



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πτυχιακή Εργασία

**<<Διαδικτυακά εργαλεία για την διευκόλυνση
διαχείρισης συνεδρίων>>**



**Του φοιτητή
Μαδένη Ιωάννη
Αρ. Μητρώου: 021911**

**Επιβλέπων καθηγητής
Σιδηρόπουλος Αντώνης**

Θεσσαλονίκη 2010

ΠΡΟΛΟΓΟΣ

Σε προηγούμενη πτυχιακή εργασία, του Δημήτρη Χατζή, έχει γίνει σχεδιασμός του συστήματος διαχείρισης συνεδρίων, *PaperReview*. Το *PaperReview* έχει σαν στόχο την συλλογή, αξιολόγηση και τέλος την επιλογή επιστημονικών διατριβών που πρόκειται να δημοσιευτούν στα συνέδρια.

Σκοπός της πτυχιακής μου εργασίας είναι ο σχεδιασμός και υλοποίηση αλγορίθμου για αυτόματη ανάθεση εργασιών προς αξιολόγηση στα μέλη της επιτροπής με βάση τις προτιμήσεις που έχουν εισάγει, όπως επίσης και η ανάπτυξη εργαλείου `command line` και στη συνέχεια διεπιφάνειας για την δημιουργία και αποστολή *mail templates* στα μέλη.

ΠΕΡΙΛΗΨΗ

Στα κεφάλαια που θα ακολουθήσουν θα γίνει αναλυτική περιγραφή των τεχνολογιών που χρησιμοποιήθηκαν για να κατασκευαστούν τα διαδικτυακά εργαλεία διευκόλυνσης της εφαρμογής *PaperReview*, όπως επίσης και η περιγραφή του αλγορίθμου αυτόματης ανάθεσης εργασιών και τέλος η περιγραφή του *command line* εργαλείου για αποστολή *XML mail templates* μαζί με την ενσωμάτωσή του στην εφαρμογή *PaperReview*.

Στο πρώτο κεφάλαιο θα αναλυθούν οι τεχνολογίες ανάπτυξης της εργασίας, όπως είναι οι *PHP*, *HTML*, *XML*, *MySQL*, *JavaScript*, αλλά και η βασική αρχιτεκτονική του διαδικτύου, ο εξυπηρετητής, το λειτουργικό του σύστημα, οι σχεσιακές βάσεις δεδομένων και οι *web clients*.

Στο δεύτερο κεφάλαιο θα αναλυθεί ο αλγόριθμος αυτόματης ανάθεσης εργασιών στα μέλη του συνεδρίου. Θα αναφερθούν όλοι οι περιορισμοί και οι φραγμοί που υπάρχουν από την εφαρμογή για την ανάθεση των εργασιών. Θα δοθεί ψευδοκώδικας της υλοποίησης του αλγορίθμου όπως επίσης και τμήματα κώδικα, και *SQL* ερωτήματα για την πληρέστερη παρουσίασή του. Τέλος θα δούμε ένα παράδειγμα και τα αποτελέσματά του.

Στο τρίτο κεφάλαιο, που αναφέρεται στο *command line* εργαλείο αποστολής *mail templates*, θα παρουσιαστεί αναλυτικά ο σκελετός του *mail template* και η δομή του, το πώς επεξεργάζεται το *script* το *mail template* όπως και το πώς αποστέλλονται τα *mail* στους παραλήπτες, μαζί παραδείγματα.

Στο τέταρτο κεφάλαιο, θα δούμε το πώς το *command line* εργαλείο αποστολής *mail template* ενσωματώνεται στην εφαρμογή *PaperReview*, το πώς μπορεί ο χρήστης να επεξεργαστεί και να δημιουργήσει νέο *mail template* μέσω φόρμας, τις επιλογή αποστολής των μηνυμάτων με *smtp server* και το πώς ελέγχει την πρόοδο της κάθε αποστολής.

ABSTRACT

At the following chapters there will be a thorough description of the technologies that were used to build the web assistant tools of the PaperReview application, as well as the description of the automatic review assign algorithm and finally the description of the command line XML mail template sending tool with its implication on the PaperReview application.

The first chapter discusses the technologies that were used to make this paper, such as PHP, HTML, XML, MySQL, JavaScript and also describes the basic architecture of the internet, the server, the operating system the relational databases and web clients.

The second chapter will analyze the automatic review assignment algorithm to the reviewers of the conference. All restrictions and limits that exist in the application will be noted. Also a pseudo-code of the algorithm will be presented, as well as pieces of the real code and some SQL queries, to make a better presentation of it. Finally, results will be presented through a simple example.

The third chapter refers to the command line mail template sending tool. The structure and form of the email template will be shown as well as how the script parses the mail template and finally how the emails are sent through the script with a simple example

The fourth chapter shows how the command line mail sending tool is intergraded into the PaperReview application, how the user can edit and create new email templates through a form, the option to use an smtp server for sending the emails, and how the sending progress is checked.

ΕΥΧΑΡΙΣΤΙΕΣ

Το παρών έργο δεν θα έφτανε ποτέ εις πέρας χωρίς την συμπαράσταση, βοήθεια, ενδιαφέρον και ατελείωτη υπομονή του Κυρίου Αντώνη Σιδηρόπουλου μέχρι και σήμερα. Ένα μεγάλο γραπτό ευχαριστώ, μιας και τα γραπτά μένουν.

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1: Τεχνολογίες ανάπτυξης διαδικτυακών εργαλείων

1.1 Εισαγωγή	10
1.2 Το διαδίκτυο	10
1.3 Βασική αρχιτεκτονική διαδικτυακών εφαρμογών	10
1.4 Ο Server.....	11
1.5 Ο Client.....	12
1.6 Τεχνολογίες ανάπτυξης εργαλείων.....	12
1.6.1 HTML.....	12
1.6.2 XML.....	13
1.6.3 PHP.....	15
1.6.4 PHP & SimpleXML.....	16
1.6.5 MySQL.....	16
1.6.6 JavaScript.....	16
1.7 Επίλογος.....	17

Κεφάλαιο 2: Αλγόριθμος αυτόματης ανάθεσης διατριβών στους αξιολογητές του συνεδρίου

2.1 Εισαγωγή.....	19
2.2 Ο σκοπός του αλγορίθμου.....	19
2.3 Αλλαγές στην εφαρμογή PaperReview.....	19
2.4 Υλοποίηση του αλγορίθμου.....	20
2.4.1 Αρχικές ενέργειες συγκέντρωσης απαραίτητων δεδομένων.....	23
2.4.2 Συγκέντρωση διαθέσιμων θέσεων για ανάθεση.....	24
2.4.3 Βελτιστοποίηση αναθέσεων.....	25
2.5 Αποτελέσματα.....	27
2.6 Παράδειγμα εφαρμογής αλγορίθμου.....	28

Κεφάλαιο 3: Εργαλείο αποστολής mail template

3.1 Εισαγωγή.....	32
3.2 Γενική δομή ενός mail template.....	32
3.2.1 Στοιχείο <smtpconfig>.....	33
3.2.2 Στοιχείο <mysqlconfig>.....	33
3.2.3 Στοιχείο <query>.....	34
3.2.4 Στοιχείο <email>.....	35
3.3 Ορισμός Σκελετού Template.....	37
3.4 Ανάλυση του script αποστολής mail template.....	39
3.4.1 Έλεγχοι ορθής εκτέλεσης του script.....	39
3.4.2 Εντολές επεξεργασίας του xml template.....	40
3.4.3 Διαδικασία αποστολής των email.....	41

Κεφάλαιο 4: Ενσωμάτωση εργαλείου αποστολής μηνυμάτων και αλλαγές περιβάλλοντος εφαρμογής

4.1 Εισαγωγή.....	43
4.2 Σκοπός του εργαλείου αποστολής mail template στην εφαρμογή PaperReview.....	45
4.3 Αλλαγές στην δομή του mail template.....	45
4.4 Αλλαγές στην εφαρμογή PaperReview.....	45
4.5 Διαδικασία αποστολής mail template στους χρήστες.....	51
4.6 Επίλογος.....	52
Βιβλιογραφία.....	53
Σημειώσεις.....	53

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ & ΠΙΝΑΚΩΝ

ΕΙΚΟΝΕΣ

Κεφάλαιο 1

Εικόνα 1.1 : Βασική αρχιτεκτονική διαδικτυακών εφαρμογών.....11

Κεφάλαιο 2

Εικόνα 2.1: Αλλαγές στην Εφαρμογή, στο μενού "Assign Papers".....20

Εικόνα 2.2: Ψευδοκώδικας για τον αλγόριθμο αυτόματης ανάθεσης.....26

Εικόνα 2.3: Δείγμα μορφής εκτύπωσης αποτελεσμάτων αλγορίθμου.....27

Κεφάλαιο 3

Εικόνα 3.1: Βασικά στοιχεία του mail template.....32

Εικόνα 3.2: Στοιχεία-παιδιά του πεδίου <smtpconfig>.....33

Εικόνα 3.3: Στοιχεία-παιδιά του πεδίου <mysqlconfig>.....34

Εικόνα 3.4: Παράδειγμα ιδιοτήτων, παιδιών του στοιχείου <query>.....34

Εικόνα 3.5: Παράδειγμα μηνύματος χωρίς subquery.....35

Εικόνα 3.6: Παράδειγμα μηνύματος με subqueries.....36

Εικόνα 3.7: Το αρχείο DTD που περιέχει την δομή του template που μπορεί να επεξεργαστεί το script.....37

Εικόνα 3.8: Ο σκελετός του template σε XML.....38

Εικόνα 3.9: Συνάρτηση αποστολής email μέσω smtp server.....41

Κεφάλαιο 4

Εικόνα 4.1: Αλλαγές στην γενική δομή του mail template.....44

Εικόνα 4.2: Αλλαγές και προσθήκες στο αρχείο DTD.....45

Εικόνα 4.3: Κύρια διεπαφή εργαλείου Email Tool.....45

Εικόνα 4.4: Λίστες με έτοιμα mail templates και δημιουργημένα email.....46

Εικόνα 4.5: Φόρμα Παραμετροποίησης SMTP Server.....47

Εικόνα 4.6: Λίστα με αρχεία καταγραφής διαδικασιών αποστολής.....49

Εικόνα 4.7: Φόρμα επεξεργασίας mail template.....50

ΠΙΝΑΚΕΣ

Κεφάλαιο 2

Πίνακας 2.1: Ο πίνακας Interest.....	21
Πίνακας 2.2: Ο πίνακας paper.....	22
Πίνακας 2.3: Ο πίνακας PaperToReviewer.....	23
Πίνακας 2.4: Πίνακας αξιολογητών/διατριβών και βαθμού ενδιαφέροντος.....	28
Πίνακας 2.5: Πίνακας με τυχαίες αναθέσεις για κάθε διατριβή.....	29
Πίνακας 2.6: Τελικός Πίνακας αναθέσεων.....	30
Πίνακας 2.7: Πίνακας συνολικών αναθέσεων σε αξιολογητές.....	30

Κεφάλαιο 4

Πίνακας 4.1: Ο πίνακας SMTP στην βάση δεδομένων.....	48
--	----

Κεφάλαιο 1: Τεχνολογίες ανάπτυξης διαδικτυακών εργαλείων

1.1 Εισαγωγή	10
1.2 Το διαδίκτυο	10
1.3 Βασική αρχιτεκτονική διαδικτυακών εφαρμογών	10
1.4 Ο Server	11
1.5 Ο Client	12
1.6 Τεχνολογίες ανάπτυξης εργαλείων	12
1.6.1 HTML	12
1.6.2 XML.....	13
1.6.3 PHP	15
1.6.4 PHP & SimpleXML	16
1.6.5 MySQL.....	16
1.6.6 JavaScript	16
1.7 Επίλογος.....	17

1.1 Εισαγωγή

Η εφαρμογή PaperReview και κατά συνέπεια τα διαδικτυακά εργαλεία που ανέπτυξα χρησιμοποιούν την γλώσσα XHTML για την σήμανση των διαφορετικών περιεχομένων που περιλαμβάνει, όπως εικόνες, σύνδεσμοι και κείμενο. Οι εντολές σε PHP διερμηνεύονται από την web server και παράγουν τον κώδικα HTML, ανάλογα με τα αιτήματα διαφορετικών χρηστών. Τα δεδομένα του συστήματος διατηρούνται οργανωμένα σε μια βάση δεδομένων MySQL , ενώ όλα τα αρχεία του βρίσκονται σε έναν εξυπηρετητή Apache.

Στο παρών κεφάλαιο θα παρουσιαστούν οι τεχνολογίες που προαναφέρθηκαν για την ανάπτυξη των εργαλείων, όπως και μια σύντομη περιγραφή της αρχιτεκτονικής του διαδικτύου και των διαδικτυακών εφαρμογών.

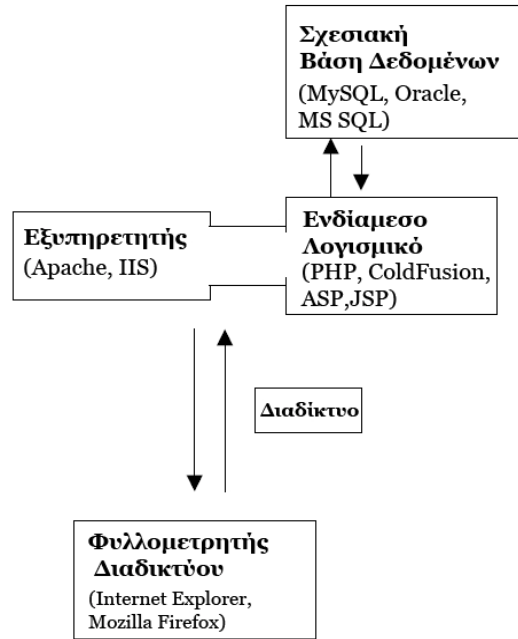
1.2 Το διαδίκτυο

Το διαδίκτυο μπορεί να οριστεί ως η τεχνολογική πλατφόρμα στην οποία ενοποιούνται, με ενσύρματα και ασύρματα μέσα σύνδεσης, διάφορες υποδομές που επικοινωνούν μεταξύ τους μέσω κοινών εφαρμογών λογισμικού. Ανάμεσα στις υποδομές (που συνεχώς αυξάνονται), κυρίαρχη είναι η θέση των υπολογιστών και των φυσικών μέσων που τους συνδέουν, καθώς αποτελούν το έρεισμα του Internet και το υποστήριγμα για οποιαδήποτε επέκταση. Το υπολογιστικό αυτό σύνολο απαρτίζεται από απλούς υπολογιστές (προσωπικούς), από υπολογιστές που αποτελούν τους κόμβους ή τους σταθμούς εργασίας ενός τοπικού δικτύου μικρής (LAN -Local Area Network) ή μεγάλης εμβέλειας (WAN -Wide Area Network) και από τους διακομιστές (servers).

