

“Προσωποποίηση λειτουργικότητας και υποστήριξη φορητότητας, μέσω smartphones και tablets, για την πλατφόρμα εκπαίδευσης M.O.L.E (Multimedia Open Learning Environment) του Πολυτεχνείου Κρήτης.”

Μαθιουδάκης Θεόδωρος



Πολυτεχνείο Κρήτης  
Τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών

Μια εργασία που παρουσιάστηκε στο Πολυτεχνείο Κρήτης για την εκπλήρωση των απαιτήσεων απόκτησης Διπλώματος στο τμήμα Ηλεκτρονικών Μηχανικών & Μηχανικών Υπολογιστών





# Πίνακας Περιεχομένων

Περίληψη	8
Ευχαριστίες	9
Κεφάλαιο 1	11
Εισαγωγή	11
1.1 Εισαγωγή	12
1.2 Στόχοι της Εργασίας	12
1.3 Δομή της Εργασίας	12
Κεφάλαιο 2	15
Φορητές Συσκευές (Mobile devices)	15
2.1 Εισαγωγή	16
2.2 Προβλήματα του μέσου	16
2.3 Πλατφόρμες, Λειτουργικά Συστήματα.	18
2.4 Native Apps vs Mobile Web Apps	18
2.5 Η ανάγκη για καθολικότητα	19
Κεφάλαιο 3	21
Τεχνολογίες και Αρχιτεκτονική Υλοποίησης	21
3.1 Εισαγωγή	22
3.2 HTML5	22
3.3 Javascript	23
3.4 jQuery Mobile	23

3.5 Αντίστοιχα <i>frameworks</i> και γιατί επιλέξαμε το JQM	30
3.6 Servlets	31
3.7 Java Servlet Pages (JSP)	31
3.8 Data Access Object (DAO)	32
3.9 Το μοτίβο BCED	33
3.9.1 Boundary Layer	33
3.9.2 Control Layer	34
3.9.3 Entity Layer	35
3.9.4 DataBase Interface Layer	35
3.10 Εργαλεία Προγραμματισμού	36
3.10.1 NetBeans IDE 6.9.1	36
3.10.2 MySQL Connector	36
3.10.3 Navicat for MySQL	36
<b>Κεφάλαιο 4</b>	<b>39</b>
Ανάλυση απαιτήσεων	39
4.1 Εισαγωγή	40
4.2 Οι χρήστες του συστήματος	40
4.3 Επιλογή ενεργειών από το ήδη υπάρχον <i>Courses</i> που θα υποστηρίζονται	43
4.3.1 Λειτουργίες με επίκεντρο συγκεκριμένο μάθημα ( <i>Course - oriented tasks</i> )	43
4.3.2 Λειτουργίες με επίκεντρο τον χρήστη ( <i>User - oriented tasks</i> )	44
4.4 Λειτουργικότητα του συστήματος	45
4.5 Διαγράμματα περιπτώσεων χρήσης και συνοπτικός πίνακας	46

4.6 Αναλυτική περιγραφή Use Cases	49
-----------------------------------	----

## Κεφάλαιο 5 55

Γραφική Διεπαφή	55
-----------------	----

5.1 Εισαγωγή	56
--------------	----

5.2 Αρχές Σχεδίασης	56
---------------------	----

5.2.1 Γενικότερο πλαίσιο (Context)	56
------------------------------------	----

5.2.2 Αρχιτεκτονική της πληροφορίας (Information Architecture)	58
----------------------------------------------------------------	----

5.2.3 Οπτική Σχεδίαση (Visual Design)	68
---------------------------------------	----

5.3 Σχεδίαση για διαφορετικές οθόνες	69
--------------------------------------	----

5.4 Ευκολία πλοήγησης	72
-----------------------	----

5.5 Χειρισμός Λαθών	73
---------------------	----

5.6 Συνέπεια	73
--------------	----

5.7 Επιλογή χρωμάτων	74
----------------------	----

5.8 Διεπαφή του χρήστη	74
------------------------	----

5.8.1 Βασικά στοιχεία της διεπαφής	74
------------------------------------	----

5.8.2 Είσοδος στο σύστημα.	79
----------------------------	----

5.8.3 Προβολή πληροφοριών μαθήματος	80
-------------------------------------	----

5.8.4 Προβολή ανακοινώσεων μαθήματος	82
--------------------------------------	----

5.8.5 Πρόσβαση στο ψηφιακό υλικό του μαθήματος	83
------------------------------------------------	----

5.8.6 Προβολή και σχολιασμός σε συζήτηση μαθήματος	84
----------------------------------------------------	----

5.8.7 Δημιουργία νέας συζήτησης μαθήματος.	85
--------------------------------------------	----

5.8.8 Προβολή βαθμολογίας μαθήματος	86
-------------------------------------	----

5.8.9 Προβολή ημερολογίου μαθήματος	87
5.8.10 Εγγραφή σε εργαστήριο	88
5.8.10 Πρόσκληση ατόμων σε ομάδα εργασίας	89
5.8.10 Αποδοχή ή απόρριψη πρόσκλησης σε ομάδα εργασίας	90
5.8.11 Πρόσβαση στις εργαστηριακές ασκήσεις	91
5.8.12 Πρόσβαση στο ψηφιακό υλικό	92
5.8.13 Προβολή προσωπικού ημερολογίου του χρήστη και πλοήγηση	93
5.8.14 Προβολή εισερχομένων και εξερχομένων μηνυμάτων	94
5.8.15 Αποστολή νέου προσωπικού μηνύματος	95
5.8.16 Προβολή νέων ανακοινώσεων στα μαθήματα του χρήστη	96
5.8.17 Προβολή προθεσμιών παράδοσης εργασιών	97
5.8.18 Προβολή νέων σχολίων σε συζητήσεις μαθημάτων του χρήστη.	98
5.9 Παράδειγμα μεταφοράς σελίδων από το την Desktop έκδοση της εφαρμογής στην mobile έκδοση.	99

## Κεφάλαιο 6 105

Αξιολόγηση της Εφαρμογής στα στάδια της υλοποίησης	105
6.1 Εισαγωγή	106
6.2 Ευριστική Αξιολόγηση (Heuristic Evaluation)	106
6.3 Πρωτότυπα χαρτιού (Paper prototypes) & Walkthrough	108
6.4 Pluralistic walkthrough	110
6.5 Think Aloud Evaluation	129

## Κεφάλαιο 7 133

Επανασχεδιασμός της δομής και της γραφικής διεπαφής του M.O.L.E. για την υποστήριξη απαιτήσεων προσωποποίησης πρόσβασης.	133
--------------------------------------------------------------------------------------------------------------------------	-----

7.1 Εισαγωγή.	134
---------------	-----

7.2 Αλλαγές που απαιτούνται στη δομή του MOLE.	134
------------------------------------------------	-----

## Κεφάλαιο 8 149

Αναλυτική περιγραφή Βάσης Δεδομένων, Data Access Objects και Entity Model για την υποστήριξη της λειτουργικότητας των προσωποκεντρικών εφαρμογών στο M.O.L.E.	149
---------------------------------------------------------------------------------------------------------------------------------------------------------------	-----

8.1 Εισαγωγή	150
--------------	-----

8.2 Ανάλυση της Βάσης Δεδομένων του συστήματος μας.	150
-----------------------------------------------------	-----

8.3 Αναλυτική περιγραφή των κλάσεων της βάσης δεδομένων.	155
----------------------------------------------------------	-----

9.4 Αναλυτική περιγραφή των DAO	158
---------------------------------	-----

## Κεφάλαιο 9 169

Ανακεφαλαίωση - Μελλοντικές Επεκτάσεις	169
----------------------------------------	-----

10.1 Ανακεφαλαίωση	170
--------------------	-----

10.2 Μελλοντικές Επεκτάσεις	171
-----------------------------	-----

Βιβλιογραφία	172
--------------	-----

# Περίληψη

Διανύουμε μια εποχή όπου η εξέλιξη της τεχνολογίας και των υπολογιστών, τόσο από πλευράς υλικού όσο και λογισμικού, έχει οδηγήσει σε μια εντελώς διαφορετική αντίληψη, λειτουργία και συμπεριφορά, σχετικά με την διαχείριση της πληροφορίας και την πρόσβαση σε αυτήν. Οι υπολογιστές που ήταν, και παραμένουν, το παράθυρο μας στη παγκόσμια κλίμακα πληροφορία έχουν αλλάξει. Έχουν γίνει πιο γρήγοροι, πιο αποδοτικοί και πολύ πιο μικροί. Τόσο μικροί που χωράνε στην τσέπη μας, δίνοντας μας την δυνατότητα να έχουμε πρόσβαση στα αρχεία μας, στα mail μας και στο διαδίκτυο όπου και αν βρισκόμαστε.

Ακολουθώντας την παγκόσμια αυτή τάση, θεωρήσαμε αναγκαίο να έχουν οι φοιτητές του τμήματος μας την δυνατότητα πρόσβασης στον ήδη υπάρχοντα ιστοχώρο του τμήματος (Courses τώρα, MOLE σύντομα) , από τα smartphones τους. Με τον τρόπο αυτό θα μπορούν να έχουν καλύτερο και πιο άμεσο έλεγχο των όσων συμβαίνουν στο ίδρυμα και τους αφορούν. Σκοπός αυτής της διπλωματικής είναι η δημιουργία μιας εφαρμογής η οποία θα παρέχει την λειτουργικότητα του Courses σε κινητές συσκευές (smartphones και tablets). Το κύριο κριτήριο που θέσαμε είναι να παρέχεται η λειτουργικότητα που θα είναι κατάλληλη για οποιαδήποτε συσκευή, ανεξάρτητα από το μέγεθος της οθόνης και της πλατφόρμας της, με τεχνολογίες διαδικτύου και όχι με τεχνολογίες συγκεκριμένων mobile application platforms που είναι ιδιαίτερες για κάθε κατασκευαστή. Οι κινητές συσκευές έχουν επίσης τυπικά διαφορετικό μοντέλο αλληλοεπίδρασης με τους χρήστες. Το μοντέλο είναι βασισμένο σε οθόνες αφής, και όχι σε ποντίκι πράγμα που βάζει περιορισμούς για το μέγεθος των αντικειμένων αλληλοεπίδρασης, τη σχετική τους απόσταση και τη σημασία των χρωμάτων. Οι κινητές συσκευές είναι επίσης κατεχοχόν προσωπικό σύστημα διαθέσιμο κάθε στιγμή της μέρας και παντού. Αυτός ήταν ο λόγος που μας οδήγησε στην επέκταση της λειτουργικότητας του M.O.L.E δίνοντας έμφαση στην παροχή προσωποποιημένης λειτουργικότητας για τους φοιτητές εστιάζοντας στο προσωπικό ημερολόγιο του φοιτητή. Όλη η λειτουργικότητα σχεδιάστηκε και υλοποιήθηκε με αρχές ανάπτυξης εφαρμογών διαδικτύου και αξιολογήθηκε λεπτομερώς με χρήστες χρησιμοποιώντας αρχές χρηστικότητας (usability engineering) για επικοινωνία ανθρώπων-υπολογιστών.

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή κ. Σταύρο Χριστοδουλάκη που μου έδωσε την ευκαιρία να κάνω την διπλωματική μου εργασία στο Εργαστήριο Διανεμημένων Πληροφοριακών Συστημάτων και Εφαρμογών, καθώς και για την έμπνευση και την καθοδήγηση που μου παρείχε καθ'όλη τη διάρκεια της εργασίας. Θα ήθελα να τον ευχαριστήσω επίσης για τις εμπειρίες και τις γνώσεις που μου προσέφερε στο εργαστήριο (M.U.S.I.C.) τόσο μέσω αυτής της διπλωματικής εργασίας, όσο και μέσω των μαθημάτων επιλογής (Human Computer Interaction και Ανάπτυξη Εφαρμογών Πληροφοριακών Συστημάτων στο Διαδίκτυο).

Επίσης θέλω να ευχαριστήσω όλους όσους βοήθησαν στο στάδιο της αξιολόγησης αυτής της διπλωματικής, προσφέροντας το χρόνο τους και τις φορητές συσκευές τους.

Τέλος ένα μεγάλο ευχαριστώ στην οικογένεια και τους φίλους που είναι πάντα δίπλα μου.





# *Κεφάλαιο 1*

## *Εισαγωγή*

## 1.1 Εισαγωγή

Διανύουμε μια εποχή όπου η εξέλιξη της τεχνολογίας και των υπολογιστών, τόσο από πλευράς υλικού όσο και λογισμικού, έχει οδηγήσει σε εντελώς διαφορετική αντίληψη, λειτουργία και συμπεριφορά, σχετικά με την διαχείριση της πληροφορίας και την πρόσβαση σε αυτήν. Οι υπολογιστές που ήταν, και παραμένουν, το παράθυρο μας στη παγκόσμια κλίμακα πληροφορία έχουν αλλάξει. Έχουν γίνει πιο γρήγοροι, πιο αποδοτικοί και πολύ πιο μικροί. Τόσο μικροί που χωράνε στην τσέπη μας, δίνοντας μας την δυνατότητα να έχουμε πρόσβαση στα αρχεία μας, στα mail μας και στο διαδίκτυο όπου και αν βρισκόμαστε.

Ακολουθώντας την παγκόσμια αυτή τάση, θεωρήσαμε αναγκαίο να έχουν οι φοιτητές του τμήματος μας την δυνατότητα πρόσβασης στον ήδη υπάρχοντα ιστοχώρο του τμήματος (Courses τώρα, MOLE σύντομα) , από τα smartphones τους. Έτσι οι φοιτητές του τμήματος μπορούν να έχουν άμεση πρόσβαση στις υπηρεσίες που τους ενδιαφέρει στο Courses καθώς επίσης και να βλέπουν συγκεντρωτικά όλα όσα συμβαίνουν στο τμήμα και τους αφορούν.

## 1.2 Στόχοι της Εργασίας

Στόχος αυτής της διπλωματικής είναι η δημιουργία μιας εφαρμογής η οποία θα παρέχει την λειτουργικότητα του Courses (MOLE) καθώς και κάποιες ακόμα πιο προσωποκεντρικές λειτουργίες, και θα είναι προσβάσιμη και λειτουργική μέσω smartphones, ανεξαρτήτως λειτουργικού συστήματος και κατασκευαστή συσκευής. Για την υλοποίηση της χρησιμοποιήθηκαν τεχνολογίες που είναι ελεύθερες προς χρήση, χωρίς κόστος, και που υποστηρίζονται από το σύνολο των σύγχρονων web browsers φορητών συσκευών.

## 1.3 Δομή της Εργασίας

Το υπόλοιπο της παρούσας διπλωματικής δομείται ως εξής:

Στο κεφάλαιο 2 περιγράφονται οι σύγχρονες τάσεις στις φορητές συσκευές. Η φύση των φορητών συσκευών, οι πλατφόρμες που υπάρχουν και παρουσιάζονται γενικότερα τα προβλήματα και οι ανάγκες που προκύπτουν από την χρήση τέτοιων συσκευών.

Στο κεφάλαιο 3 παρουσιάζονται οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για την κατασκευή του συστήματος μας.

Στο κεφάλαιο 4 γίνεται η ανάλυση των απαιτήσεων του συστήματος μας και η λειτουργικότητα του χρησιμοποιώντας διαγράμματα περιπτώσεων χρήσης.

