισσότερες από 50 εκατομμύρια εγγραφές και η ομάδα ανάπτυξης του συστήματος προσθέτει καινούρια χαρακτηριστικά ενώ συνάμα προσπαθεί να διατηρήσει τη ταχύτητα του συστήματος.
- Έχει επιλεχθεί σαν η κύρια βάση δεδομένων για τον Apache server και συνεργάζεται άψογα με την PHP, ενώ τρέχει σε περισσότερες από 20 πλατφόρμες.
- Είναι εύκολη στην εκμάθηση και με εργαλεία όπως το phpmyadmin μπορεί κάποιος να διαχειριστεί εύκολα και γρήγορα τις διάφορες βάσεις δεδομένων που έχει δημιουργήσει. [23]

3.2.3 Λοιπές Τεχνολογίες και Εργαλεία

Για την ανάπτυξη της εφαρμογής, χρησιμοποιήθηκαν και άλλες γλώσσες προγραμματισμού όπως η HTML, η Javascript και η γλώσσα μορφοποίησης CSS. Χρησιμοποιήθηκαν επίσης αρκετά εργαλεία τα οποία βοήθησαν στην ανάπτυξη αλλά και στη σωστή λειτουργία της εφαρμογής.

XHTML

Συγκεκριμένα η PHP πλαισιώθηκε από την XHTML, μια πιο αυστηρή γλώσσα σήμανσης από την HTML που ανήκει στην οικογένεια των xml γλωσσών σήμανσης.

Οι κύριες διαφορές της από την HTML είναι:

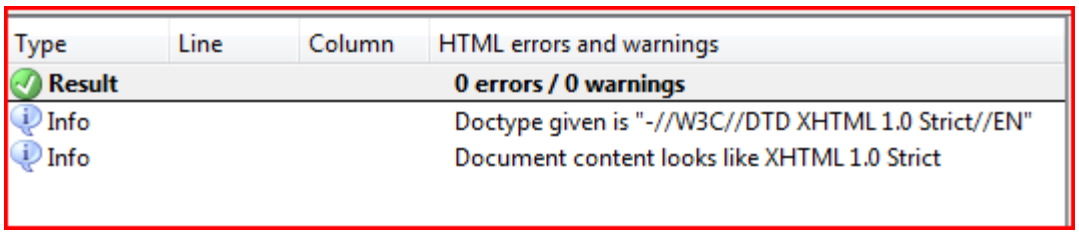
- Κάθε ιδιότητα θα πρέπει να είναι κατάλληλα εμφωλευμένη μέσα στις ετικέτες σήμανσης, π.χ.
  - Λάθος σήμανση
  - Σωστή σήμανση
- Κάθε σήμανση θα πρέπει να κλείνει κατάλληλα, π.χ.
  - Λάθος σήμανση
  - Σωστή σήμανση
- Ακόμη και τα κενά στοιχεία θα πρέπει να κλείνουν κατάλληλα, π.χ.
  - Λάθος σήμανση
  - Σωστή σήμανση
- Όλες οι σημάνσεις θα πρέπει να είναι γραμμένες σε μικρά γράμματα.
  - <BODY>
    <P> Λάθος σήμανση </P>
    </BODY>
  - <body>
     Σωστή σήμανση
    </body>
- Όλα τα στοιχεία θα πρέπει να έχουν ένα στοιχείο ρίζα, δηλαδή όλα τα στοιχεία παιδιά θα πρέπει να είναι σωστά εμφωλευμένα σε ζεύγη (άνοιγμα και κλείσιμο σήμανσης) μέσα στο στοιχείο ρίζα <html>, π.χ.
  - <html>
    <head> … </head>
    <body> … </body>

```
</html>
```

Επιπλέον, θα πρέπει να ισχύει πάντα η παραπάνω δομή.

- Δε θα πρέπει να παραλείπεται καμία βασική σήμανση από τη δομή του αρχείου, π.χ.
    - <head> Λάθος σήμανση </head>
    - <head> <title>Σωστή σήμανση </title></head>
- Κάθε XHTML αρχείο θα πρέπει να φέρει την εξής δήλωση στην πρώτη γραμμή:
    - <!DOCTYPE το Doctype τοποθετείται εδώ >
      <html xmlns="http://www.w3.org/1999/xhtml"> [5], [24]

Τα παραπάνω συντελούν τα βασικά χαρακτηριστικά ενός XHTML αρχείου. Για το έλεγχο της σωστής δομής των αρχείων χρησιμοποιήθηκε μια επέκταση του Mozilla Firefox, το HTML Validator 0.8.5.8, βλ. εικόνα 3-1.



Εικόνα 3-1 Ο Html Validator στο Mozilla Firefox.

Javascript

Η Javascript είναι μια script γλώσσα προγραμματισμού η οποία βασίζεται στο συντακτικό της γλώσσας προγραμματισμού C και χρησιμοποιείται σε συνδυασμό με τη γλώσσα σήμανσης HTMLγια την δημιουργία δυναμικών ιστοσελίδων.

Η διαφορά της από την PHP είναι, ότι ο κώδικας εκτελείται στην μεριά του χρήστη και όχι στη μεριά του server, όπως συμβαίνει με την PHP.

Ο κώδικας μπορεί να είναι ενσωματωμένος στο ίδιο το HTML αρχείο, εμφωλευμένος στην σήμανση <script>, συνήθως στο head μέρος του HTML αρχείου, π.χ.

```
<script language="javascript">
        Document.write('Hello world!');
</script>
```

ενώ υπάρχει η δυνατότητα να χρησιμοποιείται ένα αρχείο javascript, το οποίο φέρει τους χαρακτήρες .js στο τέλος του ονόματός του και καλείται από το HTML αρχείο (όπως συμβαίνει και με στην παρούσα εφαρμογή), με τον εξής κώδικα:

```
<script type='text/javascript' src='js/scripts.js'></script>
```

Πάντως η Javascript είναι περιορισμένων δυνατοτήτων και δε παρέχει σύνδεση με βάση δεδομένων.

jQuery

Η jQuery είναι μια ελαφριά Javascript βιβλιοθήκη, που επικεντρώνεται στη καλύτερη αλληλεπίδραση μεταξύ της Javascript και της HTML. Κυκλοφόρησε σαν open source script γλώσσα το 2006 από τον John Resig και δημιουργήθηκε για να δώσει μια πιο εντυπωσιακή εμφάνιση στις ιστοσελίδες, προσθέτοντας κίνηση στις αλληλεπιδράσεις και μια πιο απλοποιημένη σύνδεση με την AJAX, για την ανάπτυξη γρηγορότερων διαδικτυακών εφαρμογών.

Για να γράψει κάποιος jQuery μεθόδους, θα πρέπει πρώτα να συμπεριλάβει τη jQuery βιβλιοθήκη στο HTML αρχείο του. Πρέπει να τοποθετήσει τον εξής κώδικα στο head μέρος του HTML αρχείου.

```
<script type="application/javascript" src="jQuery.js"></script>
```

Στο παραπάνω κώδικα, jQuery.js είναι η βιβλιοθήκη jQuery και στην περίπτωση αυτή, είναι τοποθετημένη στον ίδιο κατάλογο με το HTML αρχείο. Σε άλλη περίπτωση γράφεται η σχετική ή η πλήρης διαδρομή του αρχείου. Επίσης μπορεί να χρησιμοποιηθεί και το “Google Ajax Library API”, γράφοντας τον εξής κώδικα στο head μέρος του HTML αρχείου,

```
<script type="application/javascript" src="http://www.google.com/jsapi"></script>
<script>
      google.load("jquery", "1.3.2");
</script>
```

Διάσημοι διαδικτυακοί χώροι όπως το Google.com, το Twitter.com και πολλοί άλλοι χρησιμοποιούν τη jQuery, ενώ η Microsoft την έχει συμπεριλάβει ήδη στο Visual Studio και στο ASP.NET MVC. [26]

CSS

Η CSS είναι μια γλώσσα «προγραμματισμού», που στην ουσία είναι συμπληρωματική μιας γλώσσας σήμανσης όπως η HTML και χρησιμοποιείται στον έλεγχο της εμφάνισης ενός εγγράφου, γραμμένο σε γλώσσα σήμανσης, δηλαδή είναι υπεύθυνη για το στυλ των στοιχείων που εμφανίζονται σε μια ιστοσελίδα.

Οι CSS εντολές μπορούν να γραφτούν με τρεις διαφορετικούς τρόπους.

1. Εμφωλευμένες μέσα σε μια σήμανση χρησιμοποιώντας την ιδιότητα style, π.χ.

   <td style="border-style: solid;"> Εμφανίζει ένα περίγραμμα γύρο από το κελί ενός πίνακα </td>

2. Ομαδοποιημένες συνήθως στο head μέρος ενός HTML αρχείου και εμφωλευμένες στην σήμανση <style> … </style>

3. Ή σε ξεχωριστό αρχείο (‘όπως συμβαίνει και στην παρούσα εφαρμογή) το οποίο φέρει τη κατάληξη .css και καλείται στο head μέρος του HTML αρχείου με τον εξής τρόπο:

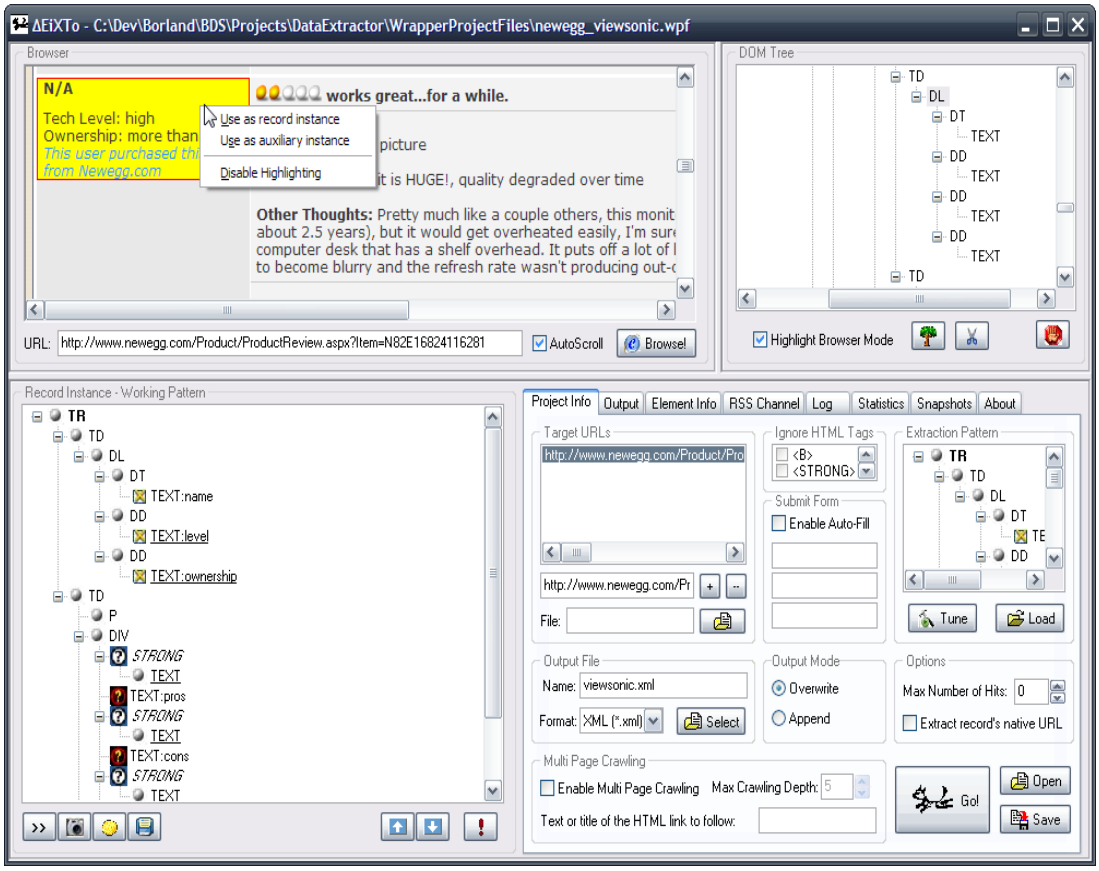
   <link rel="stylesheet" href="το path του αρχείου σε σχέση με το html αρχείο" type="text/css" />

Επίσης με την βοήθεια των CSS μπορούν να διαχειριστούν καλύτερα μεγαλύτερες ιστοσελίδες με κοινά στυλ εμφάνισης, ομαδοποιώντας τις σημάνσεις με την ιδιότητά τους, χρησιμοποιώντας ένα μοναδικό χαρακτηριστικό (id) ή ακόμη ομαδοποιώντας τις σε κλάσεις. Κατά αυτόν τον τρόπο, μία αλλαγή σε μια css ιδιότητα σημαίνει και αλλαγή με εύρος όλες τις σελίδες ενός ιστοτόπου. [5]

DEiXTo

Το DEiXTo (Data Extraction Tool – http://deixto.com) αναπτύχθηκε από τους Κ.Ντόνα και Φ.Κόκκορα (ομάδα “LPIS” του Τμήματος Πληροφορικής του ΑΠΘ) και διατίθεται σε δύο μορφές, μία για περιβάλλον Windows με γραφικό περιβάλλον και με πλήρη υποστήριξη δημιουργίας και εκτέλεσης κανόνων εξαγωγής web περιεχομένου (web content extraction rules) και μία διαπλατφορμική σε γραμμή εντολών (command line) που υποστηρίζει μόνο εκτέλεση κανόνων εξαγωγής που έχουν φτιαχτεί από την γραφική έκδοση.

Το DEiXTo είναι ένα πολύ ισχυρό εργαλείο εξαγωγής δεδομένων από ι-στοσελίδες χρησιμοποιώντας κανόνες εξαγωγής που ο χρήστης μπορεί να δημιουργήσει με τη βοήθεια του ίδιου του εργαλείου. Η διανομή του με γρα-φικό περιβάλλον είναι αρκετά φιλική προς το χρήστη. Διαθέτει μία οθόνη web browser και στο δεξιό μέρος αναπαριστά την ιστοσελίδα αυτή υπό μορφή DOM (Document Object Model) δέντρου, βλ. εικόνα 3-2.



Εικόνα 3-2 Το περιβάλλον του εργασίας του DEiXTo. Διακρίνεται ο browser (πάνω αριστερά) και το DOM δένδρο (πάνω δεξιά).

Με αυτό τον τρόπο ο χρήστης μπορεί να επιλέξει τα στοιχεία, σε μια ιστο-σελίδα, που τον ενδιαφέρουν και να δημιουργήσει το δικό του κανόνα εξαγω-γής, σχετικά εύκολα, ενώ υποστηρίζει ακόμη και χρησιμοποίηση κανονικών εκφράσεων (βλ. παράρτημα). Οι κανόνες αποθηκεύονται στον υπολογιστή του χρήστη υπό μορφή “wpf” αρχείου (αρχείο με XML δομή) και μπορούν να χρησιμοποιηθούν αναλόγως για την εξαγωγή πληροφορίας σε διάφορες μορφές αρχείων όπως, XML ή σε χωρισμένο με tab κείμενο ενώ μπορεί να δημιουργήσει και RSS έξοδο. Κατά την εκτέλεση κανόνων το λογισμικό έχει τη δυνατότητα να ακολουθεί συνδέσμους σε βάθος που μπορεί να ορίσει ο

χρήστης, κάτι που είναι ιδιαίτερα χρήσιμο σε περιπτώσεις όπου ο χρήστης ενδιαφέρεται να εξάγει πληροφορία από μία μηχανή αναζήτησης, όπως συμβαίνει στη παρούσα εργασία και τα αποτελέσματα βρίσκονται σε πολλές σελίδες [25]. Σε τέτοιες περιπτώσεις το DEiXTo επιτρέπει να ξεπεραστεί ο περιορισμός στο πλήθος αποτελεσμάτων που θέτει το Google API.

Notepad ++

Το notepad ++ ήταν ο μόνος editor που χρησιμοποιήθηκε για τη συγγραφή του κώδικα όλων των αρχείων.

Το notepad ++ είναι ένας open source text editor, ελεύθερος στην χρησιμοποίησή του για οποιονδήποτε σκοπό (ιδιωτικό ή εμπορικό) και τρέχει σε περιβάλλον MS Windows ενώ είναι μεταφρασμένος σε αρκετές γλώσσες. Στην ουσία πρόκειται για μία επέκταση του γνωστού notepad που υπάρχει σε όλες τις εκδόσεις των MS Windows, ο οποίος υποστηρίζει πολλές γλώσσες προγραμματισμού. [27]

WAMP και phpMyAdmin

Για την ανάπτυξη της εφαρμογής και καθώς πρόκειται για μία διαδικτυακή εφαρμογή, η χρησιμοποίηση και σαφώς η εγκατάσταση ενός server ήταν αναγκαία. Η τελική επιλογή ήταν το πακέτο WAMP (Windows Apache MySQL PHP ή Perl ή Python) για το λόγο ότι με μια εύκολη εγκατάσταση προσφέρει εκτός του server, τη Βάση Δεδομένων MySQL και τη γλώσσα προγραμματισμού PHP. Με μία απλή εγκατάσταση όλα τα βασικά στοιχεία που χρειάζονταν για την εφαρμογή ήταν διαθέσιμα. Επίσης, εγκαταστάθηκε μαζί με το υπόλοιπο πακέτο, το εργαλείο phpmyadmin, ένα εργαλείο με γραφικό περιβάλλον για τη διαχείριση της βάσης δεδομένων MySQL.

Ο χρήστης αποθηκεύει τα αρχεία του στο κατάλογο “www” και μπορεί να τα εμφανίσει του μέσω ενός απλού browser, γράφοντας στο πεδίο της διεύθυνσης, http://localhost ή http://127.0.0.1 και αναλόγως προσθέτοντας τη διαδρομή που είναι τοποθετημένα τα αντίστοιχα αρχεία που θέλει να εμφανίσει κάθε φορά, με κατάλογο ρίζα να είναι ο “www”. Με αυτό τον τρόπο μπορεί να έχει πρόσβαση και στο εργαλείο phpmyadmin γράφοντας την εξής διεύθυνση στον browser: http://localhost/phpmyadmin .