1.3 Βασική αρχιτεκτονική διαδικτυακών εφαρμογών

Η αρχιτεκτονική του συστήματος επηρεάζει τη λειτουργικότητα, απόδοση, αποτελεσματικότητα, συντήρηση και επέκταση του καθ' όλη τη διάρκεια ζωής του και για αυτό χρειάζεται ιδιαίτερη προσοχή στο αρχικό στάδιο σχεδίασης ώστε το σύστημα να είναι σωστά δομημένο.

Η πιο γνωστή αρχιτεκτονική τα τελευταία είκοσι χρόνια, κυρίως σε εφαρμογές βάσεων δεδομένων, είναι αυτή του πελάτη - εξυπηρετητή (client-server). Σε αυτή την αρχιτεκτονική, ο πελάτης στέλνει ένα αίτημα (request) για δεδομένα στον εξυπηρετητή και αυτός επιστρέφει την απάντηση (response), την οποία επεξεργάζεται ο πελάτης και εμφανίζει στο χρήστη τα αποτελέσματα, όπως φαίνεται στην εικόνα 1.1.



Εικόνα 1.1 : Βασική αρχιτεκτονική διαδικτυακών εφαρμογών

1.4 Ο Server

Αν και ο client κατέχει ένα μεγάλο μέρος της προσοχής, αφού με τον client αλληλεπιδρούν οι χρήστες, ο server είναι η καρδιά του client-server συστήματος. Οι servers είναι τα σημεία όπου αποθηκεύονται οι πληροφορίες και εκτελούνται οι εργασίες. Ο server που χρησιμοποιεί η εφαρμογή είναι ο γνωστός Apache.

Ο Apache Server είναι ένας αξιοσημείωτος web server και διαδραματίζει καίριο ρόλο στην αρχική ανάπτυξη του διαδικτύου. Στο 2009 έγινε ο πρώτος web server που ξεπέρασε το ορόσημο των 100 εκατομμύριων ιστοσελίδων. Ο Apache ήταν η πρώτη εναλλακτική λύση για τον Netscape Communications Server web Corporation (γνωστός σήμερα ως Sun Java System Web Server), και έκτοτε έχει εξελιχθεί σε αντίπαλος για άλλους Unix servers όσον αφορά τη λειτουργικότητα και την απόδοση. Η πλειονότητα των web servers που χρησιμοποιεί Apache τρέχει το λειτουργικό σύστημα Linux.

Ο Apache αναπτύσσεται και συντηρείται από μια ανοιχτή κοινότητα προγραμματιστών υπό την αιγίδα του Apache Software Foundation. Η εφαρμογή είναι διαθέσιμη για μια μεγάλη ποικιλία λειτουργικών συστημάτων, συμπεριλαμβανομένων των Unix, το GNU, το FreeBSD, Linux, Solaris, Novell NetWare, Mac OS X, Microsoft Windows, OS / 2, TPF, και eComStation. Βάσει της Apache License, ο Apache χαρακτηρίζεται ως λογισμικό ανοικτού κώδικα.

Από τον Απρίλιο 1996 ο Apache είναι ο πιο δημοφιλής διακομιστής HTTP για το διαδίκτυο. Μέχρι τον Σεπτεμβρίου του 2009, ο Apache χρησιμοποιείται σε πάνω από το 54,48% του συνόλου των δικτυακών τόπων και πάνω από το 66% των πολυσύχναστων ιστοσελίδων με εκατομμύρια επισκέπτες.

1.5 Ο Client

Οποιοσδήποτε web browser (Internet Explorer, Firefox, Chrome), που παίρνει τον ρόλο του client, είναι υπεύθυνος για την γραφική απεικόνιση μιας διαδικτυακής εφαρμογής. Κύρια γλώσσα που χρησιμοποιείται για τέτοιου τύπου εφαρμογές είναι η HTML(HyperText Markup Language) και με την βοήθεια των CSS(Cascading Style Sheets) έχουμε την κατάλληλη μορφοποίηση της διεπιφάνειας.

Πέρα από την HTML, μπορούμε να έχουμε διάφορα πολυμεσικά επιπρόσθετα στοιχεία, όπως είναι τα Flash animations, Windows Media ή Real Time ή Quick Time videos, streaming video και audio. Τέλος, σημαντικό ρόλο για την αλληλεπίδραση του χρήστη με την εφαρμογή παίζει η JavaScript.

1.6 Τεχνολογίες ανάπτυξης εργαλείων

Ακολουθούν αναλυτικές περιγραφές για όλες τις τεχνολογίες του χρησιμοποιήθηκαν για την ανάπτυξη των βοηθητικών εργαλείων της εφαρμογής PaperReview.

1.6.1 HTML

Τα αρχικά HTML προέρχονται από τις λέξεις HyperText Markup Language. Η html δεν είναι μια γλώσσα προγραμματισμού. Είναι μια περιγραφική γλώσσα (markup language), δηλαδή ένας ειδικός τρόπος γραφής κειμένου. Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου.

Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Ο browser αναγνωρίζει αυτόν τον τρόπο γραφής και εκτελεί τις εντολές που περιέχονται σε αυτόν. Αξίζει να σημειωθεί ότι η html είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας.

Η html χρησιμοποιεί τις ειδικές ετικέτες (τα tags) να δώσει τις απαραίτητες οδηγίες στον browser. Τα tags είναι εντολές που συνήθως ορίζουν την αρχή ή το τέλος μιας λειτουργίας. Τα tags βρίσκονται πάντα μεταξύ των συμβόλων < και >. Π.χ. <BODY> Οι οδηγίες είναι case insensitive, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία. Ένα αρχείο HTML πρέπει να έχει κατάληξη htm ή html.

1.6.2 XML

Σε ένα κόσμο όπου οι πληροφορίες παρέχονται μέσω του παγκόσμιου διαδικτύου, τα έγγραφα πρέπει να είναι εύκολα προσβάσιμα, μεταφέρσιμα και ευέλικτα. Πρέπει επίσης να είναι ανεξάρτητα οποιουδήποτε συστήματος και περιεχομένου. Οι γενικευμένες γλώσσες έχουν τέτοια χαρακτηριστικά, παρέχοντας στα έγγραφα αυτά μια δυνατότητα η οποία δεν υπάρχει σε άλλες γλώσσες περιγραφής εγγράφων. Η HTML είναι προβληματική και περιοριστική γλώσσα. Η XML έλυσε πολλά από τα προβλήματα που αντιμετώπισαν οι σχεδιαστές του web και είναι υπεύθυνη για την XHTML, μια ανασχεδιασμένη HTML. Θα χρησιμοποιείται για πολλά χρόνια επειδή προσφέρει αποτελεσματικές και δυναμικές πολυμεσικές λύσεις.

Η XML σχεδιάστηκε να ικανοποιήσει πολλές ανάγκες δίνοντας στα έγγραφα ένα μεγαλύτερο επίπεδο προσαρμοστικότητας στο στυλ και τη δομή από αυτό που υπήρχε παλαιότερα στην HTML. Η XML προσφέρει στους σχεδιαστές της HTML τη δυνατότητα να προσθέτουν περισσότερα στοιχεία στη γλώσσα. Δεν αναφέρεται μονάχα στους σχεδιαστές του web αλλά σε οποιονδήποτε ασχολείται με εκδόσεις.

Στην πραγματικότητα, η XML είναι markup γλώσσα για έγγραφα που περιέχουν δομημένες πληροφορίες. Markup γλώσσα είναι ένας μηχανισμός που καθορίζει δομές σε ένα έγγραφο. Οι δομημένες πληροφορίες περιλαμβάνουν περιεχόμενο και κάποιες διευκρινίσεις για το ρόλο που παίζει το περιεχόμενο. Σχεδόν όλα τα έγγραφα έχουν την ίδια δομή.

Η XML είναι κάτι περισσότερο από markup language είναι metalanguage, δηλαδή μια γλώσσα που χρησιμοποιείται για να καθορίσει νέες markup γλώσσες. Η XML συμπληρώνει και δεν αντικαθιστά την HTML. Ενώ η HTML χρησιμοποιείται στη διατύπωση και την εμφάνιση των δεδομένων η XML αναπαριστά τη συναφή έννοια των δεδομένων. Στην HTML τα tags είναι προκαθορισμένα ενώ η XML παρέχει τη δυνατότητα να καθορίζουν οι χρήστες τα tags και τις δομημένες μεταξύ τους σχέσεις.

Τα XML έγγραφα δεν είναι πολύπλοκα αλλά απλά και πολύ αποτελεσματικά. Το διδακτικό υλικό της well-formed XML αναλύει τη δημιουργία των XML εγγράφων, η οποία είναι κατά κάποιο τρόπο ίδια με την HTML καθώς

επιτρέπει τη μη δομημένη δημιουργία εγγράφου. Η valid XML είναι πιο σύνθετη. Απαιτεί την ύπαρξη ενός DTD(Document Type Definition) πριν να γραφεί το έγγραφο αλλά παρέχει μια γενική δομή με βάση την οποία τη δημιουργούμε.

Η γλώσσα προγραμματισμού XML περιγράφει μια κατηγορία πληροφοριών (data objects) που καλούνται XML έγγραφα (documents) καθώς επίσης περιγράφει τμηματικά τη συμπεριφορά των προγραμμάτων που τα επεξεργάζονται.

Τα XML έγγραφα αποτελούνται από μονάδες αποθήκευσης που καλούνται entities (οντότητες), οι οποίες περιέχουν πληροφορίες αναλυμένες ή μη. Οι αναλυμένες πληροφορίες αποτελούνται από χαρακτήρες (characters) οι οποίοι συνθέτουν character data και άλλοι οι οποίοι συνθέτουν markup. Η μορφή markup κωδικοποιεί την περιγραφή της τελικής αποθήκευσης του εγγράφου καθώς και τη λογική δομή.

Πρόελευση και στόχοι

Η γλώσσα XML αναπτύχθηκε από μια Ομάδα Εργασίας της XML κάτω από την καλή κηδεμονία του διεθνούς οργανισμού World Wide Web Consortium (W3C) το 1996. Εδραιώθηκε από τον John Bosak της Sun Microsystems με την ενεργή συμμετοχή μιας XML Ομάδας Ειδικού Ενδιαφέροντος (που οργανώθηκε από τον οργανισμό W3C).

Οι προσχεδιασμένοι στόχοι της XML είναι:

1. Η XML πρέπει να είναι εύχρηστη στο Internet.
2. Η XML πρέπει να υποστηρίζει μεγάλη ποικιλία από εφαρμογές.
3. Η XML πρέπει να είναι συμβατή με την SGML.
4. Θα είναι εύκολο να γράφονται προγράμματα που επεξεργάζονται XML έγγραφα.
5. Ο αριθμός των προαιρετικών χαρακτηριστικών στην XML θα είναι όσο το δυνατόν πιο μικρός, ιδανικό επίπεδο το μηδέν.
6. Τα XML έγγραφα θα πρέπει να είναι ευανάγνωστα.
7. Ο σχεδιασμός XML θα πρέπει να προετοιμάζεται γρήγορα.
8. Ο σχεδιασμός XML θα πρέπει να είναι τυπικός και περιεκτικός.
9. Τα XML έγγραφα θα πρέπει να δημιουργούνται εύκολα.
10. Η περιεκτικότητα στον XML συμβολισμό είναι μικρής σημασίας.

1.6.3 PHP

Η PHP είναι μια ευρέως χρησιμοποιημένη, γενικής χρήσης scripting γλώσσα που σχεδιάστηκε αρχικά για την ανάπτυξη διαδικτυακών εφαρμογών, για να παραγάγει δυναμικές ιστοσελίδες. Μπορεί να ενσωματωθεί σε HTML και τρέχει γενικά σε έναν server, ο οποίος πρέπει να διαμορφωθεί για να επεξεργάζεται τον κώδικα PHP και να δημιουργήσει το περιεχόμενο της ιστοσελίδας από αυτόν. Μπορεί να επεκταθεί στους περισσότερους server και σχεδόν σε κάθε λειτουργικό σύστημα και πλατφόρμα χωρίς κόστος. Η PHP είναι εγκατεστημένη σε πάνω από 20 εκατομμύρια ιστοσελίδες και 1 εκατομμύριο servers. Σημαντικές ιστοσελίδες έχουν γραφτεί σε PHP, όπως το Facebook, η Wikipedia (MediaWiki), το Yahoo!, το MyYearbook, το Digg, το Joomla, το WordPress, το YouTube στα αρχικά του στάδια, το Drupal και το Tagged.

Η PHP δημιουργήθηκε αρχικά από τον Rasmus Lerdorf το 1995 και είναι σε συνεχή ανάπτυξη από τότε. Η κύρια εφαρμογή της PHP παράγεται τώρα από το "The PHP Group" και χρησιμεύει ως το πρότυπο για την PHP δεδομένου ότι δεν υπάρχει καμία επίσημη προδιαγραφή. Η PHP ανήκει στην οικογένεια του ελεύθερου λογισμικού.

Οποιοσδήποτε κώδικας PHP σε ένα ζητούμενο αρχείο εκτελείται στον χρόνο εκτέλεσης της PHP, για να δημιουργήσει συνήθως το δυναμικό περιεχόμενο της ιστοσελίδας. Μπορεί επίσης να χρησιμοποιηθεί για εγγραφή script από γραμμή εντολών και για εφαρμογές client-side με διεπαφή (GUI - Graphical User Interface). Η PHP μπορεί να εγκατασταθεί στους περισσότερους server, πολλά λειτουργικά συστήματα και πλατφόρμες, και μπορεί να χρησιμοποιηθεί με πολλά συστήματα διαχείρισης βάσεων δεδομένων (RDBMS).

Η PHP ενεργεί πρώτιστα ως φίλτρο, παίρνοντας την εισαγωγή δεδομένων από ένα αρχείο ή ένα ρεύμα εισόδου που περιέχει κείμενο ή/και οδηγίες PHP και βγάζει έξοδο ένα άλλο ρεύμα δεδομένων που συνήθως είναι η παραγωγή HTML.

Αρχικά σχεδιασμένη για να δημιουργεί δυναμικές ιστοσελίδες, η PHP εστιάζει τώρα κυρίως στην εγγραφή server-side script και είναι παρόμοια με άλλες server-side scripting γλώσσες που παρέχουν το δυναμικό περιεχόμενο από έναν server σε έναν client, όπως η Active Server Pages της Microsoft (ASP), η JavaServer της Sun Microsystems και η mod_perl. Η PHP έχει προσελκύσει επίσης την ανάπτυξη πολλών frameworks που παρέχουν τις δομικές μονάδες και μια δομή σχεδίου για να προωθήσουν τη γρήγορη ανάπτυξη μιας εφαρμογής (RAD - rapid application development). Μερικά από αυτά είναι: CakePHP, Symfony, CodeIgniter, και Zend Framework, που προσφέρουν χαρακτηριστικά γνωρίσματα παρόμοια με άλλα frameworks για εφαρμογές Ιστού.