Στο κεφάλαιο 5 παρουσιάζεται η γραφική διεπαφή του χρήστη. Όλοι οι παράμετροι οι οποίοι συνυπολογίστηκαν και τα mock-ups για όλες τις λειτουργίες, καθώς και ένα παράδειγμα για το πως τα στοιχεία της desktop εφαρμογής μεταφέρθηκαν στο αντίστοιχο interface των mobile συσκευών.

Στο κεφάλαιο 6 παρουσιάζονται τα τέσσερα (4) είδη αξιολόγησης με τα οποία ελέγξαμε την εφαρμογή μας.

Στο κεφάλαιο 7 παρουσιάζονται οι αλλαγές που απαιτούνται στη δομή του MOLE έτσι ώστε να μπορούν να υποστηριχτούν οι προσωποκεντρικές λειτουργίες της mobile version που υλοποιήσαμε.

Στο κεφάλαιο 8 γίνεται αναλυτική περιγραφή της βάσης δεδομένων μας, του μοτιβου των DAO's και του μοντέλου οντοτήτων.

Στο κεφάλαιο 9 συνοψίζεται η διπλωματική εργασία και παρουσιάζονται μελλοντικές επεκτάσεις που μπορούν να προκύψουν από αυτή.



# *Κεφάλαιο 2*

*Φορητές Συσκευές (Mobile devices)*

## 2.1 Εισαγωγή

Οι φορητές συσκευές πέρασαν από διάφορα στάδια για να φτάσουν στην μορφή και στις δυνατότητες που έχουν σήμερα. Η εξέλιξη των microchips, η αύξηση της χωρητικότητας των μπαταριών εξαιτίας της χρήσης πιο αποδοτικών υλικών, η μαζική παραγωγή καθώς και η πιο πυκνή εγκατάσταση κεραιών οδήγησε σε πολύ μικρές, ελαφριές και με μεγάλες δυνατότητες φορητές συσκευές. Η φυσική εξέλιξη αυτού του γεγονότος ήταν η ενσωμάτωση λειτουργιών, που μέχρι τότε υπήρχαν στους φορητούς υπολογιστές, στα κινητά τηλέφωνα.

Η “Εποχή των Smartphones” ξεκίνησε στις αρχές του 2000 όπου η τεχνολογία μας επέτρεψε να χρησιμοποιήσουμε μεγαλύτερες οθόνες, καλύτερης ευκρίνειας επιτρέποντας μας ουσιαστικά να βλέπουμε περισσότερη πληροφορία σε μικρότερο χώρο. Εδώ αρχίζει η χρήση Λειτουργικών Συστημάτων με την έννοια που τα ξέρουμε από τους υπολογιστές και η ενσωμάτωση λειτουργιών όπως WiFi, Qwerty Keyboards, και εφαρμογές προσαρμοσμένες στις δυνατότητες του εκάστοτε κινητού.

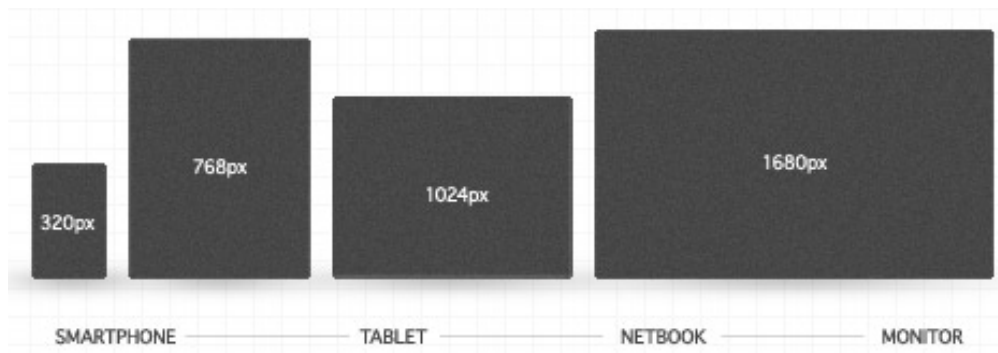
Τα πρώτα χρόνια αποτέλεσαν ουσιαστικά μια “τεχνολογική γέφυρα” καθώς τόσο η ποιότητα και η απόδοση, όσο και η λειτουργικότητα των πρώτων συσκευών ήταν κάτω του μετρίου. Η αλλαγή και η μετάβαση στην σημερινή εποχή που θα μπορούσαμε να την χαρακτηρίσουμε σαν “Εποχή της Αφής”, έγινε στις αρχές του 2007 με την έλευση του πρώτου iPhone. Από τότε είδαμε μια ραγδαία εξέλιξη τόσο στο υλικό και την ποιότητα όσο και στο λογισμικό και τις υπηρεσίες, καθώς όλες οι μεγάλες εταιρίες κατασκευής κινητών τηλεφώνων προσπάθησαν να ακολουθήσουν τις εξελίξεις προσφέροντας αντίστοιχα προϊόντα. Σήμερα όλοι οι κορυφαίοι κατασκευαστές (HTC, Samsung, Nokia, Sony Ericsson, Apple) έχουν στον “στόλο” τους smartphones με οθόνες αφής, ασύρματη πρόσβαση στο ίντερνετ, και ένα πλήθος εφαρμογών για κάθε είδους ανάγκη.

## 2.2 Προβλήματα του μέσου

Οι φορητές συσκευές σήμερα έχουν τρία μεγάλα κύρια προβλήματα.

Το πρώτο πρόβλημα είναι το μέγεθος της οθόνης. Γίνεται εύκολα αντιληπτό ότι ο τρόπος που ο χρήστης αλληλεπιδρούν με τη συσκευή αλλάζει δραστικά σε σχέση με τις εφαρμογές των υπολογιστών. Αλλάζει τόσο η δυνατότητα ομαδοποίησης της πληροφορίας όσο και το νοητικό μοντέλο του χρήστη. Η δυνατότητα ομαδοποίησης της πληροφορίας αλλάζει γιατί οι εφαρμογές πρέπει να παρουσιαστούν με τέτοιο τρόπο ώστε να είναι ξεκάθαρη η πληροφορία σε διαφορετικές διαστάσεις οθονών, οι επιτρεπτές

και μη ενέργειες καθώς και οι επιλογές που παρέχονται σε πολύ λιγότερο χώρο καθώς οι διαστάσεις των οθονών διαφέρουν πάρα πολύ.



Εικόνα 2.1 - Το πλάτος σε pixels στα πιο χαρακτηριστικά είδη οθονών διάφορων κατηγοριών συσκευών.

Το νοητικό μοντέλο είναι η πνευματική, η ψυχική και η νοητική κατάσταση του χρήστη την ώρα που χρησιμοποιεί τη συσκευή. Αυτό είναι και το δεύτερο πρόβλημα που εμφανίζεται. Σε αντίθεση με την κατάσταση του χρήστη όταν χρησιμοποιεί υπολογιστές (επιτραπέζιους ή laptop), όπου το περιβάλλον είναι μέσα σε γενικές γραμμές σταθερό (καθισμένος, χωρίς πολλά πράγματα να του αποσπούν την προσοχή), η κατάσταση όταν χρησιμοποιεί φορητή συσκευή είναι κατά κύριο λόγο απρόβλεπτη. Μπορεί να βρίσκεται στο δρόμο και να βιάζεται ή μπορεί να είναι χαλαρός μέσα σε ένα καφέ. Μπορεί να είναι απασχολημένος και αγχωμένος και να προσπαθεί μέσα από μια εφαρμογή να δει κρίσιμα πράγματα (π.χ μετοχές). Μπορεί να είναι μόνος του ή σε ένα λεωφορείο γεμάτο κόσμο (πιθανόν αδιάκριτο κόσμο που μπορεί να θέλει να ρίξει μια ματιά στο κινητό του διπλανού!). Πρέπει λοιπόν κατά τη δημιουργία κάθε εφαρμογής να συνυπολογίσουμε πάρα πολλούς παράγοντες οι οποίοι θα οδηγήσουν τον χρήστη σε λάθη (απροσεξία, δυσκολία χρήσης της συσκευής) ή θα του δημιουργήσουν προβλήματα (προφύλαξη της ιδιωτικότητας του χρήστη και των προσωπικών δεδομένων).

Το τρίτο πρόβλημα είναι η ασυμβατότητα μεταξύ συσκευών και λειτουργικών συστημάτων. Αυτό το πρόβλημα προέκυψε στη πορεία της εξέλιξης των συσκευών καθώς ο ανταγωνισμός, οι πολλοί διαφορετικοί κατασκευαστές και η μη ανάπτυξη των τεχνολογιών κάτω από μια κοινή ομπρέλα οδήγησε στη δημιουργία πολλών διαφορετικών συστημάτων και συσκευών με εντελώς διαφορετική αντίληψη κατασκευής και παροχής υπηρεσιών στον χρήστη. Αυτό δημιουργεί σίγουρα πρόβλημα στον χρήστη καθώς δεν του παρέχονται όλες οι υπηρεσίες σε όλες οι συσκευές, και όταν αυτό συμβαίνει δεν συμβαίνει με συνεπή τρόπο. Ωστόσο όμως αυτό δημιουργεί ακόμα μεγαλύτερο πρόβλημα στους προγραμματιστές και στις εταιρείες λογισμικού οι οποίες

πρέπει να πάρουν αποφάσεις μεγάλου κόστους για την ανάπτυξη εφαρμογών σε περισσότερες από μία πλατφόρμες.

## 2.3 Πλατφόρμες, Λειτουργικά Συστήματα.

Οι πλατφόρμες παρέχουν πρόσβαση στις συσκευές. Χρειάζονται για να “τρέξουν” το λογισμικό και τις υπηρεσίες σε κάθε μια από αυτές τις συσκευές. Οι κύριες κατηγορίες είναι :

- Licensed (Java ME, Windows Mobile) : οι ιδιοκτήτρια εταιρία ορίζει τους κανόνες άδειας και χρήσης.
- Proprietary ( Palm, Blackberry, iPhone) : ο κώδικας του είναι κρυφός προς τρίτους, συμπεριλαμβανομένων και των χρηστών του λογισμικού. Ακολουθείται από τις περισσότερες μεγάλες εταιρίες λογισμικού.
- Open Source (Android) : όλα τα μέρη του κώδικα των εφαρμογών και της πλατφόρμας είναι προσβάσιμα σε άτομα εκτός της εταιρίας παραγωγής τους και συνεργατών.

Τα λειτουργικά συστήματα που υπάρχουν αυτή τη στιγμή και κατέχουν το κύριο μέρος της αγοράς είναι: Symbian, Windows Mobile, Palm Os, Linux-Android, MacOSX(iOs)

## 2.4 Native Apps vs Mobile Web Apps

Ένα μεγάλο θέμα που εγείρεται από την εμφάνιση κιάλας των smartphones και που αποκτά νόημα στη σημερινή εποχή, όπου οι web browsers των κινητών αρχίζουν να αγγίζουν την δυναμικότητα των desktop web browsers είναι το εξής:

Να φτιάξω μια εφαρμογή για μια συγκεκριμένη πλατφόρμα (native application) ή μια εφαρμογή που θα τρέχει μέσα από τον web browser (mobile web application) ?

Για να αποφασίσουμε θα πρέπει να δούμε τα θετικά και τα αρνητικά της κάθε μιας υλοποίησης και να αποφασίσουμε με βάση τις απαιτήσεις που έχουμε.

### **Native Apps**

Θετικά: Προσφέρουν την καλύτερη δυνατή εμπειρία στον χρήστη, παρέχοντας του μια πλούσια σχεδίαση η οποία εκμεταλλεύεται την πλήρη δυναμικότητα της συσκευής. Επίσης - δεδομένου ότι υπάρχει η γνώση της υλοποίησης εφαρμογών στην συγκεκριμένη πλατφόρμα- είναι σχετικά εύκολο να υλοποιηθεί. Ένα ακόμα θετικό στοιχείο είναι ότι είναι συνηθισμένο και



μπορείς εύκολα να βάλεις χρέωση στην αγορά της συγκεκριμένης εφαρμογής αποκομίζοντας κέρδη.

Αρνητικά: Είναι πολύ δύσκολο και εξαιρετικά δαπανηρό να φτιάξεις, να υλοποιήσεις και ελέγξεις μια εφαρμογή για διαφορετικές πλατφόρμες. Επίσης πρέπει να εγκριθεί από τρίτους χωρίς να έχει ο δημιουργός καθόλου έλεγχο πάνω σε αυτό, καθώς και να έρθεις σε συμφωνία για να μοιραστείς πιθανά κέρδη με αυτούς τους φορείς.

## **Mobile Web Apps**

Μια εμπειρία παρόμοια με native εφαρμογή η οποία όμως “τρέχει” μέσα στον web browser και είναι ανεξάρτητη του λειτουργικού συστήματος.

Θετικά: πιο εύκολο να υλοποιηθεί κάνοντας χρήση γνωστών γλωσσών προγραμματισμού και εργαλείων σε μηχανικούς που εξειδικεύονται στη κατασκευή εφαρμογών διαδικτύου, όπως HTML, CSS, Ajax και Javascript. Είναι σχετικά απλό να υλοποιηθεί έτσι ώστε να “τρέχει” σε διαφορετικές πλατφόρμες. Οι χρήστες έχουν μια αρκετά καλή εμπειρία με τέτοιες διεπαφές στο διαδίκτυο και υπάρχουν δυνατότητες πλούσιας σχεδίασης από πλευράς λειτουργικότητας. Το περιεχόμενο είναι προσβάσιμο από όλους τους web browsers ανεξάρτητα από τη συγκεκριμένη πλατφόρμα και το λειτουργικό.

Αρνητικά: Η εμπειρία χρήστη ίσως δεν είναι η καλύτερη δυνατή (αν συγκριθεί με αυτή των native εφαρμογών), και δεν μπορούν να υποστηριχτούν και κάποιες λειτουργίες της συσκευής (χρήση χωρίς σύνδεση ίντερνετ, πρόσβαση στα αρχεία και στο υλικό που είναι αποθηκευμένο κ.τ.λ.)

## **2.5 Η ανάγκη για καθολικότητα**

Από τα παραπάνω εύκολα γίνεται κατανοητό ότι στην περίπτωση της υποστήριξης των μαθημάτων μέσω του courses η καλύτερη επιλογή είναι να δημιουργήσουμε μια web εφαρμογή η οποία θα παρέχεται σε όλους τους χρήστες ανεξαρτήτου συσκευής. Τους χρήστες δεν τους ενδιαφέρει τι συσκευή χρησιμοποιούν ή τι λειτουργικό έχουν. Θέλουν σε κάθε περίπτωση να μπορούν να έχουν ότι και ο διπλανός τους και να έχουν πρόσβαση εξίσου εύκολα με αυτόν στην αντίστοιχη πληροφορία. Η πληροφορία μας δεν είναι εξαιρετικά πολύπλοκη, όπως 3D χάρτες (Google Earth, κτλ) που να χρειάζεται υποστήριξη συγκεκριμένης πλατφόρμας. Επιπλέον τίθενται και θέματα κέρδους καθώς δεν υπάρχει αρκετό προσωπικό διαθέσιμο, αφιερωμένο στη συντήρηση της εφαρμογής, ώστε να μπορούν να υποστηρίζονται πολλές πλατφόρμες.



