



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη συστήματος Ηλεκτρονικού Χαρτοφυλακίου



Του φοιτητή

Παράσχου Χρήστου

Αρ. Μητρώου: 032451

Επιβλέπων καθηγητής

Καθ. Κλεφτούρης Δημήτριος

Θεσσαλονίκη 2012

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με το θέμα του ηλεκτρονικού χαρτοφυλακίου. Το ηλεκτρονικό χαρτοφυλάκιο αποτελεί μια ηλεκτρονική έκδοση του παραδοσιακού (hard-copy) χαρτοφυλακίου, το οποίο ο χρήστης έχει διαθέσιμο και μπορεί να παρουσιάζει σε κάθε περίπτωση.

Πιο συγκεκριμένα, στη συγκεκριμένη πτυχιακή εργασία, εξετάζουμε ένα υπάρχον σύστημα ηλεκτρονικού χαρτοφυλακίου και στη συνέχεια προχωρούμε στην ανάπτυξη και στην σχεδίαση ενός ηλεκτρονικού χαρτοφυλακίου κατάλληλο για τις ανάγκες των φοιτητών του ΤΕΙ. Ο φοιτητής έχει δυνατότητα να καταχωρεί τα μαθήματα που έχει δηλώσει μαζί με τις εργασίες και όλο το υλικό που διαθέτει. Το υλικό αυτό είναι προσβάσιμο από τους καθηγητές, οι οποίοι έχουν επιπλέον τη δυνατότητα να σχολιάζουν τα στοιχεία αυτά.

Για την ανάπτυξη της εφαρμογής έχουν αξιοποιηθεί αποκλειστικά λογισμικά και τεχνολογίες ανοιχτού κώδικα.

ABSTRACT

This thesis deals with subjects related to the e-portfolio. The electronic portfolio is an electronic version of the traditional (hard-copy) portfolio, which is online available and which the user can present at any time.

More specifically, in this thesis, an existing e-portfolio system is examined and then, the design and the development of an electronic portfolio system suitable for the needs of students of our department is performed. The student is able to track down the courses and enhance them with all related work and material. This material is accessible to tutors who are also allowed to comment on these figures.

For the application development exclusively open source technologies has been selected.

ΕΥΧΑΡΙΣΤΙΕΣ (προαιρετικά)

Τέλος θα ήθελα να ευχαριστήσω τον καθηγητή μου κ. Δημήτριο Κλεφτούρη για την σωστή καθοδήγηση που μου πρόσφερε για το πέρας της πτυχιακής μου εργασίας και τους γονείς μου που με βοήθησαν και με στήριξαν όλα αυτά τα χρόνια της φοιτητικής μου πορείας.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ.....	2
ABSTRACT	3
ΕΥΧΑΡΙΣΤΙΕΣ (προαιρετικά)	4
ΠΕΡΙΕΧΟΜΕΝΑ	5
Ευρετήριο εικόνων.....	7
ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ.....	8
ΚΕΦΑΛΑΙΟ 2 Ε-Portfolio - Ηλεκτρονικό Χαρτοφυλάκιο.....	9
ΥΠΟΚΕΦΑΛΑΙΟ 2.1 Τύποι e-portfolio.....	11
ΥΠΟΚΕΦΑΛΑΙΟ 2.2 Χρήση ενός ηλεκτρονικού χαρτοφυλακίου	11
ΚΕΦΑΛΑΙΟ 3 Προϋπάρχουσες εφαρμογές Ηλεκτρονικού Χαρτοφυλακίου.....	13
ΥΠΟΚΕΦΑΛΑΙΟ 3.1 Πρωτοβάθμια εκπαίδευση.....	13
ΥΠΟΚΕΦΑΛΑΙΟ 3.1.1 Jackie - Μαθήτρια Δημοτικού.....	13
ΚΕΦΑΛΑΙΟ 4 Σχεδιασμός Εφαρμογής Ηλεκτρονικού Χαρτοφυλακίου	18
ΥΠΟΚΕΦΑΛΑΙΟ 4.1 Απαιτήσεις της εφαρμογής	19
ΥΠΟΚΕΦΑΛΑΙΟ 4.2 Σχεδιασμός βάσης δεδομένων	20
ΥΠΟΚΕΦΑΛΑΙΟ 4.3 Σχεδιασμός Εφαρμογής.....	22
ΚΕΦΑΛΑΙΟ 5 Εργαλεία Υλοποίησης του Ηλεκτρονικού Χαρτοφυλακίου.....	23
ΥΠΟΚΕΦΑΛΑΙΟ 5.1 Εξυπηρετητής δικτυακών εφαρμογών Apache	23
ΥΠΟΚΕΦΑΛΑΙΟ 5.2 Γλώσσα Προγραμματισμού PHP.....	25
ΥΠΟΚΕΦΑΛΑΙΟ 5.3 Βάση Δεδομένων MySQL.....	27
ΥΠΟΚΕΦΑΛΑΙΟ 5.4 Πλατφόρμα CakePHP.....	28
ΥΠΟΚΕΦΑΛΑΙΟ 5.5 Μεθοδολογία ανάπτυξης Μοντέλου – Όψης – Ελεγκτή.....	30
ΥΠΟΚΕΦΑΛΑΙΟ 5.5.1 Σχέσεις μεταξύ των συνιστωσών.....	31
Προβολή σχέση ελεγκτή.....	31
Μοντέλο-Όψη	31
Μοντέλο σχέση ελεγκτή.....	32
ΥΠΟΚΕΦΑΛΑΙΟ 5.5.2 Πλεονεκτήματα από τη χρήση του MVC	33
ΥΠΟΚΕΦΑΛΑΙΟ 5.5.3 Μειονεκτήματα του MVC.....	34
ΥΠΟΚΕΦΑΛΑΙΟ 5.5.4 Αρχιτεκτονική MVC – Πλατφόρμες MVC.....	34
ΚΕΦΑΛΑΙΟ 6 Λεπτομέρειες Υλοποίησης του Ηλεκτρονικού Χαρτοφυλακίου.....	36
ΒΙΒΛΙΟΓΡΑΦΙΑ	38

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

ΠΑΡΑΡΤΗΜΑΤΑ Κώδικας της εφαρμογής	39
ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ	56

Ευρετήριο εικόνων

Εικόνα 1: Η αρχική οθόνη του e-Pearl	14
Εικόνα 2: Οι διδακτικοί στόχοι.....	15
Εικόνα 3: Μαθησιακό υλικό για την Jackie	16
Εικόνα 4: Οι εργασίες.....	17
Εικόνα 5: Το σχήμα της βάσης δεδομένων.....	21
Εικόνα 6: Το μοντέλο MVC	30
Εικόνα 7: Η MVC δομή.....	32
Εικόνα 8: Η εισαγωγική οθόνη της εφαρμογής μας.....	56
Εικόνα 9: Η οθόνη εγγραφής νέου χρήστη.....	57
Εικόνα 10: Το ηλεκτρονικό χαρτοφυλάκιο του χρήστη	58
Εικόνα 11: Εισαγωγή σχολίων.....	59

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

Τα χαρτοφυλάκια διατηρούσαν ανέκαθεν μια ιδιαίτερη κατηγορία προσωπικών εγγράφων, καθώς αποτελούσαν το μοναδικό μέσο αυτοπροώθησης και αυτοπροβολής των επαγγελματιών.

Με την έλευση της ηλεκτρονικής εποχής, την ανάπτυξη των ηλεκτρονικών υπολογιστών και του διαδικτύου, το χαρτοφυλάκιο έχει αποκτήσει πλέον και ηλεκτρονική μορφή το η-χαρτοφυλάκιο (e-portfolio) αλλά και επιπλέον χρησιμότητες, καθώς πλέον έχει αρχίσει η αξιοποίησή του από την περίοδο της εκπαιδευτικής διαδικασίας.

Στην παρούσα πτυχιακή εργασία, αρχικά εξετάζεται η έννοια του χαρτοφυλακίου δίνοντας έμφαση κυρίως στην έννοια του ηλεκτρονικού χαρτοφυλακίου, στο Κεφάλαιο 2. Στη συνέχεια, στο κεφάλαιο 3, αξιολογείται ένα ενδεικτικό ηλεκτρονικό χαρτοφυλάκιο κατάλληλο για την εκπαιδευτική διαδικασία. Η εφαρμογή που μελετάται είναι η εφαρμογή e-pearl. Στη υπόλοιπα κεφάλαια, πραγματοποιείται ο σχεδιασμός και η υλοποίηση του συστήματος του ηλεκτρονικού χαρτοφυλακίου, με χρήση λογισμικού ανοιχτού κώδικα: Εξυπηρετητής Apache, Σχεσιακή Βάση Δεδομένων MySQL, Γλώσσα προγραμματισμού PHP και πλατφόρμα υλοποίησης (PHP Framework) CakePHP.

ΚΕΦΑΛΑΙΟ 2

E-Portfolio - Ηλεκτρονικό Χαρτοφυλάκιο

Η ακριβής απόδοση του όρου portfolio στα ελληνικά σύμφωνα με γνωστά λεξικά είναι ο χαρτοφύλακας, όπου τοποθετούνται έγγραφα, είτε για προσωπική χρήση είτε για επίσημη. Ο όρος portfolio έχει συνδυαστεί με το κατάλληλα επιμελημένο χαρτοφυλάκιο που φέρει μαζί του κάθε υποψήφιος υπάλληλος σε μία ανάλογη συνέντευξη.

Το ηλεκτρονικό χαρτοφυλάκιο, το οποίο είναι επίσης γνωστό ως e-χαρτοφυλάκιο ή ψηφιακό χαρτοφυλάκιο, είναι μια συλλογή από ηλεκτρονικά τεκμήρια που έχουν συγκεντρωθεί από ένα χρήστη, συνήθως στον Παγκόσμιο ιστό (Web). Τα ηλεκτρονικά τεκμήρια αποτελούνται είτε από ένα κείμενο που έχει εισάγει ο χρήστης, είτε ηλεκτρονικά αρχεία, όπως αρχεία κειμένου, εικόνες, πολυμέσα, καταχωρήσεις από blogs και υπερσυνδέσμους. Για την ακρίβεια, το είδος των ηλεκτρονικών τεκμηρίων σε ένα ηλεκτρονικό χαρτοφυλάκιο, περιορίζεται αποκλειστικά από την φαντασία του χρήστη και τους περιορισμούς που θέτει η υπηρεσία που προσφέρει το ηλεκτρονικό χαρτοφυλάκιο στους χρήστες της. Τα ηλεκτρονικά χαρτοφυλάκια είναι σε θέση να αποτελούν ταυτόχρονα τόσο την κατάλληλη έκφραση των ικανοτήτων και της προσωπικότητας του δημιουργού τους όσο και πλατφόρμες κατάλληλες για την αποτύπωση προσωπικών απόψεων και εκφράσεων του δημιουργού τους, και, στην περίπτωση των online χαρτοφυλακίων, υπάρχουν οι δυνατότητες διαρκούς και δυναμικής ανανέωσής τους. Ορισμένα ηλεκτρονικά χαρτοφυλάκια έχουν διάφορες όψεις (views) ώστε να επιτρέπουν την κατάλληλη πρόσβαση στο κοινό. Με τον τρόπο αυτό επιτρέπεται η χρησιμοποίηση του ίδιου χαρτοφυλακίου για πολλαπλούς σκοπούς με διαφορετικές εκφάνσεις.