Τέλος, η αντίστοιχη εγκατάσταση του WAMP υπάρχει και για άλλες πλατφόρμες με τις εξής ονομασίες:

- MAMP για Apple
- LAMP για Linux
- SAMP για Solaris
- και FAMP για FreeBSD [28]

GIMP

Το GIMP (GNU Image Manipulation Program) είναι ένα πρόγραμμα για την επεξεργασία εικόνων και γραφικών. Πρόκειται για ένα open source τύπου Photoshop και πολλές από τις λειτουργίες του, βασίζονται στο ήδη ευρέως διαδεδομένο Adobe Photoshop. Το GIMP υστερεί αρκετά σε θέμα δυνατοτήτων από τον ανταγωνιστή του, το Photoshop, αλλά υπερτερεί στο θέμα τιμής καθώς διανέμεται δωρεάν για οποιαδήποτε χρήση και κάποιος μπορεί να το κατεβάσει από την επίσημη ιστοσελίδα του, http://www.gimp.org/.

Το πρόγραμμα ξεκίνησε το 1995 από τους Spencer Kimball και Peter Mattis και η εργαλειοθήκη που αρχικά σχεδιάστηκε για το GIMP, έχει βρει εφαρμογή σε πολλά περιβάλλοντα εργασίας όπως τα GNOME και Xfce. Για τις ανάγκες της εφαρμογής, χρησιμοποιήθηκε η έκδοση 2.6, ενώ κατά τη διάρκεια της συγγραφής του κειμένου της πτυχιακής εργασίας κυκλοφόρησε η 2.7 έκδοση ή οποία είναι το πρώτο βήμα για τη δημιουργία μιας σταθερής 2.8 έκδοσης που πρόκειται να ακολουθήσει. [29]

Browsers

Όταν ένας προγραμματιστής αναπτύσσει μια διαδικτυακή εφαρμογή ή μία ιστοσελίδα, αυτό που πρέπει πάντοτε να έχει κατά νου είναι, η εφαρμογή αυτή να εμφανίζεται κατά τον ίδιο τρόπο σε όλους τους browsers (multi browser εφαρμογή). Βέβαια, για το λόγο ότι αρκετοί χρήστες δεν αναβαθμίζουν τις παλιές εκδόσεις των browsers που χρησιμοποιούν ένα ακόμη πρόβλημα που δημιουργείται για τους προγραμματιστές είναι ότι η εφαρμογή πρέπει να είναι “cross browser”.

Με τον όρο “cross browser” εννοούμε μια διαδικτυακή εφαρμογή ή μία ιστοσελίδα να απεικονίζεται και να λειτουργεί σωστά με κάθε browser και κάθε έκδοση αυτού. Πρόκειται για αρκετά επίπονη διαδικασία, καθώς οι παλαιότερες εκδόσεις των browsers δεν υποστηρίζουν τις καινούριες τεχνολογίες.

Για της ανάγκες της εφαρμογή που αναπτύχθηκε, χρησιμοποιήθηκαν πέντε διαφορετικοί browsers, για να ελεγχθεί η σωστή εμφάνισή της. Οι browsers αυτοί είναι οι εξής:

- Mozilla Firefox 3.0 και στη συνέχεια 3.5

  Είναι ένας open source browser, που κυκλοφόρησε το 2004 και σύμφωνα με στατιστικές, είναι ο δεύτερος σε προτίμηση browser πίσω από τον δημοφιλή Internet Explorer της Microsoft. Ο Firefox εκτός από ένας καλός browser είναι και ένα καλό εργαλείο για τους προγραμματιστές διαδικτυακών εφαρμογών και όχι μόνο. Συνεχώς κυκλοφορούν επεκτάσεις και πρόσθετες εφαρμογές που ενσωματώνονται στο browser και βοηθούν το χρήστη σε κάθε είδους εργασία. Κατά την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν κυρίως δύο επεκτάσεις, οι οποίες βοήθησαν στην διόρθωση λαθών, το Firebug και το HTML Validator. [30]

- Internet Explorer 7 και 8

  Κυκλοφόρησε το 1995 από τη Microsoft και το 1999 ήταν ο πιο ευρέως χρησιμοποιημένος browser με ποσοστό 95%. Από το 2004 και μετά η χρήση του μειώνεται σταδιακά, χωρίς όμως να έχει χάσει τη πρωτοπορία του στην επιλογή των χρηστών. Παρόλα αυτά κατά την ανάπτυξη της εφαρμογής, προέκυψαν αρκετά προβλήματα εμφάνισης διαφόρων στοιχείων, που οφείλονται κυρίως στην CSS ιδιότητα padding που διορθώθηκε με τη χρήση μιας επιπλέον ιδιότητας, της margin και διαφόρων άλλων τεχνικών αναλόγως το πρόβλημα που προξενούσε κάθε φορά. Βέβαια τα γνωστά προβλήματα του Internet Explorer με τα CSS, διορθώθηκαν κατά κάποιο τρόπο στη $8^{η}$ έκδοση του browser, αλλά είναι πολλοί οι χρήστες που χρησιμοποιούν ακόμη την $7^{η}$ ή και την $6^{η}$ έκδοση του Internet Explorer. [31]

- Safari 4

  Κυκλοφόρησε το 2003 από την Apple και συμπεριλαμβάνεται από τότε μαζί με το λογισμικό Mac OS. Η Windows εγκατάστασή του δίνει τη δυνατότητα και στους χρήστες των Microsoft Windows να τον δοκιμάσουν, ενώ είναι ο προκαθορισμένος browser, στις συσκευές της Apple, iPhone και iPod Touch. [33]

- Chrome 3

  Ο Chrome κυκλοφόρησε από τη Google το 2008 σαν open source εφαρμογή δίνοντας τον κώδικα ακόμη και για την V8 Javascript Μηχανή του. Με την κίνηση αυτή προσπάθησαν να κάνουν παρακινήσουν τους ενδιαφερόμενους να συμμετάσχουν σε μια μεταφορά του browser και σε άλλα λογισμικά πέρα των Windows, όπως Linux και Mac OS και επίσης έχοντας κατά νου ότι οι υπόλοιποι browsers θα συμπεριλάβουν τη V8 Μηχανή, με σκοπό να βοηθηθούν κατά πολύ οι διαδικτυακές εφαρμογές. Ο Chrome είναι γρήγορος και αρκετά ασφαλής, καθώς διαθέτει δυο «μαύρες λίστες» μία για “phishing” ιστοσελίδες και μία για “malware”. Παρόλα αυτά και σαν σχετικά νέος browser έχει και τα μειονεκτήματά του, όπως το ότι δε μπορεί να διαβάσει τα RSS Feed χωρίς τη βοήθεια επεκτάσεων. [32]

- Opera 9 και 10

  Ο Opera κυκλοφόρησε το 1994 από τη Νορβηγική εταιρεία τηλεπικοινωνιών Telenor. Παρόλο που μέχρι τη $2^{η}$ έκδοσή του δεν ήταν διαθέσιμος στο κοινό, ήταν ήδη γνωστός για την ευκολία του στο να περιηγείται σε περισσότερες από μία ιστοσελίδες ταυτόχρονα και ήταν ο πρώτος που προσπάθησε να συμπεριλάβει και να ακολουθήσει κατά γράμμα όλα τα W3C πρότυπα. Παρόλα αυτά ο Opera δεν είναι ευρέως διαδεδομένος και χρησιμοποιείται κυρίως για την περιήγηση στο διαδίκτυο μέσο κινητών τηλεφώνων και άλλων φορητών συσκευών. [34]

4 Περιγραφή της Εφαρμογής

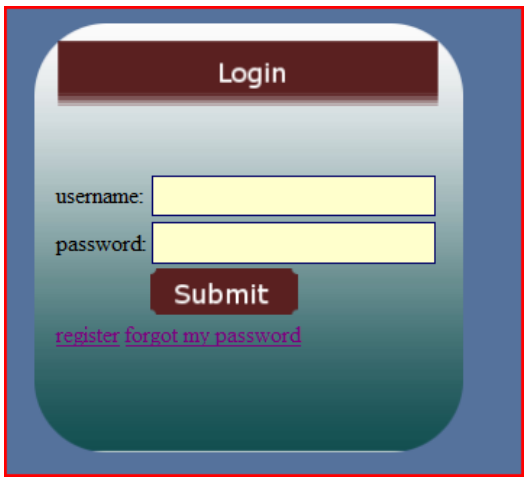
Η εφαρμογή που αναπτύχθηκε προσπαθεί να δώσει μία λύση στο πρόβλημα της διαχείρισης των βιβλιογραφικών αναφορών. Θα περιγραφεί η λειτουργία της εφαρμογής καθώς και η αρχιτεκτονική της. Θα παρουσιαστούν και θα αναλυθούν, σημαντικά κομμάτια του κώδικα, η βάση δεδομένων που αναπτύχθηκε με τη MySQL, καθώς και διάφορα συμπληρωματικά στοιχεία που χρησιμοποιούνται, όπως είναι τα Google Charts.

4.1 Λειτουργία της Εφαρμογής

Όπως ήδη αναφέρθηκε, πρόκειται για μία διαδικτυακή εφαρμογή που διαχειρίζεται έγγραφα και βιβλιογραφικές αναφορές, ενός συγγραφέα. Είναι μία πολύχρηστική (multiuser) εφαρμογή όπου ο κάθε χρήστης, μπορεί να εισέλθει στο σύστημα με τη χρήση ενός ονόματος χρήστη και ενός κωδικού.

4.1.1 Είσοδος

Ο χρήστης για την είσοδο του στο σύστημα, χρειάζεται ένα έγκυρο όνομα χρήστη και ένα κωδικό. Αυτά τα στοιχεία τα καταχωρεί ο ίδιος κατά την εγγραφή του.



Εικόνα 4-1 Login φόρμα στην αρχική σελίδα

Ο χρήστης μπορεί να κάνει την εγγραφή του στην αντίστοιχη σελίδα εγγραφών, πατώντας στο σύνδεσμο “register” (βλ. εικόνα 4-1) και συμπληρώνοντας σωστά την φόρμα εγγραφής. Σε περίπτωση που δε θυμάται το κωδικό του, μπορεί να τον ανακτήσει, πατώντας στο σύνδεσμο “forgot my password”, χρησιμοποιώντας την διεύθυνση e-mail του που χρησιμοποίησε και κατά την εγγραφή του και ένα μήνυμα με το κωδικό του, αλλά και το όνομα χρήστη, θα του σταλεί στη διεύθυνση αυτή.

Στη πραγματικότητα, δεν επανακτάται ο ίδιος κωδικός που είχε εισάγει ο χρήστης κατά της εγγραφή του, αλλά ο παλιός διαγράφεται και παράγεται ένας καινούριος. Πρόκειται για ένα ισχυρό, 12ψήφιο κωδικό. Με τον όρο ισχυρό εννοείται ένας κωδικός που περιέχει χαρακτήρες μικρούς και κεφαλαίους, αριθμούς και σύμβολα. Στη συνέχεια, ο χρήστης μπορεί να τον αλλάξει με κάποιον άλλο πιο εύκολο, για να τον απομνημονεύσει, από το μενού “profile”, βλ. εικόνα 4-2.

Μετά την είσοδό του, ο χρήστης βρίσκεται πλέον στο κύριο μέρος της εφαρμογής όπου, πλοηγούμενος από το μενού θα μπορεί να διαχειριστεί τα δημοσιευμένα του έγγραφα, τις βιβλιογραφικές αναφορές που έχουν γίνει σε αυτά, καθώς και να επωφεληθεί από τις υπόλοιπες λειτουργίες που του προσφέρει η εφαρμογή.

Εικόνα 4-2 Οι επιλογές του μενού της εφαρμογής

4.1.2 Κύρια Λειτουργία

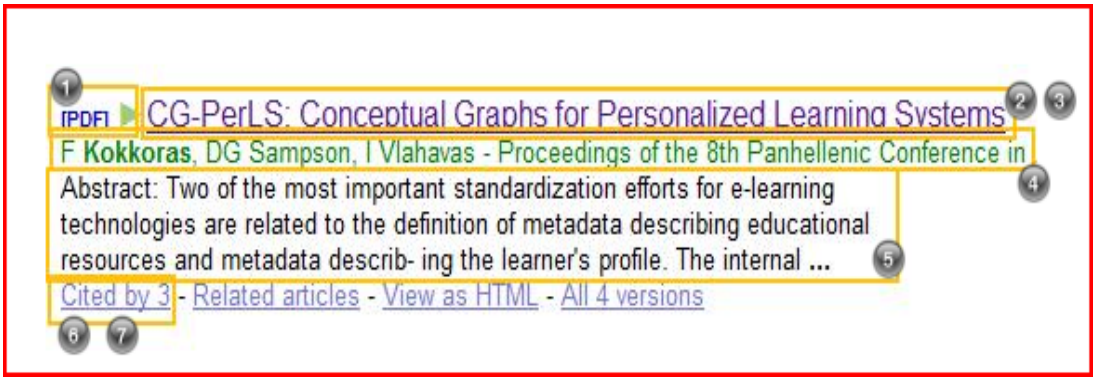
Η κύρια λειτουργία της εφαρμογής είναι να διαχειρίζεται δημοσιευμένα έγγραφα ενός συγγραφέα και τις αναφορές που έχουν γίνει σε αυτά. Ο χρήστης – συγγραφέας, πρέπει πάλι να ξεδιαλύνει τα σωστά από τα λάθος αποτελέσματα, αλλά με τη μόνη διαφορά ότι από τη δεύτερη αναζήτηση και για τις υπόλοιπες δε χρειάζεται να ασχοληθεί ξανά με αποτελέσματα που έχει δει κατά τις προηγούμενες αναζητήσεις του. Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, όταν ο συγγραφέας θελήσει να αναζητήσει αναφορές σε δημοσιεύσεις του, χρειάζεται να ξαναπάρει αποφάσεις για αποτελέσματα που έχει συναντήσει κατά το παρελθόν. Αυτή η διαδικασία του κοστίζει σε χρόνο και κόπο, ενώ υπάρχει η περί-

πτωση να υποπέσει σε λάθη, προσπαθώντας να προσπεράσει γρήγορα τα παλιά αποτελέσματα.

Η εφαρμογή αυτοματοποιεί αυτή τη διαδικασία μέσω διαφόρων ελέγχων που κάνει για να διαπιστώσει αν ένα αποτέλεσμα πρόκειται για καινούριο ή για αποτέλεσμα που έχει ξανασυναντήσει στον παρελθόν. Με τη βοήθεια του εργαλείου εξαγωγής πληροφοριών DEiXTo, αποθηκεύει τα αποτελέσματα της αναζήτησης σε ένα XML αρχείο που παράγεται από τη βοηθητική αυτή εφαρμογή.

Το XML αποθηκεύει τις εξής πληροφορίες από κάθε αρχείο:

- Το τύπο του αποτελέσματος, π.χ. BOOK, PDF, αν αυτός υπάρχει
- Το τίτλο της δημοσίευσης
- Τη διεύθυνση που οδηγεί ο σύνδεσμος του τίτλου της δημοσίευσης
- Τα συνοδευτικά στοιχεία, όπως τα ονόματα των συνεργατών συγγραφέων, ο εκδοτικός οίκος, η χρονολογία δημοσίευσης κ.α.
- Ένα γενικό κείμενο πάνω στη δημοσίευση που συνοδεύει το αποτέλεσμα και είναι πολύ σημαντικό για τη διευκρίνηση του περιεχομένου του αποτελέσματος.
- Τον αριθμό των αναφορών που έχουν γίνει στη συγκεκριμένη δημοσίευση που προσδιορίζεται ως “cited by” στο Google Scholar
- Και τέλος την διεύθυνση του συνδέσμου “cited by”, που οδηγεί σε μια σελίδα αποτελεσμάτων του Google Scholar, που εμφανίζει μια λίστα με όλες τις αναφορές που έχουν γίνει στη συγκεκριμένη δημοσίευση και πληροφορίες για τη κάθε μία από αυτές



Εικόνα 4-3 Οι πληροφορίες που συλλέγονται από το DEiXTo είναι μέσα στο πορτοκαλί πλαίσιο. [21]

Όπως φαίνεται κι από την εικόνα 4-3, όλες οι σημαντικές πληροφορίες, που μπορούν να βοηθήσουν στη διαπίστωση αν ένα αποτέλεσμα είναι βιβλιογραφική αναφορά ή όχι, συλλέγονται από το DEiXTo, και αποθηκεύονται στο XML αρχείο που εξάγει.

Ο χρήστης, στη συνεχεία πρέπει να ανεβάσει το XML αρχείο στο server για να το διαβάσει η εφαρμογή και να το επεξεργαστεί. Κατά αυτή την επεξεργασία, γίνονται διάφοροι έλεγχοι για να διαπιστωθεί αν πρόκειται για κάποιο ήδη υπάρχον αποτέλεσμα, από τη δεύτερη και μετά αναζήτηση. Αυτά τα δεδομένα αποτελούν τις εγγραφές πρώτου επιπέδου.

Για να μη συλλέγονται περιττές βιβλιογραφικές αναφορές, το σύστημα δημιουργεί έναν ακόμη κανόνα εξαγωγής στοιχείων για το DEiXTo, συλλέγοντας τους συνδέσμους “cited by”, βλ. παράρτημα κανόνα 2, από κάθε εγγραφή που είναι τύπου “my paper”. Ο χρήστης μπορεί να κατεβάσει αυτό το κανόνα στον υπολογιστή του πατώντας το κουμπί “download”, να τον τρέξει στο DEiXTo και συνέχεια να ανεβάσει εκ νέου το εξαχθέν XML αρχείο.

Μετά των έλεγχο και την εκχώρηση των αποτελεσμάτων δευτέρου επιπέδου, ο χρήστης μπορεί να βρει στο μενού “raw data” όλα τα αποτελέσματα, από την πρώτη και την δεύτερη φάση ανάγνωσης των αρχείων XML, βλ. εικόνα 4-4.