# *Κεφάλαιο 3*

*Τεχνολογίες και Αρχιτεκτονική Υλοποίησης*

### 3.1 Εισαγωγή

Στο κεφάλαιο αυτό περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής μας. Συγκεκριμένα γίνεται αναφορά στις γλώσσες προγραμματισμού που χρησιμοποιήθηκαν, στην αρχιτεκτονική και στα εργαλεία προγραμματισμού. Πιο αναλυτικά στο κεφάλαιο 3.2 γίνεται περιγραφή της HTML5, η οποία χρησιμοποιήθηκε ως markup γλώσσα στην εφαρμογή μας. Στις ενότητες 3.3 & 3.4 περιγράφεται η γλώσσα προγραμματισμού Javascript και οι βιβλιοθήκες της JQuery, JQuery Mobile οι οποίες σε συνδυασμό με την HTML5 χρησιμοποιήθηκαν για να δώσουν το αισθητικό και λειτουργικό αποτέλεσμα της διεπαφής του χρήστη μας. Στην ενότητα 3.5 περιγράφονται τα Servlets, τα οποία χρησιμοποιήθηκαν για την επικοινωνία με τον server (αποστολή αιτημάτων - λήψη απαντήσεων). Στην ενότητα 3.6 περιγράφεται η τεχνολογία Java Server Pages η οποία μας βοήθησε να χτίσουμε δυναμικές σελίδες στην εφαρμογή μας. Στην ενότητα 3.7 περιγράφεται η τεχνολογία των Data Access Object (DAO) τα οποία μας επέτρεψαν να έχουμε μια ασφαλή και επεκτάσιμη επικοινωνία με την βάση και χρήση εναλλακτικών βάσεων δεδομένων για την υποστήριξη της εφαρμογής. Αυτό είναι ιδιαίτερα σημαντικό γιατί η εφαρμογή που χτίσαμε χρησιμοποιεί ανεξάρτητη βάση από το M.O.L.E. γιατί δεν ήταν δυνατόν να χρησιμοποιήσουμε τη βάση του M.O.L.E εξαιτίας της μη ύπαρξης API για τη βάση, και μεγάλης πολυπλοκότητας του σχήματος και του μηχανισμού ασφαλείας για την πρόσβαση στο M.O.L.E. Τέλος στην ενότητα 3.8 περιγράφεται το μοτίβο αρχιτεκτονικής Boundary - Control - Entity - DB Interface Layer το οποίο χρησιμοποιήθηκε στην εφαρμογή μας.

### 3.2 HTML5

Η HTML5 είναι μια υπό ανάπτυξη markup γλώσσα για τον παγκόσμιο ιστό που όταν ετοιμαστεί θα είναι η επόμενη μεγάλη έκδοση της HTML (Γλώσσα Υπερκειμένου, HyperText Markup Language). Η ομάδα Web Hypertext Application Technology working Group (WHATWG) άρχισε δουλειά σε αυτή την έκδοση τον Ιούνιο του 2004 με το όνομα Web Applications 1.0. Η HTML5 προορίζεται για αντικατάσταση της HTML 4.01, της XHTML 1.0, και της DOM Level 2 HTML. Ο σκοπός δημιουργίας της είναι η μείωση της ανάγκης για ιδιόκτητα plug-ins (όπως το Adobe Flash, το Microsoft Silverlight, το Apache Pilot, και η Sun JavaFX) και η ανάγκη για δημιουργία πλούσιων διαδικτυακών εφαρμογών (RIA). Οι ιδέες πίσω από την HTML5 εμφανίστηκαν αρχικά το 2004 από την ομάδα WHATWG. Η HTML5 εμπεριέχει το πρότυπο Web Forms 2.0 που είναι επίσης της WHATWG. Το πρότυπο HTML5 υιοθετήθηκε ως αρχικό βήμα για τις εργασίες της νέας

ομάδας εργασίας HTML του W3C το 2007. Αυτή η ομάδα εργασίας δημοσίευσε το Πρώτο Δημόσιο Working Draft του προτύπου στις 22 Ιανουαρίου 2008. Το πρότυπο είναι ακόμη υπό ανάπτυξη, και αναμένεται να παραμείνει έτσι για πολλά χρόνια, παρόλο που όποια μέρη της HTML5 ολοκληρώνονται θα υποστηρίζονται από περιηγητές πριν το όλο πρότυπο φτάσει στη τελική του μορφή. Οι συντάκτες της HTML5 είναι ο Ίαν Χίκσον της εταιρίας Google Inc. και ο Ντέιβ Χιτ της εταιρίας Apple Inc.

### 3.3 Javascript

Η Javascript είναι η πιο δημοφιλής scripting γλώσσα για το διαδίκτυο. Σήμερα χρησιμοποιείται σχεδόν σε όλες τις ιστοσελίδες προσθέτοντας λειτουργικότητα (επικύρωση στοιχείων, διάκριση browser, επικοινωνία με τον server, αντίδραση σε συμβάντα, δημιουργία cookies, ανάγνωση και εγγραφή HTML στοιχείων κ.α.), και υποστηρίζεται από όλους τους σύγχρονους browsers (Internet Explorer, Firefox, Chrome, Opera, Safari). Ο κώδικας της μπορεί να ενσωματωθεί σε ένα HTML αρχείο ή να κληθεί από ένα ή περισσότερα εξωτερικά αρχεία. Επίσης η χρήση της είναι ελεύθερη (δεν απαιτείται άδεια για την χρήση της στις εμπορικές εφαρμογές) όπως και όλες οι τεχνολογίες που χρησιμοποιήθηκαν στην ανάπτυξη της εφαρμογής μας. Η χρήση της javascript στην εφαρμογή μας, έγκειται κυρίως στη χρήση της υπό ανάπτυξης Javascript βιβλιοθήκης, jQuery Mobile η οποία έχει δημιουργηθεί με σκοπό να εξυπηρετεί τις ανάγκες των σύγχρονων mobile browsers. Επίσης χρησιμοποιήθηκε για να μπορούμε να κάνουμε ασύγχρονες κλήσεις στη βάση (μέσω AJAX - Asynchronous Javascript and XML), προσθέτοντας αμεσότητα στην εφαρμογή μας και μείωση στο μέγεθος των δεδομένων που μεταφέρονταν κατά τη χρήση της.

### 3.4 jQuery Mobile

Η jQuery Mobile είναι μια javascript βιβλιοθήκη φτιαγμένη με σκοπό να υποστηρίζει συσκευές αφής (touch-optimized), και η οποία δημιουργήθηκε από την ομάδα δημιουργίας, της γνωστής και ευρέως χρησιμοποιούμενης, javascript βιβλιοθήκης, jQuery. Βασίζεται στη βιβλιοθήκη του jQuery και στο jQuery UI. Ο πηγαίος του κώδικας δεν απαιτεί σημαντικούς πόρους, είναι ελαφρύς, γρήγορος στην εκτέλεση, ενώ το ίδιο σύστημα είναι ευέλικτο και προσαρμόσιμο. Η υλοποίηση της στοχεύει στο να δημιουργήσει ένα πλαίσιο ανάπτυξης web εφαρμογών, συμβατό με ένα ευρύ πεδίο από Smartphones και Tablets, κάτι που κρίθηκε αναγκαίο εξαιτίας της ραγδαία αναπτυσσόμενης, αλλά και παράλληλα ετερογενούς, αγοράς των smartphones και tablets. Με την βιβλιοθήκη αυτή έχουμε στα χέρια μας ένα πολύ δυνατό εργαλείο το οποίο μας επιτρέπει να φτιάχνουμε εφαρμογές που θα τρέχουν σε όλες τις δημοφιλείς

πλατφόρμες για smartphones και tablets, μέσω του browser. Βέβαια μας εξασφαλίζει μόνο ότι έχει να κάνει με την γραφική διεπαφή του χρήστη και τις μεταβάσεις μεταξύ σελίδων για να μπορεί να δημιουργείται στο χρήστη η αίσθηση της native εφαρμογής. Είναι ακόμα σε beta έκδοση και διαπιστώσαμε ότι πολλοί λίγοι την είχαν συνδυάσει με τις τεχνολογίες που χρησιμοποιούσαμε εμείς (jsp, java-servlets κτλ). Παρ'όλα αυτά επειδή η εφαρμογή μας είναι δομημένη σε επίπεδα, τα οποία θα αναλυθούν στην συνέχεια, μπορέσαμε να τα συνδυάσουμε εκμεταλλευόμενοι τα πλεονεκτήματα όλων των τεχνολογιών. Ο βασικός λόγος που επιλέξαμε το jQuery Mobile είναι γιατί όλα δείχνουν πως θα είναι το framework που μελλοντικά θα χρησιμοποιείται για αυτό το σκοπό και επειδή γίνεται πολύ καλή δουλειά σχετικά με την υποστήριξη των συσκευών.



*Εικόνα 3.1 - Υποστηριζόμενες πλατφόρμες από το jQuery Mobile.*

Ένας ακόμα λόγος είναι ότι υποστηρίζεται απ'όλες τις μεγάλες εταιρείες (Nokia, Adobe, Blackberry, dotMobi, DeviceAtlas, palm, mozilla corporation κ.α.) δείχνοντας ότι θα καθιερωθεί μελλοντικά.

Ακολουθεί ένας πίνακας συμβατότητας του jQuery Mobile με τις πιο δημοφιλείς πλατφόρμες της αγοράς. (αναλυτικότερη περιγραφή εδώ: <http://jquerymobile.com/gbs/>)

Platform	Version	Native
iOS	v2.2.1	B
	v3.1.3, v3.2	A
	v4.0	A
Symbian S60	v3.1, v3.2	C
	v5.0	A
Symbian UIQ	v3.0, v3.1	
	v3.2	
Symbian Platform	3.0	A
BlackBerry OS	v4.5	C
	v4.6, v4.7	C
	v5.0	B
	v6.0	A
Android	v1.5, v1.6	A
	v2.1	A
	v2.2	A
Windows Mobile	v6.1	C
	v6.5.1	C
	v7.0	A
webOS	1.4.1	A
bada	1.0	A
Maemo	5.0	B
MeeGo	1.1	A

Εικόνα 3.2 - Πίνακας Συμβατότητας jQuery Mobile με εκδόσεις λειτουργικών και βαθμολογία των browser τους.  
(πηγή: <http://jquerymobile.com/gbs/> )

Στον παραπάνω πίνακα η βαθμολογία προκύπτει από την ποιότητα των browsers σε συνδυασμό με το ποσοστό χρήσης στην αγορά κινητών συσκευών.

**A-grade:** υποστηρίζουν media queries τα οποία είναι απαραίτητα για τον εντοπισμό των συσκευών και την προσαρμογή της σελίδας στο αντίστοιχο περιβάλλον. Οι browsers αυτοί έχουν ελεγχθεί εντατικά και υποστηρίζουν σχεδόν την πλήρη λειτουργικότητα του jQuery Mobile.

**B-grade:** browsers που δεν έχουν μεγάλο μερίδιο στην αγορά και για το λόγο αυτό δεν ελέγχονται εντατικά. Υποστηρίζουν τη πλήρη λειτουργικότητα αλλά υπάρχουν bugs.

**C-grade:** δεν υποστηρίζουν media queries. Δεν θα υλοποιείται το scripting από το jQuery Mobile και τα CSS. Υποβάθμιση σε απλή HTML, και CSS. (Δεν ασχοληθήκαμε με αυτή την κατηγορία).

Εμείς παρόλο που αναφερόταν ότι το framework έχει δοκιμαστεί στις παραπάνω συσκευές κατά την ανάπτυξη το δοκιμάσαμε σε iPhone 3GS (iOS4.3-iOS5.beta), HTC Wildfire (Android 1.9-2.1) Samsung Galaxy (Android 2.0-2.2) και iPad (iOS4.3) για να εξασφαλίσουμε ότι όλες οι τεχνολογίες που χρησιμοποιούμε δουλεύουν όπως πρέπει συνδυασμένα.

Ένα πλεονέκτημα του jQuery Mobile είναι ότι χρειάζεται να φορτώσουμε μόνο μία φορά τα .js και .css αρχεία κατά την εκτέλεση της εφαρμογής, και όχι σε κάθε σελίδα. Για την ακρίβεια πρέπει να τα φορτώσουμε στην πρώτη σελίδα. Έτσι έχουμε κέρδος κόστους στη μεριά του client αφού ο browser χρειάζεται να “κατεβάσει” λιγότερα δεδομένα, αφού κατεβάζει τα συγκεκριμένα αρχεία μόνο μία φορά. Αυτό συμβαίνει εξαιτίας της δομής των σελίδων και της πλοήγησης μέσω Ajax. Η δομή θα εξηγηθεί στην συνέχεια. Εδώ είχαμε δύο επιλογές. Ή να κατεβάσουμε τις βιβλιοθήκες και να τις έχουμε στον server μας ή να καλούμε την hosted version από άλλους servers, όπως η Google ή τον server του jQuery Project. Εμείς επιλέξαμε την πρώτη επιλογή, καταρχήν για να εξασφαλίσουμε ότι στην πορεία της ανάπτυξης μας δεν θα αλλάξει ο κώδικας, δεδομένου ότι το jQM είναι ακόμα σε φάση ανάπτυξης και δεν θα έχει ολοκληρωθεί μέχρι την ολοκλήρωση της διπλωματικής αυτής. Έτσι θα καταφέραμε να έχουμε μια εφαρμογή που δουλεύει με δικές της βιβλιοθήκες και η οποία αν κάποια στιγμή χρησιμοποιηθεί από το τμήμα θα μπορεί με μικρή παρέμβαση (ίσως και χωρίς καμία παρέμβαση δεδομένου ότι τα μέρη που χρησιμοποιήσαμε δεν μας παρουσίασαν πρόβλημα) να δεχτεί την ολοκληρωμένη έκδοση. Στην πορεία ωστόσο επειδή υπήρξαν προβλήματα που αφορούσαν bugs που μας επηρέαζαν αναβαθμίσαμε τις βιβλιοθήκες στην νεότερη -αλλά όχι τελική- έκδοση. Επίσης τις κατεβάσαμε για να μπορούμε να κάνουμε αλλαγές σε σημεία που χρειαζόμασταν και να μπορέσουμε να κατανοήσουμε καλύτερα τη λειτουργία τους.