Ένα e-portfolio μπορεί να χρησιμοποιηθεί σαν ένα είδος καταγραφής της πορείας ενός μαθητή. Τα αρχεία είναι στενά συνδεδεμένα με το Εκπαιδευτικό πρόγραμμα, ένα σημαντικό εργαλείο που χρησιμοποιείται για την διαχείριση της μάθησης από άτομα, ομάδες και οργανισμούς.

Οι μαθητές έχουν διδαχθεί για να δημιουργούν ψηφιακές ταυτότητες με χρήση λογισμικού παρουσίασης ή εργαλεία κατασκευής ιστοσελίδων. Πιο πρόσφατα η χρήση των εικονικών περιβαλλόντων μάθησης (Virtual Learning Environments - VLE) σε σχολεία και πανεπιστήμια έχει οδηγήσει σε αυξημένη δραστηριότητα στη δημιουργία του e-χαρτοφυλακίου για διάφορους λόγους. Τα περισσότερα από αυτά τα e-χαρτοφυλάκια, ωστόσο, διατηρούνται μέσα στο VLE και δεν είναι εύκολα προσβάσιμα εκτός αυτού. Ως εκ τούτου, αυτά τα χαρτοφυλάκια αντιμετωπίζουν σημαντικά προβλήματα σχετικά με την εξαγωγή δεδομένων και με τη διαλειτουργικότητα του συνολικού συστήματος. Μια εναλλακτική προσέγγιση είναι η χρήση συστημάτων που φιλοξενούνται εκτός ιδρυμάτων. Η συγκεκριμένη επιλογή επιτρέπει την απρόσκοπτη μετάβαση από την εκπαίδευση στην απασχόληση, ακόμη και στη συνταξιοδότηση.

Τα ηλεκτρονικά χαρτοφυλάκια, όπως περίπου και με τα παραδοσιακά, διευκολύνεται ο προβληματισμός των μαθητών σχετικά με τη μάθηση, οδηγώντας σε μεγαλύτερη ευαισθητοποίηση των στρατηγικών μάθησης και των αναγκών [1]. Σύμφωνα με μια πρόσφατη συγκριτική έρευνα, όπου συγκρίνονται τα παραδοσιακά και τα ηλεκτρονικά χαρτοφυλάκια, αποφαίνεται ότι η χρήση ηλεκτρονικού χαρτοφυλακίου οδηγεί σε καλύτερα μαθησιακά αποτελέσματα [2]

ΥΠΟΚΕΦΑΛΑΙΟ 2.1

Τύποι e-portfolio

Υπάρχουν τρεις βασικές μορφές ενός e-χαρτοφυλάκιο:

- αναπτυξιακή μορφή
- αντανακλαστική μορφή
- αναπαραστατική μορφή

Ένα αναπτυξιακό e-portfolio είναι μια αποτύπωση των πεπραγμένων του ατόμου στο οποίο αναφέρεται σε ένα συγκεκριμένο χρονικό διάστημα, και μπορεί να είναι άμεσα συνδεδεμένο με τα αποτελέσματα του ως μαθητή. Ένα αντανακλαστικό ηλεκτρονικό χαρτοφυλάκιο περιλαμβάνει τις προσωπικές απόψεις και προβληματισμούς σχετικά με το περιεχόμενο και το τι σημαίνει για την ανάπτυξη του ατόμου. Ένα αναπαραστατικό e-χαρτοφυλάκιο δείχνει τα επιτεύγματα του ατόμου στο οποίο αναφέρεται σε συγκεκριμένα έργα ή αναπτυξιακούς στόχους και είναι, συνεπώς, επιλεκτικό. Όταν χρησιμοποιείται κατά την αναζήτηση εργασίας αποκαλείται και Επαγγελματικό χαρτοφυλάκιο, ή χαρτοφυλάκιο σταδιοδρομίας .

Ασφαλώς, ορισμένα σύγχρονα ηλεκτρονικά χαρτοφυλάκια είναι δυνατό να περιλαμβάνουν περισσότερες από μια μορφές, ενώ ο κάτοχος έχει τη δυνατότητα να καθορίζει τις δυνατότητες και τα επίπεδα πρόσβασης.

ΥΠΟΚΕΦΑΛΑΙΟ 2.2

Χρήση ενός ηλεκτρονικού χαρτοφυλακίου

Σήμερα, τα ηλεκτρονικά χαρτοφυλάκια έχουν αυξημένη δημοτικότητα στους ακόλουθους κλάδους:

- Δευτεροβάθμια εκπαίδευση
- Τριτοβάθμια εκπαίδευση
- Συνεχής επαγγελματική ανάπτυξη
- Αιτήσεις εργασίας
- Ομάδες θεραπείας

- Διαπίστευση
- Αξιολόγηση
- Αναγνώριση Κεκτημένων Γνώσεων

Ορισμένα ηλεκτρονικά χαρτοφυλάκια έχουν τη δυνατότητα να χρησιμοποιηθούν για παρουσιάσεις, έναν αριθμό διαφορετικών εργασιών και πιο ευρέως, για ηλεκτρονικές τάξεις. Άλλοι μπορεί να χρησιμοποιηθεί μέσα σε ένα περιβάλλον εκπαίδευσης για την αξιολόγηση και την πιστοποίηση, όπως η θεσμική ηλεκτρονικό χαρτοφυλάκιο.

ΚΕΦΑΛΑΙΟ 3

Προϋπάρχουσες εφαρμογές Ηλεκτρονικού Χαρτοφυλακίου

Το E-PEARL [2] είναι ένα ηλεκτρονικό χαρτοφυλάκιο που δημιουργήθηκε από το Centre for the Study of Learning and Performance του Καναδά, από το απόσταγμα της σχετικής έρευνας του Ινστιτούτου, αξιοποιώντας παράλληλα τις απόψεις, τα σχόλια και τις προτάσεις των χρηστών ώστε να κερδίσει τα χαρακτηριστικά της διαρκούς ανάπτυξης και βελτίωσης. Στο κεφάλαιο αυτό εξετάζουμε τις βασικές λειτουργίες του e-Pearl και παρουσιάζουμε τη χρήση του από μαθητές και καθηγητές για διάφορα εκπαιδευτικά επίπεδα.

ΥΠΟΚΕΦΑΛΑΙΟ 3.1

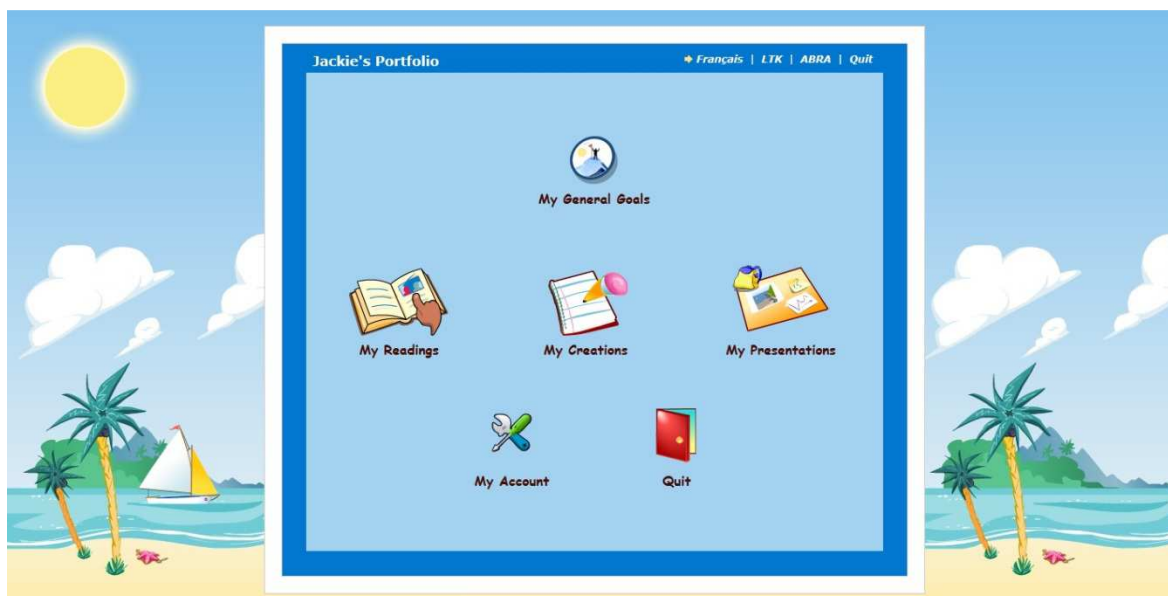
Πρωτοβάθμια εκπαίδευση

Στο σενάριο χρήσης του ePearl, η Jackie είναι μια μαθήτρια ενός καινοτόμου δημοτικού σχολείου, το οποίο έχει εισάγει τη χρήση των υπολογιστών και του διαδικτύου στην μαθησιακή διαδικασία.

ΥΠΟΚΕΦΑΛΑΙΟ 3.1.1





























Jackie - Μαθήτρια Δημοτικού

Η Jackie, από τις πρώτες ημέρες στο σχολείο, μαθαίνει για το βοηθητικό εργαλείο Στην εικόνα παρουσιάζεται η αρχική οθόνη του χαρτοφυλακίου της, ενώ στις εικόνες και , παρουσιάζονται οι οθόνες που σχετίζονται με τους διδακτικούς στόχους και τις εργασίες των μαθητών του.