Εικόνα 4-4 Τα αποτελέσματα όπως φαίνονται στο μενού “raw data”, μετά την ανάγνωση του XML αρχείου

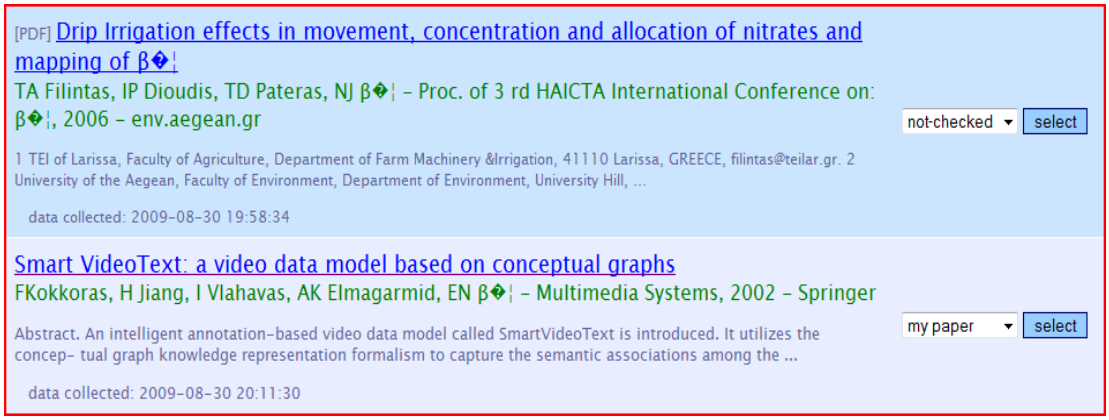
4.1.3 Κατάταξη Αποτελεσμάτων

Την πρώτη φορά που κάποιος χρήστης θα χρησιμοποιήσει το σύστημα, θα συναντήσει όλα τα εκχωρημένα στη βάση αποτελέσματα στη σελίδα “raw data”, βλ. εικόνα 4-4. Εκεί θα βρίσκει όλα τα αποτελέσματα που θα κρατά και στις επόμενες φορές που θα χρησιμοποιήσει την εφαρμογή. Θα βρίσκει στην ίδια σελίδα ακόμη και αποτελέσματα, τα οποία τα έχει αναγνωρίσει και τα έχει κατατάξει ανάλογα.

Ο χρήστης μπορεί ακόμη να ελέγξει τον σύνδεσμο ενός αποτελέσματος πατώντας στο τίτλο του και να κατατάξει τα αποτέλεσμα με ένα από τους τέσσερις τύπου αποτελεσμάτων, αναλόγως:

- Not-checked
- My paper
- My citation
- Garbage

Όλα τα αποτελέσματα στην αρχή είναι τύπου “not checked”. Τα αποτελέσματα αυτού του τύπου εμφανίζονται πρώτα, ταξινομημένα κατά φθίνουσα χρονολογική σειρά και έχουν ένα ελαφρός πιο σκούρο χρώμα φόντου, από τα από υπόλοιπα, βλ εικόνα 4-5.



Εικόνα 4-5 Η χρωματική διαφορά του φόντου, σε ένα “not checked” και ένα “checked” αποτέλεσμα

Αν ο χρήστης αλλάξει το τύπο ενός αποτελέσματος από “not checked” σε “my paper” ή “my citation”, τότε το αποτέλεσμα μεταφέρεται στο τέλος της λίστας και το χρώμα φόντου του αλλάζει επίσης και γίνεται πιο ανοιχτό. Ένα ακόμη σημαντικό στοιχείο σε αυτή τη σελίδα είναι το κουμπί “prediction”. Με το πά-

τημα αυτού του κουμπιού, η εφαρμογή χρησιμοποιώντας τον αλγόριθμο μέτρησης απόστασης και κάποιες μεθόδους, που θα αναλυθούν παρακάτω σε αυτό το κεφάλαιο, προτείνει στο χρήστη ένα τύπο για κάθε αποτέλεσμα. Η εμφάνιση της λίστας κατά αυτό τον τρόπο έχει ως σκοπό να υπενθυμίζει στο χρήστη, ότι πρέπει να πάρει αποφάσεις για τα αποτελέσματα που για αυτά που είναι καινούρια ή δεν είχε προσδιορίσει την προηγούμενη φορά.

Τα αποτελέσματα που ο χρήστης έχει προσδιορίσει σαν “my citation” ή “my paper”, μεταφέρονται επίσης στις αντίστοιχες λίστες με όλα τα άλλα αποτελέσματα που έχουν προσδιοριστεί με έναν από τους δύο αυτούς τύπους. Σε περίπτωση που ο χρήστης συναντήσει κάποια αποτελέσματα τα οποία θεωρεί ότι δεν αναφέρονται ούτε σε δημοσιεύσεις του, αλλά ούτε σε βιβλιογραφικές αναφορές πάνω σε δημοσιεύσεις του, τότε μπορεί να τα προσδιορίσει σαν “garbage” αποτελέσματα και να τα αφαιρέσει από τη λίστα των “raw data”. Αυτά τα αποτελέσματα δεν αφαιρούνται οριστικά από το σύστημα, αλλά μεταφέρονται σε μια νέα λίστα με το όνομα “garbage”. Ωστόσο αν ο χρήστης έχει προσδιορίσει λανθασμένα σαν “garbage”, ένα χρήσιμο αποτέλεσμα, τότε του δίνεται η δυνατότητα να το επαναφέρει στην “raw data” λίστα. Πρέπει πρώτα να πλοηγηθεί από το μενού στη λίστα με τα “garbage” αποτελέσματα και στη συνέχεια να πατήσει το κουμπί “restore”, που βρίσκεται δίπλα από κάθε αποτέλεσμα.

4.1.4 Δημοσιεύσεις και Αναφορές

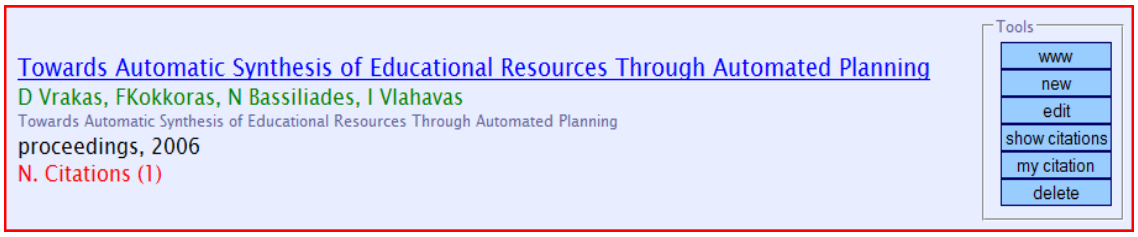
Αν ο χρήστης αναγνωρίσει ότι ένα αποτέλεσμα πρόκειται για δική του δημοσίευση, ή για αναφορά σε δική του δημοσίευση, τότε μπορεί να το προωθήσει στη κατάλληλη λίστα αποτελεσμάτων. Στην περίπτωση που πρόκειται για δημοσίευσή του, το χαρακτηρίζει ως “my paper”, ενώ στην αντίστοιχη περίπτωση που πρόκειται για βιβλιογραφική αναφορά σε δική του δημοσίευση, το χαρακτηρίζει ως “my citation”.

Αφού τα αποτελέσματα τοποθετηθούν καταλλήλως, ο χρήστης μπορεί να πλοηγηθεί από το μενού στις λίστες αυτές και να επεξεργαστεί τα αποτελέσματα περαιτέρω.

My Papers

Στο μενού “My papers”, ο χρήστης μπορεί να δει και να επεξεργαστεί όλα τα αποτελέσματα που έχει χαρακτηρίσει σαν δημοσιεύσεις του. Η μορφή που πα-

ρουσιάζονται οι δημοσιεύσεις αυτές, μοιάζει αρκετά με την μορφή που παρουσιάζονται τα αποτελέσματα στη λίστα “raw data”, έχοντας και κάποια επιπλέον συμπληρωματικά στοιχεία, βλ εικόνα 4-6.



Εικόνα 4-6 Μορφή μιας δημοσίευσης όπως παρουσιάζεται στη λίστα “my papers”

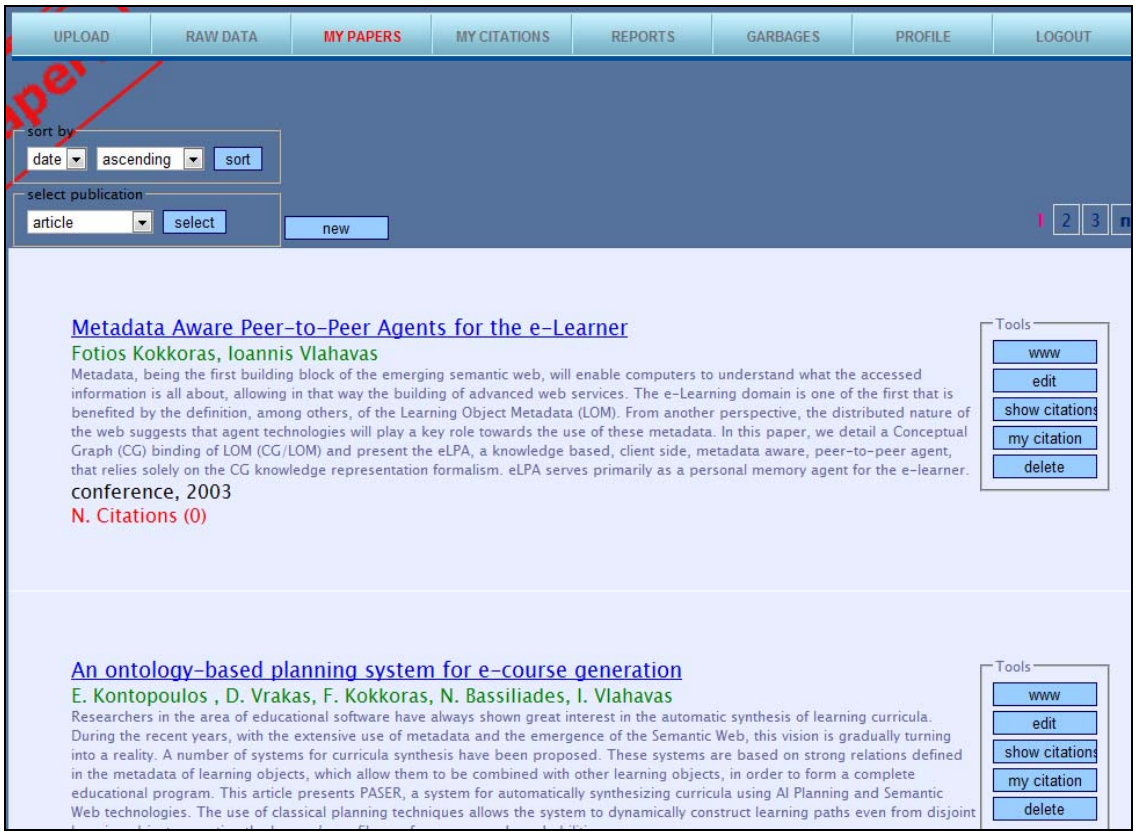
Για κάθε δημοσίευση, στο αριστερό μέρος υπάρχουν κάποιες πληροφορίες για το έγγραφο ενώ στο δεξιό μέρος υπάρχει μία σειρά από κουμπιά – εργαλεία, που βοηθούν στον έλεγχο και την επεξεργασία της δημοσίευσης ή ακόμη, σε περίπτωση λανθασμένης εκχώρησης στη λίστα “my papers”, στη διαγραφή της δημοσίευσης. Αναλυτικότερα όπως φαίνεται και στην εικόνα 4-6, στις πληροφορίες εμφανίζονται ο τίτλος της δημοσίευσης, τα ονόματα των συγγραφέων, μία γενική περιγραφή, ο τύπος έκδοσης και η χρονολογία δημοσίευσης. Επίσης στην τελευταία γραμμή των πληροφοριών, εμφανίζεται με κόκκινο χρώμα η εξής πληροφορία, “N Citations” συνοδευόμενη από ένα αριθμό που υποδεικνύει πόσες αναφορές από τη λίστα “my citations”, έχουν συσχετιστεί με τη συγκεκριμένη δημοσίευση. Στη συλλογή “tools”, υπάρχουν τα εξής εργαλεία:

- **www** - Οδηγεί σε μια διεύθυνση της δημοσίευσης, όπου ο χρήστης μπορεί να βρει περισσότερες πληροφορίες
- **edit** - Με αυτή την επιλογή, ο χρήστης μπορεί να επεξεργαστεί τις πληροφορίες των δημοσιεύσεών του
- **show citations** - Εμφανίζεται μια λίστα με τις αναφορές που έχουν συσχετιστεί με τη συγκεκριμένη δημοσίευση. Στη συγκεκριμένη λίστα παρουσιάζονται κάποιες πληροφορίες για τις αναφορές, καθώς και ένα κουμπί “remove link” δίπλα σε κάθε αναφορά για να αφαιρέσει τυχόν λανθασμένες συσχετίσεις.
- **my citation** - Σε περίπτωση που μία βιβλιογραφική αναφορά έχει τοποθετηθεί στη λίστα “my papers”, μπορεί να μεταφερθεί στη λίστα με τις βιβλιογραφικές αναφορές “my citations”, πατώντας αυτό το κουμπί.

- **delete** - Με το πάτημα αυτού του κουμπιού, ο χρήστης διαγράφει από τη λίστα με τις δημοσιεύσεις του τη συγκεκριμένη δημοσίευση.

Τέλος, ο χρήστης έχει τη δυνατότητα, να αλλάξει την εμφάνιση των δημοσιεύσεων του, ταξινομώντας τες κατά φθίνουσα ή αύξουσα σειρά, σύμφωνα με τον τίτλο τους, την ημερομηνία συλλογής των δεδομένων ή τη χρονιά έκδοσης τους. Επίσης, μπορεί να εμφανίσει μόνο τα έγγραφα ενός συγκεκριμένου τύπου δημοσίευσης, επιλέγοντας αυτόν που επιθυμεί από το αντίστοιχο drop down menu.

Ένα επίσης σημαντικό στοιχείο σε αυτή τη σελίδα είναι το κουμπί “new”. Ο χρήστης μπορεί από μόνος του να εκχωρήσει μια νέα δημοσίευση του στο σύστημα, συμπληρώνοντας τη φόρμα με τις πληροφορίες που θα εμφανιστεί, μετά το πάτημα του κουμπιού “new”. Τα παραπάνω φαίνονται στην εικόνα 4-7.



Εικόνα 4-7 Το μενού “My Papers”

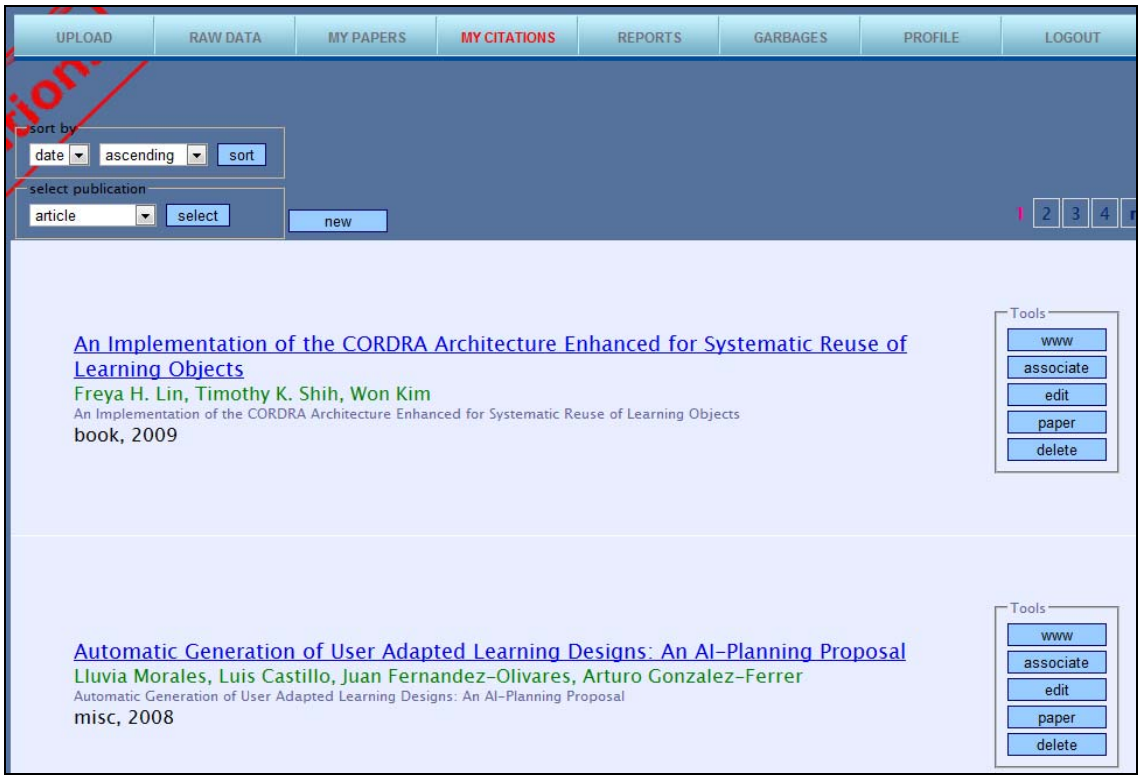
My citations

Στη λίστα “my citations”, που ο χρήστης μπορεί να βρει στο ομώνυμο μενού, έχουν τοποθετηθεί όλες οι δημοσιεύσεις που έχουν αναγνωριστεί ως βιβλιογραφικές αναφορές σε εργασίες του χρήστη - συγγραφέα.

Η παρουσίαση των πληροφοριών στο μενού “my citations” είναι ίδια με αυτή των “my papers”, όπως περιγράφηκε προηγουμένως. Τα μόνα διαφορετικά σημεία είναι, στο μέρος των πληροφοριών δε υπάρχει η τελευταία κόκκινη γραμμή με τον αριθμό των αναφορών, όπως στο “my papers” και στη πλευρά των εργαλείων - κουμπιών, αντί του “show citations”, υπάρχει το “associate”. Πιο αναλυτικά τα εργαλεία σε αυτή τη λίστα είναι τα εξής, βλ. εικόνα 4-8:

- **www** - Οδηγεί σε μια διεύθυνση της βιβλιογραφικής αναφοράς, όπου ο χρήστης μπορεί να βρει περισσότερες πληροφορίες
- **associate** - Με το πάτημα του κουμπιού αυτού, εμφανίζεται στο χρήστη μία λίστα με όλες τις δημοσιεύσεις του που υπάρχουν στη λίστα “my papers”. Ο χρήστης μπορεί να επιλέξει μία η και περισσότερες εργασίες για να συσχετίσει, καθώς υπάρχει η περίπτωση ο συγγραφέας του συγκεκριμένου εγγράφου να έχει χρησιμοποιήσει περισσότερες από μία εργασίες του χρήστη. Για να τις επιλέξει πρέπει να κρατήσει πατημένο το πλήκτρο “Ctrl” και να πατήσει με το αριστερό πλήκτρο του ποντικιού του αυτές που θέλει να συσχετίσει. Την επόμενη φορά που θα πατήσει το κουμπί “associate”, μόνο οι δημοσιεύσεις που δεν έχουν συσχετιστεί με την αναφορά θα εμφανιστούν.
- **edit** - Το κουμπί “edit”, χρησιμοποιείται για να επεξεργαστεί ο χρήστης τις πληροφορίες της βιβλιογραφικής αναφοράς.
- **paper** – Σε περίπτωση που ο χρήστης λανθασμένα έχει εισάγει μία δημοσίευσή του ως βιβλιογραφική αναφορά, τότε έχει τη δυνατότητα να της αλλάξει τύπο, πατώντας το κουμπί “paper”.
- **delete** - Με το πάτημα αυτού του κουμπιού, ο χρήστης διαγράφει από τη λίστα με τις βιβλιογραφικές αναφορές, τη συγκεκριμένη αναφορά.