Δομή: Το jQM εξαρτάται σε μεγάλο βαθμό από το χαρακτηριστικό “data-” της HTML5. Κάθε attribute που ξεκινάει με “data-” αγνοείται από την μηχανή απόδοσης (render machine) του browser και είναι διαθέσιμο για την προώθηση πληροφορίας ή οδηγιών σε μια εφαρμογή. Στην περίπτωση του jQM, το “data-” attribute βοηθάει στη δομή των σελίδων της εφαρμογής.

data-role: χρησιμοποιείται για να τακτοποιήσει τα στοιχεία σε σελίδες και τμήματα σελίδων. Κύρια συστατικά της σελίδας είναι τα : page (ορίζει την ίδια την σελίδα), header, content, footer. Ακολουθεί ένα παράδειγμα μιας σελίδας κώδικα με την αντίστοιχη γραφική διεπαφή που προκύπτει.



```

<body>
<div data-role="page" data-back-btn-text="Πίσω">

  <div data-role="header" data-theme="b" data-iconpos="notext">
    <h1>Προσωπικά Μηνύματα</h1>
    <a href="#../../controller?action=GOHOME" data-direction="reverse" data-icon="home" data-theme="e"
      class="ui-btn-right" data-iconpos="notext">Αρχική</a>
  </div><!-- /header -->

  <div data-role="content">

    <ul data-role="listview" data-counttheme="e" data-theme="d">

      <li><a href="#../../controller?action=GO2INBOX">
        
        Εισερχόμενα</a>

        <!-- if inboxUnread > 0 -->
        <span class="ui-li-count"><%=inboxUnread%></span>
        <!-- /if -->
      </li>
      <li>
        <a href="#../../controller?action=GO2OUTBOX">
          
          Εξερχόμενα</a>
        </li>
      <li>
        <a href="#../../controller?action=GO2NEWMMESSAGE">
          
          Αποστολή Νέου</a>
        </li>

      <!-- footer navigation bar -->
      &nbsp;&nbsp;&nbsp;<!--avoid slip errors-->
      <%= include file="nav-footer.jsp"%>
    </ul>
  </div><!-- /content -->
</div><!-- /page -->
</body>

```

Εικόνα 3.3 - Κώδικας της .jsp σελίδας των Προσωπικών Μηνυμάτων του χρήστη. Φαίνεται η χρήση του data-role attribute που ορίζει τη δομή της σελίδας μας.



Εικόνα 3.4 - Η σελίδα των Προσωπικών Μηνυμάτων που υλοποιείται από τον κώδικα της εικόνας 3.3.

*“data-”attributes*: Εδώ περιγράφονται τα κυριότερα “data-” attributes που χρησιμοποιήθηκαν στην υλοποίηση μας.

Μέσα στο anchor tag <a> χρησιμοποιήσαμε τα:

*data-rel*: μπορεί να πάρει τις τιμές “dialog” εμφανίζοντας την επόμενη σελίδα σαν παράθυρο διαλόγου (και δεν την αποθηκεύει στο hash list), “external” αν θέλουμε η μετάβαση να γίνει εκτός του domain που βρισκόμαστε. Το πρώτο χρησιμοποιείται στις σελίδες διαλόγου και το δεύτερο στα αρχεία που θέλουμε ο χρήστης να διαβάσει ή να κατεβάσει, καθώς και σε links σε εξωτερικά domain (π.χ. στην σελίδα του πλήρους site του MOLE).

*data-direction=“reverse”* : κάνει slide από αριστερά. Το χρησιμοποιούμε για να δημιουργήσουμε την αίσθηση στον χρήστη ότι επιστρέφει σε προηγούμενη σελίδα.

*data-transition*: επιλέγουμε τις μεταβάσεις που μοιάζουν με αυτές των native apps. Οι επιλογές εδώ είναι: slide (το χρησιμοποιούμε για τις κανονικές μεταβάσεις), slidedown (το χρησιμοποιούμε για την εμφάνιση των dialog σελίδων, slideup (για να κλείσουμε τις dialog σελίδες). Υπάρχουν και τα pop, fade, flip τα οποία δεν χρησιμοποιήσαμε όμως.

Άλλα “data-” attributes:

*data-theme*: ορίζει ποιο θέμα θα ακολουθήσει το συγκεκριμένο στοιχείο.

*data-iconpos*: ορίζει που θα τοποθετηθεί το εικονίδιο αν υπάρχει.

*data-back-button-text*: ορίζει το κείμενο του default κουμπιού για το “πίσω”, πάνω αριστερά στο header.

*data-divider-theme*: ορίζει το θέμα του στοιχείου που διαχωρίζει τις λίστες

*data-count-theme*: ορίζει το θέμα που έχουν τα notifications μου

*data-icon*: ορίζει ποιο εικονίδιο θα έχουμε

*#Hash Symbol*: Χρησιμοποιείται στα links από την εφαρμογή για να μπορέσει να δημιουργήσει μια hash list με τις σελίδες που έχουν φορτωθεί, και να μην χρειάζεται να τις ξαναφορτώνει. Επίσης έτσι μπορεί και δημιουργεί την αίσθηση της native εφαρμογής σε μία μόνο σελίδα. Όταν δεν χρησιμοποιηθεί η μετάβαση δεν διατηρείται στο hashlist.

Κλήσεις του controller: Οι ιδιαιτερότητες στη δομή των σελίδων μας δημιούργησε μεγάλο πρόβλημα στην κλήση του controller, γιατί, ουσιαστικά, μέσα στην εφαρμογή δεν μπορούσαμε να χρησιμοποιήσουμε relative paths, παρά μόνο αν έπρεπε να μπουν στο hash list. Έτσι καταλήξαμε στην χρήση των εξής:

απλά links που καλούν τον controller:

```
<a href="#../controller?action=SOMEACTION">
```

κλήσεις Ajax (για φόρτωση περισσότερων στοιχείων στη σελίδα):

```
<a href="javascript:functionName"> μέσα στο jsp.
```

```
ajaxRequest.open("GET","../controller?action=SOMEACTION", true) μέσα στο .js
```

φόρμες:

```
<form action="/controller?action=SOMEACTION" ...>
```

Φόρμες: Εδώ κάποια πράγματα παραμένουν ως έχουν και κάποια αλλάζουν ως προς το πως αντιμετωπίζουμε τις φόρμες. Ίδια παραμένουν το πως γίνεται submit μια φόρμα, τα attributes μέσα στο <form> tag, και το ότι όταν γίνεται submit τα δεδομένα φορτώνονται όπως περιμένουμε μέσα στο action της φόρμας.

Τα προβλήματα που εμφανίστηκαν, οφείλονταν και πάλι στον τρόπο με τον οποίο διαχειρίζεται τις σελίδες το jQM. Το DOM ανακατεύεται από το jQM για να δημιουργηθεί η αίσθηση της εφαρμογής σε μία μόνο σελίδα. Η προσέγγιση αυτή υλοποιείται με το να έχουμε το DOM να διαχειρίζεται πολλές σελίδες ταυτόχρονα, αντί να τις ξαναδημιουργεί σε κάθε φόρτωμα της σελίδας. Συνέπεια αυτού του γεγονότος είναι ότι το "id" χαρακτηριστικό κάθε στοιχείου πρέπει να είναι μοναδικό σε ολόκληρη την εφαρμογή, και όχι μόνο στο συγκεκριμένο html (jsp) αρχείο. Για να λύσουμε αυτό το πρόβλημα απλά ακολουθήσαμε μια σύμβαση στην ονοματολογία έτσι ώστε να μπορέσουμε να έχουμε διαφορετικά "id" σε ολόκληρη την εφαρμογή. Ένα άλλο πρόβλημα που εμφανίστηκε ήταν ότι είχαμε την λέξη "action" για λέξη κλειδί στον controller για να ξεχωρίζει τα actions που ορίζαμε. Το πρόβλημα είναι ότι μπερδεύαμε το jQM το οποίο προσπαθούσε να εντοπίσει ολόκληρο το request-query που έβλεπε μέσα στο "action" attribute του <form> και κατέληγε σε σφάλμα. Για να μπορέσουμε να το λύσουμε αυτό βάλαμε έναν έλεγχο μέσα στην κλάση RequestUtility του controller ο οποίος όταν εντοπίζει τέτοιας μορφής query το τροποποιεί κατάλληλα, έτσι ώστε να μπορεί να προχωρήσει ομαλά το request.

### 3.5 Αντίστοιχα frameworks και γιατί επιλέξαμε το JQM

Στην αγορά υπάρχουν αντίστοιχα frameworks με το jQuery Mobile. Τα πιο γνωστά είναι τα Sencha Touch, jQTouch. Παρουσιάζονται συνοπτικά στην συνέχεια για να καταλήξουμε στον λόγο που επιλέξαμε να χρησιμοποιήσουμε το jQuery Mobile.

jQTouch: αντιστοιχίζεται άμεσα με το jQuery Mobile. Είναι εύκολο στη χρήση και χρησιμοποιεί HTML, CSS, JavaScript για να καταφέρει να χτίσει iPhone-like εφαρμογές. Παρέχει ένα βασικό σύνολο από εργαλεία και animations και σχετικά εύκολα μπορεί να του εισαχθεί μια δυναμική συμπεριφορά. Παρ'όλα αυτά παρουσιάζονται αρκετά προβλήματα απόδοσης (μερικές μεταβάσεις δεν λειτουργούν όπως πρέπει για παράδειγμα) όταν εισάγεται δυναμικό περιεχόμενο στην εφαρμογή. Ωστόσο αν και το project ανάπτυξης του είναι ακόμα ενεργό, οι κύριος δημιουργός έχει φύγει και η ομάδα ανάπτυξης φαίνεται ότι σιγά σιγά το εγκαταλείπει, διατηρώντας μόνο τη υποστήριξη για τα ήδη υπάρχοντα εργαλεία. Επίσης δεν έχει ακολουθήσει τόσο τα standards τα οποία προτάσσονται, σε αντίθεση με το jQuery Mobile. Ο τελευταίος είναι και ο κύριος λόγος που επιλέξαμε το jQuery Mobile αντ'αυτού.

Sencha Touch: εξίσου καλό και δυνατό framework αλλά η κύρια διαφορά του είναι ότι αντί να υλοποιείται με βάση το προϋπάρχον DOM, δημιουργεί δικό του DOM βασισμένο σε αντικείμενα που δημιουργούνται μέσω της Javascript. Επίσης υποστηρίζει ιδιαίτερα UI χαρακτηριστικά για συσκευές σαν το iPad, αποθήκευση κ.α. Επειδή χρησιμοποιεί δικό της DOM η εφαρμογή ουσιαστικά είναι λιγότερο "διαδικτυακή" με την έννοια των τεχνολογιών που χρησιμοποιούνται. Έτσι δεν μπορούσαμε να είμαστε σίγουροι αν η εφαρμογή μας θα υποστηρίζεται απ'όλες τις επιθυμητές συσκευές. Καθώς απ'ότι φάνηκε στοχεύει σε υποστήριξη συγκεκριμένων συσκευών και όχι σε όλες μέσω των web standards (τουλάχιστον όχι ακόμα). Ένας ακόμα αποτρεπτικός παράγοντας ήταν ότι η υποστήριξη προς τους developers ξεκινούσε από τα 300\$/χρόνο, ενώ η παροχή του framework ήταν δωρεάν. Αυτό σήμαινε ότι δεν θα είχαμε τη δυνατότητα να λύσουμε προβλήματα που θα εμφανίζονταν όταν προσπαθούσαμε να το συνδυάσουμε με τις τεχνολογίες που θέλαμε, κάτι που στην περίπτωση μας έγινε μέσω συζητήσεων στον forum του jQuery Mobile. Επίσης υπάρχουν αρκετά bugs στην εισαγωγή δυναμικών χαρακτηριστικών.

### 3.6 Servlets

Τα servlets είναι μια τεχνολογία που χρησιμοποιείται για την λήψη και την απάντηση σε αιτήματα (requests) από τον web client, συνήθως μέσω του πρωτοκόλλου HTTP. Πρόκειται για μικρά κομμάτια κώδικα που εκτελούνται στην πλευρά του web-server. Κάθε servlet έχει τον ακόλουθο κύκλο ζωής: αρχικά κατασκευάζεται και αρχικοποιείται. Στη συνέχεια, αφού κληθεί, διαχειρίζεται το αίτημα που δέχεται από τον client και στέλνει μια απάντηση ως επιστροφή. Τέλος, αφού ολοκληρώσει την εργασία του, καταστρέφεται. Στην εφαρμογή μας η τεχνολογία αυτή χρησιμοποιήθηκε, στην πλευρά του server, για να απαντά στα αιτήματα που υποβάλει ο client. Για παράδειγμα αν ο χρήστης συμπληρώσει μια φόρμα, ένα servlet είναι υπεύθυνο για να αποκριθεί στο αίτημα και να επιστρέψει στο χρήστη πληροφορίες για το αν όλα πήγαν καλά και τις πιθανές επιλογές που έχει.

### 3.7 Java Servlet Pages (JSP)

Οι JSPs (JavaServer Pages) είναι μια τεχνολογία που έχει δημιουργηθεί από την εταιρεία Sun Microsystems για να μπορεί να δημιουργεί δυναμικό περιεχόμενο (dynamic content) στο Web. Πρόκειται για HTML έγγραφα (ιστοσελίδες) τα οποία αναμειγνύονται με τη γλώσσα προγραμματισμού Java, η οποία και έχει τη δυνατότητα να παρέχει (δημιουργεί) αυτό το δυναμικό περιεχόμενο.

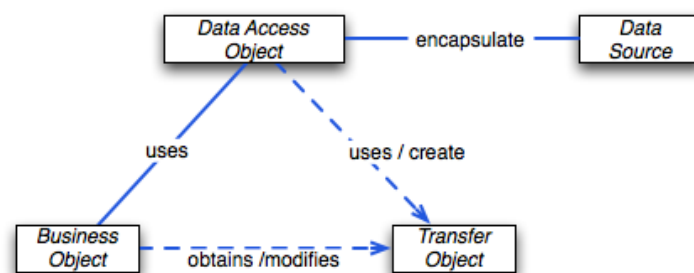
Οι JSPs είναι μια εφαρμογή στην πλευρά του server (server-side application), που σημαίνει ότι δέχονται μια αίτηση (request) και παράγουν μια απόκριση ή απάντηση (response). Σε γενικές γραμμές, οι αιτήσεις γίνονται από έναν Web client και η απόκριση είναι ένα παραγόμενο HTML έγγραφο (ιστοσελίδα) το οποίο στέλνεται πίσω στον Web client. Επειδή οι JSPs είναι μια εφαρμογή στην πλευρά του server, έχουν πρόσβαση σε πηγές (resources) στον server, όπως είναι τα Servlets, JavaBeans, EJBs, αλλά και σε βάσεις δεδομένων.

Υπάρχουν πολλά πλεονεκτήματα από τη χρήση των JavaServer Pages. Επειδή οι JSPs χρησιμοποιούν τη γλώσσα προγραμματισμού Java, ακολουθούν την πολιτική write-once, run-anywhere. Αυτό σημαίνει ότι μια JSP μπορεί να εκτελεσθεί σ' έναν οποιονδήποτε application server ο οποίος διαθέτει μια εικονική μηχανή Java (Java virtual machine) και υποστηρίζει τις JSPs χωρίς να χρειασθεί κάποια τροποποίηση στον κώδικα. Επίσης σημαντικό πλεονέκτημα των JSPs είναι η χρήση των tag libraries. Οι JSPs χρησιμοποιούν τα tags, τα οποία είναι παρόμοια με αυτά της HTML και της XML, για να εισάγουν δυναμικό περιεχόμενο (dynamic content). Τα tag libraries ορίζουν επιπλέον tags τα οποία μπορούν να χρησιμοποιηθούν για να αντικαταστήσουν τμήματα κώδικα.

Ένα άλλο σημαντικό πλεονέκτημα των JSPs είναι ο διαχωρισμός των ρόλων. Οι προδιαγραφές των JSPs επιτρέπουν να μοιραστεί το φορτίο σε δύο κατηγορίες : στο γραφικό περιεχόμενο της σελίδας και στο δυναμικό περιεχόμενο της σελίδας. Αυτό σημαίνει στην πράξη ότι η ομάδα που δεν γνωρίζει τη γλώσσα προγραμματισμού Java μπορεί να δημιουργήσει το γραφικό περιεχόμενο της σελίδας και ένας προγραμματιστής της Java να δημιουργήσει το δυναμικό περιεχόμενο της σελίδας. Και τέλος το ότι χρησιμοποιεί την ισχυρή γλώσσα προγραμματισμού Java για να δημιουργεί το δυναμικό της περιεχόμενο. Αυτό σημαίνει ότι βρίσκονται στη διάθεσή μας εκατοντάδες τάξεις (classes) και μέθοδοι (methods).