Οι εφαρμογές LAMP και WAMP έχουν γίνει δημοφιλείς στη βιομηχανία Ιστού ως τρόπος ανάπτυξης διαδικτυακών εφαρμογών. Η PHP αντιπροσωπεύει συνήθως το "P" σε αυτά τα πακέτα μαζί με Linux, Apache και MySQL, αν και το "P" μπορεί επίσης να σημαίνει Python ή Perl.

1.6.4 PHP & SimpleXML

Η οικογένεια συναρτήσεων της επέκτασης της PHP SimpleXML είναι ένας αναλυτής εγγράφων XML βασισμένος στην δενδροειδή ανάλυση, διαθέσιμος στην PHP 5. Σε σύγκριση με την επέκταση DOM, η οποία έχει μια εκτενή διεπαφή προγραμματισμού (API) και επιτρέπει πλήρη έλεγχο και χειρισμό ενός εγγράφου XML, η SimpleXML υιοθετεί μια διαφορετική μέθοδο χειρισμού του δέντρου και έχει μια πολύ μικρότερη διεπαφή προγραμματισμού και χειρίζεται τα στοιχεία και τις ιδιότητες πιο διαισθητικά.

Η επέκταση SimpleXML δημιουργήθηκε για να παρέχει έναν εύκολο, διαισθητικό τρόπο να υποβληθούν σε επεξεργασία τα έγγραφα XML. Σε πολλές περιπτώσεις, τα έγγραφα δεν είναι υπερβολικά σύνθετα και όλα αυτά που έχουμε να φροντίσουμε είναι τα στοιχεία (elements), οι ιδιότητες (attributes), και το κείμενο μέσα στους κόμβους. Επομένως η χρησιμοποίηση του DOM μπορεί να είναι υπερβολική με όλους τους διαφορετικούς τύπους επεξεργασίας του και την πληθώρα των μεθόδων του, που δεν θα χρησιμοποιήσετε ποτέ. Σε αντίθεση, η επέκταση SimpleXML έχει έναν ενιαίο τύπο κατηγορίας, τρεις λειτουργίες, και έξι μεθόδους σύνολο και αυτός είναι ο λόγος που αποκαλείται και SimpleXML.

1.6.5 MySQL

Η MySQL είναι ένα σύστημα διαχείρισης βάσεων δεδομένων (RDBMS) που έχει εγκατασταθεί σε περισσότερα από 6 εκατομμύρια συστήματα. MySQL σημαίνει "My Structured Query Language". Το πρόγραμμα τρέχει ως server που παρέχει την πρόσβαση πολλών χρηστών σε διάφορες βάσεις δεδομένων.

Η MySQL χρησιμοποιείται συχνά στα ελεύθερα προγράμματα λογισμικού που απαιτούν ένα πλήρες σύστημα διαχείρισης βάσεων δεδομένων, όπως το WordPress, το phpBB και άλλα λογισμικά που στηρίζονται στο πακέτο λογισμικού LAMP. Χρησιμοποιείται επίσης στα πολύ υψηλής κλίμακας προϊόντα του παγκόσμιου ιστού συμπεριλαμβανομένου της Wikipedia, του Google και του Facebook.

1.6.6 JavaScript

Η JavaScript είναι μια αντικειμενοστραφής scripting γλώσσα που χρησιμοποιείται για να επιτρέψει την πρόσβαση στα αντικείμενα και μέσα στην εφαρμογή του client και σε άλλες εφαρμογές. Χρησιμοποιείται πρώτιστα υπό μορφή client-side JavaScript, που εφαρμόζεται ως ενσωματωμένο στοιχείο του web browser, που επιτρέπει την ανάπτυξη ενισχυμένων διεπαφών με τον χρήστη και των δυναμικών ιστοσελίδων. Η JavaScript ακολουθεί τα πρότυπα ECMAScript και χαρακτηρίζεται ως δυναμική, πρωτότυπη γλώσσα με πρώτης τάξεως

μεθόδους. Η JavaScript επηρεάστηκε από πολλές γλώσσες και είχε ως σκοπό να μοιάσει με την Java, αλλά να είναι ευκολότερο για τους μη-προγραμματιστές να εργαστούν μαζί της.

Η JavaScript έχει γίνει μια από τις δημοφιλέστερες γλώσσες προγραμματισμού στον Ιστό. Αρχικά, εντούτοις, πολλοί επαγγελματίες προγραμματιστές δυσφήμησαν τη γλώσσα επειδή απευθυνόταν σε προγραμματιστές ιστοσελίδων. Η εμφάνιση του AJAX(asynchronous JavaScript and XML) ξανάβαλε την JavaScript στο επίκεντρο και έφερε περισσότερη προσοχή από επαγγελματίες. Το αποτέλεσμα ήταν ένας πολλαπλασιασμός των περιεκτικών frameworks και των βιβλιοθηκών, βελτίωση του πρακτικού προγραμματισμού της JavaScript, και αύξηση της χρήσης της JavaScript έξω από τον browser, όπως φαίνεται από τον πολλαπλασιασμό των server-side πλατφορμών JavaScript.

1.7 Επίλογος

Σε αυτό το κεφάλαιο είδαμε την βασική αρχιτεκτονική των διαδικτυακών εφαρμογών, μαζί με όλες τις τεχνολογίες και γλώσσες προγραμματισμού που χρησιμοποιήθηκαν έτσι ώστε να κατασκευαστούν τα βοηθητικά εργαλεία της εφαρμογής PaperReview. Στο επόμενο κεφάλαιο θα δούμε το πρώτο από αυτά, τον αλγόριθμο αυτόματης ανάθεσης εργασιών στους χρήστες της εφαρμογής.

ΚΕΦΑΛΑΙΟ 2: ΑΛΓΟΡΙΘΜΟΣ ΑΥΤΟΜΑΤΗΣ ΑΝΑΘΕΣΗΣ ΔΙΑΤΡΙΒΩΝ ΣΤΟΥΣ ΑΞΙΟΛΟΓΗΤΕΣ ΤΟΥ ΣΥΝΕΔΡΙΟΥ

2.1 Εισαγωγή	19
2.2 Ο σκοπός του αλγορίθμου	19
2.3 Αλλαγές στην εφαρμογή PaperReview	19
2.4 Υλοποίηση του αλγορίθμου	20
2.4.1 Αρχικές ενέργειες συγκέντρωσης απαραίτητων δεδομένων...23	
2.4.2 Συγκέντρωση διαθέσιμων θέσεων για ανάθεση.....24	
2.4.3 Βελτιστοποίηση αναθέσεων.....25	
2.5 Αποτελέσματα.....	27
2.6 Παράδειγμα εφαρμογής αλγορίθμου.....	28

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε το πρώτο από τα εργαλεία που κατασκευάστηκαν για την εφαρμογή PaperReview, τον αλγόριθμο αυτόματης ανάθεσης εργασιών στα μέλη του συνεδρίου. Θα αναλυθεί ο σκοπός του αλγορίθμου, η υλοποίηση και διάφοροι περιορισμοί και τέλος θα δοθεί ο ψευδοκώδικάς του, μαζί με άλλα κομμάτια κώδικα για την πληρέστερη κατανόησή του όπως και ένα αναλυτικό παράδειγμα

2.2 Ο σκοπός του αλγορίθμου

Για κάθε συνέδριο στην εφαρμογή PaperReview, έχουμε ανάθεση *chairman(s)* από τον κεντρικό διαχειριστή του συστήματος(*admin*). Ο κάθε *chairman*, μπορεί να αναθέσει όποιον αριθμό αξιολογητών(*reviewers*) θέλει για το συγκεκριμένο συνέδριο.

Αφού οριστούν οι αξιολογητές(*reviewers*), γίνεται κατάθεση υποψήφιων διατριβών από συγγραφείς(*authors*) μέσω της εφαρμογής. Ο κάθε αξιολογητής είναι υποχρεωμένος να επιλέξει βαθμό ενδιαφέροντος(*interest level*) για την κάθε διατριβή, ανάλογα βέβαια με τον τομέα που ειδικεύεται. Ο *chairman* επίσης επιλέγει τον αριθμό των αξιολογητών που πρέπει να ανατεθούν σε κάθε ξεχωριστή διατριβή.

Υπάρχουν δύο επιλογές χειροκίνητης ανάθεσης των εργασιών στην εφαρμογή, έτσι ώστε αν ο *chairman* θέλει να αναθέσει συγκεκριμένες διατριβές σε συγκεκριμένους αξιολογητές να έχει την δυνατότητα.

Βέβαια, για μεγάλο όγκο αξιολογητών και διατριβών η χειροκίνητη ανάθεση όλων των διατριβών στους αξιολογητές θα ήταν μια εξαιρετικά χρονοβόρα διαδικασία. Έτσι δημιουργήθηκε η ανάγκη για μια επιλογή αυτόματης ανάθεσης των διατριβών, φυσικά με όσο πιο γίνεται καλύτερα αποτελέσματα, ο σκοπός δηλαδή του αλγορίθμου μας.

2.3 Αλλαγές στην εφαρμογή PaperReview

Παρακάτω ακολουθεί αλλαγμένη η διεπαφή της εφαρμογής για την επιλογή "**Assign Papers**". Οι πρώτες δυο προϋπήρχαν και έχουν προστεθεί οι εξής επιλογές:

1. **"Assign reviewers to papers automatically"** : Εδώ πατώντας, τρέχουμε τον αλγόριθμο αυτόματης ανάθεσης.
2. **"Clear Automatic Assignments"** : Εδώ πατώντας, μετά από confirmation pop-up, διαγράφονται όλες οι αναθέσεις που έχει κάνει ο αλγόριθμος. Έτσι μπορούμε να κάνουμε αλλαγές και να ξανατρέξουμε τον αλγόριθμο.
3. **"Clear All Assignments"** : Εδώ πατώντας, μετά από confirmation pop-up, διαγράφονται όλες οι αναθέσεις εργασιών, αυτόματες και χειροκίνητες.



Εικόνα 2.1: Αλλαγές στην Εφαρμογή, στο μενού "Assign Papers"

2.4 Υλοποίηση του αλγορίθμου

Ο αλγόριθμος αυτόματης ανάθεσης είναι γραμμένος εξ' ολοκλήρου σε PHP, συμπεριλαμβανομένων μερικών SQL queries για την ανάκτηση δεδομένων από την βάση δεδομένων της εφαρμογής, όπως επίσης και για την εγγραφή των αποτελεσμάτων στην βάση. Στο τέλος ο αλγόριθμος εξάγει αναφορικά τα αποτελέσματα στην εφαρμογή σε HTML πίνακα.

Αρχικά, ας δούμε ποιοι πίνακες από την βάση δεδομένων θα χρησιμοποιηθούν.

Interest

Ο πίνακας interest είναι υπεύθυνος κυρίως για την αποθήκευση του βαθμού επιθυμίας αξιολόγησης των αξιολογητών για τις διατριβές που υποβλήθηκαν σε ένα συγκεκριμένο συνέδριο. Αυτό εκφράζεται με ένα βαθμό από το 1 έως το 7. Επίσης εδώ δηλώνεται αν υπάρχει κάποια σύγκρουση (conflict) του αξιολογητή με τους authors, ή αν υπάρχει σύγκρουση δηλωμένη από τον author προς κάποιον αξιολογητή. Τα πεδία αναλυτικά είναι:

- Ο κωδικός του χρήστη (user_id)
- Ο κωδικός της διατριβής στον πίνακα paper(paper_id)
- Ο κωδικός του συνεδρίου(conflict_id)
- Ο βαθμός προτίμησης του αξιολογητή για την διατριβή(level_of_interest)
- Η σύγκρουση από πλευράς αξιολογητή προς τον author της διατριβής(conflict)
- Η σύγκρουση του συγγραφέα προς τον αξιολογητή(conflict_by_author)

Το κύριο κλειδί του πίνακα είναι σύνθετο, και αποτελείται από τα πεδία user_id και paper_id, τα οποία με την σειρά τους είναι ξένα κλειδιά στους πίνακες user και paper, αντίστοιχα. Ο πίνακας επίσης περιέχει ακόμη ένα ξένο κλειδί, το conference_id του πίνακα conference.

Πίνακας 2.1: Ο πίνακας Interest

Πεδίο	Τύπος	Κενό	Προκαθορισμένο
user_id	Int(11)	Όχι	
paper_id	Int(11)	Όχι	
conference_id	Int(11)	Όχι	
level_of_interest	Varchar(0)	Ναι	NULL
conflict	Tinyint(1)	Ναι	NULL
conflict_by_author	Tinyint(1)	Ναι	NULL

Paper

Ο πίνακας Paper περιλαμβάνει όλα τα στοιχεία που περιγράφουν μια διατριβή που υποβλήθηκε στο σύστημα. Ορίζεται από τα εξής πεδία:

- Τον κωδικό της διατριβής(id)
- Τον κωδικό του χρήστη που εισάγει την διατριβή(user_id)
- Τον κωδικό του συνεδρίου στο οποίο ανήκει(conference_id)
- Τον τίτλο της(title)
- Μια σύντομη περιγραφή της(abstract)
- Τα ονόματα των συγγραφέων της(authors)
- Το επίπεδο αποδοχής της στο συνέδριο(status_code)

- Το θέμα της(subject)
- Την ημερομηνία υποβολής της στο σύστημα(submission_date)

Κύριο κλειδί του πίνακα είναι το πεδίο id, το οποίο παίρνει αυτόματα ως τιμή τον αύξοντα αριθμό της εγγραφής, και αποτελεί τον κωδικό της κάθε επιστημονικής διατριβής.

Ο πίνακας περιλαμβάνει δύο ξένα κλειδιά. Το πεδίο conference_id, ξένο κλειδί στον πίνακα conference, και το πεδίο user_id, ξένο κλειδί στον πίνακα user.

Πίνακας 2.2: Ο πίνακας paper

Πεδίο	Τύπος	Κενό	Προκαθορισμένο	Πρόσθετα
id	int(11)	Όχι		auto_increment
user_id	int(11)	Όχι		
conference_id	int(11)	Όχι		
title	varchar(100)	Όχι		
abstract	text	Όχι		
authors	text	Όχι		
status_code	int(11)	Ναι	NULL	
subject	varchar(100)	Ναι	NULL	
submission_date	datetime	Όχι	0000-00-00 00:00:00	

PaperToReviewer

Είναι ο πίνακας διασύνδεσης μεταξύ των πινάκων paper και user, όπου καθορίζονται οι διατριβές που έχουν ανατεθεί για αξιολόγηση σε κάθε μέλος της επιτροπής αξιολόγησης. Κάθε εγγραφή αποτελείται από:

- Τον κωδικό της διατριβής(paper_id)
- Τον κωδικό αξιολογητή(user_id)
- Τον κωδικό του συνεδρίου στον οποίο έγινε η ανάθεση(conference_id)

Το κύριο κλειδί του πίνακα είναι σύνθετο, και αποτελείται από τα πεδία paper_id και user_id. Τα πεδία paper_id, user_id και conference_id είναι με τη σειρά τους ξένα κλειδιά στους πίνακες paper, user και conference αντίστοιχα.