Εικόνα 1: Η αρχική οθόνη του e-Pearl

Το πρώτο σημείο που πρέπει να παρατηρήσουμε είναι η σχεδιαστική προσέγγιση του όλου εγχειρήματος. Τόσο οι σχεδιαστικές επιλογές όσο και τα επιλεγμένα γραφικά έχουν τον κατάλληλο χαρακτήρα ώστε να δελεάζουν το κοινό στο οποίο απευθύνονται (μαθητές δημοτικού), ενώ οι λειτουργίες χρησιμοποιούν μεγάλα, ευανάγνωστα και αυτοεπεξηγηματικά γραφικά πλήκτρα.

close window	
General Goals 	
	
Mes parents vont-ils connaître mes buts	Updated 06/10/11  
Je veux comprendre ce que je lis.	Updated 08/24/10  
to learn to use epearl to learn to read	Updated 04/04/10  
to use epearl	Updated 02/25/10  
Je voudrais maîtriser ePearl.	Updated 02/15/10  
Décider de l'image que je souhaite donner au travers de mon ePortFolio	Updated 01/27/10  
rit fl nam	Updated 01/22/10  
Je veux savoir écrire au passé composé	Updated 12/23/09  
i wan 2 lrn ol lts ov albet	Updated 12/10/09  
I wanna spl 20 wds rit.	Updated 12/04/09  
Je veux lire les jours de la semaine	Updated 11/15/09  
Je veux lire mon prénom	Updated 11/15/09  
Je veux lire les lettres de l'alphabet	Updated 11/15/09  

Εικόνα 2: Οι διδακτικοί στόχοι

Με τη χρήση του portfolio απλοποιούνται σημαντικές εκπαιδευτικές διεργασίες, ενώ ταυτόχρονα εξοικειώνονται οι μαθητές με τη χρήση των τεχνολογιών διαδικτύου από μικρή ηλικία. Τα χαρτοφυλακίων στην πορεία ενός μαθητή είναι εξαιρετικά σημαντικά καθώς προσφέρουν τόσο στους εκπαιδευτικούς, όσο και στους γονείς, απτές αποδείξεις της πορείας του μαθητή και των γνώσεων που έχει αποκτήσει. Η δυνατότητα πρόσβασης στις εργασίες του, η online αποτύπωση των δυσκολιών και των επιτευγμάτων του αποδεικνύονται σημαντικά για τους εκπαιδευτικούς.

Text Title			Date ▾
		Counting to Ten	10/21/10
		I Can Move Like a ...	11/18/10
		The Three Billy Goats Gruff	12/01/10
		The Little Red Hen	01/13/11
		Untitled	02/08/11
		Untitled	02/15/11
		Essai vendée	03/08/11
		Untitled	03/08/11
		Untitled	03/08/11
		Untitled	03/08/11
		Untitled	03/31/11
		Untitled	03/31/11
		Untitled	04/30/11
		Les trois boucs	04/30/11

Εικόνα 3: Μαθησιακό υλικό για την Jackie

The screenshot shows a web application interface for 'Gia's Portfolio'. At the top, there is a navigation bar with 'Français | LTK | ABRA | Quit'. Below the navigation bar is a colorful illustration of a beach scene with a panda, a girl, and a yellow cat. The main content area is titled 'My Readings' and features a table with the following data:

Text Title			Date ▾
		Race, Identity, and Representation in Education	03/09/09
		The Little Red Hen	11/29/10
		The Frogs and the Well	02/25/11
		Les trois boucs	03/09/11
		La grande aventure de Telma et Zoé	05/18/11
		La petite poule rousse	05/19/11
		Les trois boucs	05/19/11
		La petite poule rousse	05/20/11
		La petite poule rousse	05/24/11
		La petite poule rousse	05/24/11
		La grande aventure de Telma et Zoé	06/09/11
		Untitled	07/04/11

Εικόνα 4: Οι εργασίες

ΚΕΦΑΛΑΙΟ 4

Σχεδιασμός Εφαρμογής Ηλεκτρονικού Χαρτοφυλακίου

Η εφαρμογή που θα αναπτύξουμε στα πλαίσια της παρούσας πτυχιακής εργασίας αφορά ένα δικτυακό ηλεκτρονικό χαρτοφυλάκιο για τις ανάγκες του ΤΕΙ Θεσσαλονίκης.

Σε πρώτο στάδιο θα πραγματοποιήσουμε την καταγραφή των απαιτήσεων της εφαρμογής. Στη συνέχεια θα σχεδιάσουμε το σχήμα της βάση δεδομένων και θα καθορίσουμε τα βασικά δομικά στοιχεία της εφαρμογής. Τα υπόλοιπα στοιχεία θα καθοριστούν κατά την ανάπτυξη της εφαρμογής.

ΥΠΟΚΕΦΑΛΑΙΟ 4.1

Απαιτήσεις της εφαρμογής

Στο πρώτο στάδιο αναλύουμε τις απαιτήσεις της εφαρμογής. Η αρχική γνώση που διαθέτουμε είναι η ακόλουθη: Η εφαρμογή του ηλεκτρονικού χαρτοφυλακίου έχει δύο βασικούς χρήστες: τον καθηγητή και τον φοιτητή. Ο φοιτητής θα έχει τη δυνατότητα να δημιουργεί το δικό του χαρτοφυλάκιο, «ανεβάζοντας» τις εργασίες του καθώς και άλλο υλικό για τα μαθήματα που έχει δηλώσει/παρακολουθεί. Το υλικό θα είναι διαχωρισμένο ανά φοιτητή και ανά μάθημα. Ο καθηγητής θα έχει δικαίωμα να βλέπει τα αντικείμενα στο portfolio του κάθε φοιτητή και να δημιουργεί σχόλια σχετικά με αυτά. Η απόκριση του συστήματος πρέπει να είναι άμεση.

Προσπαθώντας να αποκωδικοποιήσουμε την αρχική περιγραφή, αρχικά διακρίνουμε τους βασικούς ρόλους της εφαρμογής (φοιτητής και καθηγητής), ενώ στη συνέχεια διακρίνουμε άμεσα τις ενέργειες ανέβασμα υλικού (φοιτητής) και εισαγωγή σχολίου (καθηγητής). Για να είναι εφικτό ο κάθε χρήστης να πραγματοποιεί διαφορετικές ενέργειες ανάλογα την ιδιότητά του, πρέπει να υπάρξει ένα σύστημα αυθεντικοποίησης των χρηστών μαζί με ένα σύστημα περιορισμού της πρόσβασης ανάλογα την ιδιότητά του. Έτσι, προστίθενται επιπλέον οι λειτουργίες της εγγραφής χρήστη και της επεξεργασίας των πληροφοριών του.

Επιπρόσθετα, επειδή τα αντικείμενα θα προβάλλονται ανά μάθημα, ο φοιτητής πρέπει να έχει τη δυνατότητα να επιλέγει τη σχολή την οποία φοιτά, και να προσθέτει τα μαθήματα τα οποία θέλει να συμπεριλάβει στο χαρτοφυλάκιο του. Έτσι, επιπλέον λειτουργίες που πρέπει να διαθέτει η εφαρμογή είναι η προσθήκη τμήματος (καθηγητής), προσθήκη μαθημάτων (καθηγητής), επιλογή τμήματος (φοιτητής) και επιλογή μαθημάτων (φοιτητής).

ΥΠΟΚΕΦΑΛΑΙΟ 4.2

Σχεδιασμός βάσης δεδομένων

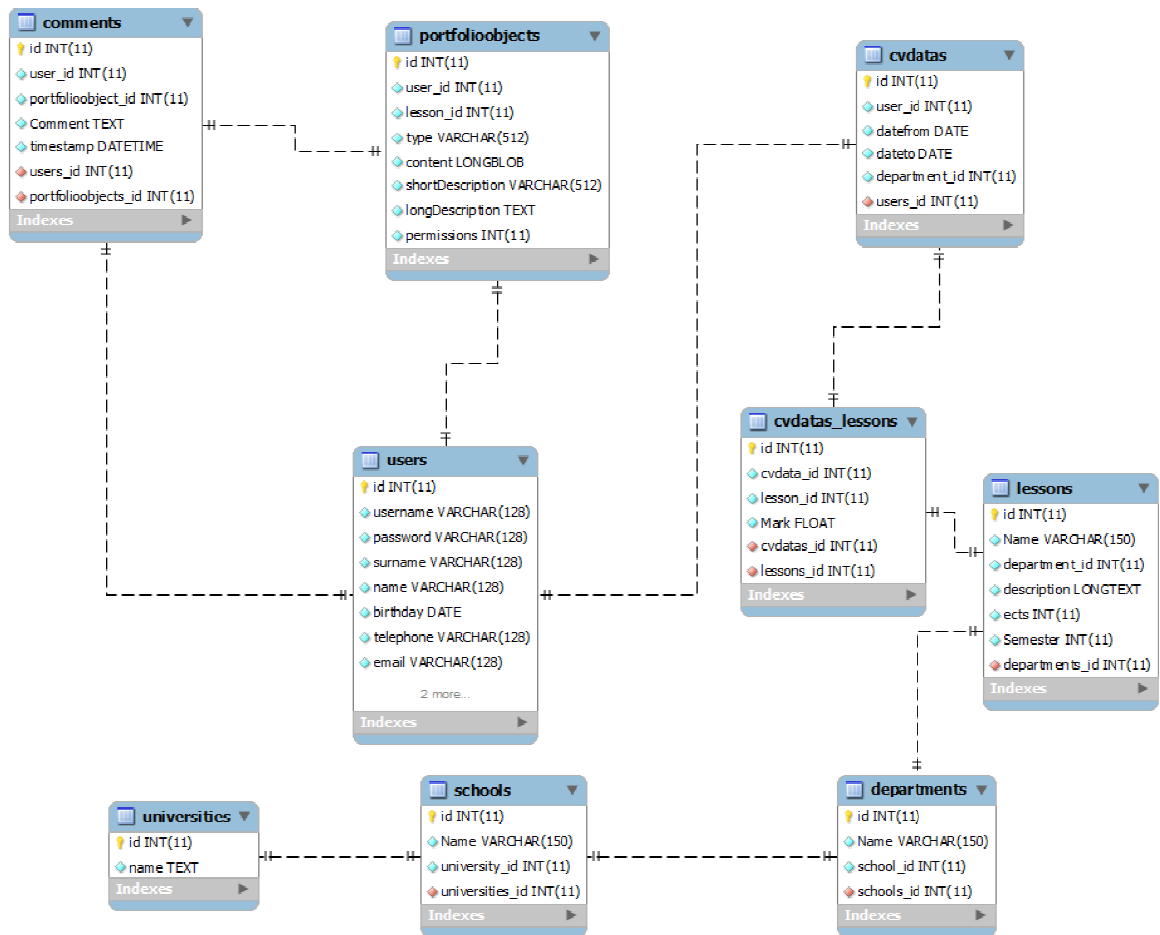
Από τις παραπάνω απαιτήσεις καταλήγουμε στο ακόλουθο σχήμα της βάσης δεδομένων.