### 3.8 Data Access Object (DAO)

Το μοτίβο Data Access Objects διευκολύνει την πρόσβαση σε μια πηγή δεδομένων (στην περίπτωση μας Βάση Δεδομένων) για την αποθήκευση και ανάκτηση της πληροφορίας. Παρέχει όλους εκείνους τους μηχανισμούς (συναρτήσεις αποθήκευσης και ανάκτησης δεδομένων) που μας χρειάζονται για την πρόσβαση στην βάση δεδομένων. Το DAO αποκρύπτει τελείως τις λεπτομέρειες υλοποίησης της πηγής δεδομένων από τους χρήστες της (clients). Σημαντικό πλεονέκτημα της συγκεκριμένης τεχνολογίας είναι ότι διαχωρίζει το business logic επίπεδο της εφαρμογής από το persistent logic. Έτσι σε περίπτωση που μελλοντικά θέλουμε να κάνουμε μια αλλαγή στην πηγή δεδομένων οι clients δεν θα επηρεαστούν. Ουσιαστικά η δουλειά του DAO είναι να λειτουργεί σαν μεσολαβητής ανάμεσα στο business logic και την πηγή δεδομένων.



Εικόνα 3.5 - Διάγραμμα κλάσεων που αναπαριστά τις συσχετίσεις για το πρότυπο DAO.

**Business Object** : είναι το αντικείμενο που χρειάζεται να έχει πρόσβαση στην πηγή δεδομένων για την ανάκτηση και την αποθήκευση των δεδομένων.

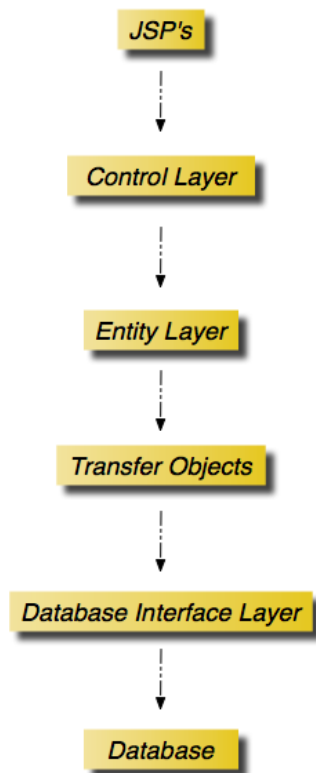
**Data Access Object** : περιγράφει το επίπεδο υλοποίησης της πρόσβασης στα δεδομένα έτσι ώστε να επιτρέπει στο business object να έχει πρόσβαση στα δεδομένα μας.

**Data Source** : αντιπροσωπεύει την πηγή των δεδομένων μου. Μπορεί να είναι μια βάση δεδομένων όπως RDBMS, OODBMS, XML repository, flat file system.

**Transer Object** : χρησιμοποιείται για τη μεταφορά των δεδομένων μας. Ένα data access object χρησιμοποιεί ένα transfer object για να επιστρέψει τα δεδομένα στο client

### 3.9 Το μοτίβο BCED

Το σχεδιαστικό μοτίβο που επιλέχτηκε για την υλοποίηση της εφαρμογής μας είναι το BCED. Το συγκεκριμένο μοτίβο αποτελείται από τέσσερα (4) επιμέρους επίπεδα (Boundary - Control - Entity - Db Interface), από τα οποία προκύπτει και το όνομα του.



Εικόνα 3.6 - Γραφική απεικόνιση της αρχιτεκτονικής BCED

#### 3.9.1 Boundary Layer

Το πρώτο επίπεδο της αρχιτεκτονικής μας, έχει σκοπό την παρουσίαση της πληροφορίας στο χρήστη μας καθώς και την λήψη πληροφορίας από αυτόν. Στο επίπεδο αυτό, λοιπόν, υπάρχουν οι JSPs, τα CSS και τα Javascript αρχεία.

### 3.9.2 Control Layer

Το επίπεδο ελέγχου σχεδιάστηκε σύμφωνα με το μοτίβο Model - View - Controller (MVC), το οποίο αποτελεί ένα σχεδιασμό αρχιτεκτονικής που χρησιμοποιείται στην τεχνολογία λογισμικού. Ο υπεύθυνος για τη σχεδίαση και την ανάπτυξη μιας εφαρμογής επιθυμεί πολλές φορές να διαχωρίσει τα δεδομένα (μοντέλα) από τη γραφική διεπαφή που εμφανίζεται στον χρήστη. Θέλουμε οποιαδήποτε αναδιοργάνωση συμβαίνει στα δεδομένα μας να μην επηρεάζει το γραφικό περιβάλλον, και αντίστροφα αλλαγές στο γραφικό περιβάλλον να μην έχουν επιπτώσεις στο επίπεδο επεξεργασίας δεδομένων. Το MVC λύνει αυτό το πρόβλημα με την αποσύνδεση της πρόσβασης και των ενεργειών που συμβαίνουν πάνω στα δεδομένα, από την παρουσίαση των στοιχείων και την αλληλεπίδραση με τους χρήστες, εισάγοντας ένα ενδιάμεσο επίπεδο ελέγχου (Control Layer). Στην εικόνα 3.5 φαίνεται η δομή του συγκεκριμένου μοτίβου.

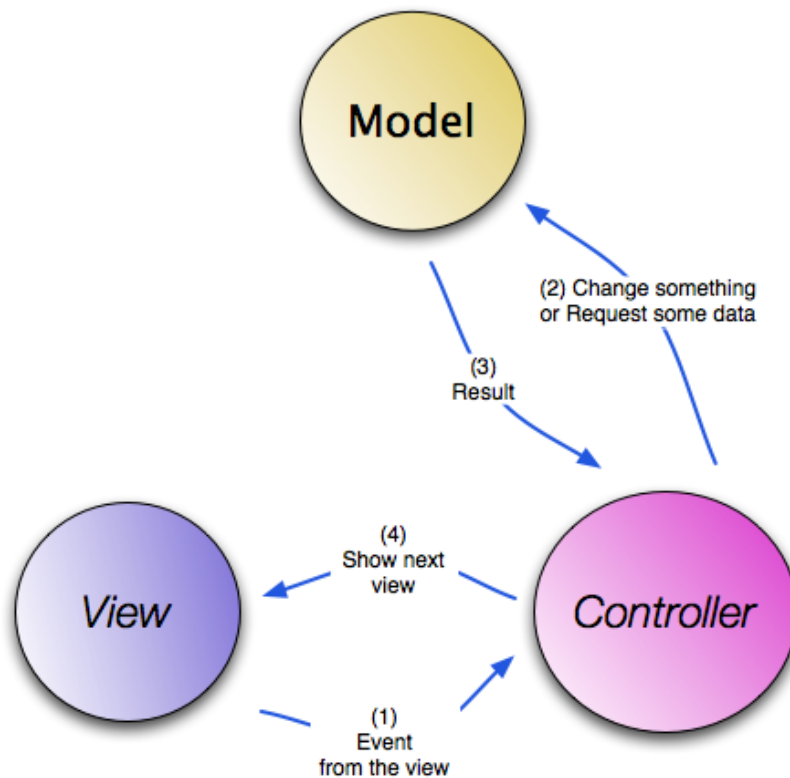
Τα τρία επίπεδα της δομής μας είναι:

**Model** : περιέχει την λογική που χρησιμοποιείται για τη διαχείριση και επεξεργασία της πληροφορίας, καθώς και την αναπαράσταση της πληροφορίας την οποία χρησιμοποιεί η εφαρμογή. Πολλά συστήματα χρησιμοποιούν μηχανισμούς αποθήκευσης πληροφορίας (persistency layer), όπως βάσεις δεδομένων ή αρχεία σε τοπικούς ή απομακρυσμένους δίσκους.

**View** : περιλαμβάνει τις γραφικές διεπαφές που χρησιμοποιούνται για την αλληλεπίδραση του χρήστη με την εφαρμογή.

**Controller** : επεξεργάζεται και αποκρίνεται σε γεγονότα, (π.χ. ενέργειες του χρήστη) και μπορεί να προκαλέσει αλλαγές στα δεδομένα του συστήματος.





Εικόνα 3.7 - Το σχεδιαστικό μοτίβο MVC.

### 3.9.3 Entity Layer

Στο επίπεδο αυτό έχουμε τις κλάσεις (entity classes) που υλοποιούν τη λογική της εφαρμογής και αναπαριστούν τα δεδομένα που εισάγονται ή εξάγονται από τη βάση δεδομένων. Οι κλάσεις αυτές περιλαμβάνουν, επίσης, τις μεθόδους που εκτελούν τις παραπάνω λειτουργίες. Επικοινωνεί με τα δύο παραπάνω επίπεδα, καθώς και με το κατώτερο που αφορά την επικοινωνία με την βάση δεδομένων.

### 3.9.4 DataBase Interface Layer

Στο τέταρτο και τελευταίο επίπεδο περιέχονται όλες αυτές οι κλάσεις, που αφορούν την αντιστοίχιση των κλάσεων οντοτήτων τους αντίστοιχους πίνακες στη βάση, έτσι ώστε να εξασφαλίζεται η επικοινωνία μεταξύ των κλάσεων οντοτήτων και της βάσης δεδομένων.

## 3.10 Εργαλεία Προγραμματισμού

### 3.10.1 NetBeans IDE 6.9.1

Το NetBeans IDE είναι ένα περιβαλλοντικό ανάπτυγμα IDE - ένα εργαλείο στους προγραμματιστές για να γράψουν, να κάνουν compile, debug και να αναπτύξουν προγράμματα. Είναι γραμμένο σε Java - αλλά μπορεί να υποστηρίξει όλες τις γλώσσες προγραμματισμού. Υπάρχει επίσης ένας μεγάλος αριθμός υπομονάδων (modules) που βοηθάνε στην επέκταση της λειτουργικότητας του NetBeans IDE. Το NetBeans IDE είναι ένα ελεύθερο προϊόν δίχως περιορισμούς στον τρόπο χρησιμοποίησής του.

### 3.10.2 MySQL Connector

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων, που επιτρέπει τη σύνδεση της βάσης με εφαρμογές που έχουν δημιουργηθεί χρησιμοποιώντας τη γλώσσα προγραμματισμού Java μέσω του JDBC (Java to Database Connectivity) driver, ο οποίος ονομάζεται MySQL Connector. Ο MySQL Connector αποτελεί έναν JDBC Type 4 driver, ο οποίος δηλώνει ότι το πρωτόκολλο της MySQL είναι γραμμένο εξ'ολοκλήρου σε γλώσσα Java και δεν βασίζεται καθόλου στις βιβλιοθήκες χρήσης της MySQL από τους χρήστες.

### 3.10.3 Navicat for MySQL

Για τη διαχείριση και υλοποίηση της βάσης δεδομένων μας χρησιμοποιήσαμε το Navicat. Ένα ισχυρό εργαλείο το οποίο μας παρέχει όλα όσα χρειαστήκαμε για τη ανάπτυξη της βάσης δεδομένων μας με MySQL.





# *Κεφάλαιο 4*

*Ανάλυση απαιτήσεων*

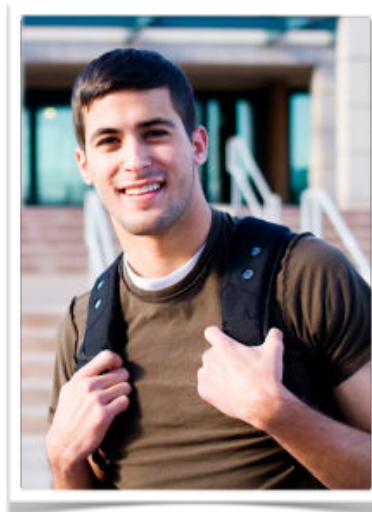
## 4.1 Εισαγωγή

Στο παρόν κεφάλαιο θα παρουσιαστούν οι απαιτήσεις του συστήματος μας, οι χρήστες του συστήματος μας καθώς και οι λόγοι για τους οποίους επιλέξαμε να υποστηρίξουμε την εκάστοτε λειτουργικότητα. Η αναλυτική περιγραφή της λειτουργικότητας θα γίνει στο κεφάλαιο 7 - Λειτουργικότητα του Συστήματος

## 4.2 Οι χρήστες του συστήματος

Η εφαρμογή μας έχει σχεδιαστεί για να υποστηρίζει μέρος της λειτουργικότητας του M.O.L.E. και κάποιες ακόμα πιο προσωπο-κεντρικές λειτουργίες για τους φοιτητές της σχολής. Οι χρήστες μας, λοιπόν, είναι άτομα από 18 μέχρι 28 ετών, οι οποίοι είναι ήδη χρήστες του M.O.L.E. και γνωρίζουν την λειτουργικότητα που παρέχεται ήδη από το σύστημα. Ακολουθούν κάποια παραδείγματα χρηστών ( personas). Τα personas αυτά καλύπτουν όλες τις μεγάλες κατηγορίες χρηστών μας. Δηλαδή είναι έμπειροι ή άπειροι χρήστες τους ιστοχώρου και αντίστοιχα έμπειροι ή άπειροι χρήστες smartphones.

### Θωμάς Τσιάκας, 24 ετών



Τελειόφοιτος φοιτητής του τμήματος ΗΜΜΥ του Πολυτεχνείου Κρήτης. Δεν χρησιμοποιεί πολύ τον ιστοχώρο. Κάνει τη διπλωματική του στον τομέα των Συστημάτων Αυτομάτου Ελέγχου. Χρησιμοποιεί λίγο το ιστοχώρο και δεν είναι εξοικειωμένος με την χρήση εφαρμογών σε smartphones. Εξοικειωμένος με τους όρους "ανακοινώσεις", "εργασίες", "εργαστήριο", "διάλεξη", "φροντιστήριο", "εργαστήριο", "προθεσμίες παράδοσης" κ.τ.λ.

### Χρησιμοποιεί το σύστημα για να:

- βλέπει τις ημερήσιες του υποχρεώσεις στην σχολή.
- ακολουθεί, όπου και αν βρίσκεται, συζητήσεις στα φόρουμ του τμήματος και να συμμετέχει άμεσα.
- έχει άμεση πρόσβαση στα μηνύματα του και να μπορεί να απαντήσει επιτόπου.
- ενημερώνεται έγκαιρα για νέες ανακοινώσεις στα μαθήματα που τον ενδιαφέρουν.

**Θεοδώρα Κεβά, 20 ετών**

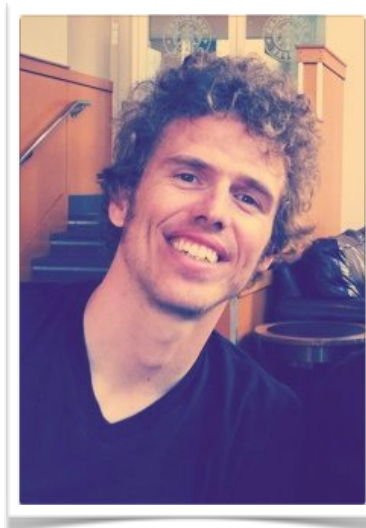


Δευτεροετής φοιτήτρια του τμήματος ΗΜΜΥ. Τακτική χρήστης του ιστοχώρου. Εξοικειωμένη με τους όρους “ανακοινώσεις”, “εργασίες”, “εργαστήριο”, “διάλεξη”, “φροντιστήριο”, “εργαστήριο”, “προθεσμίες παράδοσης” κ.τ.λ. Δεν είναι εξοικειωμένη με mobile applications παρόλο που είναι κάτοχος smartphone.