Για την σωστή λειτουργία του αλγορίθμου, έπρεπε να τροποποιηθεί ο συγκεκριμένος πίνακας, με την προσθήκη ενός ακόμη πεδίου του auto_assignment. Το πεδίο auto_assignment είναι υπεύθυνο για το αν μια διασύνδεση διατριβής-αξιολογητή έχει γίνει χειροκίνητα από κάποιον chairman, ή αν η διασύνδεση έχει γίνει αυτόματα από τον αλγόριθμο. Οι τιμές που παίρνει είναι 0 ή 1, για χειροκίνητη ή αυτόματη ανάθεση αντίστοιχα.

Πίνακας 2.3: Ο πίνακας PaperToReviewer

Πεδίο	Τύπος	Κενό	Προκαθορισμένο
paper_id	int(11)	Όχι	
user_id	int(11)	Όχι	
conference_id	int(11)	Όχι	
auto_assignment	tinyint(1)	Όχι	0

Options

Ο πίνακας options χρησιμοποιείται για την οριοθέτηση κάθε σταδίου στο χρονοδιάγραμμα λειτουργιών του συνεδρίου. Η μόνη τιμή του πίνακα που μας απασχολεί είναι η **NORPC: Number Of Reviewers for each Paper in Conference**, όπου μας δίνει τον μέγιστο αριθμό αξιολογητών για κάθε διατριβή στο συνέδριο.

2.4.1 Αρχικές ενέργειες συγκέντρωσης απαραίτητων δεδομένων

Για την υλοποίηση του αλγορίθμου, θα χρειαστούμε αρκετούς βοηθητικούς πίνακες που θα αποθηκεύουν δεδομένα καταλυτικής σημασίας για την πορεία του αλγορίθμου.

Πρώτος απαραίτητος πίνακας είναι ο πίνακας **PaperReviewerStatus**, ο οποίος είναι δισδιάστατος. Η μια διάσταση είναι οι κωδικοί αριθμοί των διατριβών(*paper_id*), που αντιστοιχούν στο τρέχον συνέδριο, τους οποίους ανακτούμε με ένα απλό ερώτημα στην βάση δεδομένων. Η άλλη διάσταση είναι οι κωδικοί αριθμοί των χρηστών(*user_id*), που έχουν οριστεί ως αξιολογητές για το τρέχον συνέδριο, τους οποίους ανακτούμε επίσης με ένα απλό ερώτημα στην βάση δεδομένων. Τα δεδομένα αυτά προέρχονται από τον πίνακα **Interest**.

Το κάθε κελί του πίνακα **PaperReviewerStatus**, μπορεί να πάρει τρεις διαφορετικές τιμές, τις οποίες ανακτούμε από την βάση δεδομένων μας. Συγκεκριμένα από τον πίνακα **Interest** βλέπουμε αν ο αξιολογητής έχει η δεν έχει δηλωμένη σύγκρουση(*conflict*) και από τον πίνακα **PaperToReviewer** για το αν έχει γίνει ήδη χειροκίνητη ανάθεση. Αναλυτικά:

- Τιμή ένα (1) που σημαίνει ότι η συγκεκριμένη διατριβή είναι δυνατόν να αξιολογηθεί από τον συγκεκριμένο αξιολογητή
- Τιμή μηδέν (0) που σημαίνει ότι η συγκεκριμένη διατριβή έχει σύγκρουση με τον συγκεκριμένο αξιολογητή

- Τιμή μείον ένα (-1) που σημαίνει ότι η συγκεκριμένη διατριβή είναι ήδη χειροκίνητα συνδεδεμένη με τον συγκεκριμένο αξιολογητή

Δεύτερος πίνακας που χρειαζόμαστε είναι ο **ReviewerAssignmentLimit**, ο οποίος είναι μονοδιάστατος. Η ετικέτα του κάθε κελιού είναι ο κωδικός του αξιολογητή(*reviewer_id*), και το κελί περιέχει τον αριθμό χειροκίνητων αναθέσεων διατριβών που του έχουν γίνει.

Τρίτος απαραίτητος πίνακας είναι ο πίνακας **PaperReviewerInterest**, ο οποίος είναι και αυτός δισδιάστατος. Η μια διάσταση είναι οι κωδικοί αριθμοί των διατριβών(*paper_id*), που αντιστοιχούν στο τρέχον συνέδριο, τους οποίους ανακτούμε με ένα απλό ερώτημα στην βάση δεδομένων. Η άλλη διάσταση είναι οι κωδικοί αριθμοί των χρηστών(*user_id*), που έχουν οριστεί ως αξιολογητές για το τρέχον συνέδριο, τους οποίους ανακτούμε επίσης με ένα απλό ερώτημα στην βάση δεδομένων.

Το κάθε κελί του πίνακα **PaperReviewerInterest**, παίρνει την τιμή που έχει δώσει ο κάθε αξιολογητής για κάθε διατριβή που υπάρχει στο συνέδριο. Οι τιμές ανακτώνται κάνοντας ένα ερώτημα στην βάση δεδομένων αναζητώντας την τιμή **level_of_interest** από τον πίνακα **Interest**. Οι τιμές κυμαίνονται από 1 έως 7, όπως έχει οριστεί από τον αρχικό σχεδιασμό της βάσης.

2.4.2 Συγκέντρωση διαθέσιμων θέσεων για ανάθεση

Σε αυτό το στάδιο πρέπει να ελέγξουμε το πόσες αναθέσεις πρέπει να γίνουν για την κάθε διατριβή, σύμφωνα με το όριο που έχει ορίσει ο διαχειριστής του συνεδρίου, το οποίο ανακτούμε με ένα απλό ερώτημα στην βάση δεδομένων, ζητώντας το πεδίο **NORPC** από τον πίνακα **Options**.

Αρχικά πρέπει να υπολογίσουμε το πόσες αξιολογήσεις πρέπει να κάνει ο κάθε αξιολογητής. Αρκεί να ακολουθήσουμε τον τύπο:

$$ReviewsPerReviewer = \left\lfloor \frac{NumberOfPapers * NORPC}{NumberOfReviewers} \right\rfloor$$

Στην συνέχεια, για κάθε διατριβή, υπολογίζεται πόσες χειροκίνητες αναθέσεις της έχουν γίνει, έτσι ώστε να υπολογίσουμε πόσες αυτόματες αναθέσεις πρέπει να κάνει ο αλγόριθμος για την συγκεκριμένη διατριβή ακόμη. Σύμφωνα με αυτόν τον αριθμό, θα δημιουργηθεί μια λίστα με υποψήφιους τυχαίους αξιολογητές για κάθε διατριβή.

Οι αξιολογητές αναθέτονται τυχαία σε αυτό το στάδιο, ως εξής: Χρησιμοποιούμε τον πίνακα **ReviewerAssignmentLimit**, ο οποίος ταξινομείται με αύξουσα σειρά. Έτσι πάντα πρώτος στον παραπάνω πίνακα είναι αυτός που έχει τις λιγότερες αξιολογήσεις. Στην συνέχεια, για κάθε διατριβή, ελέγχουμε πόσες ακόμη αναθέσεις πρέπει να γίνουν, και για κάθε θέση ανάθεσης ελέγχουμε αν ο πρώτος από τους αξιολογητές στον πίνακα **ReviewerAssignmentLimit**, είναι ικανός να αξιολογήσει την διατριβή (από τον πίνακα **PaperReviewerStatus**, ελέγχεται η τιμή ένα(1) για το τρέχον ζευγάρι αξιολογητή-διατριβής). Σε περίπτωση ανικανότητας, ελέγχεται ο επόμενος, μέχρις ότου να βρεθεί κάποιος και αφού βρεθεί, ο αριθμός των αναθέσεων του συγκεκριμένου αξιολογητή αυξάνεται κατά ένα, ο πίνακας **ReviewerAssignmentLimit** ταξινομείται και πάλι και προχωράμε στην επόμενη κενή θέση ανάθεσης.

Με αυτόν τον τρόπο καταφέρνουμε μια ομοιόμορφη κατανομή των διατριβών στους αξιολογητές. Σε μερικές περιπτώσεις, όταν έχουμε μεγάλο αριθμό αναθέσεων για κάθε διατριβή (*NORPC*) και πολλές συγκρούσεις(*conflicts*), κάποιοι αξιολογητές αναμένεται να πάρουν λιγότερες ή περισσότερες αξιολογήσεις από τους άλλους. Το εύρος της διαφοράς τους είναι **ReviewsPerReviewer \pm 1**.

Έτσι λοιπόν θα δημιουργηθεί ο πίνακας **CleanAssignment**, ο οποίος θα περιέχει σε κάθε κελί του, του οποίου ετικέτα είναι ο κωδικός αριθμός της διατριβής(*paper_id*), μια λίστα με τους τυχαίους υποψήφιους αξιολογητές της συγκεκριμένης διατριβής.

2.4.3 Βελτιστοποίηση αναθέσεων

Αυτό είναι το τελευταίο στάδιο του αλγορίθμου. Εδώ γίνεται η προσπάθεια για την καλύτερη ανάθεση διατριβών στους αξιολογητές σύμφωνα με τις προτιμήσεις τους. Παίρνουμε ζευγάρια αξιολογητών-διατριβών και ελέγχουμε αν αντιμετωπίσουμε τις διατριβές για τους συγκεκριμένους αξιολογητές αν θα έχουμε καλύτερο άθροισμα των τιμών ενδιαφέροντος(τιμές από τον πίνακα **PaperReviewerInterest**) που έχουν δηλώσει οι αξιολογητές. Προϋπόθεση για αυτό το σημείο είναι ότι όλοι οι αξιολογητές έχουν δηλώσει προτιμήσεις για τις διατριβές του συνεδρίου. Σε περίπτωση που δεν έχει γίνει κάτι τέτοιο, ορίζεται η default τιμή 1. Ακολουθεί ψευδοκώδικας για το τελευταίο στάδιο του αλγορίθμου.

Έστω τα σύνολα:

- R το σύνολο των reviewer

- P το σύνολο των paper

Έστω οι πίνακες:

- $RI[i,j]$ που περιέχει το βαθμό του ενδιαφέροντος του reviewer i για το paper j
- $RA[i,j]$ που περιέχει το αν ο reviewer i είναι ικανός να πάρει το paper j
- $CA[i,j]$ που περιέχει τις τυχαίες αναθέσεις που έγιναν για το paper i στην λίστα j

```
Repeat until no swap is made {
  Swapcounter=0;
  Swapflag=false;
  Foreach paper as outerPaper{
    Foreach paper as innerPaper {
      If (innerPaper==outerPaper)
        continue;

      Foreach user as outerUser in CA[outerpaper, j] {
        Foreach user as innerUser in CA[innerpaper, j] {
          If (outerUser==innerUser)
            continue;

          If (innerUser is already assigned for outerPaper OR outerUser is already
assigned for innerPaper)
            continue;

          If ( $RI[innerUser, innerPaper] + RI[outerUser, outerPaper] <$ 
 $RI[innerUser, outerPaper] + RI[outerUser, innerPaper]$  and check if innerUser is able to
review outerPaper and check if outerUser is able to review innerPaper {
            Exchange reviewers for current outerPaper and innerPaper;
            Flag=true;
            Swapcounter++;
            Go to next user;
          }
        }
      }
      if(flag==true)
        Go to next paper;
    }
  }
  if (swapcounter==0)
    Break;
}
```

Εικόνα 2.2: Ψευδοκώδικας για τον αλγόριθμο αυτόματης ανάθεσης

Βλέπουμε ότι ο αλγόριθμος δεν θα μπει σε ατέρμον βρόγχο, μιας και η ανισότητα στην συνθήκη ελέγχου ανταλλαγής αξιολογητών θα μας δίνει μόνιμα false όταν ο αλγόριθμος φτάσει στο μέγιστο άθροισμα που μπορεί να δώσει

ανταλλάσσοντας ζεύγη αξιολογητών-διατριβών. Έτσι το *swarcouter* θα πάρει την τιμή 0 και θα τερματίσει ο αλγόριθμος.

2.5 Αποτελέσματα

Αφού τελειώσει η βελτιστοποίηση ο αλγόριθμος τυπώνει τα αποτελέσματα σε έναν πίνακα HTML σε σελίδα της εφαρμογής μας. Χρειαζόμαστε πολλά πεδία από την βάση δεδομένων. Ονόματα συγγραφέων, τίτλος διατριβής, ονόματα των αξιολογητών, το επίπεδο προτίμησης και τον συνολικό αριθμό διατριβών που έχουν ανατεθεί, τα οποία εκτυπώνονται με το κατάλληλο ερώτημα. Επίσης πρέπει να φαίνεται ποια ανάθεση είναι αυτόματη και ποια χειροκίνητη. Ακολουθεί ένα δείγμα των αποτελεσμάτων του αλγορίθμου.

AUTOMATIC ASSIGNMENT RESULTS

Below are the results of the automatic assignment. Note that rows in teal are automatic assignments, and in grey are manual assignments made previously by the chairman.

AUTHORS	PAPER TITLE	REVIEWERS, INTEREST LEVEL AND NUMBER OF REVIEWS
James Doe,	Databases	Adam Strange 7 4
		Matt Fraction 5 4
		Brian Wood 7 4
Maden Madenis,	Databases And Web	Adam Strange 7 4
		Matt Fraction 3 4
		Brian Wood 7 4
Matt Fraction,	Web & Distributed Systems	Scott Pilgrim 1 4
		Logan Howlett 1 5
		Brian Wood 5 4
Brian Lee O'malley,	Operating Systems	John Madenis 7 4
		Matt Murdock 4 4
		Giannis Madenis 7 4

Εικόνα 2.3: Δείγμα μορφής εκτύπωσης αποτελεσμάτων αλγορίθμου

2.6 Παράδειγμα Εφαρμογής Αλγορίθμου

Έστω ότι έχουμε ένα συνέδριο, το οποίο έχει 14 διατριβές για αξιολόγηση (*papers*), 10 αξιολογητές (*reviewers*), και 3 αξιολογητές ανά διατριβή. Ο βαθμός ενδιαφέροντος κάθε αξιολογητή για κάθε διατριβή φαίνεται στον παρακάτω πίνακα:

Πίνακας 2.4: Πίνακας αξιολογητών/διατριβών και βαθμού ενδιαφέροντος

	john	adam	matt	david	scott	paul	logan	brian	Ed	bob
Databases	X	7	5	1	1	1	1	7	1	1
Databases and web	X	7	3	4	1	1	1	7	1	1
Web & distributed systems	5	1	X	1	1	1	1	5	1	1
Operating systems	7	1	1	1	4	4	1	1	1	7
Operating systems & network	7	1	1	7	X	3	5	1	1	7
Informatics	3	1	4	2	1	X	3	4	3	2
Software Engineering	1	1	1	1	6	6	X	1	1	1
Software Updating	1	1	1	1	7	7	X	1	1	1
Software Revolution	1	1	1	1	7	7	X	1	3	5
Wireless networks	1	1	1	4	1	1	6	1	2	1
Network Evolution	1	1	1	7	1	1	7	1	1	1
Hardware and limits	1	3	7	1	1	1	1	1	7	1
Future Hardware	1	3	7	1	1	1	1	1	7	1
Wi-fi evolution	1	1	1	4	1	1	6	2	X	1

Βλέπουμε ότι κάθε αξιολογητής έδωσε μεγαλύτερο βαθμό ενδιαφέροντος ανάλογα με το πεδίο που ενδιαφέρεται-ειδικεύεται. Στην περίπτωση που κάποιος αξιολογητής είναι και συγγραφέας διατριβής, έχουμε στο κελί την τιμή **X**.