Οι πίνακες είναι οι εξής:

1. Users: Ο πίνακας με τα βασικά στοιχεία κάθε χρήστη
2. Cnvdats: Ο πίνακας που περιλαμβάνει τη διασύνδεση του κάθε φοιτητή με το Τμήμα στο οποίο φοιτά και αποθηκεύει την περίοδο στην οποία ήταν ο φοιτητής ενεργός. Επιπρόσθετα έχει διασύνδεση πολλά προς πολλά με τον πίνακα μαθημάτων
3. Universities: Ο πίνακας περιλαμβάνει τα πανεπιστήμια/ΤΕΙ που χρησιμοποιούν την εφαρμογή
4. Schools: Ο πίνακας που περιλαμβάνει τις σχολές κάθε εγγραφής του πίνακα Universities
5. Departments: Ο πίνακας που περιλαμβάνει τα τμήματα της κάθε σχολής.
6. Lessons: Ο πίνακας περιλαμβάνει τα μαθήματα του τμήματος
7. Portfolioobjects: Εδώ αποθηκεύονται τα αντικείμενα του ηλεκτρονικού χαρτοφυλακίου
8. Comments: Ο πίνακας των σχολίων του κάθε αντικειμένου

Για την διευκόλυνση της λειτουργίας της εφαρμογής πραγματοποιήθηκαν οι κατάλληλες συνδέσεις μεταξύ των πινάκων, όπως φαίνεται στο ακόλουθο σχήμα.

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου



Εικόνα 5: Το σχήμα της βάσης δεδομένων

ΥΠΟΚΕΦΑΛΑΙΟ 4.3

Σχεδιασμός Εφαρμογής

Το τελευταίο στάδιο του σχεδιασμού αφορά την διάκριση των βασικών τμημάτων λογισμικού που θα περιλαμβάνει η εφαρμογή. Αυτές είναι οι ακόλουθες:

- Διαχείριση Χρήστη
- Διαχείριση Μαθημάτων Τμήματος
- Διαχείριση αντικειμένων Χρήστη και σχολίων

Ακολουθώντας τις επιταγές της MVC αντιμετώπισης και του CakePHP framework, σχεδιάζουμε τις βασικές κλάσεις (Χρήστης, Μαθήματα, Αντικείμενα Portfolio), καθώς και τις βοηθητικές κλάσεις (Ίδρυμα, Σχολή, Βιογραφικό αντικείμενο, Σχόλιο). Οι ιδιότητες της κάθε κλάσης συνάδουν με τα πεδία των αντίστοιχων πινάκων της βάσης δεδομένων. Οι ενέργειες που ορίζουμε για την εφαρμογή μας είναι οι ακόλουθες:

1. Εγγραφή χρήστη
2. Αυθεντικοποίηση Χρήστη
3. Έξοδος από το σύστημα
4. Προσθήκη Βιογραφικού στοιχείου
5. Επεξεργασία Βιογραφικού στοιχείου
6. Προσθήκη Μαθημάτων σε Βιογραφικό στοιχείο
7. Επεξεργασία Μαθημάτων σε Βιογραφικό στοιχείο
8. Προσθήκη Αντικειμένων Portfolio σε Μάθημα
9. Επεξεργασία Αντικειμένων Portfolio
10. Εισαγωγή σχολίων
11. Αναζήτηση χρήστη

ΚΕΦΑΛΑΙΟ 5

Εργαλεία Υλοποίησης του Ηλεκτρονικού Χαρτοφυλακίου

Για την υλοποίηση της εφαρμογής που σχεδιάσαμε στο προηγούμενο κεφάλαιο, χρησιμοποιήσαμε την πλατφόρμα ανάπτυξης CakePHP [4], η οποία λειτουργεί σε έναν Web server Apache [5] που έχει τη δυνατότητα εκτέλεσης PHP [6], και υποστηρίζεται από μια βάση δεδομένων MySQL [6]. Πριν προχωρήσουμε στις λεπτομέρειες της ανάπτυξης, θα παρουσιάσουμε συνοπτικά τα Apache, PHP, MySQL και CakePHP.

ΥΠΟΚΕΦΑΛΑΙΟ 5.1

Εξυπηρετητής δικτυακών εφαρμογών Apache

Ο Apache HTTP Server (συνήθως αποκαλείται απλώς Apache) [5], είναι ένα σύγχρονο λογισμικό το οποίο λειτουργεί σαν εξυπηρετητής δικτυακών εφαρμογών (web server) και αποτελεί από το 2009, τον πρώτο Web Server ο οποίος ξεπέρασε τον αριθμό ρεκόρ των 100 εκατομμυρίων ενεργών εγκαταστάσεων. Ο Apache αποτέλεσε την πρώτη εναλλακτική λύση στον Netscape Communications Corporation web server, ο οποίος ήταν μια από τις καθιερωμένες λύσεις της εποχής, και από τότε έχει εξελιχθεί σε αξιοσημείωτο βαθμό έτσι ώστε να επικρατήσει του ανταγωνισμού, τόσο από άποψη λειτουργικότητας και όσο και απόδοσης. Ο Apache μπορεί να εκτελεστεί στην συντριπτική πλειονότητα από τα υπάρχοντα λειτουργικά συστήματα όπως: Unix, FreeBSD, Linux, Solaris, Novell NetWare, AmigaOS, το Mac OS X, Microsoft Windows, OS/2, XAT, και eComStation.

Το μεγαλύτερο συγκριτικό πλεονέκτημα στην ανάπτυξη και την καθιέρωση του Apache ως defacto web server, αποτελεί το γεγονός ότι αναπτύσσεται και συντηρείται σύμφωνα με τις αρχές του ανοιχτού λογισμικού και λογισμικού ανοιχτού κώδικα, από μια ανοιχτή κοινότητα προγραμματιστών, υπό την αιγίδα του Ιδρύματος Apache. Ο κώδικας του Apache βασίστηκε αρχικά στον προϋπάρχοντα κώδικα του NCSA HTTPd, ο οποίος όμως στην πορεία της εξέλιξη του εξυπηρετητή έχει αφαιρεθεί.

Μια άλλη εξίσου σημαντική παράμετρος που οδήγησε στην καθιέρωση του Apache, αποτελεί ο μεγάλος αριθμός των χαρακτηριστικών που υποστηρίζει. Η αρχιτεκτονική σχεδίαση του Apache έχει λάβει υπόψη της την επεκτασιμότητα, και επιτρέπει την προσθήκη νέων χαρακτηριστικών στην βασική εφαρμογή είτε κατά το «χτίσιμο» του εξυπηρετητή (build) είτε μετέπειτα με την μορφή προσθέτων. Αυτές κυμαίνονται από υποστήριξη server-side γλώσσων προγραμματισμού έως υποστήριξη για συστήματα ελέγχου ταυτότητας χρηστών. Μερικές πολύ γνωστές επεκτάσεις του Apache προσφέρουν υποστήριξη στις γλώσσες Perl, Python, Tcl και PHP. Δημοφιλές συστήματα αυθεντικοποίησης και ελέγχου πρόσβασης είναι τα mod_access, mod_auth, mod_digest, και mod_auth_digest, ο διάδοχος του mod_digest. Ένα δείγμα από τα άλλα χαρακτηριστικά είναι οι ασφαλείς συνδέσεις Secure Sockets Layer (mod_ssl), η μονάδα ενδιάμεσου εξυπηρετητή (mod_proxy), οι δυνατότητες επανεγγραφής URL (mod_rewrite), προσαρμοσμένα αρχεία καταγραφής (mod_log_config), και η υποστήριξη φίλτρων (mod_include και mod_ext_filter).

Επιπρόσθετα, ο Apache περιλαμβάνει την εξωτερική μονάδα συμπίεσης, mod_gzip, με σκοπό να βοηθήσουν στη μείωση του μεγέθους των δεδομένων ιστοσελίδων και την μονάδα ModSecurity, μια μηχανή πρόληψης και ανίχνευσης εισβολών για την προστασία των διαδικτυακών εφαρμογών. Η δυνατότητα Virtual hosting επιτρέπει σε μία εφαρμογή Apache να εξυπηρετεί ταυτόχρονα πολλές διαφορετικές ιστοσελίδες.

Παρά το γεγονός ότι ο κύριος στόχος του σχεδιασμού του Apache δεν είναι να είναι ο "ταχύτερος" web server, ο Apache παρουσιάζει επιδόσεις παρόμοιες με άλλους "υψηλής απόδοσης" web servers, καθώς αξιοποιεί τις δυνατότητες της πολυνηματικής επεξεργασίας.

ΥΠΟΚΕΦΑΛΑΙΟ 5.2

Γλώσσα Προγραμματισμού PHP

Η PHP [6] είναι μια γενικής χρήσης server-side scripting γλώσσα προγραμματισμού η οποία έχει αρχικά σχεδιαστεί για την ανάπτυξη δικτυακών εφαρμογών και την παραγωγή δυναμικών ιστοσελίδων και αποτελεί μια από τις πρώτες του είδους της. Οι server-side scripting γλώσσες ενσωματώνονται σε ένα HTML αρχείο και ο κώδικας ερμηνεύεται από τον Apache ο οποίος διαθέτει μια μονάδα-επέκταση για τον χειρισμό των αρχείων PHP. Η εκτέλεση του κώδικα PHP παράγει την τελική ιστοσελίδα η οποία αποστέλλεται στο χρήστη. Η PHP μπορεί να εγκατασταθεί σχεδόν στην πλειονότητα των διακομιστών Web, όπως επίσης και με τη μορφή της αυτόνομης γλώσσας σχεδόν σε κάθε λειτουργικό σύστημα. Οι δύο κυριότεροι ανταγωνιστές της PHP είναι οι Active Server Pages-ASP (Microsoft) και Java Server Pages (JSP). Το βασικό πλεονέκτημα της PHP έναντι τους εντοπίζεται στην άδεια χρήσης και στον περιορισμένο αριθμό από server που μπορούν να εγκατασταθούν καθώς απαιτούν εξειδικευμένο server (IIS και Tomcat για ASP και JSP αντίστοιχα).

Η PHP δημιουργήθηκε αρχικά από Rasmus Lerdorf το 1995 και πλέον παράγεται και συντηρείται από τον Όμιλο της PHP. Η PHP είναι ελεύθερο λογισμικό που διατίθεται βάσει της Άδειας PHP, η οποία είναι ασυμβίβαστη με την GNU Γενικής Δημόσιας Άδειας (GPL), λόγω των περιορισμών στη χρήση του όρου PHP.

Το συντακτικό της γλώσσας PHP θυμίζει αρκετά το συντακτικό των γλωσσών C/C++ και Java. Σημαντικός περιορισμός, στις παλαιότερες εκδόσεις της γλώσσας PHP, αποτέλεσε το γεγονός ότι στερούνταν των αρχών του αντικειμενοστραφούς προγραμματισμού. Από την έκδοση 3, αρχικά, και, κυρίως, με την 5^η έκδοση της γλώσσας, ο περιορισμός αυτός έχει αρθεί, καθώς πλέον η PHP γλώσσα μπορεί να θεωρηθεί μια αντικειμενοστραφής γλώσσα. Παρόλα αυτά, ακόμη επιτρέπεται η σύνταξη κώδικα με τον παλιό interpreted τρόπο.