**Χρησιμοποιεί το σύστημα για να:**

- ενημερώνεται έγκαιρα για νέες ανακοινώσεις στα μαθήματα που την ενδιαφέρουν.
- παρακολουθεί τις προθεσμίες λήξης εργασιών.
- συμμετέχει στις γενικές συζητήσεις του τμήματος.
- βλέπει τις ημερήσιες της υποχρεώσεις στην σχολή.

**Παναγιώτης Κοτσαρίνης, 19 ετών**



Πρωτοετής φοιτητής του τμήματος ΗΜΜΥ. Δεν έχει κάνει εκτεταμένη χρήση του ιστοχώρου καθώς είναι νέος στο τμήμα. Πολύ εξοικειωμένος με mobile applications και χρησιμοποιεί καθημερινά το smartphone του για να μπαίνει στο ίντερνετ. Όχι ιδιαίτερα εξοικειωμένος με τους όρους της σχολής.

#### Χρησιμοποιεί το σύστημα για να:

- βλέπει τις ημερήσιες του υποχρεώσεις στην σχολή.
- χρησιμοποιεί τις υπηρεσίες των μαθημάτων (βαθμολογία, πρόσβαση σε υλικό μαθήματος)
- ακολουθεί συζητήσεις στα φόρουμ του τμήματος και να συμμετέχει άμεσα.
- έχει άμεση πρόσβαση στα μηνύματα του καθώς έχει ανάγκη την προσωπική επικοινωνία για να αντιμετωπίσει προβλήματα που παρουσιάζονται στη σχολή μέχρι να προσαρμοστεί.
- ενημερώνεται έγκαιρα για νέες ανακοινώσεις στα μαθήματα που τον ενδιαφέρουν.

#### Ελένη Ιωαννίδου, 22 ετών



4ο-ετής φοιτήτρια του τμήματος ΗΜΜΥ. Θέλει να ακολουθήσει τον τομέα Πληροφορικής και να ασχοληθεί μελλοντικά με την ανάπτυξη πληροφοριακών συστημάτων και λογισμικού. Κάνει πολύ συχνή χρήση του ιστοχώρου. Πολύ εξοικειωμένη με τους όρους της σχολής που υπάρχουν στον ιστοχώρο, "ανακοινώσεις", "εργασίες", "εργαστήριο", "διάλεξη", "φροντιστήριο", "εργαστήριο", "προθεσμίες παράδοσης" κ.τ.λ. Έχει smartphone το οποίο το χρησιμοποιεί συχνά είναι εξοικειωμένη με mobile εφαρμογές. Τόσο native όσο και web applications.

#### Χρησιμοποιεί το σύστημα για να:

- ενημερώνεται έγκαιρα για νέες ανακοινώσεις στα μαθήματα που την ενδιαφέρουν.
- παρακολουθεί τις προθεσμίες λήξης εργασιών της.
- συμμετέχει στις γενικές συζητήσεις του τμήματος καθώς και σε αυτές των μαθημάτων που την ενδιαφέρουν.
- βλέπει τις ημερήσιες της υποχρεώσεις στην σχολή.
- χρησιμοποιεί τις υπηρεσίες των μαθημάτων (βαθμολογία, πρόσβαση σε υλικό μαθήματος)



## 4.3 Επιλογή ενεργειών από το ήδη υπάρχον Courses που θα υποστηρίζονται

Οι mobile εκδόσεις των desktop εφαρμογών συνήθως υποστηρίζουν ένα μέρος της λειτουργικότητας δίνοντας έμφαση και σε άλλες λειτουργίες που περιστρέφονται γύρω από τον χρήστη. Οι λόγοι που το επιτάσσουν αυτό, είναι τόσο οι περιορισμοί των φορητών συσκευών (μέγεθος οθόνης, λειτουργικότητα που μπορεί να υποστηριχτεί από τις συσκευές, χρεώσεις με βάση τον όγκο των δεδομένων) όσο και η κατάσταση του χρήστη όταν χρησιμοποιεί τις εκάστοτε εφαρμογές.

Για να επιλέξουμε ποιες λειτουργίες έπρεπε να μεταφέρουμε στην δική μας εφαρμογή κάναμε τα εξής. Αρχικά ελέγξαμε τα στατιστικά του ιστοχώρου για να δούμε τις πιο “δημοφιλείς” λειτουργίες. Στη συνέχεια κάναμε μια έρευνα μέσω του φόρουμ του Courses (προκάτοχος του MOLE) για να στοχεύσουμε σε χρήστες που ήδη χρησιμοποιούν την εφαρμογή και ξέρουν ποια λειτουργικότητα είναι πιο χρήσιμη. Τέλος ρωτήσαμε διάφορους χρήστες, κάποιοι από τους οποίους στην συνέχεια συμμετείχαν και στην αξιολόγηση της εφαρμογής μας.

Γενικά τις λειτουργίες μπορούμε να τις χωρίσουμε σε δύο κατηγορίες. Αυτές που σχετίζονται με κάποιο συγκεκριμένο μάθημα και αυτές που έχουν συγκεντρωθεί γύρω από τον χρήστη μας.

### 4.3.1 Λειτουργίες με επίκεντρο συγκεκριμένο μάθημα (Course - oriented tasks)

1. Προβολή πληροφοριών μαθήματος
2. Ανακοινώσεις μαθήματος
3. Πρόσβαση στο Ψηφιακό Υλικό : Διαλέξεις, σημειώσεις φροντιστηρίου, χρήσιμα έγγραφα.
4. Προβολή και συμμετοχή στις συζητήσεις του μαθήματος
5. Προβολή βαθμολογίας μαθήματος
6. Προβολή Ημερολογίου
7. Χώροι εργασίας : πρόσβαση σε πολλαπλούς χώρους εργασίας, προβολή περιγραφής, πρόσβαση στο υλικό εργαστηρίου, πρόσβαση στις εργαστηριακές

ασκήσεις, ομάδες εργασίας (δημιουργία, εγγραφή, πρόσκληση και αποδοχή πρόσκλησης).

#### 4.3.2 Λειτουργίες με επίκεντρο τον χρήστη (User - oriented tasks)

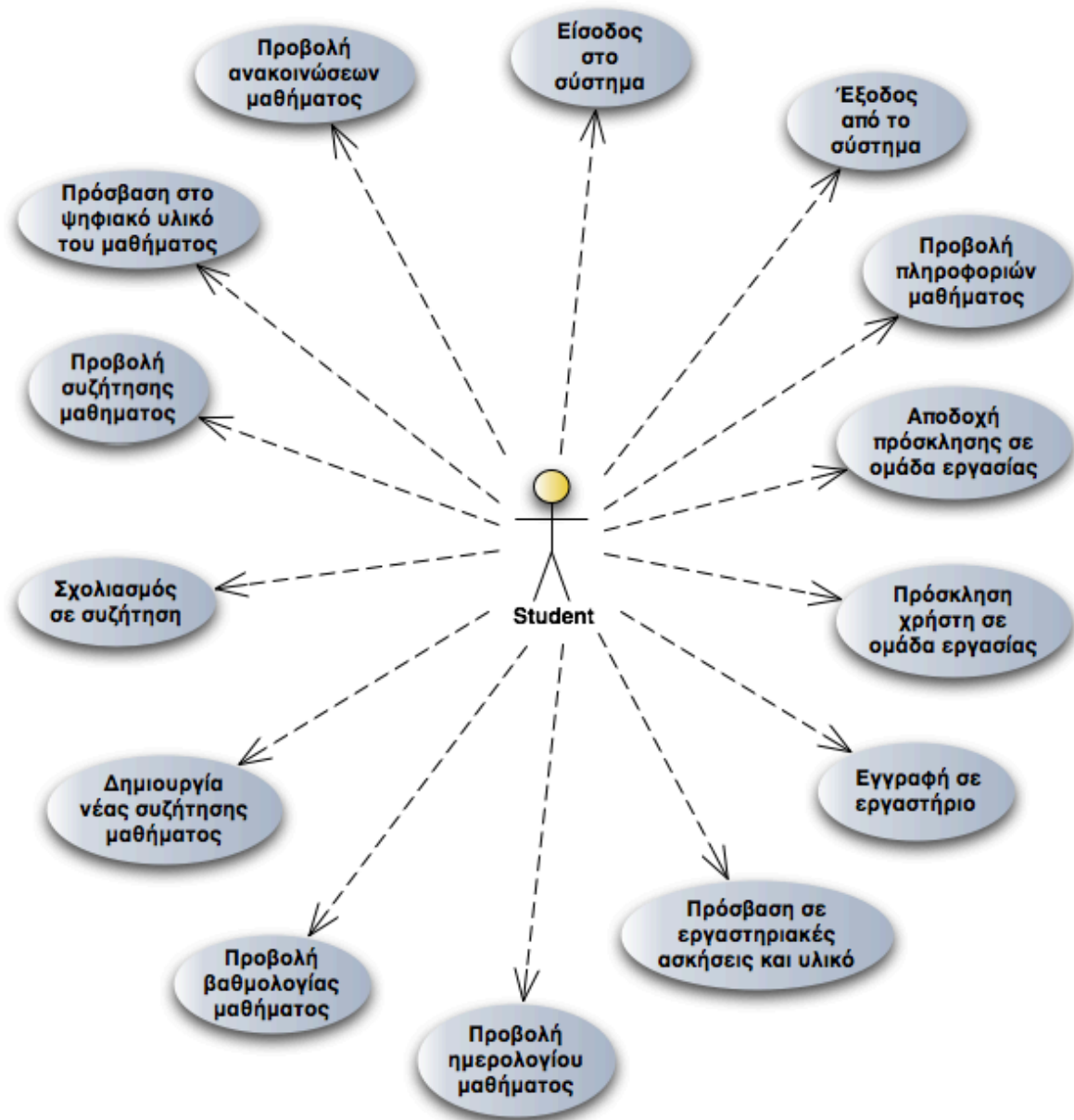
Όλες οι παρακάτω λειτουργίες συγκεντρώνουν πληροφορία από τα μαθήματα στα οποία είναι εγγεγραμμένος ο χρήστης και υπάρχουν στην αρχική σελίδα του χρήστη έτσι ώστε να υπάρχει άμεση πρόσβαση. Επίσης εμφανίζουν μία επισήμανση (notification) για νέα πληροφορία.

1. Προσωπικό Ημερολόγιο του χρήστη : Δείχνει τις υποχρεώσεις του συγκεκριμένου φοιτητή την τρέχουσα μέρα. Περιλαμβάνει Διαλέξεις, Φροντιστήρια, Εργαστήρια, καθώς και υπενθυμίσεις για λήξεις προθεσμίας εγγραφής σε εργαστήριο και παράδοσης εργασιών.
2. Προσωπικά μηνύματα του χρήστη : Εισερχόμενα, Εξερχόμενα, Αποστολή νέου
3. Νέες ανακοινώσεις : νέες ανακοινώσεις στα μαθήματα που είναι εγγεγραμμένος ο χρήστης.
4. Προθεσμίες : οι προθεσμίες παράδοσης εργασιών στα μαθήματα που είναι εγγεγραμμένος ο χρήστης.
5. Συζητήσεις μαθημάτων : νέες απαντήσεις σε συζητήσεις μαθημάτων που είναι εγγεγραμμένος ο χρήστης.

## 4.4 Λειτουργικότητα του συστήματος

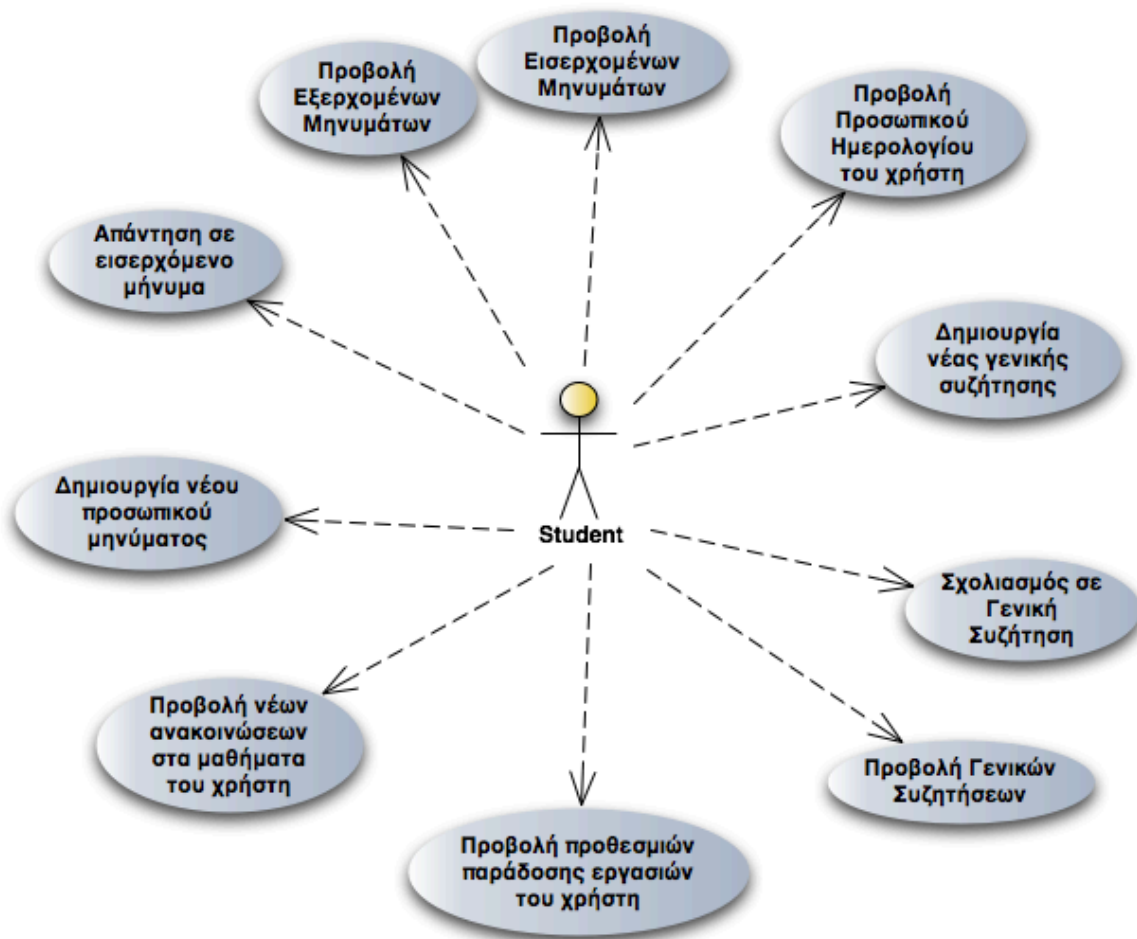
Μέχρι το σημείο αυτό έχει γίνει εκτενής ανάλυση απαιτήσεων του συστήματος μας. Περιγράφηκαν οι χρήστες του συστήματος καθώς και όλες οι λειτουργίες που θέλουμε να υποστηρίζονται από το σύστημα μας. Στην συνέχεια θα αναλύσουμε τη λειτουργικότητα με τη μέθοδο των Περιπτώσεων Χρήσης (Use Cases). Μια περίπτωση χρήσης λαμβάνει μια σύμβαση μεταξύ των ενδιαφερομένων μερών σχετικά με τη συμπεριφορά του συστήματος. Η περίπτωση χρήσης περιγράφει τη συμπεριφορά του συστήματος κάτω από διαφορετικές συνθήκες, δεδομένου ότι ανταποκρίνεται σε κάποιο αίτημα ενός από τα συμβαλλόμενα μέρη, το οποίο ονομάζεται πρωταρχικός παράγοντας. Ο πρωταρχικός παράγοντας ξεκινάει μια αλληλεπίδραση με το σύστημα προκειμένου να ολοκληρωθεί κάποιος στόχος. Το σύστημα ανταποκρίνεται ικανοποιώντας τις προσδοκίες των συμβαλλόμενων μερών. Μπορούν να ακολουθηθούν διαφορετικές εκδοχές της συμπεριφοράς των σεναρίων, ανάλογα με τα αιτήματα και τις συνθήκες που τα περιβάλλουν. Οι περιπτώσεις χρήσης είναι κατά βάση μια μορφή κειμένου, παρότι μπορούν να αναπαρασταθούν χρησιμοποιώντας διάφορα διαγράμματα, δίκτυα ή γλώσσες προγραμματισμού. Υπό κανονικές συνθήκες, χρησιμεύουν για την επικοινωνία με τους υπόλοιπους χρήστες του συστήματος, συνήθως με άτομα που δεν έχουν ειδική εκπαίδευση. Τις περισσότερες φορές το απλό κείμενο αποτελεί την καλύτερη επιλογή. Ακολουθούν ένα διάγραμμα με τα use cases για το χρήστη μας, ένας πίνακας που αντιπροσωπεύει τις περιπτώσεις χρήσης και περιέχει πληροφορίες σχετικά με το id, το όνομα, τον πρωταγωνιστή και το στόχο καθώς και μια σύντομη περιγραφή για την κάθε περίπτωση χρήσης.