Ακόμη, για το παράδειγμά μας, ο chairman έχει δηλώσει σύγκρουση του reviewer "**John**" με το paper *Web & distributed systems* (δηλαδή δεν μπορεί να του γίνει ανάθεση). Επίσης έχουν γίνει κάποιες χειροκίνητες αναθέσεις διατριβών σε αξιολογητές. Ο αξιολογητής "**Paul**" έχει αναταθεί να αξιολογήσει τις διατριβές: "*Software Engineering*", "*Software Updating*", "*operating Systems*" και "*operating Systems & network*". Ο αξιολογητής "**Brian**" έχει αναταθεί να αξιολογήσει τις διατριβές "*Databases*" και "*Wi-fi evolution*", και τέλος ο "**Matt**" την "*Databases and web*".

Έτσι, ο πίνακας με τις τυχαίες αναθέσεις διατριβών σε αξιολογητές θα έχει ως εξής:

Πίνακας 2.5: Πίνακας με τυχαίες αναθέσεις για κάθε διατριβή

Databases	Adam(7)	Matt(5)	
Databases and web	Logan(1)	Ed(1)	
Web & distributed systems	David(1)	Scott(1)	Bob(1)
Operating systems	John(7)	Adam(1)	
Operating systems & network	John(7)	Bob(7)	
Informatics	Logan(3)	Ed(3)	David(2)
Software Engineering	Scott(6)	Brian(1)	
Software Updating	Ed(1)	David(1)	
Software Revolution	John(1)	Bob(5)	Scott(7)
Wireless networks	Matt(1)	Brian(1)	Bob(1)
Network Evolution	Adam(1)	David(7)	Logan(7)
Hardware and limits	Matt(7)	Scott(1)	Ed(7)
Future Hardware	John(1)	David(1)	Logan(1)
Wi-fi evolution	Logan(6)	Adam(1)	

Στην παρένθεση βλέπουμε τον βαθμό ενδιαφέροντος του κάθε αξιολογητή για τη συγκεκριμένη διατριβή. Βλέπουμε ότι ο "Paul" δεν έχει καμιά τυχαία ανάθεση, αφού του έγιναν 4 από τον chairman.

Τέλος, έχουμε τον πίνακα μετά από την εφαρμογή του αλγορίθμου και τις αντιμεταθέσεις διατριβών-αξιολογητών που έκανε.

Πίνακας 2.6: Τελικός Πίνακας αναθέσεων

Databases	Adam(7)	Matt(5)	Brian(7)
Databases and web	Adam(7)	Brian(7)	Matt(3)
Web & distributed systems	Logan(1)	Scott(1)	Brian(5)
Operating systems	John(7)	Bob(7)	Paul(4)
Operating systems & network	David(7)	Bob(7)	Paul(3)
Informatics	Logan(3)	Ed(3)	John(3)
Software Engineering	Scott(6)	John(1)	Paul(6)
Software Updating	Scott(7)	David(1)	Paul(7)
Software Revolution	Ed(3)	Bob(5)	Scott(7)
Wireless networks	David(4)	Logan(6)	Bob(1)
Network Evolution	John(1)	David(7)	Logan(7)
Hardware and limits	Matt(7)	Scott(1)	Ed(7)
Future Hardware	Adam(3)	Matt(7)	Ed(7)
Wi-fi evolution	Logan(6)	David(4)	Brian(2)

Εδώ βλέπουμε τα αποτελέσματα του αλγορίθμου. Τα κελιά με γκρι φόντο είναι οι χειροκίνητες αναθέσεις από τον chairman σε αξιολογητές. Ακόμη βλέπουμε ότι οι αναθέσεις σε κάθε αξιολογητή κυμαίνονται από 4 έως 5.

Πίνακας 2.7: Πίνακας συνολικών αναθέσεων σε αξιολογητές.

john	adam	matt	david	scott	paul	logan	brian	Ed	bob
4	4	4	5	4	4	5	4	4	4

ΚΕΦΑΛΑΙΟ 3: ΕΡΓΑΛΕΙΟ ΑΠΟΣΤΟΛΗΣ MAIL TEMPLATE

3.1 Εισαγωγή	32
3.2 Γενική δομή ενός mail template	32
3.2.1 Στοιχείο <smtpconfig>	33
3.2.2 Στοιχείο <mysqlconfig>	33
3.2.3 Στοιχείο <query>	34
3.2.4 Στοιχείο <email>	35
3.3 Ορισμός Σκελετού Template.....	37
3.4 Ανάλυση του script αποστολής mail template.....	39
3.4.1 Έλεγχοι ορθής εκτέλεσης του script.....	39
3.4.2 Εντολές επεξεργασίας του xml template.....	40
3.4.3 Διαδικασία αποστολής των email.....	41

3.1 Εισαγωγή

Στο κεφάλαιο αυτό θα δούμε το αναλυτικά το script αποστολής mail template, καθώς και αναλυτικά τον σκελετό και την δομή του προτύπου mail template. Το script είναι γραμμένο σε PHP, και είναι σχεδιασμένο να τρέχει από command line σε Apache server ενώ το mail template είναι γραμμένο σε XML. Φυσικά δεν απουσιάζει η ύπαρξη μιας βάσεως δεδομένων και εντολών SQL για συλλογή δεδομένων. Στο επόμενο κεφάλαιο θα γίνει η ενσωμάτωσή του στην εφαρμογή *PaperReview*.

3.2 Γενική δομή ενός mail template

Για να κατανοήσουμε καλύτερα το script, που θα αναλυθεί αργότερα, σε αυτό το υποκεφάλαιο θα ακολουθήσει η ανάλυση της γενικής δομής ενός mail template. Ποια πεδία υπάρχουν, ποια είναι ιδιότητες (*attributes*), ποια είναι στοιχεία (*elements*) και τι περιέχει το καθένα από αυτά.

Κάθε mail template αρχίζει με την δήλωση της έκδοσης (*version*) και της κωδικοποίησης (*encoding*) των χαρακτήρων του. Στην εφαρμογή μας χρησιμοποιείται η έκδοση 1.0 και για κωδικοποίηση χαρακτήρων το σύνολο utf-8.

Αρχικό στοιχείο (*element*) κάθε template είναι το **<template>**, το οποίο εσωκλείει μέσα του όλα τα υπόλοιπα στοιχεία ενός mail template. Παιδιά του είναι τα στοιχεία **<smtpconfig>**, **<mysqlconfig>**, και **<query>**.

```
<?xml version="1.0" encoding="utf-8"?>
<template>
  <smtpconfig>
    [...]
  </smtpconfig>
  <mysqlconfig>
    [...]
  </mysqlconfig>
  <query>
    [...]
  </query>
</template>
```

Εικόνα 3.1: Βασικά στοιχεία του mail template

Ακολουθεί ανάλυση κάθε στοιχείου-παιδιού του **<template>**.

3.2.1 Στοιχείο <smtpconfig>

Το στοιχείο <smtpconfig> έχει τέσσερα στοιχεία-παιδιά. Είναι τα εξής:

1. **<smtpserver>** : αυτό το στοιχείο έχει σαν περιεχόμενο την διεύθυνση του SMTP Server που επιθυμεί να χρησιμοποιήσει ο chairman για να αποστείλει τα email στους συμμετέχοντες του συνεδρίου.
2. **<smtpusername>** : αυτό το στοιχείο έχει σαν περιεχόμενο το username που έχει ο chairman στο δηλωμένο SMTP Server του.
3. **<smtppassword>** : αυτό το στοιχείο περιέχει το password του chairman για τον δηλωμένο SMTP Server του.
4. **<smtpport>** : αυτό το στοιχείο περιέχει τον αριθμό της θύρας που χρησιμοποιεί ο SMTP Server για να αποστείλει τα emails.

```
<smtpconfig>
  <smtpserver> ssl://mysmtpserver.com /smtpserver>
  <smtpusername> smtpuser </smtpusername>
  <smtppassword> smtppass </smtppassword>
  <smtpport> 25 </smtpport>
</smtpconfig>
```

Εικόνα 3.2: Στοιχεία-παιδιά του πεδίου <smtpconfig>

Τα πεδία αυτά συμπληρώνονται από τον εκάστοτε δημιουργό του mail template σε περίπτωση που θέλει να χρησιμοποιήσει κάποιον SMTP Server για την αποστολή των μηνυμάτων. Στην περίπτωση που μείνουν κενά χρησιμοποιείται ο server στον οποίο τρέχει το script.

3.2.2 Στοιχείο <mysqlconfig>

Το στοιχείο <mysqlconfig> έχει τέσσερα στοιχεία-παιδιά. Είναι τα εξής:

1. **<mysqlserver>**: αυτό το στοιχείο περιέχει την διεύθυνση του MySQL Server που είναι αποθηκευμένα τα δεδομένα της εφαρμογής του συνεδρίου.
2. **<mysqluser>**: αυτό το στοιχείο περιέχει το όνομα χρήστη που χρησιμοποιείται για την σύνδεση με τον εξυπηρετητή που έχει την βάση δεδομένων του συνεδρίου.
3. **<mysqlpassword>**: αυτό το στοιχείο περιέχει το συνθηματικό για την σύνδεση με τον εξυπηρετητή που έχει την βάση δεδομένων του συνεδρίου.
4. **<mysqladb>**: αυτό το στοιχείο περιέχει το όνομα της βάσεως δεδομένων στον εξυπηρετητή που συνδέεται η εφαρμογή μας.

```

<mysqlconfig>
  <mysqlserver> localhost </mysqlserver>
  <mysqluser> sqluser </mysqluser>
  <mysqlpassword> sqlpassword </mysqlpassword>
  <mysqldb> sqldatabase </mysqldb>
</mysqlconfig>

```

Εικόνα 3.3: Στοιχεία-παιδιά του πεδίου <mysqlconfig>

Τα παραπάνω στοιχεία συμπληρώνονται από τον εκάστοτε δημιουργό του mail template και χρησιμοποιούνται από το script για την σύνδεση και συλλογή δεδομένων από την επιθυμητή βάση δεδομένων.

3.2.3 Στοιχείο <query>

Το στοιχείο <query> έχει δυο ιδιότητες, την "SQL" και την "NAME" και ένα στοιχείο παιδί, το <email>. Η ιδιότητα SQL περιέχει μέσα της το ερώτημα SQL που θα γίνει στην βάση δεδομένων και είναι διαμορφωμένο ανάλογα με το σε ποιους χρήστες και ποια δεδομένα θα πρέπει να συλλεχθούν για να αποσταλούν σε αυτούς. Η ιδιότητα NAME περιέχει ένα προσδιορισμό του SQL ερωτήματος, που βοηθάει στον διαχωρισμό δεδομένων-sql ερωτήματος. Στο ερώτημα που υπάρχει στο στοιχείο <query>, καλό είναι να δίνεται ο χαρακτηρισμός "main".

```

<query SQL="SELECT name, email, address FROM user WHERE type = 'reviewer'"
  NAME="main">
  <email >
  [...]
  </email>
</query>

```

Εικόνα 3.4: Παράδειγμα ιδιοτήτων, παιδιών του στοιχείου <query>

Το παραπάνω sql ερώτημα θα μας δώσει ένα σύνολο από μεταβλητές που μπορούν να χρησιμοποιηθούν μέσα στο κείμενο του πεδίου <email> που θα δούμε στην συνέχεια. Ανάλογα με το ποιιά πεδία ζητάμε από την βάση στο ερώτημα, τόσες διαφορετικές μεταβλητές μπορούμε να χρησιμοποιήσουμε μέσα στο κείμενο, στις ιδιότητες του <email> και στο θέμα του μηνύματος. Στην προκειμένη περίπτωση θα έχουμε τις μεταβλητές **#[main.name]**, **#[main.email]**, **#[main.address]**.

3.2.4 Στοιχείο <email>

Το στοιχείο <email> είναι αυτό που περιέχει όλες τις πληροφορίες για την αποστολή των μηνυμάτων. Έχει έξι ιδιότητες, τις "TO", "BCC", "FROM", "REPLYTO", "RETURN-PATH" και "SUBJECT". Η ιδιότητα "TO" μας λέει σε ποιόν αποστέλλεται το μήνυμα, η "BCC" στέλνει ένα ακριβές αντίγραφο του μηνύματος στην διεύθυνση που περιέχει, η "FROM" περιέχει τον αποστολέα του μηνύματος, η "REPLYTO" μια διεύθυνση που ο παραλήπτης μπορεί να απαντήσει, η "RETURN-PATH" περιέχει την διεύθυνση που θα γυρίσει το μήνυμα αν δεν μπορεί να αποσταλεί στην διεύθυνση παραλήπτη και η "SUBJECT" που περιέχει το θέμα του μηνύματος.

Πέρα από τις ιδιότητες του στοιχείου <email>, έχει σαν περιεχόμενο το κείμενο αποστολής του μηνύματος που προορίζεται για τους χρήστες, και προαιρετικά έχει στοιχεία-παιδιά, ονομαζόμενα <subquery>. Στην περίπτωση που υπάρχουν **subqueries**, μέσα στο κείμενο του <email> υπάρχουν κάποια *text-tags*, όπως **\$Sub1Start**, **\$Sub1Text**, **\$Sub1End**, τα οποία κατατοπίζουν το script που υπάρχει το κείμενο του <subquery> και που θα γίνουν οι αλλαγές μεταβλητών του <subquery> μέσα στο γενικό κείμενο του <email>.