Ο κώδικας PHP εσωκλείεται μέσα σε αρχεία HTML, ξεκινώντας με τη σήμανση `<?php` και ολοκληρώνεται με τη σήμανση `?>`. Ο compiler της PHP εκτελεί μόνο τον κώδικα ο οποίος περικλείεται από την παραπάνω σήμανση και αγνοεί τα υπόλοιπα

τμήματα. Τα τμήματα του κώδικα PHP αντικαθίστανται από το αποτέλεσμα της εκτέλεσης του κώδικα από τον compiler. Ακολουθεί ένα αντιπροσωπευτικό παράδειγμα.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>PHP Test</title>
  </head>
  <body>
    <?php
    echo 'Hello World';
    ?>
  </body>
</html>
```

ΥΠΟΚΕΦΑΛΑΙΟ 5.3

Βάση Δεδομένων MySQL

Η MySQL [7] είναι το πιο συχνά χρησιμοποιούμενο σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (RDBMS) στον κόσμο. Λειτουργεί ως ένα διακομιστής που παρέχει ταυτόχρονη πρόσβαση σε πολλούς χρήστες στις βάσεις δεδομένων που εξυπηρετεί.

Η MySQL αναπτύσσεται με διπλή άδεια: είτε σαν εφαρμογή ανοιχτού κώδικα διαθέσιμη υπό τους όρους της GNU General Public License, είτε με εμπορική άδεια. Η MySQL ανήκε και χρηματοδοτείται από μια μη κερδοσκοπική εταιρεία, την σουηδική εταιρεία MySQL AB, η οποία εξαγοράστηκε από την Sun και πλέον ανήκει στην Oracle Corporation. Η MySQL χρησιμοποιείται σε πολλές υψηλού προφίλ, μεγάλης κλίμακας World Wide Web προϊόντα, συμπεριλαμβανομένου του Wikipedia, της Google (όχι για αναζητήσεις), του Facebook, και του Twitter.

Η MySQL έχει αναπτυχθεί στις γλώσσες C και C++ ενώ ο μεταφραστής SQL στο yacc. Η MySQL λειτουργεί σε πολλές και διαφορετικές πλατφόρμες, συμπεριλαμβανομένων των AIX , BSDi , FreeBSD , HP-UX , eComStation , i5/OS , IRIX , Linux , Mac OS X , Microsoft Windows , το NetBSD , Novell NetWare , OpenBSD , OpenSolaris , OS / 2 Warp, QNX , Solaris , Symbian , SunOS , OpenServer SCO , SCO UnixWare , Sanos και Tru64.

Πολλές γλώσσες προγραμματισμού με τη ειδικά APIs περιλαμβάνουν τις κατάλληλες βιβλιοθήκες για την πρόσβαση σε βάσεις δεδομένων MySQL, όπως ο MySQL Connector/Net για την παροχή διασύνδεση με το Visual Studio και του οδηγού JDBC για Java. Επιπλέον, υπάρχει και διασύνδεση για ODBC (διεπαφή MyODBC) η οποία επιτρέπει επιπλέον γλώσσες προγραμματισμού να υποστηρίξουν τη MySQL με χρήση διασύνδεσης ODBC. Ο διακομιστής MySQL και οι επίσημες βιβλιοθήκες υλοποιούνται κατά κανόνα σε ANSI C / ANSI C++.

Η MySQL χρησιμοποιεί το πρότυπο SQL. Η υποστήριξη του όμως, δεν είναι πλήρης και αρκετές παραβιάσεις του έχουν σημειωθεί, η σημαντικότερη εκ των οποίων είναι η απουσία υποστήριξης ξένων κλειδιών.

ΥΠΟΚΕΦΑΛΑΙΟ 5.4

Πλατφόρμα CakePHP

Η πλατφόρμα CakePHP [4] είναι μια δωρεάν, ανοιχτού κώδικα , πλατφόρμα ταχείας ανάπτυξης για την γλώσσα PHP. Είναι μια θεμελιώδης δομή που αξιοποιούν οι προγραμματιστές να δημιουργήσουν εφαρμογές web. Η πλατφόρμα παρέχει όλα τα εργαλεία που χρειάζεται ένας προγραμματιστής για να ξεκινήσει την κωδικοποίηση του κώδικα που πραγματικά χρειάζεται να κατασκευάσει: Τον κώδικα της λογική της συγκεκριμένης εφαρμογής. Περιλαμβάνει όλα τα βοηθητικά εκείνα τμήματα που απαιτούνται ώστε η ανάπτυξη της εφαρμογής να ολοκληρωθεί γρήγορα μετά τον σχεδιασμό της εφαρμογής.

Μια λίστα με τα χαρακτηριστικά της πλατφόρμας CakePHP είναι η ακόλουθη:

- Ενεργή και φιλική κοινότητα: Σημαντική παράμετρος στην χρησιμοποίηση ενός εργαλείου από έναν προγραμματιστή αποτελεί η ύπαρξη σημαντικής μερίδας χρηστών που είναι πρόθυμες να προσφέρουν υποστήριξη όποτε χρειαστεί. Η CakePHP διαθέτει μια από τις μεγαλύτερες κοινότητες χρηστών.
- Ευέλικτη άδεια χρήσης: Η άδεια χρήσης της CakePHP είναι η MIT: Ενώ και η MIT είναι ανοιχτού κώδικα όπως η GPL, δίνει τη δυνατότητα να ενσωματωθεί σε εμπορικές εφαρμογές κλειστού κώδικα
- Συμβατό με τις εκδόσεις 4 και 5 της PHP:
- Ενσωματωμένες δυνατότητες για την αλληλεπίδραση βάσεων δεδομένων: Η CakePHP προσφέρει διασύνδεση με όλες τις γνωστές βάσεις δεδομένων (MySQL, Microsoft SQL, Oracle, SQLite, Postgres, ODBC), με έναν ενιαίο τρόπο. Αυτό διευκολύνει την μεταφερσιμότητα των εφαρμογών μεταξύ διαφορετικών διακομιστών που χρησιμοποιούν (ενδεχομένως) διαφορετικά συστήματα βάσεων δεδομένων, με απλή αλλαγή του αρχείου ρυθμίσεων της πλατφόρμας. Επίσης προσφέρει ενιαίες εντολές για δημιουργία, ανάγνωση, ανανέωση και διαγραφή των εγγραφών από τη βάση δεδομένων.
- Αυτόματη δημιουργία κώδικα κοινών συναρτήσεων: Η πλατφόρμα διαθέτει εργαλείο το οποίο παράγει απλό κώδικα για το χειρισμό των

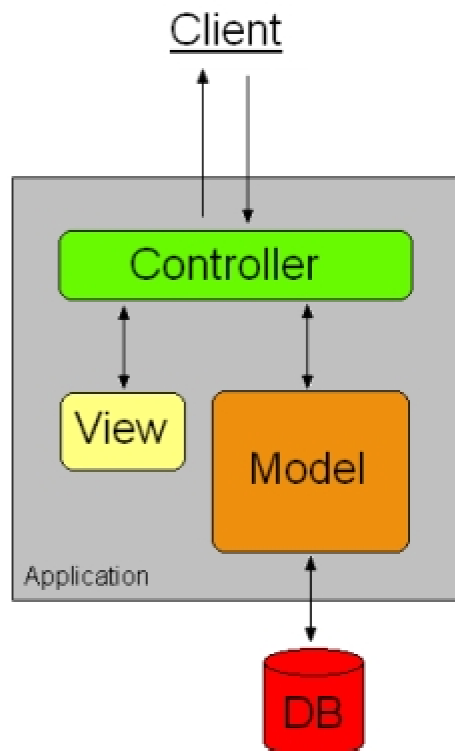
δεδομένων. Αυτή η διαδικασία μπορεί να επιταχύνει σημαντικά την ανάπτυξη της εφαρμογής, καθώς ο προγραμματιστής θα εστιάσει την προσοχή του στα τμήματα του κώδικα που αφορούν τη λογική του προγράμματος και όχι στις συνηθισμένες και τετριμμένες διαδικασίες

- MVC αρχιτεκτονική: Η πλατφόρμα χωρίζει τον κώδικα σύμφωνα με την αρχιτεκτονική MVC. Επειδή η MVC αρχιτεκτονική θα αξιοποιηθεί σημαντικά και αποτελεί σημαντική επιλογή, λεπτομέρειες σχετικά με αυτή θα αναφερθούν στη συνέχεια.
- Ενσωματωμένη επικύρωση των δεδομένων: Η επικύρωση των δεδομένων είναι πολύ σημαντική παράμετρος για τη σωστή λειτουργία της κάθε εφαρμογής

ΥΠΟΚΕΦΑΛΑΙΟ 5.5

Μεθοδολογία ανάπτυξης Μοντέλου - Όψης - Ελεγκτή

Το Model-View-Controller (MVC) είναι ένα αρχιτεκτονικό μοτίβο που χρησιμοποιείται στην τεχνολογία λογισμικού για το σχεδιασμό και την ανάπτυξη των εφαρμογών. Η τεχνική αυτή χρησιμοποιείται συχνά από εφαρμογές που απαιτούν τη δυνατότητα να παρουσιάζουν στο χρήστη πολλαπλές προβολές των ίδιων δεδομένων. Το MVC περιγράφηκε για πρώτη φορά το 1879 αρχικά για τις διεπαφές χρήστη σε Smalltalk από τον Trygve Reenskaug [1]. Στόχος του MVC είναι ένας ευέλικτος σχεδιασμός του προγράμματος που θα διευκολύνει τις μελλοντικές τροποποιήσεις ή επεκτάσεις του και θα δίνει την δυνατότητα επαναχρησιμοποίησης όλων ή ορισμένων εξαρτημάτων του. Μια γενική αναπαράσταση του τρόπου διεξαγωγής του σχεδιασμού MVC φαίνεται στην παρακάτω εικόνα:



Εικόνα 6: Το μοντέλο MVC

Το πρότυπο αυτό διαχωρίζει την εφαρμογή κατά την μοντελοποίηση, σε τρεις διαφορετικές διαφορετικές έννοιες οι οποίες περιγράφονται αναλυτικά στη συνέχεια:

- **Το μοντέλο (Model):** Το μοντέλο διαχειρίζεται την συμπεριφορά και τη διασύνδεση της εφαρμογής με τη βάση δεδομένων, ενώ ανταποκρίνεται σε αιτήματα για λήψη πληροφοριών σχετικά με την κατάσταση του (συνήθως από την όψη) και στις οδηγίες για αλλαγή των δεδομένων (συνήθως από τον ελεγκτή).
- **Την όψη/προβολή (View):** Η όψη/προβολή αναφέρεται στην αισθητική εμφάνιση των πληροφοριών.
- **Τον ελεγκτή (Controller):** Ο ελεγκτής ερμηνεύει τις εισόδους του χρήστη φροντίζοντας να ενημερώνει το μοντέλο και την προβολή ή και τα δυο μαζί για να αλλάξουν ανάλογα με την περίπτωση.

ΥΠΟΚΕΦΑΛΑΙΟ 5.5.1

Σχέσεις μεταξύ των συνιστωσών

Προβολή σχέση ελεγκτή

Στην παραδοσιακή μεθοδολογία Smalltalk MVC οι όψεις και ελεγκτές είναι στενά συνδεδεμένες. Κάθε όψη συνδέεται με ένα ενιαίο και μοναδικό ελεγκτή και αντιστρόφως. Ο ελεγκτής χρησιμοποιεί την όψη/προβολή για την είσοδο των δεδομένων, ενώ η όψη/προβολή είναι επίσης υπεύθυνη για τη δημιουργία νέων όψεων και των ελεγκτών. Είναι λογικό ότι οι όψεις και ελεγκτές είναι τμήματα που είναι στενά συνδεδεμένα μεταξύ τους όπως ακριβώς η είσοδος και έξοδος σε ένα υπολογιστικό σύστημα. Στις περισσότερες γραφικές διεπαφές (GUI) που έχουν υλοποιηθεί με βάση τις αρχές του MVC, η προβολή και ο ελεγκτής μπορούν απλά να συγχωνευτούν σε ένα αντικείμενο. Αυτό ονομάζεται προβολή εγγράφων.

Μοντέλο-Όψη

Η όψη εξαρτάται από το μοντέλο. Αλλαγές στη διασύνδεση μοντέλο απαιτεί παράλληλες αλλαγές στην όψη. Είναι πολύ δύσκολο να επιτευχθεί ένας αυστηρός διαχωρισμός μεταξύ του μοντέλου. Για παράδειγμα σκεφτείτε την απαίτηση “Εμφάνισε τα αρνητικά υπόλοιπα με κόκκινο χρώμα”. Με μια πρώτη ματιά αυτό

φαίνεται να είναι αυστηρά μια απαίτηση εξόδου και ένα τεστ που μπορεί να τοποθετηθεί στην προβολή σε περίπτωση αυτή την μορφή:

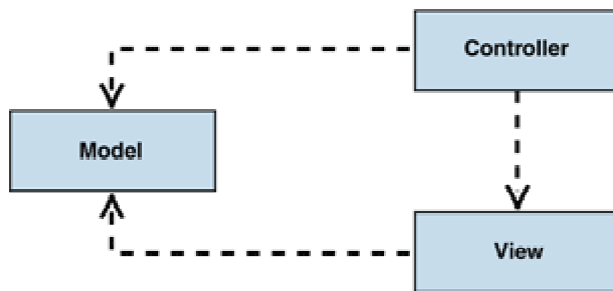
```
If balance < 0 then red
```

Αυτό θα παραβίαζε τον διαχωρισμό των περιορισμών του προτύπου MVC. Μετά από περαιτέρω ανάλυση αποδεικνύεται ότι η πραγματική απαίτηση είναι “εμφάνισε τα ακάλυπτα υπόλοιπα σε κόκκινο” και ο ορισμός του «ακάλυπτος = (ισοζύγιο < 0)» πρέπει να τοποθετηθεί στο μοντέλο.

Μοντέλο σχέση ελεγκτή

Ο υπεύθυνος επεξεργασίας ελεγκτής έχει άμεση εξάρτηση από το μοντέλο. Έτσι, αλλαγές στη διασύνδεση του μοντέλου μπορεί να προκαλέσει παράλληλες αλλαγές στον ελεγκτή.

Το Σχήμα 1 απεικονίζει τη μεταξύ τους δομική σχέση των τριών αυτών αντικειμένων.



Εικόνα 7: Η MVC δομή.

Πρέπει να σημειώσουμε στο σημείο αυτό ότι τόσο η όψη όσο και ο ελεγκτής εξαρτώνται από το μοντέλο. Ωστόσο το μοντέλο δεν εξαρτάται ούτε από την όψη ούτε από τον ελεγκτή και αυτό είναι ένα από τα βασικά πλεονεκτήματα του διαχωρισμού αυτού: επιτρέπει στο μοντέλο που θα κατασκευαστεί να δοκιμαστεί ανεξάρτητα από την οπτική παρουσίαση. Ο διαχωρισμός αυτός μεταξύ της όψης και του ελεγκτή δεν είναι αυστηρά καθορισμένος σε πολλές εφαρμογές επιτραπέζιων υπολογιστών. Από την άλλη πλευρά σε εφαρμογές Web ο διαχωρισμός μεταξύ της προβολής και του ελεγκτή είναι πολύ καλά καθορισμένος.

Το Model-View-Controller είναι ένα θεμελιώδες πρότυπο για τον διαχωρισμό της λογικής διεπαφής χρήστη από την λογική της εφαρμογής. Η δημοτικότητα του πρότυπου έχει οδηγήσει σε μια σειρά από διαφορετικές περιγραφές. Ειδικά ο όρος

“Ελεγκτής” έχει χρησιμοποιηθεί για να σημαίνει διαφορετικά πράγματα σε διαφορετικά πλαίσια. Ευτυχώς οι εφαρμογές Web έχουν βοηθήσει στην επίλυση ορισμένων ασαφειών, διότι ο διαχωρισμός μεταξύ της όψης και του ελεγκτή είναι πρόδηλη.

ΥΠΟΚΕΦΑΛΑΙΟ 5.5.2

Πλεονεκτήματα από τη χρήση του MVC

- **Υποστηρίζει πολλαπλές όψεις.** Επειδή η κάθε όψη είναι διαχωρισμένη από το μοντέλο και δεν υπάρχει άμεση εξάρτηση από αυτό, το γραφικό περιβάλλον που παρουσιάζεται στο χρήστη μπορεί να τροποποιείται πολύ απλά, χρησιμοποιώντας πολλαπλές όψεις των ίδιων δεδομένων. Για παράδειγμα πολλές σελίδες σε μια εφαρμογή Web μπορεί να χρησιμοποιούν το ίδιο μοντέλο αντικειμένων. Άλλο παράδειγμα είναι μια εφαρμογή Web που επιτρέπει στον χρήστη την αλλαγή στην εμφάνιση των σελίδων. Αυτές οι σελίδες εμφανίζουν τα ίδια στοιχεία από το κοινό μοντέλο αλλά τα δείχνουν με διαφορετικό τρόπο.
- **Επιτρέπει αλλαγές στις όψεις.** Οι απαιτήσεις της διεπαφής χρήστη τείνουν να αλλάζουν ταχύτερα από τη λογική της εφαρμογής. Οι χρήστες μπορούν να επιλέγουν διαφορετικά χρώματα, γραμματοσειρές, διάταξη της οθόνης, και τα επίπεδα στήριξης για τις νέες συσκευές όπως κινητά τηλέφωνα ή PDAs. Επειδή το μοντέλο δεν εξαρτάται από τις όψεις, προσθέτοντας νέους τύπους όψεων, η λειτουργικότητα του συστήματος δεν τροποποιείται.

ΥΠΟΚΕΦΑΛΑΙΟ 5.5.3

Μειονεκτήματα του MVC

- **Στενή διασύνδεση μεταξύ των απόψεων και των ελεγκτών για το μοντέλο:** Αλλαγές στη διασύνδεση μοντέλο ενδέχεται να προκαλέσουν παράλληλες αλλαγές στην όψη και μπορεί να απαιτηθούν πρόσθετες αλλαγές στον ελεγκτή.
- **Μεγάλη σύζευξη μεταξύ όψης και ελεγκτή:** Ο αυστηρός διαχωρισμός είναι δύσκολος αν όχι αδύνατος.

ΥΠΟΚΕΦΑΛΑΙΟ 5.5.4

Αρχιτεκτονική MVC – Πλατφόρμες MVC

Παρά το γεγονός ότι το MVC κατά κανόνα συνδέεται με πλατφόρμες είναι ουσιαστικά μια αρχιτεκτονική ανάπτυξης εφαρμογών. Αυτό σημαίνει ότι μπορεί να εφαρμοστεί ακόμη και χωρίς τη χρήση μια αντικειμενοστραφούς γλώσσας προγραμματισμού ή σε μιας συγκεκριμένης ιεραρχίας κλάσεων.

Η Smalltalk MVC εφαρμογή ενέπνευσε πολλές MVC πλατφόρμες ανάπτυξης τόσο παραδοσιακών εφαρμογών για επιτραπέζιους υπολογιστές όσο και πλατφόρμες για ανάπτυξη δικτυακών εφαρμογών. Μερικές από τις πιο γνωστές πλατφόρμες που υλοποιούν το πρότυπο MVC είναι οι ακόλουθες:

- AppFlower
- XPages
- GNUstep
- GTK+
- JFace
- Oracle Application
- Microsoft Foundation Class Library
- Microsoft ASP.NET
- Microsoft Composite Block

- Qt
- Java Swing
- Apache Pivot
- Adobe Flex
- WAVEMAKER
- CakePHP
- CodeIgniter
- Zend Framework

ΚΕΦΑΛΑΙΟ 6

Λεπτομέρειες Υλοποίησης του Ηλεκτρονικού Χαρτοφυλακίου

Σε πρώτο στάδιο, αναπτύσσεται η βάση δεδομένων. Στη συνέχεια, ακολουθεί η υλοποίηση μας στην CakePHP, ακολουθώντας τις προδιαγραφές της αρχιτεκτονικής MVC ξεκινώντας με την δημιουργία των Μοντέλων. Έτσι, δημιουργήθηκαν τα ακόλουθα μοντέλα:

1. Χρήστης
2. Ίδρυμα
3. Σχολή
4. Τμήμα
5. Μάθημα
6. Σχόλιο
7. Βιογραφικό αντικείμενο

Παρατηρούμε ότι δημιουργήθηκε ακριβώς ένα μοντέλο για κάθε βασικό πίνακα της βάσης δεδομένων (εκτός από τους συνδετικούς πίνακες).

Στη συνέχεια, προχωρήσαμε στη δημιουργία των ελεγκτών. Δημιουργήσαμε έναν ελεγκτή για κάθε μοντέλο. Όλοι οι ελεγκτές διαθέτουν τις βασικές λειτουργίες (προβολή, εισαγωγή, μεταβολή στοιχείων). Επιπλέον, υλοποιήθηκαν οι ενέργειες που έχουν προβλεφθεί κατά το σχεδιασμό της εφαρμογής. Αναλυτικά ο πλήρης κώδικας των ελεγκτών υπάρχει στο παράρτημα. Ο κάθε χρήστης έχει συγκεκριμένα δικαιώματα ανάλογα τον ρόλο του. Κατ' αρχήν διαθέτει απεριόριστα δικαιώματα στο προφίλ του, ενώ δεν έχει δικαίωμα να τροποποιήσει το προφίλ κάποιου άλλου χρήστη. Έχει δικαίωμα να βλέπει τα προφίλ όλων των χρηστών και, στην περίπτωση που είναι καθηγητής, μπορεί να εισάγει σχόλια στα αντικείμενα των χρηστών.

Ο χρήστης που έχει το ρόλο του καθηγητή δεν έχει δικαίωμα να έχει χαρτοφυλάκιο στο σύστημα, αλλά είναι ο μόνος που μπορεί να διαχειριστεί τα μαθήματα, τα τμήματα και τα ιδρύματα από τα οποία μπορεί να επιλέξει ο χρήστης.

Σημαντική σχεδιαστική εκλογή αποτελεί η χρήση των κατάλληλων πεδίων στην πλευρά του χρήστη, οι οποίες να ελαχιστοποιούν τις πιθανότητες εσφαλμένων

εισαγωγών. Έτσι, όπου αυτό είναι εφικτό, χρησιμοποιούνται drop-down λίστες (combo box).

Στο τέλος υλοποιήθηκαν οι όψεις. Οι όψεις αφορούν κυρίως το αισθητικό τμήμα της εφαρμογής και εμφανίζουν κατάλληλα τις πληροφορίες που αποστέλλονται από τον ελεγκτή. Σύμφωνα με τις καλές δικτυακές πρακτικές σχετικά με την ανάπτυξη ιστοσελίδων, προτείνεται η χρήση δύο ειδών αρχείων: Τα αρχεία δεδομένων (αρχεία HTML) που περιλαμβάνουν το σύνολο των δεδομένων χωρίς καμία είδους μορφοποίηση, και τα αρχεία μορφοποίησης (CSS), τα οποία περιλαμβάνουν όλες εκείνες τις εντολές μορφοποίησης. Ακολουθώντας αυτήν την καλή πρακτική, οι όψεις (οι οποίες παράγουν τον HTML κώδικα) περιλαμβάνουν τον απαραίτητο HTML κώδικα και έχουν κατάλληλες συνδέσεις με τα αρχεία μορφοποίησης.

Οι όψεις και τα αρχεία CSS που υλοποιήσαμε ακολουθούν κατά βάση την HTML4 και την CSS2, με χρήση όμως ορισμένων εξελεγμένων χαρακτηριστικών του νεότερου προτύπου CSS3. Για τον λόγο αυτό, απαιτούν την ύπαρξη σχετικά καινούριων browser που να τα υποστηρίζουν. Από τους ελέγχους που πραγματοποιήσαμε, η εφαρμογή παίζει χωρίς πρόβλημα στις τελευταίες εκδόσεις Firefox, Chrome, Opera, και Internet Explorer.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] J. Moon, Guide for Busy Academics No. 4: Learning through reflection, The Higher Education Academy.
- [2] M. v. Wesel και A. Prop, «The influence of Portfolio media on student perceptions and learning outcomes,» σε *Student Mobility and ICT: Can E-LEARNING overcome barriers of Life-Long learning Conference*, Maastricht, 2008.
- [3] E. Meyer, P. C. Abrami, A. Wade, O. Aslan και L. Deault, «Improving literacy and metacognition with electronic portfolios: Teaching and learning with ePEARL,» *Computers & Education*, τόμ. 55, αρ. 1, pp. 84-91, 2010.
- [4] «CakePHP Framework,» [Ηλεκτρονικό]. Available: <http://www.cakephp.org>.
- [5] «Apache HTTP Server Project,» 2012. [Ηλεκτρονικό]. Available: <http://httpd.apache.org/>.
- [6] «PHP,» 2012. [Ηλεκτρονικό]. Available: <http://www.php.net/>.
- [7] «MySQL Community Database,» Oracle, 2012. [Ηλεκτρονικό]. Available: <http://www.mysql.com/>.

ΠΑΡΑΡΤΗΜΑΤΑ

Κώδικας της εφαρμογής

Ο κώδικας του ελεγκτή του χρήστη