Επίσης, όπως και στη σελίδα “my papers”, ο χρήστης έχει τη δυνατότητα να ταξινομήσει ή να παρουσιάσει τις βιβλιογραφικές αναφορές ενός συγκεκριμένου τύπου δημοσίευσης, χρησιμοποιώντας τα drop down menus, που υπάρχουν στο πάνω μέρος της σελίδας. Ακόμη μπορεί να εισάγει μία καινούρια αναφορά, που έχει εντοπίσει ο ίδιος, πατώντας το κουμπί “new”. Αυτή η λειτουργία μπορεί να φανεί πολύ χρήσιμη στο χρήστη, κυρίως στη περίπτωση που έχει χρησιμοποιήσει μία δημοσίευσή του σαν βιβλιογραφική αναφορά σε κάποια άλλη, επίσης δικιά του.



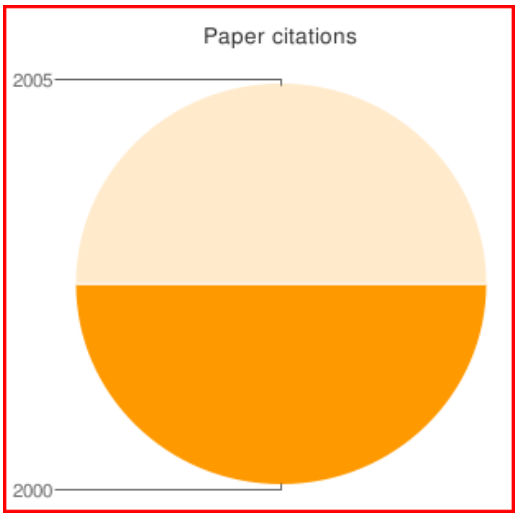
Εικόνα 4-8 Το μενού “My citations”

4.1.5 Λοιπές Λειτουργίες

Στις λοιπές λειτουργίες συμπεριλαμβάνονται η σελίδα με τις αναφορές “Reports” και η επεξεργασία των στοιχείων του από το μενού “Profile”.

Στο μενού “profile”, ο χρήστης μπορεί να αλλάξει τα στοιχεία του όπως, το όνομα χρήστη του, τη διεύθυνση e-mail του καθώς και το κωδικό του. Η διεύθυνση e-mail, πρέπει να είναι έγκυρη, για το λόγο ότι είναι ο μόνος τρόπος για να ανακτήσει ο χρήστης τον κωδικό του, σε περίπτωση που δε το θυμάται.

Στο μενού “reports”, παρουσιάζονται συγκεντρωτικά οι δημοσιεύσεις του χρήστη και οι βιβλιογραφικές αναφορές σε αυτές. Συγκεκριμένα παρουσιάζονται ταξινομημένες σε χρονολογική σειρά οι δημοσιεύσεις, με τα ονόματα των συγγραφέων να παρουσιάζονται πρώτα, στη συνέχεια ο τίτλος της δημοσίευσης και τέλος ο τύπος δημοσίευσης. Από κάτω παρατάσσονται σε μορφή λίστας οι βιβλιογραφικές αναφορές που έχουν γίνει στη κάθε δημοσίευση. Σε κάθε βιβλιογραφική αναφορά εμφανίζεται, ο τίτλος της αναφοράς, ο τύπος και η χρονιά δημοσίευσής της. Δίπλα σε κάθε εγγραφή της λίστας “report”, υπάρχει ένα κουμπί “citation chart”, που εμφανίζει μία γραφική παράσταση για τις βιβλιογραφικές αναφορές που έχουν γίνει με βάση τη χρονιά δημοσίευσής τους (εικ. 4-9)



Εικόνα 4-9 Παράδειγμα γραφικής παράστασης, με δύο βιβλιογραφικές αναφορές, μία το 2000 και μία το 2005.

Εικόνα 4-10 Η εργασίες με τις αναφορές που έχουν γίνει σε αυτές από τρίτους, ταξινομημένες κατά χρονιά έκδοσης, όπως παρουσιάζονται στο μενού “Reports”

Σε αντίθεση με τις άλλες σελίδες που έχουν αποτελέσματα σε μορφή λίστας, εδώ δεν υπάρχει σελιδοποίηση. Αυτό εξυπηρετεί την αντιγραφή στο πληκτρο-

λόγιο των δεδομένων για την μετέπειτα χρησιμοποίησή τους σε κάποιο βιογραφικό σημείωμα ή και στη προσωπική ιστοσελίδα του χρήστη.

Επιπλέον, παρόμοια με τα drop down menus στις σελίδες “my papers” και “my citations”, έτσι κι εδώ υπάρχουν drop down menus που μπορούν να αλλάξουν την παρουσίαση των αποτελεσμάτων. Ο χρήστης μπορεί να επιλέξει εμφάνιση των αποτελεσμάτων μόνο μιας συγκεκριμένης χρονιάς από αυτές που υπάρχουν στο μενού “select by year” ή να επιλέξει να παρουσιαστούν δημοσιεύσεις μόνο ενός συγκεκριμένου τύπου, επιλέγοντας τον από το μενού “select by publication”, βλ. εικόνα 4-10.

Τέλος ο χρήστης μπορεί να αποσυνδεθεί από το σύστημα πατώντας το πλήκτρο “logout”, στο τέλος του μενού.

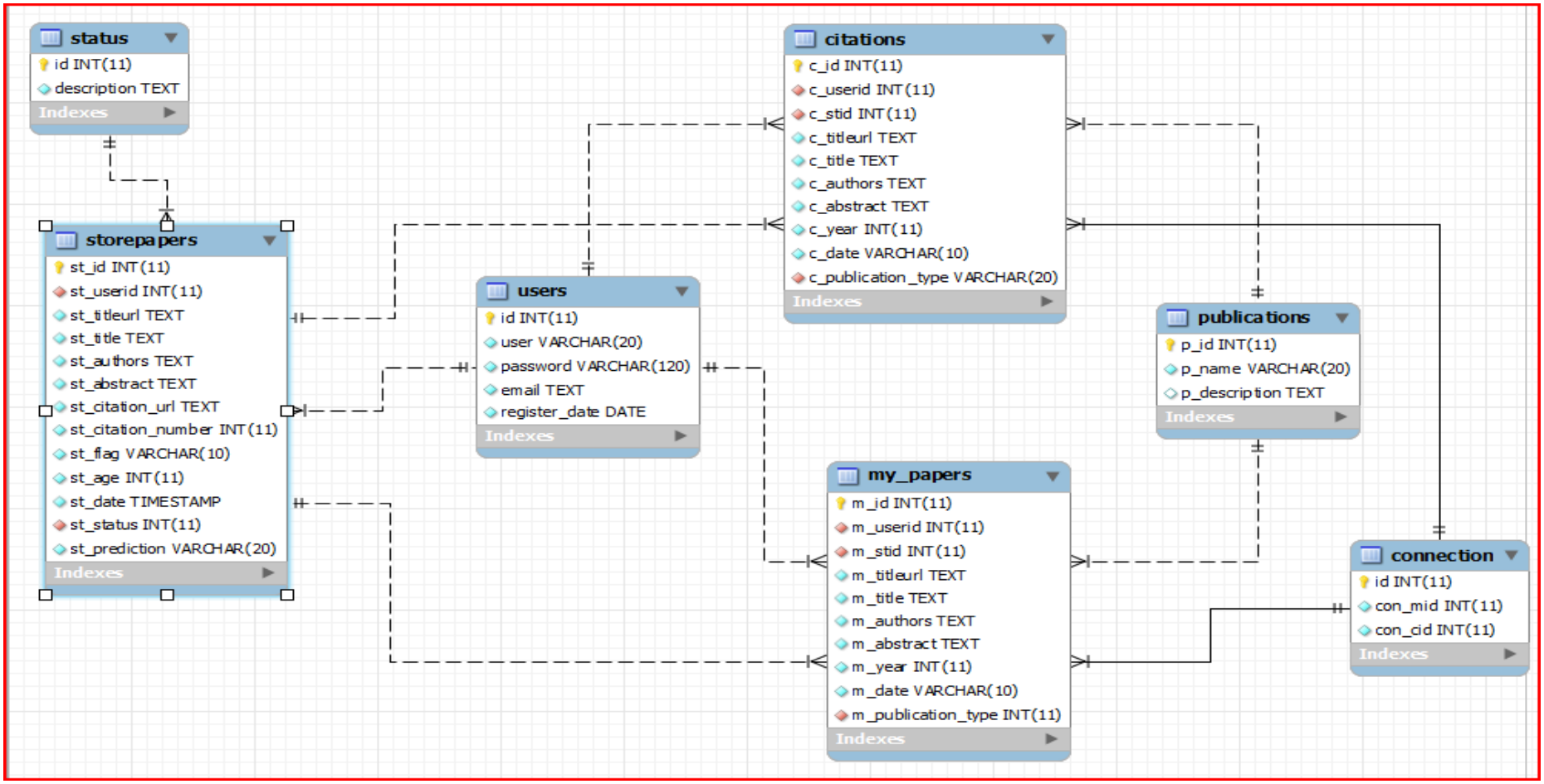
4.2 Η Βάση Δεδομένων

Το σύστημα διαχείρισης βάσεων δεδομένων που χρησιμοποιήθηκε στη συγκεκριμένη εφαρμογή είναι το MySQL και η βάση δεδομένων αναπτύχθηκε με το εργαλείο phpmyadmin. Όπως αναφέρθηκε και στο κεφάλαιο 3, το MySQL είναι ένα open source σύστημα διαχείρισης βάσεων δεδομένων και μπορεί να υποστηρίξει μεγάλο αριθμό εγγραφών, κάτι που το κάνει ιδιαίτερα δημοφιλές. Στις παρακάτω παραγράφους θα περιγραφεί η δομή και η λειτουργία της βάσης.

4.2.1 Δομή της Βάσης Δεδομένων

Η βάση δεδομένων αποτελείται από 7 πίνακας, που είναι υπεύθυνοι για την αποθήκευση και συσχέτιση των δεδομένων. Όπως φαίνεται και από την εικόνα 4-11, οι πίνακες αυτοί είναι:

- users – Περιέχει όλα τα στοιχεία των χρηστών
- storepapers - Όλα τα έγγραφα που περνούν τον έλεγχο κατά την ανάγνωση του XML αρχείου βρίσκονται εδώ
- status - Περιέχει τους τύπους κατάταξης των εγγράφων
- citations - Τα έγγραφα που είναι βιβλιογραφικές αναφορές βρίσκονται σε αυτό το πίνακα
- my papers - Τα έγγραφα που είναι δημοσιεύσεις του χρήστη βρίσκονται σε αυτό το πίνακα



Εικόνα 4-11 Διάγραμμα EER, της βάσης δεδομένων της εφαρμογής, στο MySQL Workbench

- publications - Όλοι ο τύποι δημοσίευσης, σύμφωνα με το πρότυπο BibTeX βρίσκονται εδώ
- connection - Οι συσχετίσεις των δημοσιεύσεων του χρήστη, με τις βιβλιογραφικές αναφορές που έχουν γίνει σε αυτές

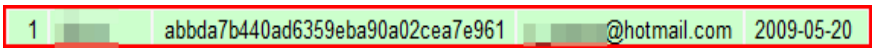
4.2.2 Λειτουργία της Βάσης Δεδομένων

Στις επόμενες παραγράφους θα ακολουθήσει μια αναλυτική περιγραφή των πινάκων, που αναφέρθηκαν στη προηγούμενη ενότητα και της λειτουργίας τους.

users

Ο πίνακας “users” είναι υπεύθυνος για τη διαχείριση και την αποθήκευση των στοιχείων των χρηστών. Διαθέτει ένα πεδίο user που μπορεί να έχει μήκος μέχρι και 20 χαρακτήρες ένα password με μήκος μέχρι 120 χαρακτήρες, ένα email τύπου Text και ένα register_date τύπου Date.

Όλοι οι πίνακες στη βάση δεδομένων έχουν ένα πεδίο μοναδικού αριθμού ταυτοποίησης id, ο οποίος είναι και το κύριο κλειδί σε όλους τους πίνακες. Ωστόσο ο πίνακας “users”, έχει ακόμη δύο πεδία που δε δέχονται διπλοεγγραφές. Αυτά είναι τα πεδία email και user, που αποθηκεύουν τις διευθύνσεις email και τα ονόματα χρήστη αντίστοιχα. Επίσης ο κωδικός (password), έχει αρκετά μεγάλο μήκος χαρακτήρων για να μπορεί να αποθηκεύει τον κωδικό του χρήστη κρυπτογραφημένο και θα περιγραφεί παρακάτω σε αυτό το κεφάλαιο το πώς ακριβώς γίνεται αυτή η κρυπτογράφηση. Το τελευταίο πεδίο του πίνακα, register_date, είναι καθαρά για στατιστικούς λόγους και καταγράφει την ημερομηνία εγγραφής του χρήστη στο σύστημα, με τον τρόπο που η MySQL κρατά τις ημερομηνίες, ΧΡΟΝΙΑ-ΜΗΝΑΣ-ΜΕΡΑ, βλ. εικόνα 4-12.



Εικόνα 4-12 Μία εγγραφή στο πίνακα users

storepapers

Όπως δηλώνει και το όνομα του, ο “storepapers” αποθηκεύει κάθε είδους έγγραφο. Εδώ τοποθετούνται όλα τα καταταγμένα και μη έγγραφα που περνούν από τον έλεγχο κατά την ανάγνωση του XML αρχείου και πρόκειται επίσης, για το πίνακα που χρησιμοποιείται στον έλεγχο των δεδομένων μετά την αναζήτηση.

Σε αυτό το πίνακα, όπως και στους υπόλοιπους, εκτός από το πίνακα “users”, χρησιμοποιείται ένα πρόθεμα (prefix) στα ονόματα των πεδίων, που τα χαρακτηρίζει μοναδικά για τη καλύτερη διαχώριση όμοιων πεδίων. Εκτός λοιπόν από το πεδίο st_id, που είναι και το κύριο κλειδί του, περιέχει και άλλα σημαντικά πεδία που θα αναλυθούν παρακάτω.

- st_userid: Τύπου int. Αποθηκεύει για κάθε εγγραφή το id του χρήστη, ούτως ώστε να παρουσιάζει στο χρήστη μόνο τα δικά του αποτελέσματα. Πρόκειται για το ξένο κλειδί του πίνακα που τον συσχετίζει με το πίνακα “users”
- st_titleurl: Τύπου text. Περιέχει τη διεύθυνση url που βρίσκεται πίσω από κάθε τίτλο-σύνδεσμο στα αποτελέσματα του google scholar
- st_title: Τύπου text. Εδώ περιέχεται ο τίτλος της κάθε δημοσίευσης
- st_authors: Τύπου text. Περιέχει τα ονόματα των συγγραφέων καθώς και κάποιες συμπληρωματικές πληροφορίες. Στην ουσία πρόκειται για τα πράσινα γράμματα των αποτελεσμάτων του google scholar.
- st_abstract: Τύπου text. Πρόκειται για το γενικό κείμενο που εμφανίζεται στα αποτελέσματα του google scholar.
- st_ciation_url: Τύπου text. Περιέχει το url του συνδέσμου “cited by”, των αποτελεσμάτων του google scholar.
- st_citation_number: Τύπου int. Περιέχει τον αριθμό των βιβλιογραφικών αναφορών που υπάρχουν πίσω από το σύνδεσμο “cited by”.
- st_flag: Τύπου varchar. Περιέχει το προσδιοριστικό που υπάρχει συνήθως στα αποτελέσματα του google scholar και δηλώνει, π.χ. αν μία δημοσίευση είναι βιβλίο ή PDF.
- st_date: Τύπου timestamp. Περιέχει την ημερομηνία που και την ώρα που συλλέγονται τα δεδομένα.
- st_status: Τύπου int. Το συγκεκριμένο πεδίο είναι ξένο κλειδί του πίνακα “status” και περιέχει ένα αριθμό που αντιστοιχεί στο κύριο κλειδί του.
- st_prediction: Τύπου varchar. Περιέχει την πρόβλεψη για τον τύπο κάθε εγγραφής. Το πεδίο αυτό ανανεώνεται κάθε φορά που ο χρήστης πατά το κουμπί prediction στο μενού “raw data”.

status

Ο πίνακας “status”, περιέχει δύο πεδία και τέσσερις μόνο εγγραφές. Το ένα περιέχει ένα μοναδικό αριθμό id και το άλλο το όνομα του τύπου που αντιστοιχεί στο μοναδικό αριθμό. Οι εγγραφές που περιέχει είναι οι τέσσερις τύποι που μπορεί να καταταχθεί ένα έγγραφο, not-checked, my paper, my citation και garbage.

my_papers

Ο πίνακας “my_papers”, έχει περίπου την ίδια δομή με τον πίνακα “storepapers”. Διατηρεί όλα τα απαραίτητα πεδία από τον “storepapers”, ενώ έχει και δύο επιπλέον πεδία, το m_year και το m_publication_type. Το πρώτο είναι τύπου int και περιέχει τη χρονιά έκδοσης του κάθε εγγράφου και το δεύτερο είναι επίσης τύπου int και πρόκειται για το ξένο κλειδί του πίνακα “publications”.

Τα πεδία που «κληρονομούνται», από το πίνακα “storepapers” είναι τα m_userid, m_stid, m_titleurl, m_title, m_authors, m_abstract και m_date. Αυτά τα πεδία έχουν τον ίδιο τύπο και περιεχόμενο με τα αντίστοιχα του πίνακα “storepapers”, με τη μόνη διαφορά να είναι στο πρόθεμα “m_”.