Τα στοιχεία <subquery> με την σειρά τους έχουν δυο ιδιότητες, την "SQL" και την "NAME". Η ιδιότητα **SQL** περιέχει ένα **SQL** ερώτημα το οποίο συνήθως στοχεύει στον τρέχον παραλήπτη του μηνύματος που παίρνουμε από το κεντρικό ερώτημά μας (από το **SQL** του <query> δηλαδή), με σκοπό να πάρει περισσότερες πληροφορίες για το γενικό θέμα του μηνύματος. Η ιδιότητα **NAME** περιέχει ένα προσδιορισμό για το **SQL** ερώτημα, που στην περίπτωση των **subquery** καλό είναι να ονομάζονται **Sub1**, **Sub2** κ.ο.κ.

Όπως και με το κύριο ερώτημα sql που είχαμε στο στοιχείο <query>, έτσι και εδώ θα δημιουργηθεί ένα σύνολο από μεταβλητές που μπορούν να χρησιμοποιηθούν από τον χρήστη μέσα στο κείμενο. Πρέπει να σημειωθεί ότι οι μεταβλητές που δημιουργούνται από ένα **subquery** έχουν δυνατότητα χρησιμοποίησης μόνο μέσα στα όριά του, που ορίζονται από τα *text-tags* που αναφέρθηκαν παραπάνω.

```
<email TO="${main.email}"
      BCC="blindcarboncopy@someone.com"
      FROM="scriptuser@scriptserver.net"
      REPLYTO="boss@scriptserver.net"
      RETURN-PATH="errormail@scriptserver.net"
      SUBJECT="Hello ${main.name}">

Hello ${main.name}! Your Address is ${main.address}.
</email>
```

Εικόνα 3.5: Παράδειγμα μηνύματος χωρίς subquery

Στο παραπάνω παράδειγμα, όλοι οι χρήστες που επιλέχθηκαν από το κύριο sql ερώτημα θα λάβουν ένα mail με το όνομα τους και την διεύθυνσή τους.

```
<email TO="${main.email}"
  BCC="blindcarboncopy@someone.com"
  FROM="scriptuser@scriptserver.net"
  REPLYTO="boss@scriptserver.net"
  RETURN-PATH="errormail@scriptserver.net"
  SUBJECT="Hello ${main.name}">

  Hello ${main.name}! Your Address is ${main.address}.
  You like these:
  ${Sub1TEXT}
  ${Sub1START}
  ---
  <subquery SQL="SELECT hobbies FROM user_interests WHERE name = ${main.name}"
  NAME="Sub1">
  ${Sub1.hobbies}
  </subquery>
  ---
  ${Sub1END}
  Goodbye!
</email>
```

Εικόνα 3.6: Παράδειγμα μηνύματος με subqueries

Στο παραπάνω παράδειγμα ο εκάστοτε παραλήπτης θα λάβει ένα μήνυμα με τα ενδιαφέροντά του. Π.χ:

```
Subject:"Hello John"
Hello John! Your Address is Fabierou 6.
You like these:
---
Basketball
Football
---
Goodbye!
```

```
Subject:"Hello Mary"
Hello Mary! Your Address is Dimokritou 5.
You like these:
---
Fashion
Modeling
Vacations
---
Goodbye!
```

3.3 Ορισμός Σκελετού Template

Για τον ορισμό του σκελετού του template που μπορεί να επεξεργαστεί το script μας θα μας βοηθήσει ένα αρχείο DTD(Document Type Definition) με το οποίο ορίζουμε τα στοιχεία, τις ιδιότητες και την μορφή που πρέπει να έχει ένα δημιουργημένο template για να επεξεργαστεί σωστά από το script.

```

<!ELEMENT template (smtpconfig, mysqlconfig, query)>

<!ELEMENT smtpconfig (smtpserver, smtpusername, smtppassword, smtpport)>
<!ELEMENT smtpserver      (#PCDATA)>
<!ELEMENT smtpusername    (#PCDATA)>
<!ELEMENT smtppassword    (#PCDATA)>
<!ELEMENT smtpport        (#PCDATA)>

<!ELEMENT mysqlconfig (mysqlserver, mysqluser, mysqlpassword, mysqlpdb)>
<!ELEMENT mysqlserver     (#PCDATA)>
<!ELEMENT mysqluser       (#PCDATA)>
<!ELEMENT mysqlpassword   (#PCDATA)>
<!ELEMENT mysqlpdb        (#PCDATA)>

<!ELEMENT query (email)>

<!ELEMENT email (#PCDATA|subquery)*>

<!ELEMENT subquery (#PCDATA|subquery)*>

<!ATTLIST query SQL CDATA          #REQUIRED>
<!ATTLIST query NAME CDATA         #REQUIRED>

<!ATTLIST email TO CDATA           #REQUIRED>
<!ATTLIST email BCC CDATA          #REQUIRED>
<!ATTLIST email FROM CDATA         #REQUIRED>
<!ATTLIST email SUBJECT CDATA      #REQUIRED>
<!ATTLIST email REPLYTO CDATA      #IMPLIED>
<!ATTLIST email RETURN-PATH CDATA  #REQUIRED>

<!ATTLIST subquery SQL CDATA       #REQUIRED>
<!ATTLIST subquery NAME CDATA      #REQUIRED>

```

Εικόνα 3.7: Το αρχείο DTD που περιέχει την δομή του template που μπορεί να επεξεργαστεί το script.

Εδώ βλέπουμε τον ορισμό του template μας. Βλέπουμε ότι πρέπει να ξεκινάει πάντα με το στοιχείο <template> το οποίο έχει σαν στοιχεία τα <smtpconfig> , <mysqlconfig> και <query>. Τα στοιχεία αυτά με την σειρά τους έχουν άλλα στοιχεία-παιδιά. Για το <smtpconfig> έχουμε τα <smtpserver>, <smtpusername>, <smtppassword>, <smtpport>. Για το <mysqlconfig> έχουμε τα <mysqlserver>, <mysqluser>, <mysqlpassword>, <mysqldb> και για το <query> έχουμε το <email>. Τα στοιχεία των <smtpconfig> και <mysqlconfig> περιέχουν χαρακτήρες (**#PCDATA**), ενώ το <email> περιέχει χαρακτήρες και προορατικά ένα στοιχείο παιδί, το <subquery>. Το subquery με την σειρά του μπορεί να περιέχει χαρακτήρες και προορατικά ένα ακόμη <subquery>.

Το στοιχείο <email> έχει έξι ιδιότητες, TO, BCC, FROM, SUBJECT, REPLYTO (προαιρετικό), RETURN-PATH, ενώ το <query> και το <subquery> έχουν δυο ιδιότητες, την SQL και την NAME.

```
<?xml version="1.0" encoding="utf-8"?>
<template>
  <smtpconfig>
    <smtpserver> </smtpserver>
    <smtpusername> </smtpusername>
    <smtppassword> </smtppassword>
    <smtpport> </smtpport>
  </smtpconfig>
  <mysqlconfig>
    <mysqlserver> </mysqlserver>
    <mysqluser> </mysqluser>
    <mysqlpassword> </mysqlpassword>
    <mysqldb> </mysqldb>
  </mysqlconfig>
  <query SQL="SELECT ..." NAME="main">
    <email TO="someone@nowhere.com"
      BCC="noone@nowhere.com"
      FROM="noreply@nowhere.com"
      REPLYTO=" noone@nowhere.com""
      RETURN-PATH=" noone@nowhere.com""
      SUBJECT="something">
      Blah blah.
    </email>
  </query>
</template>
```

Εικόνα 3.8: Ο σκελετός του template σε XML.

3.4 Ανάλυση του script αποστολής mail template

Σε αυτό το σημείο αναλύεται το script που επεξεργάζεται τα mail templates σταδιακά, ξεκινώντας από τον κώδικα για την ορθή εκτέλεσή του από τον εξυπηρετητή, συνεχίζοντας με την επεξεργασία του xml template από το script για να παραχθούν όλα τα emails που είναι προς αποστολή και τελειώνοντας με την αποστολή των email στους χρήστες.

3.4.1 Έλεγχοι ορθής εκτέλεσης του script

Το script είναι σχεδιασμένο να εκτελείται από γραμμή εντολών σε περιβάλλον Unix. Είναι σχεδιασμένο να δέχεται διάφορα flags και ένα αρχείο XML. Τα flags μπορεί να είναι:

- -s (send) για αποστολή του mail template
- -t (test) για εκτύπωση όλων των διαφορετικών email που θα αποστέλλονταν
- -v (verbose) για εκτύπωση επιπλέον πληροφοριών για την αποστολή των email, όπως χρόνος εκκίνησης της αποστολής, ποσοστό ολοκλήρωσης, αν η αποστολή έγινε επιτυχώς, κ.α.

Τέλος, πρέπει να ακολουθεί ένα αρχείο με κατάληξη XML. Παράδειγμα:

```
./EmailSendingScript.php -s -v myTemplate.xml > log.txt
```

Εδώ το script θα αποστείλει το template *myTemplate.xml* σε όλους τους χρήστες που ορίστηκαν από το εσωτερικό sql ερώτημα, και τα αποτελέσματα από στολής θα αποθηκευτούν στο αρχείο *log.txt* για παρακολούθηση της διαδικασίας.

Αφού ορίσουμε στην αρχή του script ότι ακολουθεί κώδικας σε PHP, ελέγχουμε τον αριθμό των ορισμάτων που δόθηκαν από τον χρήστη, όπως και αν ο συνδυασμός ή η σειρά των ορισμάτων είναι σωστή και αν δόθηκε αρχείο προς επεξεργασία.

Τέλος, γίνεται ένας έλεγχος του XML αρχείου με το αρχείο DTD που δόθηκε παραπάνω, ώστε να δούμε αν το δοθέν XML αρχείο συμμορφώνεται με τις απαιτήσεις που έχει η γενική μορφή του email template που μπορεί να επεξεργαστεί το script μας.

3.4.2 Εντολές επεξεργασίας του xml template

Αφού έχουν γίνει οι έλεγχοι ορθότητας, αρχίζει η επεξεργασία του xml αρχείου. Εδώ θα επεξεργαστούν τρία στοιχεία του template, αυτά που έχουν τα δεδομένα για να αποσταλεί το email. Τα στοιχεία είναι τα **<smtpconfig>**, **<mysqlconfig>** και το **<query>**.

Τα δεδομένα από το **<smtpconfig>** θα χρειαστούν αργότερα, όταν είναι να γίνει αποστολή των μηνυμάτων. Παρόλα αυτά τα δεδομένα θα αποκομιστούν σε αυτόν τον χρόνο. Εάν υπάρχουν στοιχεία για SMTP Server τότε θα παραχθεί από το script ένας πίνακας με όλα τα δεδομένα που έχουν δοθεί (*server name, username, password, port*). Εάν πάλι δεν έχουν δοθεί δεδομένα, ο πίνακας θα είναι κενός.

Τα δεδομένα από το **<mysqlconfig>** είναι τα απαραίτητα για να συνδεθούμε με την βάση δεδομένων που επιθυμούμε. Όλα τα στοιχεία πρέπει να δοθούν στα αντίστοιχα πεδία (**<mysqlserver>**, **<mysqluser>**, **<mysqlpassword>**, **<mysqladb>**) για να πραγματοποιήσει το script μια σύνδεση με την βάση δεδομένων για την άντληση δεδομένων από το SQL ερώτημα που υπάρχει στο στοιχείο **<query>** και ενδεχομένως και στο **<subquery>**.

Το στοιχείο **<query>** περιέχει το SQL ερώτημα του οποίου τα αποτελέσματα, σε συνδυασμό με την ιδιότητα NAME δίνουν τα ονόματα των μεταβλητών που υπάρχουν μέσα στο κείμενο και στο xml αρχείο γενικότερα. Έτσι, το script πρώτα εκτελεί το SQL ερώτημα και ανάλογα με το πόσα **rows** θα επιστραφούν, τόσα διαφορετικά **emails** θα παραχθούν. Έτσι, θα γίνουν τόσες επαναλήψεις αντικατάστασης μεταβλητών όσες και τα **rows** που επιστράφηκαν. Πέρα από αυτό, δημιουργείται ένας πίνακας σε κάθε επανάληψη που έχει την αντιστοιχία των πεδίων που επιλέχτηκαν από το SQL ερώτημα και των τιμών που αντιπροσωπεύουν (*[\$[main.name]="john", \$[main.address]="Fabierou", ...*). Για κάθε επανάληψη, αντικαθίστανται οι μεταβλητές που υπάρχουν στο κείμενο και στις ιδιότητες του στοιχείου-παιδιού του **<query>**, **<email>**. Αυτό επιτυγχάνεται με την μετατροπή του xml στοιχείου **<email>** σε string.

Στην περίπτωση που έχουμε **<subquery>**, δηλαδή ένα εμφωλευμένο ερώτημα SQL μέσα στο κείμενο του email μας, ακολουθείται η ίδια περίπτωση διαδικασία. Παίρνουμε το SQL ερώτημα που υπάρχει σαν ιδιότητα και το εκτελούμε, μόνο που αυτή τη φορά τα αποτελέσματα που παίρνουμε αφορούνε τον τρέχον παραλήπτη του μηνύματος. Έτσι τα rows που επιστρέφονται από το ερώτημα SQL του **<subquery>**, αποθηκεύονται προσωρινά σε πίνακα και αφού τελειώσει η εκτέλεσή του, αντικαθίσταται το *text-tag* που υπάρχει στο κείμενο, π.χ. *\$Sub1Text* με το αποτέλεσμα του ερωτήματος.

Αφού τελειώσουν οι αντικαταστάσεις των μεταβλητών του κυρίως και των δευτερευόντων SQL queries, το email μετατρέπεται και πάλι σε XML και

αποθηκεύεται σε έναν πίνακα. Η διαδικασία συνεχίζεται μέχρις ότου τελειώσει η επανάληψη που ορίστηκε από το αποτέλεσμα του κυρίως SQL ερωτήματος. Στην συνέχεια, αρχίζει η αποστολή των email από τον πίνακα που δημιουργήθηκε.

3.4.3 Διαδικασία αποστολής των email

Αφού έχουμε όλα τα emails με αντικατεστημένες τις μεταβλητές, σε μορφή XML, είναι ώρα να αρχίσουμε την αποστολή τους. Αρχίζουμε μια επανάληψη για τον αριθμό των email, και για κάθε ένα από αυτά αντλούμε τα εξής στοιχεία: "**TO**", "**FROM**", "**BCC**", "**RETURN-PATH**" και "**SUBJECT**". Όλες αυτές τις ιδιότητες μαζί με το κυρίως κείμενο και μαζί με τον πίνακα SMTP που δημιουργήθηκε προηγουμένως, τα περνάμε στην συνάρτηση αποστολής του email.