```
<?php
class UsersController extends ApplicationController {
    var $name = "Users";
    var $components = array('Auth', 'Session');
    var $helpers = array('Html', 'Form', 'Session', 'Javascript');
    var $uses = array('User', 'Cvdata', 'Lesson', 'CvdatasLesson');

    function beforeFilter() {
        parent :: beforeFilter();
        $this->Auth->allow('add', 'search', 'view', 'listPortfolios', 'viewrandom');
    }
    function index() {
        $this->redirect(array (
            'action' => 'view'
        ));
    }

    function add() {
        if (!empty ($this->data)) {
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
        if ($this->data['User']['password'] == $this->Auth->password($this-
>data['User']['password_confirm'])) {
            $this->User->create();
            if($this->User->save($this->data))
            {
                $this->Session->setFlash('Registration Successful','information');
                $this->redirect('/');
                die();
            }
            else
            {
                $this->Session->setFlash('The registration cannot be completed right now. Please try
again','error');
            }
        } else {
            $this->Session->setFlash('Please correct the errors','warning');
        }
    }
}
function isAuthorized() {
    if ($this->Session->read('Auth.User'))
        return true;
    return false;
}
function login() {
    if ($this->Session->read('Auth.User')) {
```


Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$this->Session->setFlash('You are logged in!','information');
$this->redirect(array (
    'action' => 'index'
));
}
}
function logout() {
    $this->Session->setFlash('Logout','information');
    $this->redirect($this->Auth->logout());
}

function view($uid = null) {

    $usr = $this->Session->read('Auth.User');
    $uid1 = $usr['id'];
    if($uid==null || $uid == $uid1)
    {
        if($usr['mode']!=0)
            $this->redirect(array('action'=>'browse'));
        $uid = $uid1;
        $usr = $this->User->find('all', array('recursive'=>4,'conditions' => array('User.id'=>$uid)) );
        $this->set('profile',$usr[0]);
        $this->set('readonly',false);
        $this->set('viewingUserId',-2);
    }
    else
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
{

    $usr = $this->User->find('all', array('recursive'=>4,'conditions' => array('User.id'=>$uid)) );
    $this->set('profile',$usr[0]);
    $this->set('readonly',true);
    $this->set('viewingUserId',-1);

}

}

function browse()
{
    $list = $this->User->find('all',array('conditions'=>array('mode'=>0)));
    $this->Session->write('list',$list);
    $this->redirect(array('action'=>'listPortfolios'));
}

function addPortfolioObject($cvdataid=null)
{
    if($cvdataid==null)
        $this->redirect(array('action'=>'view'));
    $this->set('myCvdatId',$cvdataid);
    $usr = $this->Session->read('Auth.User');
    $id = $usr['id'];
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$this->set('myUserId', $id);
$thisuser = $this->User->read(null, $id);
$thiscvdata = $this->Cvdata->read(null, $cvdataid);

//debug($thisuser);
//debug($thiscvdata);
$les = array();
foreach($thiscvdata['Lesson'] as $ls)
    $les[$ls['id']]=$ls['Name'];
$this->set('lessons', $les);
}

function addCvdata($param)
{
    $usr = $this->Session->read('Auth.User');
    $id = $usr['id'];
    $this->set('myUserId', $id);
    $this->set('param', $param);
}

function edit() {

    // ensure user is logged in
    $usr = $this->Session->read('Auth.User');
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$id = $usr['id'];
if ($this->isAuthorized()) {
    if (!empty ($this->data) && $id==$this->data['User']['id']) {

        if ($this->data['User']['password_change'] == $this->data['User']['password_confirm']) {
            $this->data['User']['password']=$this->data['User']['password_change'];
            $fieldList = array();
            $this->User->id=$id;
            if($this->data['User']['password']!="")
                $this->User->saveField('password', $this->data['User']['password']);

            $this->User->saveField('surname', $this->data['User']['surname']);
            $this->User->saveField('name', $this->data['User']['name']);
            $this->User->saveField('telephone', $this->data['User']['telephone']);
            $this->User->saveField('email', $this->data['User']['email']);
            $this->User->saveField('birthday', $this->data['User']['birthday']);
            $this->Session->setFlash('Your Account has been saved.', 'information');
            $this->redirect(array ('action' => 'index'));

        }
        else
            $this->Session->setFlash('Please correct the errors', 'warning');

    }
}
// display logged in user info to edit
$user = $this->User->read(NULL, $this->Auth->user('id'));
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
        $this->data = $user;
    }
}

function download($id)
{
    //debug($this->User);
    $usr = $this->User->portfolioobject->find('first', array('conditions' =>
array('portfolioobject.id'=>$id)) );
    Configure::write('debug', 0);
    //print_r($usr);
    header('Content-type: ' . $usr['portfolioobject']['type']);
    echo $usr['portfolioobject']['content'];
    //exit();
    //$this->set('file',$usr);

    // we'll use a new layout, file, that will allow custom headers
    $this->render(null,'usr');

}

function listPortfolios()
{
    $list = $this->Session->read('list');
    if(empty($list))
    {
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
        $this->Session->setFlash("No results has been found",'information');
    }
    $this->set('list',$list);
}

function search()
{
    if(!empty($this->data))
    {
        $terms = $this->data['User']['Criteria'];
        if(strlen($terms)<3)
        {
            $this->Session->setFlash("The query should have at least 3 characters",'warning');
        }
        else
        {
            $orcon = array();
            $i=0;
            $str = strtok($terms," ");
            do
            {
                if(strlen($str)<3)
                    continue;
                $orcon[$i] = array('User.Name like' => '%'.$str.'%');
                $i++;
                $orcon[$i] = array('User.Surname like' => '%'.$str.'%');
            }
            while($str = strtok($terms," "));
        }
    }
}
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
        $i++;
    }while(strtok(" "));
    if($i>0)
    {
        $conditions = array("OR"=>$orcon);
        $list = $this->User->find('all', array('conditions'=>$conditions));
        $this->Session->write('list',$list);
        //$log = $this->User->getDataSource()->getLog(false, false);
        //debug($log);

        //die();
        $this->redirect(array('action'=>'listPortfolios'));
        die();
    }
    else
    {
        $this->Session->setFlash('No term more with 3 characters or more has been
found', 'warning');
    }
}

function viewrandom()
{
    $list = $this->User->find('all');
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$max = count($list);
$i = rand(0,$max-1);

$this->redirect(array('action'=>'view',$list[$i]['User']['id']));
}

function addlesson($cvdataid=null)
{
    if(!empty($ths->data))
    {

    }
    if($cvdataid==null)
        $this->redirect(array('action'=>'view'));
    $userData = $this->Session->read('Auth.User');
    $theUser = $this->Cvdata->find('first',array('conditions'=>array('Cvdata.id'=>$cvdataid)));
    if($theUser['Cvdata']['user_id']!=$userData['id'])
    {
        $this->Session->setflash('You are not authorized to do this action','error');
        $this->redirect($this->referer());
        die();
    }
    // debug($theUser['Cvdata']);
}
```


Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$this->data['Cvdata']=$this->data['CvdatasLesson'];
if($theUser['Cvdata']['user_id']!=$userData['id'])
{
    $this->Session->setflash('You are not authorized to do this action','error');
    $this->redirect($this->referer());
    die();
}

$lessons=$this->Lesson-
>find('list',array('conditions'=>array('Department_id'=>$theUser['Cvdata']['department_id']));
    $this->set('cvid',$cvdataid);
    $this->set(compact('lessons'));
}

}
?>
```

Ο ελεγκτής των αντικειμένων του Ηλεκτρονικού χαρτοφυλακίου.