Η ανάγκη για διαφορετικό πίνακα για κάθε έγγραφο που είναι τύπου my papers όπως και για αυτά που είναι τύπου citation, προέκυψε για την αποθήκευση των επεξεργασμένων εγγράφων και την εν συνεχεία καλύτερη παρουσίασή τους από την εφαρμογή.

citations

Ο πίνακας “citations” είναι ίδιος με τον πίνακα “my_papers”. Η επεξεργασία και η αποθήκευση των εγγράφων είναι η αιτία ύπαρξης και αυτού του πίνακα και η μόνη του διαφορά από τον πίνακα “my_papers” είναι το πρόθεμα. Αντί για το πρόθεμα “m_” που έχουν τα πεδία του “my_papers”, τα πεδία του “citations” έχουν το πρόθεμα “c_”.

publications

Ο πίνακας “publications” περιέχει τους τύπους δημοσίευσης, σύμφωνα με το πρότυπο BibTeX. Εκτός του id έχει δύο ακόμη πεδία, το p_name και το p_description. Το p_name περιέχει 13 συγκεκριμένους τύπους δημοσίευσης που είναι οι article, book, booklet, conference, inbook, incollection, inproceedings, manual, masterthesis, misc, phdthesis, proceedings και techreport. Για

κάθε ένα τύπο δημοσίευσης υπάρχει μια μικρή περιγραφή στο πεδίο p_description.

connection

Ο πίνακας connection συσχετίζει τους πίνακες “citations” και “my_papers”. Περιέχει ένα κύριο κλειδί id και δύο ξένα κλειδιά con_mid και con_cid, που συσχετίζονται με τα ids του πίνακα “my_papers” και του πίνακα “citations” αντίστοιχα. Με τη συσχέτιση κάποιας βιβλιογραφικής αναφοράς από το κουμπί “associate” στο μενού “my citations”, γίνεται εκχώρηση στο πίνακα, το id του “citations” και τα id όλων των id, όλων των συσχετιζόμενων με την αναφορά, δημοσιεύσεων του χρήστη.

Επίσης είναι ο πίνακας που χρησιμοποιείται στο μενού “reports”, για την παρουσίαση των βιβλιογραφικών αναφορών σε μορφή λίστας κάτω από κάθε δημοσίευση.

4.3 Ανάλυση του Κώδικα

Σε αυτή την ενότητα περιγράφονται τα σημαντικότερων σημείων της εφαρμογής σε επίπεδο κώδικα και γενικότερα υλοποίησης.

4.3.1 Κρυπτογράφηση και Αποστολή Κωδικού

Όπως αναφέρθηκε προηγουμένως, πριν ο χρήστης εισέλθει στο σύστημα πρέπει να πληκτρολογήσει το όνομα χρήστη του και το κωδικό του. Αν δεν διαθέτει αυτά τα στοιχεία θα πρέπει πρώτα να γραφτεί στο σύστημα.

Κατά την εγγραφή του χρήστη το μόνο ενδιαφέρον κομμάτι σε θέμα κώδικα είναι η διατήρηση των στοιχείων στις θέσεις τους σε περίπτωση λάθους. Αν δηλαδή ο χρήστη κάνει λάθος στην επιβεβαίωση του κωδικού θα ήταν ενοχλητικό για το χρήστη να πληκτρολογήσει ξανά το όνομα χρήστη ή τη διεύθυνση e-mail του. Με τη χρήση της μεταβλητής $_POST όμως, η κάθε εισαγωγή συγκρατείται και διαγράφεται μόνο η λανθασμένη.

Ο χρήστης μπορεί να χρησιμοποιεί τον ίδιο κωδικό και όνομα χρήστη και σε άλλους διαδικτυακούς χώρους, οπότε θα ήταν αρκετά ασφαλές να υπήρχε κάποια κρυπτογράφηση του κωδικού του. Στη παρούσα εφαρμογή ο κωδικός κρυπτογραφείται σε md5 hash μορφή, με τη βοήθεια της PHP μεθόδου md5().

Η μέθοδος αυτή, παίρνει ως παράμετρο μία συμβολοσειρά (το κωδικό του χρήστη στη προκειμένη περίπτωση) και τη μετατρέπει σε ένα δεκαεξαδικό αριθμό 32 χαρακτήρων. Οπότε, κάθε φορά που ο χρήστης εισάγει τον κωδικό του, αυτός μετατρέπεται σε md5 hash μορφή και ελέγχεται η ομοιότητα του με αυτόν που βρίσκεται στη βάση δεδομένων. Στη περίπτωση όμως που ο χρήστης δε θυμάται το κωδικό του, η αποκρυπτογράφηση δεν είναι δυνατή. Η εφαρμογή χρησιμοποιεί μία μέθοδο για να παράγει ένα τυχαίο δωδεκαψήφιο αριθμό, βλ. εικόνα 4-13, τον κρυπτογραφεί με τον ίδιο τρόπο και αντικαθιστά το προηγούμενο στη βάση δεδομένων, ενώ την ίδια στιγμή στέλνει ένα μήνυμα με το νέο κωδικό, στη κανονική του μορφή, στην e-mail διεύθυνση του χρήστη.

```php
//generates a 12digit strong password
function randomPassword($length = 12,
    $allow = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789!@#$%^&*()_") {

    $i = 1;
    while ($i <= $length) {

        $max  = strlen($allow)-1;
        $num  = rand(0, $max);
        $temp = substr($allow, $num, 1);
        $ret  = $ret . $temp;
        $i++;
    }
    return $ret;
}
```

Εικόνα 4-13 ο τυχαίος κωδικός που παράγεται περιέχει χαρακτήρες, μικρούς και κεφαλαίους, αριθμούς και σύμβολα

4.3.2 Εισαγωγή Αποτελεσμάτων

Στη συνέχεια και αφού ο χρήστης έχει εισέλθει επιτυχώς στο σύστημα, θα πρέπει να χρησιμοποιήσει το μενού “upload” για να ανεβάσει το XML αρχείο με τα αποτελέσματα (Εικόνα 4-14). Η εφαρμογή ανεβάζει και μετονομάζει το αρχείο, χρησιμοποιώντας ένα μοναδικό αναγνωριστικό όπως είναι το id του χρήστη και τέλος καλεί τη μέθοδο xmlToDatabase(), με παράμετρο το path του αρχείου.

Στη μέθοδο xmlToDatabase() γίνεται ο έλεγχος των αποτελεσμάτων καθώς και η εισαγωγή των πληροφοριών στη βάση δεδομένων. Το διάβασμα και ο χειρισμός ενός XML αρχείου είναι μια εύκολη υπόθεση για την PHP, λόγο των βιβλιοθηκών που διαθέτει. Κάθε πληροφορία που εξάγεται από το XML αρχείο, αποθηκεύεται προσωρινά σε πίνακες αναλόγως το είδος της. Στη συνέχεια γί-

νονται κάποιες μετατροπές στα δεδομένα και καλείται η μέθοδος checkDB() για να ελέγξει αν τα αποτελέσματα έχουν εισαχθεί σε παλιότερες αναζητήσεις, βλ εικόνα 4-15.

```php
$to = "upload/project.xml";
if(isset($_POST['submit'])) {
    if ($_FILES["file"]["error"] > 0){
        echo "Return Code: " . $_FILES["file"]["error"] . "<br />";
    } else {
        $to = "upload/" . $_FILES["file"]["name"];
        //moves the file to the server
        move_uploaded_file($_FILES["file"]["tmp_name"], $to);
        //and calls directly the function xmlToDatabase to read the xml file
        //and insert the data in the database
        xmlToDatabase($to);
```

Εικόνα 4-14 Ανέβασμα του αρχείου στο server. Χρησιμοποιείται η μεταβλητή $_FILES για το χειρισμό του αρχείου

```php
        if($titlos_no[$i] == NULL){
            $checking = checkDB($userID, $dieuthinsi[$i], $titlos[$i], $cited[$i]);
            $replace= "'";
            $with=" ";
            //removes from the text the single quote ' for not confusing the sql query
            $titlos[$i] = str_replace($replace, $with, $titlos[$i], $cited[$i]);
            $sigrafeas[$i] = str_replace($replace, $with, $sigrafeas[$i]);
            $afirimeno[$i] = str_replace($replace, $with, $afirimeno[$i]);
            if($checking){
                $sqlDataInsert = "INSERT INTO storepapers (st_userid, st_titleurl, st_title, st_a
                                                    VALUES ( '" . $userID . "', '" . $dieuthinsi[$i]
                mysql_query($sqlDataInsert);
            }
        } else {
            $checking = checkDB($userID, ' ', $titlos_no[$i], $cited[$i]);
            $replace= "'";
            $with=" ";
            $sigrafeas[$i] = str_replace($replace, $with, $sigrafeas[$i]);
            $titlos_no[$i] = str_replace($replace, $with, $titlos_no[$i]);
            if($checking){
                $sqlDataInsert = "INSERT INTO storepapers (st_userid, st_titleurl, st_title, st_a
                                                    VALUES ( '" . $userID . "', '', '" . $titlos_no[$
                mysql_query($sqlDataInsert);
            }
        }
    }
}
//deletes the file after insertign the data into the database
unlink($xmlFile);
```

Εικόνα 4-15 Έλεγχος των στοιχείων και εν συνεχεία εισαγωγή στη βάση δεδομένων αυτό που πέρασαν επιτυχώς τον έλεγχο

Η μέθοδος checkDB(), ελέγχει τη βάση για προηγούμενα ίδια αποτελέσματα και επιστρέφει μία τιμή false αν αντιστοιχίσει το αποτέλεσμα και true εάν όχι. Στη συνέχεια γίνεται έλεγχος της επιστρεφόμενης τιμής και αν ένα αποτέλεσμα περάσει τον έλεγχο τότε αποθηκεύεται στη βάση δεδομένων, αλλιώς απορρίπτεται. Για τον έλεγχο αυτό χρησιμοποιείται ο αλγόριθμος μέτρησης απόστασης

Levenshtein. Σε μια περίπτωση βέβαια που μία εγγραφή υπάρχει ήδη στη βάση και αποκλειστεί από την εφαρμογή δε θα μπορούσε να υπάρχει κάποια ανανέωση των πληροφοριών.

Για τον παραπάνω λόγο γίνεται ένας επιπλέον έλεγχος που αφορά στον αριθμό των βιβλιογραφικών αναφορών μιας δημοσίευσης. Αν παραδείγματος χάριν, μία δημοσίευση χρησιμοποιηθεί σαν βιβλιογραφική αναφορά, από κάποιες εργασίες, μετά την εισαγωγή της εγγραφής στη βάση δεδομένων, ο χρήστης θα ήθελε να ειδοποιείται για αυτή την αλλαγή ούτως ώστε να ανανεώσει τη βάση δεδομένων.

```
//checks the database
function checkDB($userId, $url, $title, $number){
    $sqlCheck = "SELECT * FROM storepapers WHERE st_userid='" . $userId . "'";
    $resCheck = mysql_query($sqlCheck);
    $checkCounter = 0;
    while($check = mysql_fetch_array($resCheck)){
        $lev = levenshtein($check['st_title'], $title);
        if(($url == $check['st_titleurl']) || ($title == $check['st_title'])){
            if($number == $check['st_citation_number']){
                $checkCounter++;
            } else {
                $sqlUpdate = "UPDATE storepapers SET st_citation_number='" . $number . "',
                        st_status='1', st_flag='[UPDATED]' WHERE st_id='" . $check['st_id'] . "'";
                mysql_query($sqlUpdate);
                $checkCounter++;
            }
        } elseif ($lev < 50){
            $checkCounter++;
        }
    }
    if($checkCounter > 0){
        return false;
    } else {
        return true;
    }
}
```

Εικόνα 4-16 Η μέθοδος checkDB()

Η μέθοδος checkDB() (εικόνα 4-16), αρχικά ελέγχει αν ένας τίτλος υπάρχει ήδη στη βάση δεδομένων και αν υπάρχει συγκρίνει των αριθμό των βιβλιογραφικών αναφορών της εγγραφής που ελέγχεται και της όμοιας της που βρέθηκε στη βάση δεδομένων. Αν παρουσιαστεί διαφορετικό αριθμός, τότε ανανεώνει τη βάση, αλλάζει τη κατάσταση της εγγραφής σε “not-checked” και στη θέση flag που συνήθως φαίνεται το είδος του εγγράφου, εμφανίζεται το χαρακτηριστικό “[UPDATED]”. Ο χρήστης τότε ξέρει ότι πρέπει να ανανεώσει τη συγκεκριμένη εγγραφή αναλόγως.

Για τα αποτελέσματα που θα περάσουν τον έλεγχο, καλείται η μέθοδος citationList() (εικόνα 4-17), η οποία δημιουργεί μία λίστα με τις βιβλιογραφικές αναφορές αποτελεσμάτων και τις τοποθετεί στο κομμάτι target url του κανόνα που χρησιμοποιεί το DEiXTo.

```
//adds a list of citation pages to the wrapper
function citationList($userId){
    $file = "google.wpf";
    $replace = 'cambialo';
    $list = "";
    $sqlCitations = "SELECT * FROM storepapers WHERE st_userid='" . $userId . "' AND st_age='0'";
    $resCitations = mysql_query($sqlCitations);
    //if there are no new records will keep the zero
    $counter=0;
    while($citation = mysql_fetch_array($resCitations)){
        $list = $list . "<URL Address='" . $citation['st_citation_url']. "'/>";
        $counter++;
        $sqlUpdateStore = "UPDATE storepapers SET st_age='1' WHERE st_id='" . $citation['st_id'] . "'";
        mysql_query($sqlUpdateStore);
    }

    $store = file_get_contents($file);
    //fixing the rule by placing the list with the citation's url addresses in the rule
    $store2 = str_replace($replace, $list, $store);
    $newFile = "scholar" . $userId . ".wpf";
    //stores the changed file in a scholar.wpf file
    file_put_contents($newFile, $store2);
    //returns zero or another for later check
    return $counter;
}
```

Εικόνα 4-17 Μέθοδος δημιουργίας του κανόνα για το DEiXTo, για τη συλλογή των αποτελεσμάτων "πίσω" από το σύνδεσμο “cited by”.

Αν υπάρχει έστω και μία σελίδα αναφορών ή ακόμη και μία βιβλιογραφική αναφορά, τότε εμφανίζεται ένα κουμπί “download”, για το κατέβασμα του καινούριου κανόνα.

4.3.3 Πρόβλεψη Τύπου Αποτελεσμάτων

Ο χρήστης μέσο του μενού μπορεί να περιηγηθεί και να επεξεργαστεί τα αποτελέσματα της αναζήτησης. Αρχικά θα υπάρχουν αποτελέσματα μόνο στο μενού “raw data”, τα οποία θα πρέπει να κατατάξει καταλλήλως. Στο μενού “raw data” υπάρχει επίσης και ένα κουμπί που με την ονομασία “prediction”. Με το πάτημα αυτού του κουμπιού η εφαρμογή προσπαθεί να προβλέψει τον τύπο του αποτελέσματος. Για να κάνει αυτή τη πρόβλεψη, η εφαρμογή χρησιμοποιεί μια σειρά από μεθόδους και τον αλγόριθμο μέτρησης απόστασης Levenshtein.

Η PHP διαθέτει στη βιβλιοθήκη της τη μέθοδο levenshtein() η οποία μπορεί να μετρήσει την απόσταση δύο συμβολοσειρών που δεν έχουν μέγεθος, μεγα-

λύτερο των 250 χαρακτήρων. Για το λόγο αυτό χρησιμοποιήθηκε μια τροποποιημένη levenshtein μέθοδος που ονομάστηκε levenshtein2(), βλ. εικόνα 14-18. Υπολογίζει και επιστρέφει την απόσταση δύο συμβολοσειρών, χωρίς να βάζει περιορισμούς στο μέγεθός τους.

```php
//counts the levenshtein distance without size limits
function levenshtein2($str1, $str2, $cost_ins = null, $cost_rep = null, $cost_del = null) {
    $d = array_fill(0, strlen($str1) + 1, array_fill(0, strlen($str2) + 1, 0));
    $ret = 0;

    for ($i = 1; $i < strlen($str1) + 1; $i++)
        $d[$i][0] = $i;
    for ($j = 1; $j < strlen($str2) + 1; $j++)
        $d[0][$j] = $j;

    for ($i = 1; $i < strlen($str1) + 1; $i++)
        for ($j = 1; $j < strlen($str2) + 1; $j++) {
            $c = 1;
            if ($str1{$i - 1} == $str2{$j - 1})
                $c = 0;
            $d[$i][$j] = min($d[$i - 1][$j] + 1, $d[$i][$j - 1] + 1, $d[$i - 1][$j - 1] + $c);
            $ret = $d[$i][$j];
        }

    return $ret;
}
```

Εικόνα 4-18 Η μέθοδος Levenshtein2, χρησιμοποιεί τον ομώνυμο αλγόριθμο μέτρησης απόστασης μεταξύ δύο συμβολοσειρών

Χρειάζεται επίσης να υπάρχουν δεδομένα στους πίνακες “my_papers” και “citations”, καθώς γίνεται έλεγχος αυτών των αποτελεσμάτων, για τη δημιουργία της πρόβλεψης. Η μέθοδος levenshtein2() καλείται από δύο άλλες μεθόδους που υπολογίζουν το ποσοστό του να είναι η εγγραφή δημοσίευση του χρήστη ή βιβλιογραφική αναφορά, βλ. εικόνες 4-19 και 4-20. Οι μέθοδοι αυτές ονομάζονται, citationPercentage() και mypaperPercentage().