Πριν από οτιδήποτε άλλο, δημιουργούμε τα *email headers*. Αυτά είναι κοινά και για την αποστολή του email μέσω του server της εφαρμογής, αλλά και μέσω ενός SMTP Server και περιέχουν τις πληροφορίες αποστολής του μηνύματος. Το header του email περιέχει το "**TO**", "**FROM**", "**BCC**" και "**RETURN-PATH**". Στην συνέχεια ελέγχουμε αν ο πίνακας με τα δεδομένα για τον SMTP Server είναι κενός. Εάν είναι, τότε χρησιμοποιούμε την συνάρτηση *mail()* της PHP, δίνοντάς την σαν ορίσματα τα *headers*, τον παραλήπτη, το θέμα του μηνύματος και το κυρίως κείμενο. Στην περίπτωση που έχουμε δεδομένα στον πίνακα του SMTP Server, χρησιμοποιούμε την συνάρτηση *mail()* της επέκτασης *PEAR* της PHP, που υποστηρίζει πρωτόκολλα *SSL(Secure Sockets Layer)* και *TLS(Transport Layer Security)*. Δίνεται παρακάτω:

```
require_once "Mail.php";
$host = $smtp["smtp_server"];
$port = $smtp["smtp_port"];
$username = $smtp["smtp_username"];
$password = $smtp["smtp_password"];
$headers = array ('From' => $from,
                 'To' => $to,
                 'Subject' => $subject,
                 'Return-Path' => $return-path);
$smtp = Mail::factory('smtp',
                    array ('host' => $host,
                          'port' => $port,
                          'auth' => true,
                          'username' => $username,
                          'password' => $password));
$mail = $smtp->send($to, $headers, $body);
if (PEAR::isError($mail)) {
    echo("<p>" . $mail->getMessage() . "</p>");
} else {
    echo("<p>Message successfully sent!</p>");
}
```

Εικόνα 3.9: Συνάρτηση αποστολής email μέσω smtp server.

ΚΕΦΑΛΑΙΟ 4: ΕΝΣΩΜΑΤΩΣΗ ΕΡΓΑΛΕΙΟΥ ΑΠΟΣΤΟΛΗΣ ΜΗΝΥΜΑΤΩΝ ΚΑΙ ΑΛΛΑΓΕΣ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΕΦΑΡΜΟΓΗΣ

4.1 Εισαγωγή	43
4.2 Σκοπός του εργαλείου αποστολής mail template στην εφαρμογή PaperReview	43
4.3 Αλλαγές στην δομή του mail template	43
4.4 Αλλαγές στην εφαρμογή PaperReview	45
4.5 Διαδικασία αποστολής mail template στους χρήστες.....	51
4.6 Επίλογος	52

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε το πώς ενσωματώνεται το εργαλείο αποστολής mail template στην εφαρμογή μας PaperReview και τις απαραίτητες αλλαγές και προσθήκες που έγιναν στις διεπιφάνειες της εφαρμογής μας, όπως επίσης και απαραίτητες προσθήκες στο γενικό mail template που χρησιμοποιείται για αποστολή μηνυμάτων.

4.2 Σκοπός του εργαλείου αποστολής mail template στην εφαρμογή PaperReview

Στην εφαρμογή μας, PaperReview, υπήρχε η ανάγκη για ενημέρωση των διαφόρων ομάδων χρηστών μέσω μηνυμάτων. Έπρεπε να ενημερωθούν οι χρήστες για ένα ανερχόμενο συνέδριο, να ενημερωθούν οι αξιολογητές για τις διατριβές που τους έχουν ανατεθεί, οι συγγραφείς για την ημερομηνία ληξιπρόθεσμης αποστολής συγγραμμάτων και άλλες παρόμοιες περιπτώσεις.

Έτσι λοιπόν δημιουργήθηκαν μερικά mail templates σε xml, με γενικό χαρακτήρα. Ο εκάστοτε chairman, μπαίνοντας στο εργαλείο μπορεί να διαλέξει από την λίστα των προτύπων μηνυμάτων όποιο ταιριάζει στην περίπτωση του, να περάσει στην επεξεργασία του θέματος του μηνύματος και του κυρίου κειμένου, δημιουργώντας έτσι ένα μήνυμα με προσωπικό ύφος και χαρακτήρα.

Κάθε mail template περιέχει ενσωματωμένο μέσα του ένα ερώτημα SQL με το οποίο αντλεί δεδομένα από την βάση της εφαρμογής, και με μορφή μεταβλητών μπορούν τα δεδομένα να ενσωματωθούν στο θέμα και κυρίως κείμενο του μηνύματος, δίνοντας έτσι μια προσωπική αναφορά σε κάθε ξεχωριστό παραλήπτη. Οι μεταβλητές ποικίλουν ανάλογα με το θέμα του mail template και για κάθε mail template δίνεται η αναλυτική λίστα των μεταβλητών που μπορούν να ενσωματωθούν κατά την δημιουργία-επεξεργασία του μηνύματος.

Αφού τελειώσει η επεξεργασία του μηνύματος, ο chairman δημιουργεί το δικό του email. Πλέον μπορεί να προχωρήσει στην αποστολή του, χρησιμοποιώντας τον τοπικό server της εφαρμογής ή να χρησιμοποιήσει τον δικό του SMTP Server για μεγαλύτερη ευκολία και σιγουριά.

4.3 Αλλαγές στην δομή του mail template

Για την ενσωμάτωση του εργαλείου αποστολής mail template στην εφαρμογή μας, ήταν απαραίτητο να προστεθούν μερικά ακόμη στοιχεία στην δομή

του γενικού μας mail template που είδαμε στο προηγούμενο κεφάλαιο. Είναι τα παρακάτω:

Στοιχείο <templateinfo>

Το στοιχείο <templateinfo> περιέχει μια σύντομη περιγραφή του template για το ποια περίπτωση είναι η χρησιμότητα του. Συμπληρώνεται κατά την κατασκευή του template, και υπάρχει δυνατότητα αλλαγής του κατά την δημιουργία ενός νέου, όπως θα δούμε παρακάτω.

Στοιχείο <templatecreator>

Το στοιχείο <templatecreator> περιέχει το όνομα του τελικού δημιουργού του mail template. Είναι αρχικά κενό, και συμπληρώνεται αυτόματα με την δημιουργία ενός mail template από τον chairman που το δημιούργησε.

Στοιχείο <fieldsinfo>

Το στοιχείο <fieldsinfo> περιέχει έναν αριθμό στοιχείων-παιδιών με όνομα <field>, ανάλογα με το πόσα μεταβλητά πεδία έχουμε από τα SQL ερωτήματα στο template μας. Το κάθε στοιχείο-παιδί έχει μια ιδιότητα (*attribute*) η οποία είναι το όνομα μιας από τις μεταβλητές που υπάρχουν στο κείμενο του template, και το περιεχόμενο του στοιχείου είναι μια σύντομη περιγραφή της εκάστοτε μεταβλητής. Παράδειγμα: <field name="\$[main.lname]">Email's receiver last name.</field> <field name="\$[Sub1.degree]">Value of receivers degrees</field>.

Άρα η γενική δομή του mail template γίνεται ως εξής:

```
<?xml version="1.0" encoding="utf-8"?>
<template>
  <templateinfo>[...]</templateinfo>
  <templatecreator>[...]</templatecreator>
  <fieldsinfo>
    <field name="...">Description</field>
    [...]
  </fieldsinfo>
  <smtpconfig>
    [...]
  </smtpconfig>
  <mysqlconfig>
    [...]
  </mysqlconfig>
  <query>
    [...]
  </query>
</template>
```

Εικόνα 4.1: Αλλαγές στην γενική δομή του mail template

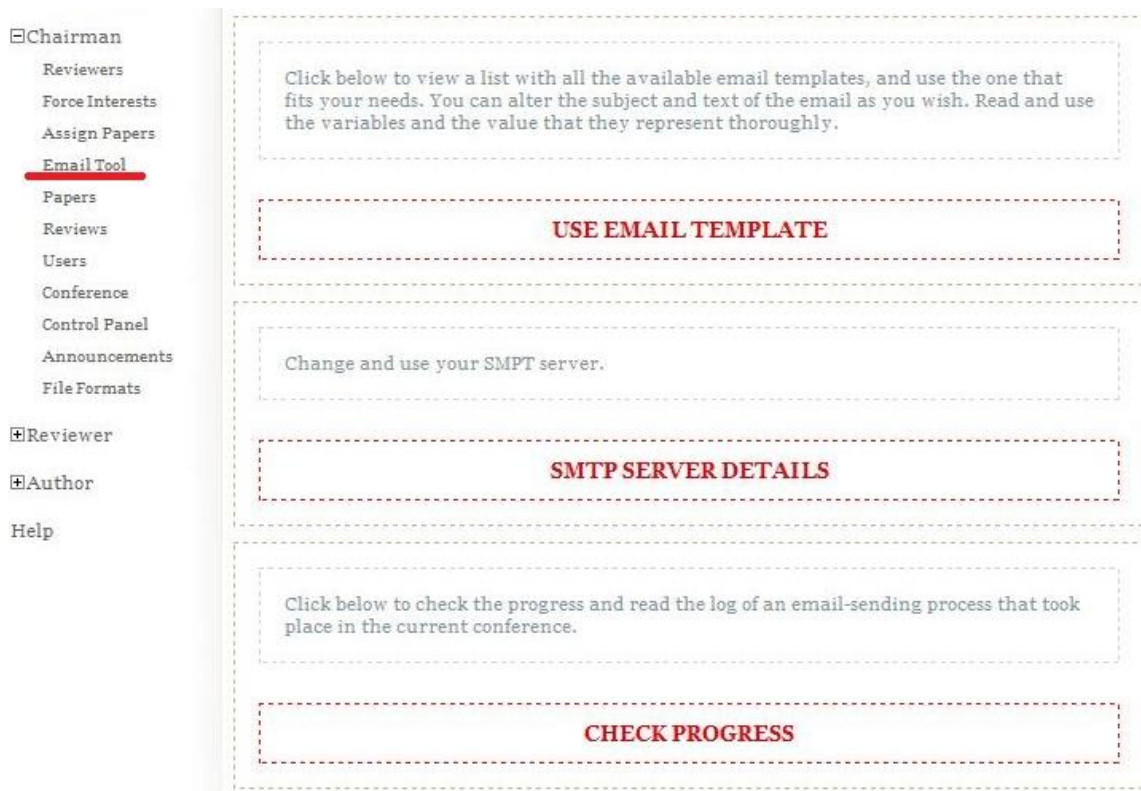
Τέλος, θα έχουμε και κάποιες αλλαγές στο DTD αρχείο μας:

```
<!ELEMENT template (templateinfo, templatecreator, fieldsinfo, smtpconfig,
mysqlconfig, query)>
<!ELEMENT templateinfo      (#PCDATA)>
<!ELEMENT templatecreator  (#PCDATA)>
<!ELEMENT fieldsinfo       (field*)>
<!ELEMENT field             (#PCDATA)>
[...]
<!ATTLIST field name CDATA #REQUIRED>
```

Εικόνα 4.2: Αλλαγές και προσθήκες στο αρχείο DTD

4.4 Αλλαγές στην εφαρμογή PaperReview

Η κύρια αλλαγή που έγινε στην εφαρμογή PaperReview ήταν η προσθήκη της επιλογής του "Email Tool" στο μενού επιλογών του chairman. Εκεί είναι η κεντρική σελίδα του εργαλείου όπου ο chairman έχει τρεις επιλογές: Να δημιουργήσει ένα νέο mail template από μερικά ήδη υπάρχοντα mail templates, να ορίσει κάποιον SMTP Server για την αποστολή των μηνυμάτων στους παραλήπτες, και τέλος να δει την πορεία μιας διεργασίας αποστολής μηνυμάτων μέσω αρχείων καταγραφής(logs).



Εικόνα 4.3: Κύρια διεπαφή εργαλείου Email Tool.

Η πρώτη επιλογή μας βγάξει στην παρακάτω οθόνη:

The screenshot shows a web interface for 'PAPER REVIEW' with a user profile for 'JOHN MADENIS'. The main content area displays two lists of XML templates. The first list, 'Ready XML Templates List', contains three entries: 'CallForPapers', 'ReviewsToMake', and 'WelcomePC', each with a description and a 'Use' button. The second list, 'Created XML Templates List', contains two entries: 'CallForPapers' and 'MadenisWelcomes', each with creation details and buttons for 'Send', 'View', 'Edit', and 'Del'. A 'RETURN TO TOP' link is located at the bottom right of the content area.

NAME	DESCRIPTION	
CallForPapers	With This Template You Can Send A Message (CFP) To Everybody Registered In This System In Any Conference	Use
ReviewsToMake	With This Template You Can Send A Message To Reviewers Of This Conference With Their Assignements.	Use
WelcomePC	With This Template You Can Send A Welcome Message To Everybody Registered As A Reviewer In This Conference	Use

NAME	INFO				
CallForPapers	Created On: April 05 2010 12:11:03. Created By: Madenis John Template Info: Call For Papers For Algorith Conference By John Madenis	Send	View	Edit	Del
MadenisWelcomes	Created On: April 05 2010 12:14:13. Created By: Madenis John Template Info: With This Template You Can Send A Welcome Message To Everybody Registered As A Reviewer In This Conference	Send	View	Edit	Del

Εικόνα 4.4: Λίστες με έτοιμα mail templates και δημιουργημένα email

Εδώ βλέπουμε δυο λίστες, η πρώτη "*Ready XML Templates List*" είναι αυτή που περιέχει όλα τα mail templates που είναι έτοιμα για επεξεργασία και δημιουργία μέσω αυτών των καινούργιων mail template. Η δεύτερη "*Created XML Templates List*" είναι αυτή με όλα τα κατασκευασμένα από κάποιον chairman mail template. Βλέπουμε ότι για κάθε κατασκευασμένο email έχουμε το όνομα του chairman που το κατασκεύασε και την ημερομηνία. Επίσης υπάρχει η δυνατότητα να διαγραφεί το email (*Del*), η επιλογή της διόρθωσης (*Edit*), όπως και να γίνει μια γρήγορη προεπισκόπηση (*View*), για να δούμε αν είναι το επιθυμητό αποτέλεσμα.

Για να αποσταλεί το email, αρκεί να πατήσουμε Send. Θα γίνει κλήση στο script της αποστολής των μηνυμάτων, και αυτό με την σειρά του θα δημιουργήσει ένα αρχείο καταγραφής της αποστολής, με το οποίο ελέγχουμε την διαδικασία αποστολής, αν παραδόθηκαν τα μηνύματα σωστά και σε ποια διεύθυνση, τον χρόνο παράδοσης και το ποσοστό ολοκλήρωσης της διαδικασίας αποστολής.

Η δεύτερη επιλογή της κεντρικής οθόνης μας βγάξει στην παρακάτω φόρμα:

The screenshot shows a web interface for 'PAPER REVIEW'. At the top, it says 'USER ~ JOHN MADENIS' and has 'profile' and 'logout' links. A navigation menu on the left includes 'Home', 'Announcements', 'Chairman' (with sub-items: Reviewers, Force Interests, Assign Papers, Email Tool, Papers, Reviews, Users, Conference, Control Panel, Announcements, File Formats), 'Reviewer', 'Author', and 'Help'. The main content area shows the date 'Wednesday, 13th of January 2010' and the title 'EMAIL TOOL'. Below this is a dashed box containing a message: 'Here you can fill in the form for SMTP server configuration. If you want to use the site's server, simply leave the form blank.' The form itself is titled 'SMTP Server Config' and has four input fields: 'SMTP Server:', 'SMTP Username:', 'SMTP Password:', and 'SMTP Server Port:'. A 'Submit' button is located below the fields. At the bottom right of the page, there is a 'RETURN TO TOP' link.