```
<?php
class PortfolioobjectsController extends ApplicationController {

    var $name = 'Portfolioobjects';
    var $components = array('Auth', 'Session');
    var $helpers = array('Html', 'Form', 'Session', 'Javascript');
    var $uses=array('Portfolioobject','User');
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
function beforeFilter()
{
    parent::basicStuff();
}

function index($lessonid=null,$uid=null) {
    if($lessonid==null)
        $this->redirect(array('controller'=>'users','action' => 'view'));
    //$this->Portfolioobject->recursive = 0;
    $usr = $this->Session->read('Auth.User');
    if($uid==null)
    {

        $uid = $usr['id'];
    }
    $conditions = array('lesson_id'=>$lessonid,'user_id'=>$uid);
    $returnedValues = $this->Portfolioobject->find('all',array('conditions'=>$conditions));
    $this->set('portfolioobjects',$returnedValues);
    $this->set('readonly',$uid!=$usr['id']);
}

function view($id = null) {
    if (!$id) {
        $this->Session->setFlash(__('Invalid portfolioobject', true));
    }
}
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
        $this->redirect(array('action' => 'index'));
    }
    $this->Portfolioobject->recursive = 1;
    $portobj = $this->Portfolioobject->read(null, $id);
    foreach($portobj['Comment'] as &$cm)
        $cm['User']=$this->User->read(null, $cm['user_id']);
    $this->set('portfolioobject', $portobj);
    $usr = $this->Session->read('Auth.User');
    $this->set('readonly', $portobj['Portfolioobject']['user_id']!= $usr['id']);
}

function add() {
    if (!empty($this->data)) {
        $this->Portfolioobject->create();
        if ($this->Portfolioobject->save($this->data)) {
            $this->Session->setFlash('The portfolioobject has been saved', 'information');
            $this->redirect(array('action' => 'index'));
        } else {
            $this->Session->setFlash('The portfolioobject could not be saved. Please, try again.',
'error');
        }
    }
    $users = $this->Portfolioobject->User->find('list');
    $lessons = $this->Portfolioobject->Lesson->find('list');
    $this->set(compact('users', 'lessons'));
```

```
}  
  
function edit($id = null) {  
    if (!$id && empty($this->data)) {  
        $this->Session->setFlash('Invalid portfolioobject', 'error');  
        $this->redirect(array('action' => 'index'));  
    }  
    if (!empty($this->data)) {  
        if ($this->Portfolioobject->save($this->data)) {  
            $this->Session->setFlash('The portfolioobject has been saved', 'information');  
            $this->redirect(array('action' => 'index'));  
        } else {  
            $this->Session->setFlash('The portfolioobject could not be saved. Please, try again.',  
'error');  
        }  
    }  
    if (empty($this->data)) {  
        $this->data = $this->Portfolioobject->read(null, $id);  
    }  
    $users = $this->Portfolioobject->User->find('list');  
    $lessons = $this->Portfolioobject->Lesson->find('list');  
    $this->set(compact('users', 'lessons'));  
}  
  
function delete($id = null) {  
    if (!$id) {
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$this->Session->setFlash(__('Invalid id for portfolioobject', true));
$this->redirect(array('action'=>'index'));
}
if ($this->Portfolioobject->delete($id)) {
    $this->Session->setFlash(__('Portfolioobject deleted', true));
    $this->redirect(array('action'=>'index'));
}
$this->Session->setFlash(__('Portfolioobject was not deleted', true));
$this->redirect(array('action' => 'index'));
}
function addPortfolioObject()
{
    //debug($this->data);
    if(!empty($this->data))
    {
        $this->data['Portfolioobject']['content']=fread(fopen($this->data['Portfolioobject']['File']['tmp_name'], "r"), $this->data['Portfolioobject']['File']['size']);
        $this->data['Portfolioobject']['type']=$this->data['Portfolioobject']['File']['type'];
        if($this->Portfolioobject->save($this->data))
        {
            $this->Session->setFlash('Portfolio object has been successfully saved','information');
            //$this->redirect(array ('controller'=>'Users','action' => 'view'));
        }
        else
        {
```

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

```
$this->Session->setFlash('Portfolio object has not been succefully saved. Please try
again.', 'error');

    }
    $this->redirect(array ('controller'=>'cvdatas', 'action' => 'view', $this-
>data['Portfolioobject']['cvdata_id'] ));
    }
}

function download($id)
{
    $usr = $this->Portfolioobject->find('first', array('conditions' => array('Portfolioobject.id'=>$id)) );
    //debug($usr);
    header('content-type: '.$usr['Portfolioobject']['type']);
    echo $usr['Portfolioobject']['content'];
    exit();
}
}
```

ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

Η χρήση της συγκεκριμένης εφαρμογής είναι παρόμοια με τη χρήση των περισσότερων Web Εφαρμογών. Ο χρήστης, κατά την είσοδό του, βρίσκεται στην ακόλουθη Εικόνα:

Αλεξάνδρειο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Θεσσαλονίκης

Home Random portfolio Register

Welcome to e-portfolio.
The e-portfolio application has been created as an final Thesis in the Department of Information Technology of ATEI of Thessaloniki
In this e-portfolio application, any user can register as a normal user, or as a professor of the Department
This page is for educational uses only

Search

Search

Login

Username

Password

Login

© 2011 | Alexander Technological Educational Institute of Thessaloniki | All Rights Reserved

Εικόνα 8: Η εισαγωγική οθόνη της εφαρμογής μας

Στο επάνω τμήμα υπάρχει το μενού πλοήγησης της εφαρμογής, το οποίο είναι διαφορετικό για τους εγγεγραμμένους χρήστες. Επίσης είναι διαφορετικό ανάλογα το είδος του χρήστη (Φοιτητής - Καθηγητής).

Στο δεξί τμήμα της εφαρμογής υπάρχουν δύο φόρμες: Η φόρμα αναζήτησης και η φόρμα εισόδου στο σύστημα. Από την φόρμα αναζήτησης ο χρήστης μπορεί να αναζητήσει το Portfolio του χρήστη που επιθυμεί. Στη φόρμα εισοδου, ο χρήστης μπορεί να εισάγει το όνομα χρήστη και τον κωδικό του, ώστε να μπορέσει να εισέλθει στο σύστημα.

Ο χρήστης μπορεί να εγγραφεί στο σύστημα από την εντολή Register επάνω δεξιά. Έτσι, εμφανίζεται η ακόλουθη εικόνα, όπου ο χρήστης μπορεί να δηλώσει τα στοιχεία του.

Αλεξάνδρειο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Θεσσαλονίκης

Home Random portfolio Register

Register

Username

Password

Password Confirm

Surname

Name

Email

Date of birth
12 - February - 1999

Telephone

Mode
Student

Submit

Search

Search

Login

Username

Password

Login

© 2011 | Alexander Technological Educational Institute of Thessaloniki | All Rights Reserved

Εικόνα 9: Η οθόνη εγγραφής νέου χρήστη

Ο χρήστης μπορεί να εγγραφεί είτε ως καθηγητής, είτε ως φοιτητής. Ο φοιτητής μπορεί να παρουσιάσει μια ολοκληρωμένη διαδικτυακή παρουσία για τον εαυτό του, εισάγοντας τα μαθήματα που έχει παρακολουθήσει ως εξής:

Πτυχιακή εργασία του φοιτητή Παράσχου Χρήστου

Αλεξάνδρειο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Θεσσαλονίκης

[Home](#)
[Random portfolio](#)
[Portfolio](#)
[Edit Profile](#)

Details

A/A	Title	ECTS	Mark	Actions
1	Μαθηματική Ανάλυση	6	5	
2	Ψηφιακά Συστήματα - Θεωρία	4	5	
3	Δεξιότητες Επικοινωνίας - Θεωρία	4	8	
4	Δεξιότητες Επικοινωνίας - Εργαστήριο	2	5	
5	Εισαγωγή στη Πληροφορική - Θεωρία	4	6	
6	Εισαγωγή στη Πληροφορική - Εργαστήριο	2	8	
7	Προγραμματισμός Η/Υ I - Θεωρία	4	5	
8	Προγραμματισμός Η/Υ I - Εργαστήριο	2	5	
9	Προγραμματισμός Η/Υ II - Θεωρία	4	5.6	
10	Προγραμματισμός Η/Υ II - Εργαστήριο	2	8	
11	Διακριτά Μαθηματικά	6	6	
12	Ψηφιακά Συστήματα - Εργαστήριο	2	7	
13	Οργάνωση και Αρχιτεκτονική Υπολ. Συστημάτων - Θεωρία	4	5.5	
14	Οργάνωση και Αρχιτεκτονική Υπολ. Συστημάτων - Εργαστήριο	2	5.5	
15	Αγγλική Ορολογία	6	6	
16	Οργάνωση-Διοίκηση Επιχειρήσεων και Οικονομία των Επιχειρήσεων	6	8	

Total Courses: 16 Average: 6.123333333333333
[Add new lesson](#)

Search

Welcome User christos

[View your portfolio](#)
[Logout](#)

© 2011 | Alexander Technological Educational Institute of Thessaloniki | All Rights Reserved

Ο χρήστης που έχει τα δικαιώματα του καθηγητή μπορεί να σχολιάσει τα αντικείμενα του ηλεκτρονικού χαρτοφυλακίου του χρήστη ως εξής:

The screenshot shows a web interface for a portfolio. At the top, there is a navigation bar with links: Home, Random portfolio, Admin, and Edit Profile. The main content area is divided into two columns. The left column, titled 'Portfolioobject', displays the following information: User: ΧΡΗΣΤΟΣ; Lesson: Ψηφιακά Συστήματα - Εργαστήριο; Title: Εργασία; Description: Επισυνάπτεω την εργασία που έκανα στο πρώτο εξάμηνο για το μάθημα των ψηφιακών συστημάτων. Κύριο γνωστικό πεδίο έχει τους πίνακες καρνό και απο όλα βλέπω οι περισσότερες εργασίες που σας βάζουν τα τελευταία χρόνια είναι με πίνακες καρνό. Below the description is a small green icon. The right column contains a 'Search' box with a search button and a 'Welcome User adam' section with links for 'View your portfolio' and 'Logout'. At the bottom of the main content area, there is a 'Related Comments' section with an 'Add Comment' form containing a text input field and a submit button. The footer of the page reads: © 2011 | Alexander Technological Educational Institute of Thessaloniki | All Rights Reserved.