Οι μέθοδοι αυτές καλούνται με τη σειρά τους από τη βασική μέθοδο για τη πρόβλεψη που είναι η statusPrediction(). Η μέθοδος αυτή κάνει τον έλεγχο αν το ποσοστό που έχει υπολογισθεί στις προηγούμενες μεθόδους είναι μεγαλύτερο από το όριο που έχει τεθεί, ενώ σε αντίθετη περίπτωση δε μπορεί να κάνει πρόβλεψη για τη συγκεκριμένη εγγραφή και εισάγει στο πεδίο πρόβλεψης της βάσης το λατινικό χαρακτήρα “?” ερωτηματικό. Σε περίπτωση που μόνο ένα από τα δύο ποσοστά είναι αποδεκτό, τότε ενημερώνει τη βάση με τις λέξεις “My Paper” ή “My Citation” αντιστοίχως για τις δημοσιεύσεις του χρήστη και τις βιβλιογραφικές αναφορές. Η τελευταία περίπτωση που υπάρχει είναι και τα δύο

ποσοστά που υπολογίστηκαν να ξεπερνούν το όριο που έχει τεθεί. Τότε γίνεται σύγκριση των δύο ατών ποσοστών και το μεγαλύτερο επικρατεί και η προσδιοριστική φράση που αντιστοιχεί σε αυτό εκχωρείται στη βάση δεδομένων.

```php
//citation possibility
function citationPercentage($title){
    $maxPercent = 0;

    $sqlCitations = "SELECT * FROM citations WHERE c_userid='" . $_SESSION['usersID'] . "'";
    $resCitations = mysql_query($sqlCitations);

    while($citation = mysql_fetch_array($resCitations)){
        $leven = levenshtein2($title, $citation['c_title']);
        $strSize = strlen($title) + strlen($citation['c_title']);
        $percent = 100 - ($leven / $strSize) * 100;
        if($percent >= $maxPercent){
            $maxPercent = $percent;
        }
    }

    return $maxPercent;
}
```

Εικόνα 4-19 Η μέθοδος υπολογισμού του ποσοστού η εγγραφή να είναι βιβλιογραφική αναφορά

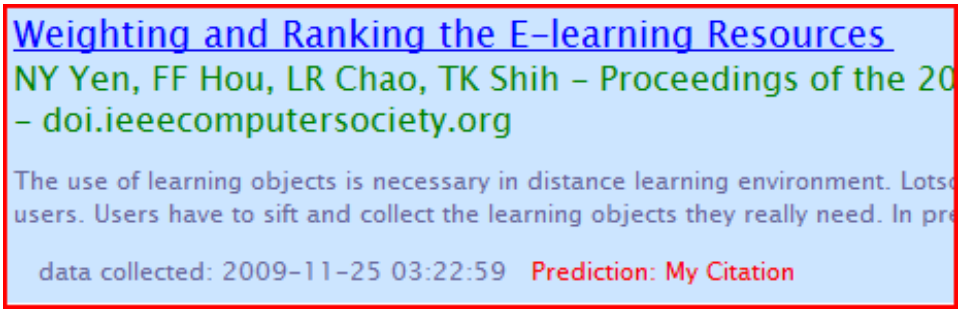
```php
//my paper possibility
function mypaperPercentage($title){
    $maxPercent = 0;

    $sqlMyPapers = "SELECT * FROM my_papers WHERE m_userid='" . $_SESSION['usersID'] . "'";
    $resMyPapers = mysql_query($sqlMyPapers);

    while($mypaper = mysql_fetch_array($resMyPapers)){
        $leven = levenshtein2($title, $mypaper['m_title']);
        $strSize = strlen($title) + strlen($mypaper['m_title']);
        $percent = 100 - ($leven / $strSize) * 100;
        if($percent >= $maxPercent){
            $maxPercent = $percent;
        }
    }
    return $maxPercent;
}
```

Εικόνα 4-20 Η μέθοδος υπολογισμού του ποσοστού η εγγραφή να είναι δημοσίευση του χρήστη

Με το πάτημα του κουμπιού “prediction”, εμφανίζεται στο URL ένα επιπλέον προσδιοριστικό, το “&prediction=1”, βλ. εικόνα 4-21.

Εικόνα 4-21 Πέρασμα τιμής για την εμφάνιση της πρόβλεψης μέσο του URL

Η εφαρμογή δεσμεύει από την αρχή κάποιο χώρο στις πληροφορίες που παρουσιάζονται για τα αποτελέσματα της πρόβλεψης. Μόλις κάνει την εμφάνιση του το &prediction στο URL, η εφαρμογή συλλέγει και ελέγχει το περιεχόμενό του, με τη βοήθεια της PHP μεταβλητής $_GET['prediction']. Αν το περιεχόμενο αυτής της μεταβλητής ισούται με 1, τότε εμφανίζεται η πρόβλεψη, με κόκκινα γράμματα, στις πληροφορίες του αποτελέσματος, βλ. εικόνα 4-22.



Εικόνα 4-22 Εμφάνιση της πρόβλεψης (κόκκινα γράμματα) σε ένα αποτέλεσμα

4.3.4 Παρουσίαση Αποτελεσμάτων

Στην εφαρμογή υπάρχουν τέσσερις σελίδες παρουσίασης αποτελεσμάτων. Για τα ανεπεξέργαστα αποτελέσματα, για τα μη σχετικά αποτελέσματα, για τις δημοσιεύσεις του χρήστη και για τις βιβλιογραφικές αναφορές. Σε κάθε κατηγορία υπάρχει ένα όριο για τα αποτελέσματα που παρουσιάζονται. Στα μενού "raw data" και "garbage" το όριο είναι 10 αποτελέσματα ενώ στα μενού "my papers" και "my citations", το όριο είναι 5 αποτελέσματα. Τα υπόλοιπα δε χάνονται, αλλά όπως συμβαίνει και με στις εφαρμογές ηλεκτρονικού ταχυδρομείου, σελιδοποιούνται.

Για την υλοποίηση της σελιδοποίησης, αρχικά γίνονται οι ταξινομήσεις στις εγγραφές που πρόκειται να παρουσιαστούν από τη βάση δεδομένων, ενώ στο ίδιο ερώτημα στη βάση τοποθετείται ένα όριο αποτελεσμάτων, βλ εικόνα 4-19.

Το επόμενο βήμα είναι η εμφάνιση και η λειτουργία των κουμπιών για τη σελιδοποίηση. Το σύνολο των σελίδων που θα παρουσιαστούν είναι εύκολο να υπολογιστεί, καθώς είναι επίσης εύκολο να βρεθεί το σύνολο των εγγραφών με ένα ερώτημα SQL και είναι γνωστός ο αριθμός των αποτελεσμάτων που θα εμ-

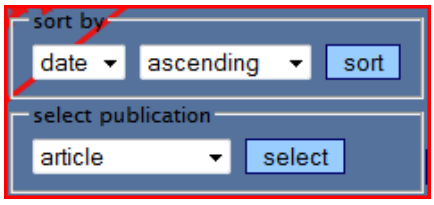
φανίζονται ανά σελίδα. Στη συνέχεια γίνεται έλεγχος της μεταβλητής current-page που γίνεται GET από το URL, για να εντοπιστεί ποια σελίδα είναι ενεργή και για το αν θα πρέπει να εμφανιστούν τα κουμπιά “previous” και “next”. Αν ο χρήστης βρίσκεται στη πρώτη σελίδα δεν εμφανίζεται το “previous” κουμπί, ενώ αντίστοιχα αν ο χρήστης βρίσκεται στη τελευταία σελίδα αποτελεσμάτων, δεν εμφανίζεται το κουμπί “next”, βλ. εικόνα 4-24.

```php
//count the number of rows that are gonna be displayed
$sqlnumberOfRows = "SELECT COUNT(m_id) FROM my_papers WHERE m_userid='" . $_SESSION['usersID'] . "'";
$numberOfRows = mysql_result(mysql_query($sqlnumberOfRows), 0);
// number of rows to show per page
$rowsPerPage = 5;
//total pages
//ceil() : Returns the next highest integer value by rounding up value  if necessary
$totalpages = ceil($numberOfRows / $rowsPerPage);

if (isset($_GET['currentpage'])) {
    $currentpage = $_GET['currentpage'];
} else {
    $currentpage = 1;
}

// if current page is greater than total pages then set as last page
if ($currentpage > $totalpages) {
    $currentpage = $totalpages;
}
// if current page is less than first page set as first page
if ($currentpage < 1) {
    $currentpage = 1;
}

// the offset of the list based on current page
$offset = ($currentpage - 1) * $rowsPerPage;
// range of num links to show
$range = 3;

if(isset($_GET['sort'])){
    $sqlMyPapers = "SELECT * FROM my_papers WHERE m_userid='" .  $_SESSION['usersID']  . "'
                ORDER BY " . $_GET['sort'] . " " . $_GET['by'] . " LIMIT " .$offset . ", " . $rowsPerPage;
}else {
    $sqlMyPapers = "SELECT * FROM my_papers WHERE m_userid='" .  $_SESSION['usersID']  . "'
                ORDER BY m_date DESC LIMIT " .$offset . ", " . $rowsPerPage;
}

$resMyPapers = mysql_query($sqlMyPapers);
```

Εικόνα 4-23 Η σελιδοποίηση όπως υλοποιείται για το μενού “my papers”.
Το όριο των αποτελεσμάτων βρίσκεται στη μεταβλητή $rowsPerPage

Εικόνα 4-24 Οι τρεις περιπτώσεις σελιδοποίησης

Το τρίτο και πολύ σημαντικό στοιχείο που υπάρχει μόνο στα μενού “my papers” και “my citation” είναι τα drop down menus για την ταξινόμηση των αποτελεσμάτων ή για τη παρουσίαση μόνο αποτελεσμάτων ενός συγκεκριμένου τύπου δημοσίευσης, βλ εικόνα 4-25.

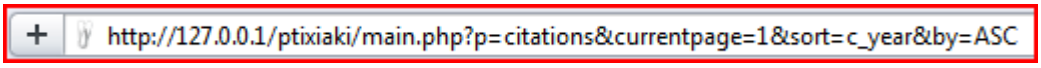


Εικόνα 4-25 Drop down menus για τη ταξινόμηση ή τη παρουσίαση αποτελεσμάτων ενός τύπου δημοσίευσης

Με την επιλογή ενός είδους ταξινόμησης, π.χ. κατά χρονιά έκδοσης σε φθίνουσα σειρά και το πάτημα του κουμπιού sort η εφαρμογή με τη βοήθεια μιας μεθόδου Javascript βλ εικόνα 4-26, περνά τις τιμές στο URL, βλ εικόνα 4-27.

```
function sortThemP(page){
    sort = document.getElementById('sorting').value;
    by = document.getElementById('sortby').value;
    link = 'main.php?p=mypapers&currentpage=' + page + '&sort=' + sort + '&by=' + by;
    window.open(link, '_self');
}
```

Εικόνα 4-26 Η javascript μέθοδος που παράγει το URL με τις μεταβλητές για την ταξινόμηση



Εικόνα 4-27 Η μεταβλητές για την ταξινόμηση στο URL

Η εφαρμογή κάνει GET τις τιμές των μεταβλητών sort και by και τις τοποθετεί στο SQL ερώτημα που είναι υπεύθυνο για τη παρουσίαση των αποτελεσμάτων.

```
function publicationP(){
    publ = document.getElementById('public').value;
    link = './paperpublications.php?pub=' + publ;
    window.open(link, 'myWindow', height='500', width='800');
}
```

Εικόνα 4-28 Η Javascript μέθοδος που ανοίγει ένα καινούριο παράθυρο με τα αποτελέσματα ενός συγκεκριμένου τύπου δημοσίευσης

Για την παρουσίαση αποτελεσμάτων ενός επιλεγμένου τύπου δημοσίευσης, χρησιμοποιείται επίσης μια Javascript μέθοδος που περνά τις μεταβλητές (το είδος της δημοσίευσης) στο URL. Η διαφορά αυτής της περίπτωσης, από την ταξινόμηση, είναι ότι τα αποτελέσματα εμφανίζονται σε καινούριο pop up παρά-

θυρο. Οπότε εκτός από το πέρασμα των μεταβλητών η μέθοδος ανοίγει και ένα καινούριο παράθυρο, συγκεκριμένου μήκους, στο browser, βλ. εικόνα 4-28.

4.3.5 Reports

Στο μενού “reports” παρουσιάζονται όλα τα αποτελέσματα σε μορφή λίστας χωρίς σελιδοποίηση. Αυτό γίνεται για την ευκολότερη αντιγραφή των αποτελεσμάτων και την μετέπειτα χρησιμοποίησή τους, από τον χρήστη.

Κάθε εγγραφή έχει στοιχεία από δύο διαφορετικούς πίνακες στη βάση δεδομένων, τον πίνακα “my_papers” και το πίνακα “citations”, χρησιμοποιώντας τον πίνακα “connection” σαν ενδιάμεσο και εμφανίζει τα αποτελέσματα ταξινομημένα κατά χρονιά δημοσίευσης.

Ένα αποτέλεσμα εμφανίζει πρώτα το τίτλο μιας δημοσίευσης του χρήστη και από κάτω μία λίστα με όλες τις βιβλιογραφικές αναφορές που έχουν γίνει σε αυτή τη δημοσίευση, ταξινομημένες με φθίνουσα σειρά κατά τη χρονιά δημοσίευσής τους. Πιο αναλυτικά, με βάση το κώδικα, αρχικά γίνονται δύο έλεγχοι, βλ. εικόνα 4-29.

- Πρώτον ελέγχεται αν υπάρχει η μεταβλητή year στο URL (για την εμφάνιση αποτελεσμάτων μιας συγκεκριμένης χρονιάς που ο χρήστης έχει επιλέξει)
- Και δεύτερον αν υπάρχει η μεταβλητή ptype στο URL (για την εμφάνιση αποτελεσμάτων ενός συγκεκριμένου τύπου δημοσίευσης που έχει επιλέξει ο χρήστης)

Αν αποτύχει να περάσει και τους δύο ελέγχους, υπάρχει και μία τρίτη περίπτωση (default περίπτωση), όπου τα αποτελέσματα εμφανίζονται όπως αναφέραμε πιο πάνω.

```
if(isset($_GET['year'])){
    $sqlReport = "SELECT * FROM my_papers WHERE m_year='" . $_GET['year'] . "' AND m_userid='" . $_SESSION['usersID'] . "'";
} elseif(isset($_GET['ptype'])) {
    $sqlReport = "SELECT * FROM my_papers WHERE m_publication_type='" . $_GET['ptype'] . "' AND m_userid='" . $_SESSION['usersID'
} else {
    $sqlReport = "SELECT * FROM my_papers WHERE m_userid='" . $_SESSION['usersID'] . "' ORDER BY m_year DESC";
}
$resReport = mysql_query($sqlReport);
```

Εικόνα 4-29 Οι δύο έλεγχοι και η τρίτη περίπτωση, που εκτελούν ένα από τα τρία SQL ερωτήματα

Στη default περίπτωση ένα SQL ερώτημα συλλέγει όλες τις εγγραφές του χρήστη στο πίνακα “my_papers” ταξινομημένες κατά χρονιά δημοσίευσης, σε φθίνουσα σειρά. Στη συνέχεια μέσα σε ένα while βρόχο, συλλέγονται όλα τα ids των βιβλιογραφικών αναφορών, από το πίνακα “connection”, που συσχετίζονται με τη κάθε δημοσίευση του χρήστη. Έπειτα και μέσα στο βρόχο, εκτελείται ένας δεύτερος εμφωλευμένος βρόχος, ο οποίος για κάθε βιβλιογραφική αναφορά που το id της αντιστοιχεί σε αυτό που έχει ήδη συλλεχθεί. Στη συνέχεια συλλέγει, για κάθε εγγραφή του πίνακα “citations”, που περνά αυτό τον έλεγχο ταυτοποίησης, τα στοιχεία της εγγραφής αυτή και τα αποθηκεύει στο πίνακα $storeCitations (πίνακας στη PHP), βλ. εικόνα 4-31. Στην ουσία δημιουργεί ένα πίνακα πινάκων, μιας και η PHP συλλέγει κάθε εγγραφή από το SQL ερώτημα, σε μορφή πίνακα που είτε μπορεί να τον διαβάσει είτε χρησιμοποιώντας την θέση του κάθε στοιχείου, είτε χρησιμοποιώντας το όνομα κάθε πεδίου, του πίνακα της βάσης δεδομένων.

Μετά την έξοδο από τους βρόχους, καλείται η μέθοδος της PHP usort() με δύο παραμέτρους, τον πίνακα $storeCitations και τη μέθοδο cmpi(), η οποία συγκρίνει δύο τιμές σε ένα πίνακα. Μιας και ο $storeCitations είναι ένας πίνακας πινάκων η usort() χρησιμοποιεί τη cmpi(), για να προσδιορίσει ως προς ποιο στοιχείο θα γίνει η ταξινόμηση, βλ εικόνα 4-30.

```
// compare function
function cmpi($a, $b) {
    $sort_field = 7; //the year field
    return strcmp($a[$sort_field], $b[$sort_field]);
}
```

Εικόνα 4-30 Η υλοποίηση της cmpi()

Αφού προσδιοριστεί το πεδίο και γίνει η σύγκριση δύο διαδοχικών εγγραφών, usort() αλλάζει ολόκληρη την εγγραφή και όχι μόνο τα πεδία που χρησιμοποιήθηκαν για να γίνει η σύγκριση. Η usort(), δεν επιστρέφει κάποια τιμή, αλλά αλλάζει οριστικά τη διάταξη του πίνακα. Στη συνέχεια του κώδικα, διαβάζεται ο ταξινομημένος πίνακας με τη βοήθεια του βρόχου foreach και εμφανίζονται τα αποτελέσματα στην οθόνη. [36]

Στην αρχή της ενότητας αυτής, αναφέρθηκε ότι γίνονται ακόμη δύο έλεγχοι πριν εκτελεστεί το αρχικό ερώτημα SQL, για το ποιες δημοσιεύσεις θα εμφανιστούν, βλ εικόνα 4-29. Αυτοί οι έλεγχοι έχουν να κάνουν με τις επιλογές του

χρήστη στα δύο drop down menus που υπάρχουν στο πάνω μέρος της σελίδας “reports”. Στο πρώτο drop down menu, select by year, εμφανίζονται μόνο οι χρονιές που ο χρήστης έχει κάνει κάποια δημοσίευση και η δημοσίευση αυτή έχει καταχωρηθεί στο σύστημα. Με την επιλογή μιας χρονιάς μόνο τα αποτελέσματα από δημοσιεύσεις της συγκεκριμένης χρονιάς θα εμφανιστούν. Αντιστοίχως στο δεύτερο drop down menu, select by publication, ο χρήστης επιλέγει συγκεκριμένο τύπο δημοσίευσης και εμφανίζονται μόνο τα αποτελέσματα του συγκεκριμένου τύπου.

```php
while($report = mysql_fetch_array($resReport)){
    $sqlConPaper = "SELECT * FROM connection WHERE con_mid='" . $report['m_id'] . "'";
    $resConPaper = mysql_query($sqlConPaper);

    $sqlPubl = "SELECT * FROM publications WHERE p_id='" . $report['m_publication_type'] . "'";
    $resPubl = mysql_query($sqlPubl);
    $publ = mysql_fetch_array($resPubl);

    $storeCitations = array();
    $i=0;

    while($conPaper = mysql_fetch_array($resConPaper)){
        $sqlCitations = "SELECT * FROM citations WHERE c_id='" . $conPaper['con_cid'] . "'";
        $resCitations = mysql_query($sqlCitations);
        $citation = mysql_fetch_array($resCitations);
        $storeCitations[$i] = $citation;
        $i++;
    }

    // do the array sorting
    usort($storeCitations, 'cmpi');
```

Εικόνα 4-31 Η δημιουργία του πίνακα πινάκων $storeCitations και η κλήση της usort() μετά τη έξοδο από τους βρόχους.