Εικόνα 4.5: Φόρμα Παραμετροποίησης SMTP Server

Εδώ βλέπουμε μια φόρμα με 4 πεδία:

1. **SMTP Server:** Συμπληρώνουμε με την διεύθυνση του SMTP Server που θέλουμε να χρησιμοποιήσουμε, αναλόγως και με την προστασία που παρέχει(SSL ή TLS) π.χ. ssl://user.mysmtpserver.com
2. **SMTP Username:** Συμπληρώνουμε με το username που έχουμε στον παραπάνω server.
3. **SMTP Password:** Συμπληρώνουμε με το password που έχουμε στον παραπάνω server.
4. **SMTP Server Port:** Συμπληρώνουμε με το port που επικοινωνεί ο server για το SMTP protocol.

Εδώ, αν ένας chairman έχει και θέλει να χρησιμοποιήσει τον SMTP server του, συμπληρώνει την φόρμα και οι διαδικασίες αποστολής μηνυμάτων που θα ξεκινάει θα αποστέλλονται μέσω του SMTP server, χρησιμοποιώντας επιπλέον πρωτόκολλα ασφαλείας, όπως SSL και TLS. Στην περίπτωση που δεν υπάρχουν τιμές σε όλα τα πεδία, τα μηνύματα αποστέλλονται μέσω του server της εφαρμογής.

Για να διασφαλιστεί ότι τα πεδία θα συμπληρωθούν, η φόρμα έχει έναν JavaScript validator από πίσω της, με σκοπό την ορθή συμπλήρωσή της. Ο validator ελέγχει για τα εξής: να είναι όλα τα πεδία συμπληρωμένα ή όλα κενά, τα τρία πρώτα πεδία να έχουν αλφαριθμητικούς χαρακτήρες, και το τελευταίο μόνο αριθμητικούς, έτσι ώστε να εισαχθούν σωστά δεδομένα στον πίνακα αποθήκευσης ρυθμίσεων SMTP.

Για την σωστή διαχείριση των δεδομένων, δημιουργήθηκε στην βάση μας ένας καινούργιος πίνακας, ο "**SMTP**". Τα πεδία του πίνακα είναι: *chairman_id*, *smtp_server*, *smtp_username*, *smtp_password*, *smtp_port*, *smtp_flag*. Το κύριο κλειδί του πίνακα είναι το *chairman_id*.

Πίνακας 4.1: Ο πίνακας SMTP στην βάση δεδομένων

Πεδίο	Τύπος	Κενό	Προκαθορισμένο
<i>chairman_id</i>	int(11)	Όχι	
<i>smtp_server</i>	char(40)		
<i>smtp_username</i>	char(20)		
<i>smtp_password</i>	char(20)		
<i>smtp_port</i>	int(11)		
<i>smtp_flag</i>	tinyint(1)		

Σε αυτόν τον πίνακα αποθηκεύονται οι πληροφορίες που δίνει ο κάθε *chairman* στην περίπτωση που θέλει να χρησιμοποιήσει κάποιον SMTP Server. Το πεδίο *chairman_id* αντιπροσωπεύει το id του *chairman* που εισήγαγε δεδομένα στην βάση. Το πεδίο *smtp_server* περιέχει τον επιθυμητό server που θέλει να χρησιμοποιήσει ο *chairman* για την αποστολή των μηνυμάτων. Το πεδίο *smtp_username* και *smtp_password* αντίστοιχα περιέχουν το όνομα χρήστη και τον κωδικό πρόσβασης στον παραπάνω δηλωμένο server. Το πεδίο *smtp_port* περιέχει τον αριθμό του port που ακούει ο συγκεκριμένος server, και τέλος το πεδίο *smtp_flag* βοηθάει για τον διαχωρισμό εισαγωγής ή διόρθωσης των δεδομένων που έδωσε ο *chairman*. Από αυτόν τον πίνακα θα αντλήσει αργότερα το script τις πληροφορίες για τον τρόπο αποστολής των μηνυμάτων.

Η τρίτη επιλογή του κεντρικού μενού μας βγάζει στην επόμενη οθόνη:

The screenshot shows a web interface for 'PAPER REVIEW'. At the top, it says 'USER ~ JOHN MADENIS' and has links for 'profile' and 'logout'. A navigation menu on the left includes 'Home', 'Announcements', 'Chairman', 'Reviewers', 'Force Interests', 'Assign Papers', 'Email Tool', 'Papers', 'Reviews', 'Users', 'Conference', 'Control Panel', 'Announcements', 'File Formats', 'Reviewer', 'Author', and 'Help'. The main content area is titled 'Monday, 05th of April 2010' and 'EMAIL TOOL'. It contains a text box stating: 'Below you can read the log files to gather info for any e-mail sending process that took place for current conference.' Below this is a section titled 'List of Log Files' containing a table with three columns: 'NAME', 'INFO', and 'View'.

NAME	INFO	View
Madenis_John_CallForPapers	Process Stared On: January 13 2010 19:53:47.	View
Doe_james_ReviewsToMake	Process Stared On: October 01 2009 19:45:09.	View
Doe_james_Welcome	Process Stared On: October 01 2009 19:43:25.	View

At the bottom right of the page, there is a link that says 'RETURN TO TOP'.

Εικόνα 4.6: Λίστα με αρχεία καταγραφής διαδικασιών αποστολής

Εδώ βλέπουμε μια λίστα από αρχεία καταγραφής διαδικασιών αποστολής email. Βλέπουμε ότι κάθε αρχείο έχει το ονοματεπώνυμο του chairman που άρχισε την διαδικασία, ποιο mail template χρησιμοποίησε μαζί με έναν αύξον αριθμό δίπλα, στην περίπτωση που χρησιμοποιηθεί το ίδιο template πάνω από μία φορά. Επίσης δίπλα υπάρχει η πληροφορία της ημερομηνίας έναρξης της διαδικασίας. Πατώντας "View" μας ανοίγει καινούργιο παράθυρο με τις πληροφορίες αποστολής.

Τέλος έχουμε την φόρμα επεξεργασίας του mail template.

The screenshot shows a web interface for editing email templates. On the left is a navigation menu with items like Home, Announcements, Chairman, Reviewers, etc. The main content area is titled 'EMAIL TOOL' and contains a warning about subqueries, a table of variables, and a form to create a template.

Monday, 05th of April 2010
EMAIL TOOL

Here you can see the ready email template form, and apply any changes desired. Below are the variables that can be inserted in the email's text.
WARNING! Some Email templates contain subqueries. For them to work fields as **\$\$Sub1START, \$Sub1TEXT, \$Sub1END**, must remain unchanged! Altering them may cause parsing failures.

VARIABLE	VALUE THAT REPRESENTS
`\${main.lname}`	Email's Reciever Last Name.
`\${main.fname}`	Email's Reciever First Name.
`\${main.email}`	Email's Reciever Email Address.
`\${main.name}`	Conference Name.
`\${main.alias}`	Conference Alias.
`\${main.place}`	Conference Place.
`\${main.date_conference_held}`	Conference Dates.
`\${main.contact_email}`	Conference Contact Email.
`\${main.contact_phone}`	Conference Contact Phone.
`\${main.website}`	Conference Website.
`\${main.comments}`	Conference Comments.
`\${main.deadline}`	Conference Deadline.
`\${main.abstracts_deadline}`	Submission Abstracts Deadline.
`\${main.manuscripts_deadline}`	Submission Manuscripts Deadline.
`\${main.camera_ready_deadline}`	Submission CR Deadline.
`\${main.preferences_deadline}`	PC Preferences Deadline.
`\${main.reviews_deadline}`	PC Reviews Deadline.

Template Name: CallForPapers

Template Info: With this template you can send a message (CFP) to everybody registered in this system in any conference

Subject: CFP: `\${main.name}`

Text:

```

=====
`${main.alias}` - Call for Papers
`${main.name}`
`${main.date_conference_held}` `${main.place}`
`${main.website}`
In conjunction with XXXXX Conference
=====
Dear `${main.lname}` `${main.fname}`,
This mail was sent to you by WRITE YOUR NAME HERE.
This is a call for papers, for `${main.name}` (`${main.alias}`).
PLACE HERE YOUR CFP TEXT.
SUBMISSION INSTRUCTIONS:
Submitted papers must not substantially overlap with papers that have
been
published or that are simultaneously submitted to a journal or a
conference
with proceedings. Submitted papers can be of two types:
- Regular Research Papers: these papers should report original research
results or significant case studies. They should be at most 10 pages.
=====

```

Create

Εικόνα 4.7: Φόρμα επεξεργασίας mail template

Εδώ βλέπουμε αρχικά μια λίστα από μεταβλητά πεδία τιμών που παίρνουμε από το SQL ερώτημα του template μας, και το τι αντιπροσωπεύει το καθένα. Αυτές είναι οι πληροφορίες που βρίσκονται στα πεδία <field ...> στο mail

template μας. Από κάτω βλέπουμε το πεδίο "*Template Name*" με το οποίο δίνουμε ένα όνομα στο mail template που θα δημιουργήσουμε. Για το συγκεκριμένο πεδίο υπάρχει JavaScript validator που ελέγχει το όνομα μόνο για αλφαριθμητικούς χαρακτήρες έτσι ώστε να μην υπάρχει πρόβλημα στην δημιουργία αρχείου. Έπειτα βλέπουμε το πεδίο "*Template Info*", όπου βλέπουμε τις πληροφορίες για το συγκεκριμένο template. Ο chairman μπορεί να το αλλάξει ανάλογα με το ύφος και θέμα που θα δώσει στο μήνυμά του. Στην συνέχεια έχουμε το θέμα του μηνύματος, το οποίο μπορούμε να τροποποιήσουμε, όπως και πιο κάτω το κυρίως κείμενο του μηνύματος που θα αποστείλουμε. Πατώντας "*Create*" δημιουργούμε ένα xml αρχείο το οποίο θα εμφανιστεί στην λίστα των δημιουργημένων αρχείων email, και θα είναι έτοιμο προς αποστολή.

4.5 Διαδικασία αποστολής mail template στους χρήστες

Η διαδικασία αποστολής του τελικού mail template είναι απλή. Αφού τελειώσει ο chairman με την διαδικασία σύνταξης του mail template του, και έχει διαλέξει αν θα χρησιμοποιήσει SMTP server η τον τοπικό server της εφαρμογής, επιλέγει "*Send*" και ξεκινά η διαδικασία αποστολής.

Στην εφαρμογή μας βέβαια, ο τρόπος εκτέλεσης του script είναι τυποποιημένος, και έχει την εξής μορφή που είναι γραμμένη στο κομμάτι της αποστολής των email:

```
./SendEmails.php -s -v MailTemplate.xml
```

όπου το MailTemplate.xml αλλάζει ανάλογα με το πιο template επιλέγουμε για αποστολή.

Κατά την διάρκεια της αποστολής, δημιουργείται και ένα αρχείο καταγραφής (log file), του οποίου τίτλος είναι το όνομα του chairman που άρχισε την αποστολή μαζί με το ποιο template χρησιμοποιήθηκε. Π.χ.

```
John_Madenis_WelcomeMessage.txt
```

Το αρχείο αυτό ενημερώνεται συνεχώς με κάθε επιτυχή ή μη αποστολή μηνύματος. Το ποσοστό ολοκλήρωσης εμφανίζεται μέσα στο αρχείο. Η μορφή της ενημέρωσης έχει ως εξής:

```
The e-mail has been(or has been not) delivered to: sarkyman@hotmail.com  
successfully.
```

```
Number of email in queue: 1
```

```
Total number of emails to be send: 13
```

```
Percentage of completion of process: 7.69%
```

```
Timestamp: 19:44, 1st October
```

4.6 Επίλογος

Το εργαλείο αποστολής mail template είναι ένας απλός και εύχρηστος τρόπος ενημέρωσης των χρηστών που βρίσκονται στην εφαρμογή PaperReview. Το βασικότερο μειονέκτημα είναι ότι ο chairman δεν έχει την δυνατότητα δημιουργίας δικού του SQL ερωτήματος, για να επιλέξει πεδία και αποτελέσματα που δεν του προσφέρουν τα ήδη υπάρχοντα templates. Από την άλλη βέβαια θα μπορούσε να εξάγει δεδομένα που δεν τον αφορούνε, και να τα χρησιμοποιήσει με κακόβουλο τρόπο. Έτσι, εάν ο chairman χρειάζεται κάποιο άλλο template, πρέπει να επικοινωνήσει με τον administrator του συστήματος για να το δημιουργήσει εκείνος.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Quentin Zervaas, Practical Web 2.0 Applications with PHP.
Εκδόσεις Apress, 2008

Jason Gerner, Elizabeth Naramore, Morgan L. Owens, Matt Warden, Professional LAMP: Linux®, Apache, MySQL®, and PHP5 Web Development
Εκδόσεις Wiley Publishing, Inc., 2006

Welling Luke, Thomson Laura, Ανάπτυξη Web Εφαρμογών με PHP και MySQL.
Μετάφραση: Γκλαβά Μαρία. Εκδόσεις Γκιούρδας, 2005

ΙΣΤΟΣΕΛΙΔΕΣ

<http://www.it.uom.gr/>
<http://www.wikipedia.org>
<http://w3schools.com>
<http://php.net>

ΣΗΜΕΙΩΣΕΙΣ

Στο CD που εσωκλείεται, περιέχονται όλα τα αρχεία της εφαρμογής. Τα αρχεία που είναι δημιουργημένα-τροποποιημένα από εμένα για τα διαδικτυακά εργαλεία είναι τα εξής:

AlterEmail.php
assign_papers.php
assignAlgorithm.php
automaticAssignmentResults.php
CheckProgress.php
CreateEmail.php
deleteAllAssignments.php
deleteAllAssignments.php
deleteAssignments.php
DeleteEmail.php
EditEmail.php
emailTool.php
PreviewEmail.php
SendEmail.php
SMTPInsertValues.php
SMTPServerValues.php
UseExistingTemplates.php
ViewEmail.php
ViewLog.php
/mail_sender/EmailSender.php
/mail_sender/mailHelpFunctions.php
/mail_sender/mail-validator.js
/mail_sender/smtp-validator.js

/mail_sender/template.dtd
/email_templates/Ready/CallForPapers.xml
/email_templates/Ready/ReviewsToMake.xml
/email_templates/Ready/WelcomePC.xml
/include/layoutinc.php