#### 4.5 Διαγράμματα περιπτώσεων χρήσης και συνοπτικός πίνακας



Όλες οι ενέργειες προαπαιτούν ο χρήστης να είναι συνδεδεμένος στο σύστημα.

Εικόνα 4.1 - Use Case Diagram με τα Use Cases που σχετίζονται με κάποιο συγκεκριμένο μάθημα.



Όλες οι ενέργειες προαπαιτούν ο χρήστης να είναι συνδεδεμένος στο σύστημα.

Εικόνα 4.2 - Use Case Diagram με τα Use Cases που σχετίζονται με τον συγκεκριμένο χρήστη.

Στον πίνακα που ακολουθεί σε όλα τα tasks ο πρωταγωνιστής είναι ο “φοιτητής” αφού η εφαρμογή μας υποστηρίζει λειτουργικότητα που σχετίζεται μόνο με τους φοιτητές.

#	Επίπεδο	Στόχος	Περιγραφή
UC1		Εισαγωγή στο σύστημα.	Ο χρήστης μπαίνει στο σύστημα.
UC2		Έξοδος από το σύστημα	Ο χρήστης αποσυνδέεται από το σύστημα
UC3		Προβολή ανακοινώσεων μαθήματος	Ο χρήστης μπαίνει στη σελίδα με τις ανακοινώσεις μαθήματος
UC4		Πρόσβαση στο ψηφιακό υλικό του μαθήματος.	Ο χρήστης θέλει να δει κάποια διάλεξη, σημείωση ή κάποιο χρήσιμο έγγραφο από το ψηφιακό υλικό του μαθήματος.
UC5		Προβολή συζήτησης μαθήματος	Ο χρήστης μπαίνει να διαβάσει μια συζήτηση σε κάποιο μάθημα.

#	Επίπεδο	Στόχος	Περιγραφή
UC6		Σχολιασμός σε συζήτηση μαθήματος.	Ο χρήστης κάνει κάποιο σχόλιο σε μία συζήτηση μαθήματος.
UC7		Δημιουργία νέας συζήτησης μαθήματος	Ο χρήστης δημιουργεί μια νέα συζήτηση μαθήματος.
UC8		Προβολή βαθμολογίας μαθήματος	Ο χρήστης μπαίνει να δει τη βαθμολογία του σε κάποιο μάθημα.
UC9		Προβολή ημερολογίου μαθήματος	Ο χρήστης μπαίνει να δει το ημερολόγιο ενός μαθήματος
UC10		Πρόσβαση στο υλικό εργαστηρίου	Ο χρήστης θέλει να δει κάποια εργαστηριακή άσκηση ή κάποιο έγγραφο στο εργαστηριακό υλικό.
UC11		Εγγραφή σε εργαστήριο	Ο χρήστης κάνει εγγραφή σε κάποιο εργαστήριο.
UC12		Πρόσκληση χρήστη σε ομάδα εργασίας	Ο χρήστης καλεί κάποιον άλλον χρήστη για να συμμετέχει στην ομάδα εργασίας του σε κάποιο εργαστήριο.
UC13		Αποδοχή πρόσκλησης σε ομάδα εργασίας	Ο χρήστης αποδέχεται την πρόσκληση για κάποια ομάδα εργασίας.
UC14		Προβολή πληροφοριών μαθήματος	Ο χρήστης βλέπει τις πληροφορίες μαθήματος
UC15		Προβολή Εισερχομένων Μηνυμάτων	Ο χρήστης μπαίνει να δει κάποιο εισερχόμενο μήνυμα, στα προσωπικά του μηνύματα
UC16		Προβολή Εξερχομένων Μηνυμάτων	Ο χρήστης μπαίνει να δει κάποιο εξερχόμενο μήνυμα, στα προσωπικά του μηνύματα
UC17		Απάντηση σε εισερχόμενο μήνυμα	Ο χρήστης απαντάει σε κάποιο εισερχόμενο μήνυμα
UC18		Δημιουργία και αποστολή νέου προσωπικού μηνύματος	Ο χρήστης συντάσσει και αποστέλλει ένα νέο προσωπικό μήνυμα
UC19		Προβολή νέων ανακοινώσεων στα μαθήματα του χρήστη	Ο χρήστης μπαίνει να δει αν υπάρχουν νέες ανακοινώσεις στα μαθήματα που είναι εγγεγραμμένος
UC20		Προβολή προθεσμιών παράδοσης εργασιών	Ο χρήστης μπαίνει να δει τι προθεσμίες εκκρεμούν για παραδόσεις εργασιών στα μαθήματα στα οποία είναι εγγεγραμμένος
UC21		Προβολή γενικών συζητήσεων	Ο χρήστης μπαίνει να διαβάσει μια γενική συζήτηση
UC22		Σχολιασμός σε γενική συζήτηση	Ο χρήστης κάνει κάποιο σχόλιο σε μία γενική συζήτηση
UC23		Δημιουργία νέας γενικής συζήτησης	Ο χρήστης δημιουργεί μια νέα γενική συζήτηση
UC24		Προβολή προσωπικού ημερολογίου του χρήστη	Ο χρήστης μπαίνει στο προσωπικό του ημερολόγιο για να δει τις ημερήσιες υποχρεώσεις του

## 4.6 Αναλυτική περιγραφή Use Cases

### Περίπτωση χρήσης 1: Είσοδος στο σύστημα

Ο χρήστης συμπληρώνει την φόρμα εισόδου. Αν τα στοιχεία είναι έγκυρα μπαίνει στο σύστημα μας. Αν δεν είναι του εμφανίζεται μήνυμα λάθους και τον επιστρέφει πίσω στη φόρμα.

### Περίπτωση χρήσης 2: Έξοδος από το σύστημα

Ο χρήστης βρίσκεται στην αρχική σελίδα και επιλέγει “Έξοδο”. Του εμφανίζεται ένα μήνυμα επιβεβαίωσης. Αν ο χρήστης επιβεβαιώσει τότε αποσυνδέεται, αλλιώς επιστρέφει στην αρχική σελίδα.

### Περίπτωση χρήσης 3: Προβολή ανακοινώσεων μαθήματος

Ο χρήστης μπαίνει στη σελίδα του μαθήματος και επιλέγει “Ανακοινώσεις”. Εναλλακτικό σενάριο είναι να φτάσει στη σελίδα ανακοινώσεων του μαθήματος από τη σελίδα που έχει όλες τις ανακοινώσεις των μαθημάτων του χρήστη (δες Use Case 19).

### Περίπτωση χρήσης 4: Πρόσβαση στο ψηφιακό υλικό του μαθήματος

Ο χρήστης μπαίνει στη σελίδα μαθήματος και επιλέγει “Ψηφιακό Υλικό”. Εκεί μπορεί να επιλέξει “Διαλέξεις”, “Σημειώσεις Φροντιστηρίου” ή “Χρήσιμα Έγγραφα” και να δει τα αντίστοιχα έγγραφα.

*Σημείωση: Ανάλογα με το λειτουργικό της συσκευής η διαχείριση των εγγράφων θα γίνει με διαφορετικό τρόπο. Κάποιοι browsers έχουν την δυνατότητα να ανοίξουν έγγραφα (pdf, doc, xls) και άλλοι κάνουν λήψη των αρχείων αυτών στην συσκευή για να τα ανοίξουν με κάποια εφαρμογή.*

### Περίπτωση χρήσης 5: Προβολή συζήτησης μαθήματος

Ο χρήστης μπαίνει στη σελίδα του μαθήματος και επιλέγει “Συζήτηση”. Εκεί βλέπει σε λίστα τις πρώτες συζητήσεις του μαθήματος. Επιλέγει ποια θέλει να δει και πηγαίνει στην αντίστοιχη σελίδα. Αν θέλει να φορτώσει παλαιότερες συζητήσεις επιλέγει το “Περισσότερες Συζητήσεις”.

### Περίπτωση χρήσης 6: Σχολιασμός σε συζήτηση μαθήματος

Ο χρήστης έχει ήδη μπει σε μία συζήτηση ενός μαθήματος. Εκεί επιλέγει “Απάντηση” και του εμφανίζεται μια φόρμα. Την συμπληρώνει και πατάει “Απάντηση”. Του εμφανίζεται ένα μήνυμα ότι όλα έγιναν σωστά και επιστρέφει στην σελίδα της συζήτησης.

#### *Περίπτωση χρήσης 7: Δημιουργία νέας συζήτησης μαθήματος*

Ο χρήστης βρίσκεται στη σελίδα με τις συζητήσεις των μαθημάτων. Εκεί επιλέγει “Δημιουργία νέας συζήτησης”. Του εμφανίζεται η αντίστοιχη φόρμα, την συμπληρώνει και πατάει δημιουργία. Του εμφανίζεται μήνυμα ότι όλα πήγαν καλά και επιστρέφει στη σελίδα με τις συζητήσεις.

#### *Περίπτωση χρήσης 8: Προβολή βαθμολογίας μαθήματος*

Ο χρήστης μπαίνει στη σελίδα μαθήματος και επιλέγει “Βαθμολογία”, μεταβαίνοντας στην σελίδα με τις βαθμολογίες του στο αντίστοιχο μάθημα.

#### *Περίπτωση χρήσης 9: Προβολή ημερολογίου μαθήματος*

Ο χρήστης μπαίνει στη σελίδα μαθήματος και επιλέγει “Ημερολόγιο”. Μεταβαίνει στη σελίδα με το ημερολόγιο του μαθήματος, στο οποίο βλέπει το ημερολόγιο για τον τρέχον μήνα. Μπορεί να πλοηγηθεί σε επόμενους και προηγούμενους μήνες.

#### *Περίπτωση χρήσης 10: Πρόσβαση στο υλικό εργαστηρίου*

Ο χρήστης μπαίνει στη σελίδα μαθήματος. Επιλέγει “Χώροι Εργασίας” και πηγαίνει στη σελίδα με τα εργαστήρια του μαθήματος. Εκεί επιλέγει το εργαστήριο που θέλει και μεταβαίνει στην αντίστοιχη σελίδα του εργαστηρίου. Εκεί μπορεί να επιλέξει “Υλικό Εργαστηρίου” και να μεταβεί στα βοηθητικά έγγραφα του εργαστηρίου ή να επιλέξει “Ασκήσεις - Εργασίες” και να δει τις εργαστηριακές ασκήσεις.

#### *Περίπτωση χρήσης 11: Εγγραφή σε εργαστήριο*

Ο χρήστης φτάνει όπως και πριν στην σελίδα του εργαστηρίου. Εκεί μπαίνει στο “Ομάδες Εργασίας” και επιλέγει “Εγγραφή”. Αυτόματα του ανοίγουν οι επιλογές για πρόσκληση ή αποδοχή πρόσκλησης σε περίπτωση που τον έχει καλέσει κάποιος χρήστης. Αν έχει λήξει η προθεσμία εγγραφής στο εργαστήριο δεν εμφανίζεται καμία επιλογή.

#### *Περίπτωση χρήσης 12: Πρόσκληση χρήστη σε ομάδα εργασίας*

Ο χρήστης έχει εγγραφεί στο συγκεκριμένο εργαστήριο και πατάει “Πρόσκληση χρήστη”. Του εμφανίζεται μια φόρμα για να επιλέξει το χρήστη. Αν τα στοιχεία είναι έγκυρα του εμφανίζεται μήνυμα επιτυχημένης αποστολής πρόσκλησης. Αν όχι του εμφανίζεται το αντίστοιχο ενημερωτικό μήνυμα και τον επιστρέφει στην προηγούμενη σελίδα για να διορθώσει το λάθος του.



### *Περίπτωση χρήσης 13: Αποδοχή πρόσκλησης σε ομάδα εργασίας*

Ο χρήστης έχει εγγραφεί στο συγκεκριμένο εργαστήριο και πατάει αποδοχή πρόσκλησης από κάποιον χρήστη. Το σύστημα τον καταχωρεί στην ομάδα του χρήστη που τον κάλεσε και διαγράφει τις προσκλήσεις που έχουν γίνει στον χρήστη μας από άλλες ομάδες. Μπορεί να τις απορρίψει όλες

### *Περίπτωση χρήσης 14: Προβολή πληροφοριών μαθήματος*

Ο χρήστης μπαίνει στη σελίδα του μαθήματος και επιλέγει “Πληροφορίες Μαθήματος”, μεταβαίνοντας στη σελίδα με τις πληροφορίες του μαθήματος.

### *Περίπτωση χρήσης 15: Προβολή εισερχομένων μηνυμάτων*

Ο χρήστης μπαίνει στα προσωπικά του μηνύματα επιλέγοντας “Προσωπικά Μηνύματα” στην αρχική σελίδα. Εκεί επιλέγει “Εισερχόμενα” και μπαίνει στη σελίδα με τα εισερχόμενα μηνύματα του. Του εμφανίζονται τα πιο πρόσφατα μηνύματα και ο χρήστης επιλέγει ποιο θέλει να δει. Αν ο χρήστης θέλει να δει περισσότερα μηνύματα επιλέγει “Περισσότερα μηνύματα” και το σύστημα, με χρήση του Ajax τραβάει από τη βάση έναν αριθμό από παλαιότερα μηνύματα. Η διαδικασία μπορεί να επαναληφθεί μέχρι να τραβήξει ο χρήστης όλα τα μηνύματα του.