Ένα ακόμη, πολύ σημαντικό στοιχείο της σελίδας “reports” είναι το κουμπί “citation chart” που υπάρχει στο αριστερό μέρος κάθε αποτελέσματος. Με το πάτημα αυτού του κουμπιού ένα νέο pop up παράθυρο εμφανίζεται και παρουσιάζει μια γραφική παράσταση σε μορφή πίτας, για τις βιβλιογραφικές αναφορές που έχουν γίνει στη συγκεκριμένη δημοσίευση με βάση τη χρονολογία δημοσίευσης των αναφορών.

Για τη δημιουργία της γραφικής παράστασης αυτή χρησιμοποιήθηκε η έτοιμη PHP κλάση googlecharts που αναπτύχθηκε από τους Ludwig Pettersson και Fredrik Holmström το 2008. Ο κώδικας αυτός είναι ελεύθερος για οποιονδήποτε

χρήστη και για οποιαδήποτε χρήση. Χρησιμοποιεί το API της Google και χρησιμοποιώντας επίσης κάποια δεδομένα που της περνάει ο χρήστης, σχηματίζει και εμφανίζει μια γραφική παράσταση.

Για τις ανάγκες της παρούσας εφαρμογής, χρειάστηκε να δημιουργηθεί ένας πίνακας που να περιέχει σε κάθε σειρά του, δύο κελιά, ένα για τη χρονιά δημοσίευσης μιας αναφορά και ένα με το συνολικό αριθμό αναφορών για τη συγκεκριμένη δημοσίευση, τη συγκεκριμένη χρονιά. Αυτό επιτεύχθηκε με τη χρησιμοποίηση του πίνακα $storeCitations περιέχει όλες τις απαραίτητες πληροφορίες, βλ. εικόνα 4-32.

```
if($storeCitations != NULL){
    $data = array();
    $datatemp = array();
    $k = 0;
    foreach($storeCitations as $value){
        $chartYear = $value;
        $counter = 0;
        if(!in_array($chartYear['c_year'], $datatemp)){
            $datatemp[$k] = $chartYear['c_year'];
            foreach($storeCitations as $value2){
                $innerYear = $value2;
                if($datatemp[$k] == $innerYear['c_year']){
                    $counter++;
                }
            }
            $xronia = $datatemp[$k];
            $data[$xronia] = $counter;
        }
        $k++;
    }
    $s_name = "chart_" . $report['m_id'];
    $_SESSION[$s_name] = $data;
```

Εικόνα 4-32 Ο κώδικας για τη συλλογή των ζευγαριών χρονιάς και συνόλου βιβλιογραφικών αναφορών τη χρονιά αυτή.

Όπως φαίνεται και στην παραπάνω εικόνα, ο πίνακας με τα ζευγάρια, χρονιά και σύνολο βιβλιογραφικών αναφορών τη χρονιά αυτή, αποθηκεύεται σε μια $_SESSION μεταβλητή, με μοναδικό όνομα που περιέχει το id της κάθε δημοσίευσης του χρήστη. Έτσι με το πάτημα του κουμπιού “citation chart” στην εφαρμογή, περνιέται σαν παράμετρος σε μία Javascript μέθοδο το id της δημοσίευσης. Στη συνέχεια η μέθοδος περνά το id αυτό μέσω του URL, στο pop up παράθυρο που εμφανίζει τη γραφική παράσταση. Στον κώδικα του pop up παραθύρου, γίνεται GET από το URL και τη μεταβλητή chart, το id. Έπειτα συνθέτεται το όνομα της $_SESSION που περιέχει τα δεδομένα προσθέτοντας μπρο-

στά από το id τη συμβολοσειρά chart_ και εξάγεται ο πίνακας με τα δεδομένα για τη γραφική παράσταση. Εν συνεχεία τα δεδομένα περνιούνται στο Google charts και τέλος παρουσιάζεται η γραφική παράσταση στην οθόνη του χρήστη, βλ εικόνα 4-33.

```
// Creates chart
$chart = new GoogChart();
if(isset($_GET['chart'])){
    $name = "chart_" . $_GET['chart'];
    $data = $_SESSION[$name];
    echo "<!DOCTYPE html PUBLIC '-//W3C//DTD XHTML 1.0 Strict//EN' 'http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd'>
    <html xmlns='http://www.w3.org/1999/xhtml' xml:lang='en' lang='en'>
        <head>
            <title>Paper Chart</title>
        </head>
        <body class='color'>";
    echo '<h2>Citations</h2>';
    //passes the chart parameters
    $chart->setChartAttrs( array(
        'type' => 'pie',
        'title' => 'Paper citations',
        'data' => $data,
        'size' => array( 400, 300 )
        ));
    // Prints chart
    echo $chart;
    echo "</body>
    </html>";
```

Εικόνα 4-33 Ο κώδικας του pop up παραθύρου για τη παρουσίαση της γραφικής παράστασης

5 Μελέτη Περιπτώσεων

Σε αυτό το κεφάλαιο θα ελεγχθεί η λειτουργία της εφαρμογής. Θα συγκριθούν οι χρόνοι που χρειάζεται ένας χρήστης για να ελέγξει τα αποτελέσματα από του Google scholar και της εφαρμογής.

5.1 Εύρεση Αναφορών με Google Scholar

Σε αναζήτηση των εργασιών και βιβλιογραφικών αναφορών του κ. Φώτη Κόκκορα, με κλειδί αναζήτησης το επίθετό του “kokkoras”, το Google Scholar επέστρεψε 106 αποτελέσματα, σε 11 σελίδες αποτελεσμάτων.

Δειγματοληπτικά παίρνουμε τα 10 πρώτα αποτελέσματα που επιστρέφει το Google Scholar. Με μια γρήγορη ματιά, αμέσως αποκλείουμε δύο αποτελέσματα για το λόγο ότι αναφέρονται στο τομέα της γεωπονίας, ενώ ο συγγραφέας που αναζητούμε ειδικεύεται στο τομέα της πληροφορικής.

Στη συνέχεια ελέγχοντας τα υπόλοιπα 8 αποτελέσματα, διαπιστώνουμε ότι τα 7 πρόκειται για εργασίες του συγγραφέα, ενώ ένα πρόκειται για βιβλιογραφική αναφορά. Ακολουθώντας τον σύνδεσμο κάθε αποτελέσματος εντοπίστηκε το όνομα του συγγραφέα στην αρχή της εργασίας συνοδευόμενο από τα ονόματα των υπολοίπων συγγραφέων και όχι στις βιβλιογραφικές αναφορές. Για αυτή τη διαδικασία χρειάστηκαν περίπου ένα με δύο λεπτά, για το άνοιγμα των συνδέσμων και τον έλεγχο του ονόματος. Για τον έλεγχο 10 αποτελεσμάτων χρειάζονται 10 με 15 λεπτά.

Οι σύνδεσμοι “cited by” αυτών των αποτελεσμάτων υποδεικνύουν ότι υπάρχουν συνολικά 45 βιβλιογραφικές αναφορές γι αυτά. Για τον έλεγχο και των 45 βιβλιογραφικών αναφορών δαπανήθηκε περίπου 85 λεπτά και χρησιμοποιήθηκε ένας απλός κειμενογράφος, όπως το notepad, για να κρατούνται κάποιες σημειώσεις που θα βοηθούσαν στον έλεγχο. Με μια γρήγορη ματιά διαπιστώθηκε ότι από τις 45 αναφορές, 35 ήταν μοναδικά αποτελέσματα. Κάποιες ήταν η ίδια εργασία που παρουσιαζόταν διαφορετικά από το Google Scholar, ενώ υπήρχαν και εργασίες του ίδιου του συγγραφέα που υπήρχαν στα αρχικά 7 απο-

τελέσματα. Υπήρχαν επίσης και εργασίες του συγγραφέα, οι οποίες δεν υπήρχαν στα αρχικά αποτελέσματα και καταμετρήθηκαν κανονικά στα 35 μοναδικά αποτελέσματα. Από 35 αποτελέσματα, στα 25 εξακριβώθηκε ότι το όνομα του συγγραφέα υπήρχε στις βιβλιογραφικές αναφορές. Για τις υπόλοιπες 10 δεν ήταν δυνατή η ανάκτηση αντιγράφου της εργασίας για το λόγο ότι ζητήθηκε χρηματικό αντίτιμο από την ηλεκτρονική βιβλιοθήκη όπου βρισκόταν ή υπήρχε πρόβλημα με τον σύνδεσμο του αποτελέσματος.

Για κάθε ένα αποτέλεσμα από τα 35, δαπανήθηκαν 1 με 10 λεπτά για να ελεγχθούν. Καθώς συλλέγονταν όλο και περισσότερα το πρόβλημα που προέκυψε ήταν ότι πρέπει κανείς να ελέγξει όλες τα προηγούμενα που έχει στο σημειωματάριο του για τυχόν ομοιότητες, ούτως ώστε να τα αποκλείσει.

Όλα τα παραπάνω δείχνουν ότι παρόλο που τα αποτελέσματα του Google Scholar είναι αρκετά καλά, χρειάζεται μια επιπλέον βοήθεια στο χρήστη για να διαχειριστεί αποκλειστικά "με το χέρι" τα αποτελέσματα αυτά και κυρίως κάτι που θα βοηθήσει τη μνήμη του, ούτως ώστε να μη χρειάζεται να δαπανήσει τον ίδιο ή και περισσότερο χρόνο σε μια δεύτερη αναζήτηση του.

5.2 Εύρεση Αναφορών με την Εφαρμογή

Η παρούσα εφαρμογή αναπτύχθηκε, για να συμπληρώσει το Google Scholar και για να βοηθήσει τους χρήστες του να εντοπίσουν και να διαχειριστούν καλύτερα τις βιβλιογραφικές αναφορές και τις εργασίες τους.

Με το ίδιο κλειδί αναζήτησης και με τη βοήθεια του DEiXTo, εξάγουμε τα αποτελέσματα από το Google Scholar. Πρώτα αλλάζουμε τον αριθμό των σελίδων που πρόκειται να προσπελάσει από 10 σε 0, για να έχουμε ακριβώς το ίδιο δείγμα, όπως και στην αναζήτηση μόνο με το Google Scholar. Ουσιαστικά απενεργοποιούμε το check box “Enable Multi Page Crawling” και κατά αυτό τον τρόπο του δίνουμε την εντολή να διαβάσει μόνο τη πρώτη σελίδα των αποτελεσμάτων.

Στη δεύτερη φάση, περνάμε το XML αρχείο, που έχει εξαχθεί από το DEiXTo, στην εφαρμογή και μετά από τους ελέγχους που γίνονται, τα αποτελέσματα αποθηκεύονται αρχικά στη βάση και στη συνέχεια παρουσιάζονται στο χρήστη, στη σελίδα “raw data”. Αυτό που βλέπουμε είναι μία σελίδα αποτελε-

σμάτων καθώς όπως και το Google Scholar, η εφαρμογή παρουσιάζει μέχρι 10 αποτελέσματα σε κάθε σελίδα της λίστας “raw data”.

Όπως και προηγουμένως, ο χρήστης μπορεί να διαπιστώσει με μια γρήγορη ματιά τα δύο αποτελέσματα που φέρουν ένα μη σχετικό τίτλο και αναφέρονται σε άλλο συγγραφέα. Τα δύο αυτά αποτελέσματα, μαρκάρονται ως “garbage” και μεταφέρονται στην αντίστοιχη λίστα. Τα υπόλοιπα ελέγχονται κατά τον ίδιο τρόπο όπως και με την απλή μέθοδο και έτσι ο χρόνος που δαπανάται είναι ο ίδιος.

Καταλήγοντας στο συμπέρασμα ότι από τα 8 εναπομείναντα αποτελέσματα τα 7 είναι εργασίες του συγγραφέα, τα μαρκάρουμε ως έγγραφα “my papers” και ένα πρόκειται για βιβλιογραφική αναφορά, το μαρκάρουμε ως “my citation”. Για να συλλέξουμε τις βιβλιογραφικές αναφορές από αυτά τα έγγραφα, πηγαίνουμε στο μενού “upload” και πατάμε το κουμπί “download” για να κατεβάσουμε το καινούριο κανόνα, βλ. παράρτημα. Τοποθετούμαι το καινούριο κανόνα στο DEiXTo και αυτό εξάγει ένα νέο XML αρχείο με αποτελέσματα που είναι βιβλιογραφικές αναφορές στις εργασίες που έχουμε ήδη καταχωρήσει.

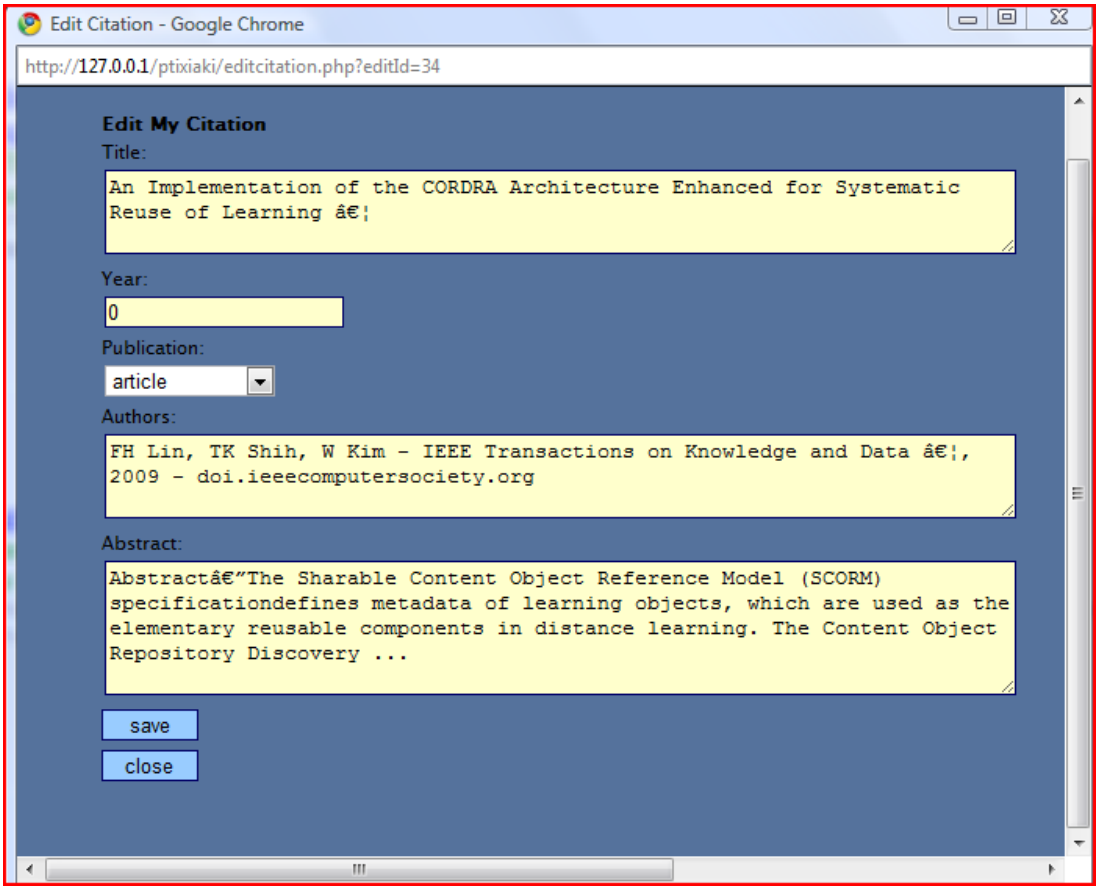
Ήδη κατά την εκχώρηση των αποτελεσμάτων του XML αρχείου, 6 αποτελέσματα από τα συνολικά 45 αποκλείστηκαν από την εφαρμογή, για το λόγο ότι δε πληρούσαν τις προϋποθέσεις που θέτουν οι διάφοροι έλεγχοι της εφαρμογής. Άλλα 3 αποτελέσματα εντοπίστηκαν με ευκολία από το ανθρώπινο μάτι. Αυτά λόγο της κακή δομής που είχαν πέρασαν από τον έλεγχο της εφαρμογής ως διαφορετικά αποτελέσματα. Για τα υπόλοιπα έπρεπε να γίνει ο ίδιος έλεγχος που έγινε και προηγουμένως, στην απλή αναζήτηση με το Google Scholar. Χωρίς να χρειάζεται η μεταφορά των αποτελεσμάτων σε κάποιο κειμενογράφο και παρουσιάζοντας καλύτερα τα αποτελέσματα δαπανήθηκαν περίπου 65 λεπτά για τον έλεγχο των τελικών 35 αποτελεσμάτων που απέμειναν στη λίστα. Από αυτά 25 αναγνωρίστηκαν σαν βιβλιογραφικές αναφορές και προωθήθηκαν στη λίστα “my citations”, ενώ τα υπόλοιπα 10 παρέμειναν με το χαρακτηριστικό “not-checked”, για να θυμίζουν στο χρήστη ότι πρέπει να ασχοληθεί ξανά με τα συγκεκριμένα.

Έπειτα και με τη βοήθεια του αποτελεσμάτων του συνδέσμου “cited by”, οι βιβλιογραφικές αναφορές συσχετίστηκαν με τις δημοσιεύσεις που είναι βιβλιογραφικές αναφορές.

5.3 Επιπλέον Λειτουργίες

Μετά την τοποθέτηση όλων αυτών των αποτελεσμάτων στις αντίστοιχες λίστες, ο χρήστης πρέπει να προβεί σε κάποιες πρόσθετες εργασίες. Θα πρέπει να επεξεργαστεί τις πληροφορίες των αποτελεσμάτων, για την καλύτερη εμφάνισή τους.

Παρόλο που αυτή η εργασία μοιάζει να προσθέτει επιπλέον χρόνο στο χρήστη από ότι θα συνέβαινε αν χρησιμοποιούσε τον απλό τρόπο, στην ουσία του εξοικονομεί αρκετό χρόνο. Σε κάθε περίπτωση ο χρήστης θα έπρεπε να συλλέξει και να τακτοποιήσει τα αποτελέσματα σε ένα, αρχείο κειμένου, σε ένα αρχείο excel ή σε κάποια άλλη μορφή όπως ένα XML αρχείο. Επίσης θα προέβαινε σε κάποια επεξεργασία των πληροφοριών, που θα βοηθούσαν στην ομοιομορφία των αποτελεσμάτων. Όλα τα παραπάνω μπορεί να τα κάνει συμπληρώνοντας τις φόρμες επεξεργασίας εγγράφου και βιβλιογραφικών αναφορών, που αυτοματοποιούν και επιταχύνουν αυτή τη διαδικασία, βλ. εικόνα 5-1.



Εικόνα 5-1 Η φόρμα για την επεξεργασία μιας βιβλιογραφικής αναφοράς.

Με το άνοιγμα της φόρμας, σε κάθε πεδίο εμφανίζονται οι υπάρχουσες πληροφορίες όπως φορτώνονται από τη βάση δεδομένων.

Κατά τη διαδικασία της επεξεργασίας των αποτελεσμάτων διαπιστώθηκε ότι, εκτός από τους λανθασμένους χαρακτήρες σε διάφορες πληροφορίες και τους κακογραμμένους τίτλους, υπήρχαν λάθη στα ονόματα των συγγραφέων καθώς και σε κάποιες εργασίες παραλείπονταν ονόματα συγγραφέων. Επίσης βελτιώθηκε το abstract κείμενο και προστέθηκαν ο τύπος και η χρονιά δημοσίευσης των εγγράφων.

Ύστερα από την ολοκλήρωση αυτής της διαδικασίας, ο χρήστης έχει μια πλήρη και καλογραμμένη εικόνα όλων των δημοσιεύσεων του με τις βιβλιογραφικές αναφορές σε αυτές, στο μενού “reports”, βλ εικόνα 5-2.

Εικόνα 5-2 Report του συστήματος

Όλες αυτές οι λειτουργίες εξοικονόμησαν αρκετό χρόνο από το χρήστη και αυτοματοποίησαν επαναληπτικές εργασίες που τυχόν θα έκανε ο χρήστης, την συλλογή και επεξεργασία των αποτελεσμάτων σε μια απλή αναζήτηση με το Google Scholar.

Η αποθήκευση όλων των πληροφοριών στη βάση δεδομένων βοηθά στην αυτοματοποίηση κάποιων εργασιών. Ο χρήστης θα έπρεπε να συντηρεί αρκετές λίστες με αποτελέσματα, σε αρχεία κειμένου ή άλλα αρχεία όπου θα μπορούσε να αποθηκεύσει κείμενο και σε περίπτωση μιας καινούριας προσθήκης, θα χρειαζόταν να ενημερώσει αρκετά αρχεία. Στην εφαρμογή όλα τα στοιχεία φορτώνονται από τη βάση δεδομένων και οτιδήποτε το καινούριο εντάσσεται αυτομάτως σε όλες τις λίστες και επίσης διατηρεί μία καλή εμφάνιση των αποτελεσμάτων, βλ. εικόνα 5-2.

Ωστόσο **το μεγαλύτερο πλεονέκτημα της εφαρμογής γίνεται αντιληπτό σε μια δεύτερη, τρίτη ή *n* - οστή, αναζήτηση**. Όσο μεγαλώνει ο αριθμός των αποτελεσμάτων, τόσο πιο δύσκολη γίνεται για το χρήστη, η αναγνώριση των αποτελεσμάτων και ο έλεγχος αυτών. Σε κάθε περίπτωση η εφαρμογή έχει αναπτυχθεί με σκοπό να εξοικονομεί αρκετό χρόνο στο χρήστη κάνοντας αυτόματα κάποιους ελέγχους και κυρίως υπενθυμίζοντάς του που αποτελέσματα αναζήτησης έχει διαλευκάνει. Με το σκεπτικό ότι ο χρήστης θα χρησιμοποιήσει πάνω από μία φορές την εφαρμογή, εξοικονομεί όλο και περισσότερο χρόνο αναλόγως με τον αριθμό των αποτελεσμάτων που υπάρχουν στη βάση δεδομένων για κάθε χρήστη. Όσο μεγαλύτερος είναι ο αριθμός των δημοσιεύσεων και των βιβλιογραφικών αναφορών σε αυτές, τόσο περισσότερο χρόνο μπορεί να εξοικονομήσει από τη δεύτερη αναζήτησή του και μετέπειτα.

6 Συμπεράσματα

Στην παρούσα πτυχιακή εργασία μελετήθηκε το πρόβλημα της αναζήτησης και διαχείρισης βιβλιογραφικών αναφορών (citations). Εξετάστηκαν διάφορες μηχανές αναζήτησης και ψηφιακές βιβλιοθήκες, όπως το citeseer^{x}, το DBLP και το Google Scholar, ενώ παρουσιάστηκαν τα πλεονεκτήματα και τα μειονεκτήματά τους κατά την παρουσίαση ή τη συσχέτιση των αποτελεσμάτων. Επιπλέον αναλύθηκαν τα προβλήματα που προκύπτουν σε μια αναζήτηση βιβλιογραφικών αναφορών και παρουσιάστηκαν διάφοροι μέθοδοι που μπορούν να δώσουν λύση.

Παρόλο που οι σχετικές μηχανές αναζήτησης βοηθούν στο έργο του εντοπισμού βιβλιογραφικών αναφορών, εντούτοις απαιτούν εκτενή χειρωνακτική εργασία από τον χρήστη, καθώς τα αποτελέσματα περιέχουν αρκετά λάθη και "θόρυβο". Το πρόβλημα εντείνεται κατά πολύ εξαιτίας της ανάγκης επανάληψης της διαδικασίας σε τακτικά χρονικά διαστήματα.

Για το λόγο αυτό αναπτύχθηκε μία διαδικτυακή εφαρμογή για την επεξεργασία (αποθήκευση, διαχείριση και φιλτράρισμα) των αποτελεσμάτων που επιστρέφει το Google Scholar. Η εφαρμογή προσδίδει με αυτό τον τρόπο ένα είδος μνήμης στη μηχανή αναζήτησης καθώς πλέον ο συνδυασμός Google Scholar και εφαρμογής έχει τη δυνατότητα να θυμάται την κρίση του χρήστη για τα διάφορα αποτελέσματα αναζήτησης. Σε κάθε νέα αναζήτηση αναφορών, το πρόγραμμα φιλτράρει τα αποτελέσματα χρησιμοποιώντας μία βάση δεδομένων όπου αποθηκεύονται όλες τις σχετικές πληροφορίες και οι κρίσεις που έκανε ο χρήστης για παλαιότερα αποτελέσματα αναζήτησης και παρουσιάζει στο χρήστη μόνο αυτά που ικανοποιούν τους ελέγχους. Για ένα ενεργό ερευνητή, η περιοδικά επαναλαμβανόμενη χρήση της εφαρμογής στα ολοένα και περισσότερα αποτελέσματα που θα επιστρέφει το Google Scholar, θα του εξοικονομήσει πολύ χρόνο και θα του επιτρέψει να τηρεί σχετικά εύκολα, ένα ενημερωμένο αρχείο αναφορών.

Θα ήταν δυνατό, σε μια βελτιωμένη έκδοση της εφαρμογής, να συμπεριληφθούν και άλλες μηχανές αναζήτησης βιβλιογραφικών αναφορών όπως το citeseer^{x} κ.α. Σε αυτή τη περίπτωση θα χρειαστεί να δημιουργηθούν καινούριοι κανόνες εξαγωγής στο DEiXTo, που θα μπορούν να εξάγουν πληροφορίες από τις σελίδες αποτελεσμάτων αυτών των μηχανών. Έτσι θα είναι δυνατή η περαιτέρω επεξεργασία τους από την εφαρμογή και εν συνεχεία η αποθήκευση τους στη βάση δεδομένων.

Μια άλλη βελτίωση που μπορεί να γίνει είναι η προσθήκη επιπλέον μεθόδων επίλυσης των προβλημάτων που αναφέρθηκαν, πέρα του αλγορίθμου μέτρησης απόστασης Levenshtein.

Τέλος θα είχε αξία να δημιουργηθεί κάποιος περισσότερο έξυπνος αλγόριθμος χειρισμού των αποτελεσμάτων, που θα ταξινομεί αυτόματα περισσότερα αποτελέσματα, εξοικονομώντας επιπλέον χρόνο στο χρήστη, ειδικά κατά τις πρώτες χρήσεις της εφαρμογής.

Βιβλιογραφία

ΑΡΘΡΑ ΚΑΙ ΒΙΒΛΙΑ

[1] Dongwon Lee, Byung Won On, Jaewoo Kang, Sanghyun Park - Effective and Scalable Solutions for Mixed and Split Citation Problems in Digital Libraries, 2005, Baltimore, MD, USA

[2] Dongwon Lee, Jaewoo Kang, Prasenjit Mitra, C. Lee Giles, Byung Won On - Are your citations clean? - Communications of the ACM, December 2007/ Vol. 50, No 12, 33-38

[3] Hanna Pasula, Bhaskara Marthi, Brian Milch, Stuart Russell, Ilya Shpitser - Identity Uncertainty and Citation Matching, - Advances in Neural Information Processing Systems, 2003

[4] M.V. Dodson - Biochemical and Biophysical Research Communications, 387 2009, 625-626

[5] Δρ Μιχάλης Σαλαμπάσης - Εισαγωγή στον προγραμματισμό διαδικτυακών εφαρμογών, Πρώτη έκδοση, 2005

ΙΣΤΟΣΕΛΙΔΕΣ

[6] Οδηγίες για γραπτές εργασίες: http://www.ionio.gr/depts/music/download.php?f=st_write_an_essay.doc

[7] Levenshtein Distance, http://en.wikipedia.org/wiki/Levenshtein_distance

[8] Levenshtein Distance, http://www.dcs.shef.ac.uk/~sam/stringmetrics.html#Levenshtein

[9] Jaro-Winkler Distance, http://en.wikipedia.org/wiki/Jaro_distance

[10] Cosine Similarity, http://www.dcs.shef.ac.uk/~sam/stringmetrics.html#cosine

[11] Cosine Similarity, http://en.wikipedia.org/wiki/Cosine_similarity

[12] TF-IDF weight on Wikipedia, http://en.wikipedia.org/wiki/TF_IDF

[13] TFIDF or TF/IDF, http://www.dcs.shef.ac.uk/~sam/stringmetrics.html#tfidf

[14] DBLP - Digital Bibliography & Library Project on Wikipedia http://en.wikipedia.org/wiki/DBLP

[15] DBLP - Digital Bibliography & Library Project, http://dblp.uni-trier.de/

[16] Microsoft Academic, http://academic.research.microsoft.com/

[17] Microsoft Live Search Academic on Wikipedia
http://en.wikipedia.org/wiki/Live_Search_Academic

[18] CiteSeer, http://citeseer.ist.psu.edu/

[19] CiteSeer^{x}, http://citeseerx.ist.psu.edu/

[20] CiteSeer^{x} on Wikipedia, http://en.wikipedia.org/wiki/Citeseer

[21] Google Scholar, http://scholar.google.com/

[22] Google Scholar on Wikipedia, http://en.wikipedia.org/wiki/Google_Scholar

[23] UCL INFORMATION SERVICES DIVISION WEB SERVICES - PHP/ MySQL, http://www.ucl.ac.uk/web-services/web-technologies/php-mysql

[24] XHTML, http://www.w3schools.com/XHTML/xhtml_html.asp

[25] DEiXTo – Data Extraction Tool, http://deixto.com/

[26] jQuery, http://jquery.com/

[27] Notepad ++, http://notepad-plus.sourceforge.net/uk/site.htm

[28] WAMP on Wikipedia, http://en.wikipedia.org/wiki/WAMP

[29] GIMP - Image Manipulation Program, http://en.wikipedia.org/wiki/GIMP

[30] Mozilla Firefox on Wikipedia, http://en.wikipedia.org/wiki/Firefox

[31] Microsoft Internet Explorer on Wikipedia
http://en.wikipedia.org/wiki/Internet_Explorer

[32] Google Chrome on Wikipedia
http://en.wikipedia.org/wiki/Google_Chrome

[33] Apple Safari on Wikipedia
http://en.wikipedia.org/wiki/Safari_%28browser%29

[34] Opera web browser on Wikipedia
http://en.wikipedia.org/wiki/Opera_browser

[35] PHP on Wikipedia, http://en.wikipedia.org/wiki/Php

[36] Description of usort function, PHP,
http://us2.php.net/manual/en/function.usort.php

Παράρτημα

Παρακάτω παραθέτονται οι δύο κανόνες του DEiXTo (DEiXTo extraction rules) που χρησιμοποιεί το σύστημα για να εξάγει δομημένα δεδομένα από το Google Scholar. Ο δεύτερος κανόνας διαμορφώνεται δυναμικά από το σύστημα καθώς οι web διευθύνσεις στις οποίες στοχεύει συγκεντρώνονται από τον πρώτο κανόνα.

ΚΑΝΟΝΑΣ 1: Χρησιμοποιείται για το χειρισμό ενός τυπικού αποτελέσματος από το Google Scholar.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Project SYSTEM "wpf.dtd">
<Project>
  <InputFile Filename=""/>
  <TargetUrls>
    <URL Address="http://scholar.google.com/scholar?q=kokkoras&amp;hl=en&amp;lr=&amp;btnG=Search"/>
  </TargetUrls>
  <MultiplePage Enabled="false" ContainsText="" MaxCrawlDepth="5"/>
  <SubmitForm Enabled="false" FormName="" InputName="" Term=""/>
  <ExtractionPattern>
  <Node tag="BODY" stateIndex="grayed">
    <Node tag="H3" stateIndex="grayed">
      <Node tag="FONT" stateIndex="grayed_implied">
        <Node tag="TEXT" stateIndex="grayed"/>
      </Node>
      <Node tag="A:fulltext_url" stateIndex="checked_implied">
        <Node tag="TEXT:title" stateIndex="checked"/>
      </Node>
      <Node tag="TEXT:title_no_link" stateIndex="checked_implied"/>
    </Node>
  <Node tag="FONT" stateIndex="grayed">
      <Node tag="SPAN" stateIndex="grayed">
```

```
        <Node tag="TEXT:author" stateIndex="checked" IsRoot="true"/>
      </Node>
      <Node tag="TEXT:abstract" stateIndex="checked_implied" CareAboutSO="1" so_start="1" so_step="0"/>
      <Node tag="SPAN" stateIndex="grayed">
        <Node tag="A:citations_url" stateIndex="checked_implied" regexpr=".*cites=.*">
          <Node tag="TEXT:citations_number" stateIndex="checked" regexpr="Cited by (\d+)"/>
          </Node>
        </Node>
      </Node>
    </Node>
  </ExtractionPattern>
  <IgnoredTagsList>
    <IgnoredTag Label="&lt;B&gt;"/>
    <IgnoredTag Label="&lt;STRONG&gt;"/>
    <IgnoredTag Label="&lt;I&gt;"/>
    <IgnoredTag Label="&lt;EM&gt;"/>
    <IgnoredTag Label="&lt;U&gt;"/>
    <IgnoredTag Label="&lt;BR&gt;"/>
    <IgnoredTag Label="&lt;NOBR&gt;"/>
    <IgnoredTag Label="&lt;HR&gt;"/>
    <IgnoredTag Label="&lt;DIV&gt;"/>
    <IgnoredTag Label="&lt;WBR&gt;"/>
  </IgnoredTagsList>
  <OutputFile Filename="" Format="XML" FileMode="Overwrite"/>
</Project>
```

ΚΑΝΟΝΑΣ 2: Δημιουργείται δυναμικά από την εφαρμογή, αναπροσαρμόζοντας έναν κατά τ' άλλα στατικό κανόνα. Η αναπροσαρμογή συνίσταται στον επαναπροσδιορισμό της ενότητα <TargetUrls> που περιέχει τις διευθύνσεις που επιστρέφει το Google Scholar πίσω από συνδέσμους [cited by] σε πρωτογενή αποτελέσματα αναζήτησης.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Project SYSTEM "wpf.dtd">
   <Project>
```

```
<InputFile Filename=""/>
<TargetUrls>
    <URL Address='http://scholar.google.com/scholar?cites=4113692542956863988&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=530667403726906714&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=5810164935030940058&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=10172568255371072024&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=707778294537428054 3&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=11943990019458492996&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=4264485119372406601&amp;hl=en'/>
    <URL Address='http://scholar.google.com/scholar?cites=1601487484450010385 0&amp;hl=en'/>
</TargetUrls>
<MultiplePage Enabled="false" ContainsText="" MaxCrawlDepth="5"/>
<SubmitForm Enabled="false" FormName="" InputName="" Term=""/>
<ExtractionPattern>
    <Node tag="BODY" stateIndex="grayed">
        <Node tag="H3" stateIndex="grayed">
            <Node tag="FONT" stateIndex="grayed_implied">
                <Node tag="TEXT" stateIndex="grayed"/>
            </Node>
            <Node tag="A:fulltext_url" stateIndex="checked_implied">
                <Node tag="TEXT:title" stateIndex="checked"/>
            </Node>
            <Node tag="TEXT:title_no_link" stateIndex="checked_implied"/>
        </Node>
        <Node tag="FONT" stateIndex="grayed">
            <Node tag="SPAN" stateIndex="grayed">
                <Node tag="TEXT:author" stateIndex="checked" IsRoot="true"/>
            </Node>
            <Node tag="TEXT:abstract" stateIndex="checked_implied" CareAboutSO="1" so_start="1" so_step="0"/>
            <Node tag="SPAN" stateIndex="grayed">
                <Node tag="A:citations_url" stateIndex="checked_implied" regexpr=".*cites=.*">
                    <Node tag="TEXT:citations_number" stateIndex="checked" regexpr="Cited by (\d+)"/>
                </Node>
            </Node>
        </Node>
    </Node>
</ExtractionPattern>
```

```
    <IgnoredTagsList>
        <IgnoredTag Label="&lt;B&gt;"/>
        <IgnoredTag Label="&lt;STRONG&gt;"/>
        <IgnoredTag Label="&lt;I&gt;"/>
        <IgnoredTag Label="&lt;EM&gt;"/>
        <IgnoredTag Label="&lt;U&gt;"/>
        <IgnoredTag Label="&lt;BR&gt;"/>
        <IgnoredTag Label="&lt;NOBR&gt;"/>
        <IgnoredTag Label="&lt;HR&gt;"/>
        <IgnoredTag Label="&lt;DIV&gt;"/>
        <IgnoredTag Label="&lt;WBR&gt;"/>
    </IgnoredTagsList>
    <OutputFile Filename="" Format="XML" FileMode="Overwrite"/>
</Project>
```