### *Περίπτωση χρήσης 16: Προβολή εξερχομένων μηνυμάτων*

Ο χρήστης μπαίνει στα προσωπικά του μηνύματα επιλέγοντας “Προσωπικά Μηνύματα” στην αρχική σελίδα. Εκεί επιλέγει “Εξερχόμενα” και μπαίνει στη σελίδα με τα εξερχόμενα μηνύματα του. Του εμφανίζονται τα πιο πρόσφατα μηνύματα και ο χρήστης επιλέγει ποιο θέλει να δει. Αν ο χρήστης θέλει να δει περισσότερα μηνύματα επιλέγει “Περισσότερα μηνύματα” και το σύστημα, με χρήση του Ajax τραβάει από τη βάση έναν αριθμό από παλαιότερα μηνύματα. Η διαδικασία μπορεί να επαναληφθεί μέχρι να τραβήξει ο χρήστης όλα τα μηνύματα του.

### *Περίπτωση χρήσης 17: Απάντηση σε εισερχόμενο μήνυμα*

Ο χρήστης μπαίνει σε κάποιο εισερχόμενο μήνυμα και επιλέγει “Απάντηση”. Του εμφανίζεται η φόρμα για την αποστολή μηνύματος με έτοιμο τον παραλήπτη. Συμπληρώνει τα στοιχεία και πατάει “Αποστολή”. Εμφανίζεται μήνυμα έγκυρης αποστολής και ο χρήστης μπορεί να επιλέξει αν θέλει να επιστρέψει στην “Αρχική Σελίδα” ή στα μηνύματα του.

#### *Περίπτωση χρήσης 18: Δημιουργία και αποστολή νέου προσωπικού μηνύματος*

Ο χρήστης μπαίνει στα “Προσωπικά Μηνύματα” και επιλέγει “Αποστολή Νέου”. Του εμφανίζεται η φόρμα και ο χρήστης συμπληρώνει τα στοιχεία: Προς (παραλήπτης), Τίτλος, Περιεχόμενο. Πατάει “Αποστολή” και στέλνει το μήνυμα. Αν τα στοιχεία του παραλήπτη δεν είναι έγκυρα εμφανίζεται ένα ενημερωτικό μήνυμα λάθους και ο χρήστης επιστρέφει πίσω. Αν τα στοιχεία είναι έγκυρα τότε εμφανίζεται μήνυμα έγκυρης αποστολής και ο χρήστης μπορεί να επιλέξει αν θέλει να επιστρέψει στην “Αρχική Σελίδα” ή στα μηνύματα του.

#### *Περίπτωση χρήσης 19: Προβολή νέων ανακοινώσεων στα μαθήματα του χρήστη*

Ο χρήστης βρίσκεται στην αρχική σελίδα και επιλέγει “Νέες Ανακοινώσεις”. Με την επιλογή αυτή μεταβαίνει στη σελίδα που δείχνει συγκεντρωτικά τις ανακοινώσεις του σε μαθήματα στα οποία είναι εγγεγραμμένος καθώς και το πλήθος ανακοινώσεων σε κάθε μάθημα. Επιλέγει το μάθημα που επιθυμεί και μεταβαίνει στη σελίδα ανακοινώσεων του αντίστοιχου μαθήματος.

#### *Περίπτωση χρήσης 20: Προβολή προθεσμιών παράδοσης εργασιών*

Ο χρήστης βρίσκεται στην αρχική σελίδα και επιλέγει “Οι προθεσμίες μου”. Εκεί του εμφανίζονται συγκεντρωτικά όλες οι προθεσμίες παράδοσης εργασιών που έχει. Επιλέγοντας κάποια από αυτές μεταβαίνει στη σελίδα

#### *Περίπτωση χρήσης 21: Προβολή γενικών συζητήσεων*

Ο χρήστης βρίσκεται στην αρχική σελίδα και επιλέγει “Γενικές Συζητήσεις”. Εκεί βλέπει σε λίστα τις πρώτες γενικές συζητήσεις. Επιλέγει ποια θέλει να δει και πηγαίνει στην αντίστοιχη σελίδα. Αν θέλει να φορτώσει παλαιότερες συζητήσεις επιλέγει το “Περισσότερες Συζητήσεις”.

#### *Περίπτωση χρήσης 22: Σχολιασμός σε γενική συζήτηση*

Ο χρήστης έχει ήδη μπει σε μία συζήτηση. Εκεί επιλέγει “Απάντηση” και του εμφανίζεται μια φόρμα. Την συμπληρώνει και πατάει “Απάντηση”. Του εμφανίζεται ένα μήνυμα ότι όλα έγιναν σωστά και επιστρέφει στην σελίδα της συζήτησης.

#### *Περίπτωση χρήσης 23: Δημιουργία νέα γενικής συζήτησης*

Ο χρήστης βρίσκεται στη σελίδα με τις γενικές συζητήσεις. Εκεί επιλέγει “Δημιουργία νέας συζήτησης”. Του εμφανίζεται η αντίστοιχη φόρμα, την συμπληρώνει και πατάει

δημιουργία. Του εμφανίζεται μήνυμα ότι όλα πήγαν καλά και επιστρέφει στη σελίδα με τις συζητήσεις.

#### *Περίπτωση χρήσης 24: Προβολή προσωπικού ημερολογίου χρήστη*

Ο χρήστης βρίσκεται στη σελίδα μαθήματος και επιλέγει “Το Ημερολόγιο Μου”. Μεταβαίνει στη σελίδα με το προσωπικό ημερολόγιο του, στο οποίο βλέπει τις υποχρεώσεις του. Μπορεί να πλοηγηθεί σε επόμενους και προηγούμενους μήνες.



# *Κεφάλαιο 5*

*Γραφική Διεπαφή*

## 5.1 Εισαγωγή

Στο κεφάλαιο αυτό γίνεται αναλυτική περιγραφή των αρχών σχεδίασης που ακολουθήθηκαν, της αρχιτεκτονικής της πληροφορίας μας καθώς και γενικότερη περιγραφή ολόκληρης της γραφικής σχεδίασης (visual design) της εφαρμογής.

## 5.2 Αρχές Σχεδίασης

Ένα σημείο στο οποίο έπρεπε να δώσω ιδιαίτερη προσοχή κατά την ανάπτυξη της εφαρμογής μου ήταν η σχεδίαση και οι αρχές που έπρεπε να ακολουθηθούν. Οι φορητές συσκευές έχουν πολλές ιδιαιτερότητες τις οποίες πρέπει να λάβουμε υπόψη μας. Τα τρία στοιχεία πάνω στα οποία στηρίχτηκε η σχεδίαση μας είναι τα εξής:

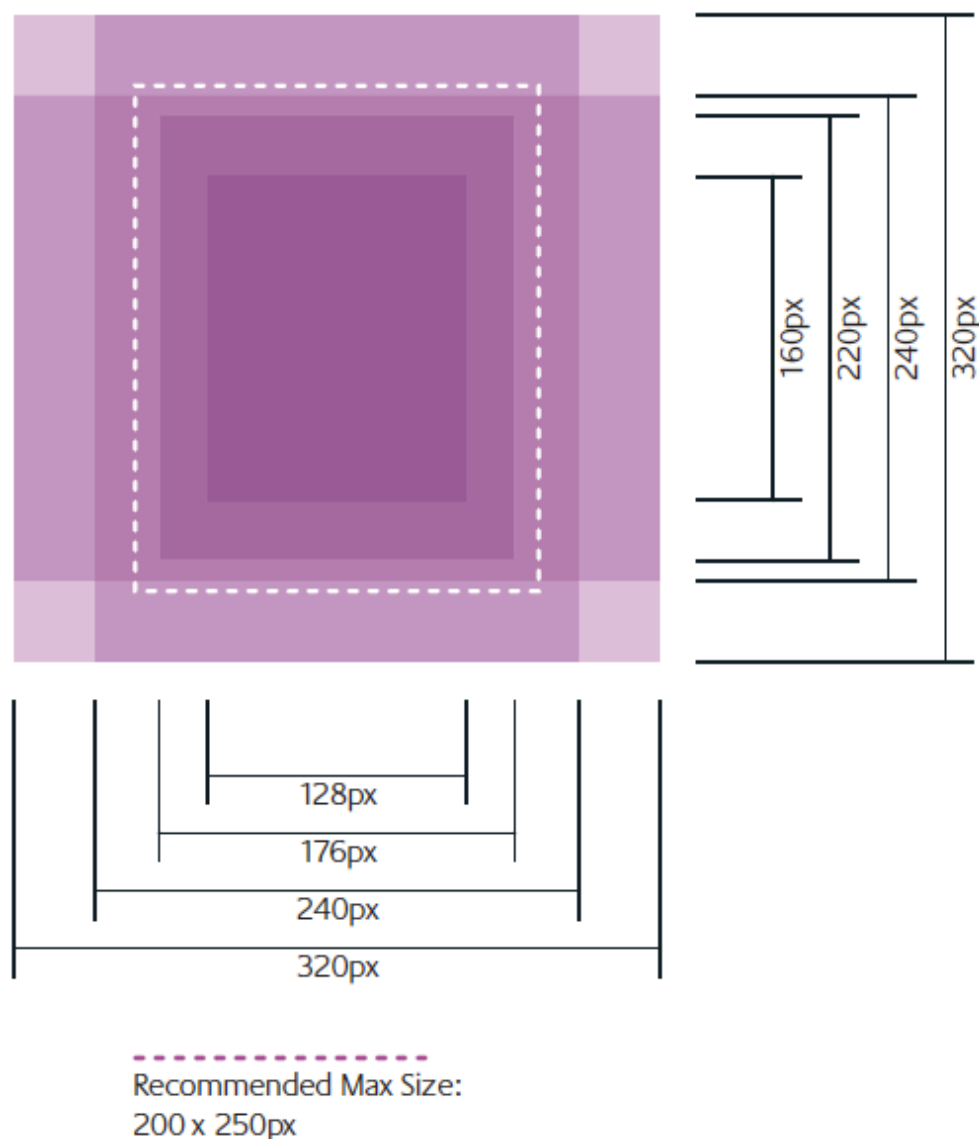
1. Γενικότερο πλαίσιο (Context)
2. Αρχιτεκτονική της πληροφορίας (Information Architecture)
3. Οπτική Σχεδίαση (Visual Design)

### 5.2.1 Γενικότερο πλαίσιο (Context)

Το γενικότερο πλαίσιο αναφέρεται στο νοητικό μοντέλο το οποίο οι χρήστες αποκτούν για να κατανοήσουν, και το μέσο ή το περιβάλλον μέσα στο οποίο θα εκτελέσουν τις εκάστοτε ενέργειες, καθώς και τις συνθήκες κάτω από τις οποίες πρέπει οι χρήστες να κατανοήσουν τι και πως πρέπει να το κάνουν. Και εδώ υπάρχουν τρεις τύποι που μας ενδιαφέρουν.

1. Παρούσα τοποθεσία (physical context) : στην περίπτωση μας οι χρήστες μπορούν να έχουν πρόσβαση στην εφαρμογή μας από οπουδήποτε. Κυρίως όμως η χρήση της εφαρμογής μας θα γίνεται όταν οι φοιτητές μας δεν έχουν πρόσβαση σε κανονικό υπολογιστή. Αυτό σημαίνει ότι δεν μπορούμε να προβλέψουμε τις συνθήκες που θα επικρατούν την ώρα της χρήσης. Για παράδειγμα δεν ξέρουμε αν θα είναι νύχτα ή μέρα και αν ο φωτισμός είναι επαρκής ή όχι. Αυτό ήταν μία από τις παραμέτρους, η οποία μας καθοδήγησε στην επιλογή των χρωμάτων που θα χρησιμοποιούσαμε, έτσι ώστε να έχουμε ένα interface που θα παραμένει ευανάγνωστο ανεξάρτητα από τις εξωτερικές συνθήκες.
2. Παρούσα συσκευή (media context) : δεδομένου ότι στοχεύαμε σε smartphones και tablets ξέραμε το μέσο με το οποίο οι χρήστες μας θα χρησιμοποιούσαν την εφαρμογή

μας. Αυτό ήταν μια δεύτερη παράμετρος που καθόρισε τη σχεδίαση μας. Η χρήση οθόνης αφής έκανε σαφές το ότι έπρεπε οι επιλογές του χρήστη να είναι ευδιάκριτες, με μεγάλα κουμπιά και ξεκάθαρους τίτλους, και η αντικατάσταση του κανονικού πληκτρολογίου από το touch keyboard των φορητών συσκευών, ότι έπρεπε να διευκολύνουμε το χρήστη στην περίπτωση εισαγωγής κειμένου. Επίσης η έλλειψη ενός εύχρηστου πληκτρολογίου, γεγονός που αναμφίβολα θα οδηγούσε σε λάθη, έκανε αναγκαία την ύπαρξη σωστών και άμεσων μηνυμάτων λάθους, τα οποία θα βοηθούσαν τον χρήστη να διορθώσει το λάθος του. Κάτι ακόμα που έπρεπε να σκεφτούμε ήταν ότι το interface μας έπρεπε να προσαρμόζεται στην αλλαγή προσανατολισμού της οθόνης μας, δηλαδή portrait ή landscape mode (Δες: Ενότητα 6.2 - Σχεδίαση για διαφορετικές οθόνες).



Εικόνα 5.1 - Portrait και Landscape mode. Φαίνεται για διάφορα ενδεικτικά μεγέθη οθονών, το τι βλέπει ο χρήστης και τι περιθώρια υπάρχουν για να χωρέσει το interface μας.

3. Παρούσα πνευματική κατάσταση (modal context) : όπως στις περισσότερες mobile εφαρμογές έτσι και στη δική μας, είναι ιδιαίτερα δύσκολο να προβλέψουμε την πνευματική κατάσταση του χρήστη όταν πάει να χρησιμοποιήσει την εφαρμογή μας. Ο χρήστης μας μπορεί να είναι χαλαρός στο σπίτι του και να χαζεύει τις υποχρεώσεις του στη σχολή για τις επόμενες μέρες (μέσω του προσωπικού ημερολόγιου) ή μπορεί να τρέχει αγχωμένος να προλάβει την παράδοση μιας εργασίας ή κάποιο εργαστήριο και να θέλει να δει τι άλλο έχει να κάνει στη σχολή. Αυτό ήταν ακόμα μια παράμετρος που μας υπέδειξε τους συνδυασμούς χρωμάτων που θα χρησιμοποιήσουμε καθώς θέλαμε να είναι ξεκάθαρο το τι κάνει κάθε κουμπί και τι επιλογές υπάρχουν, διατηρώντας πάντα μια συνέπεια, καθώς επίσης και το οι επιλογές αυτές πρέπει να είναι εύκολα προσβάσιμες αποτρέποντας το χρήστη να κάνει λάθη λόγω βιασύνης (slip errors). Τα παραπάνω περιγράφονται με λεπτομέρεια στην οπτική σχεδίαση της εφαρμογής.

*“Οι χρήστες δεν ενδιαφέρονται για το μέσο, την πνευματική τους κατάσταση ή τις υπάρχουσες συνθήκες. Μια καλή σχεδίαση πρέπει να κάνει όλα αυτά τα στοιχεία αόρατα στο χρήστη.” ~ Brian Fling*

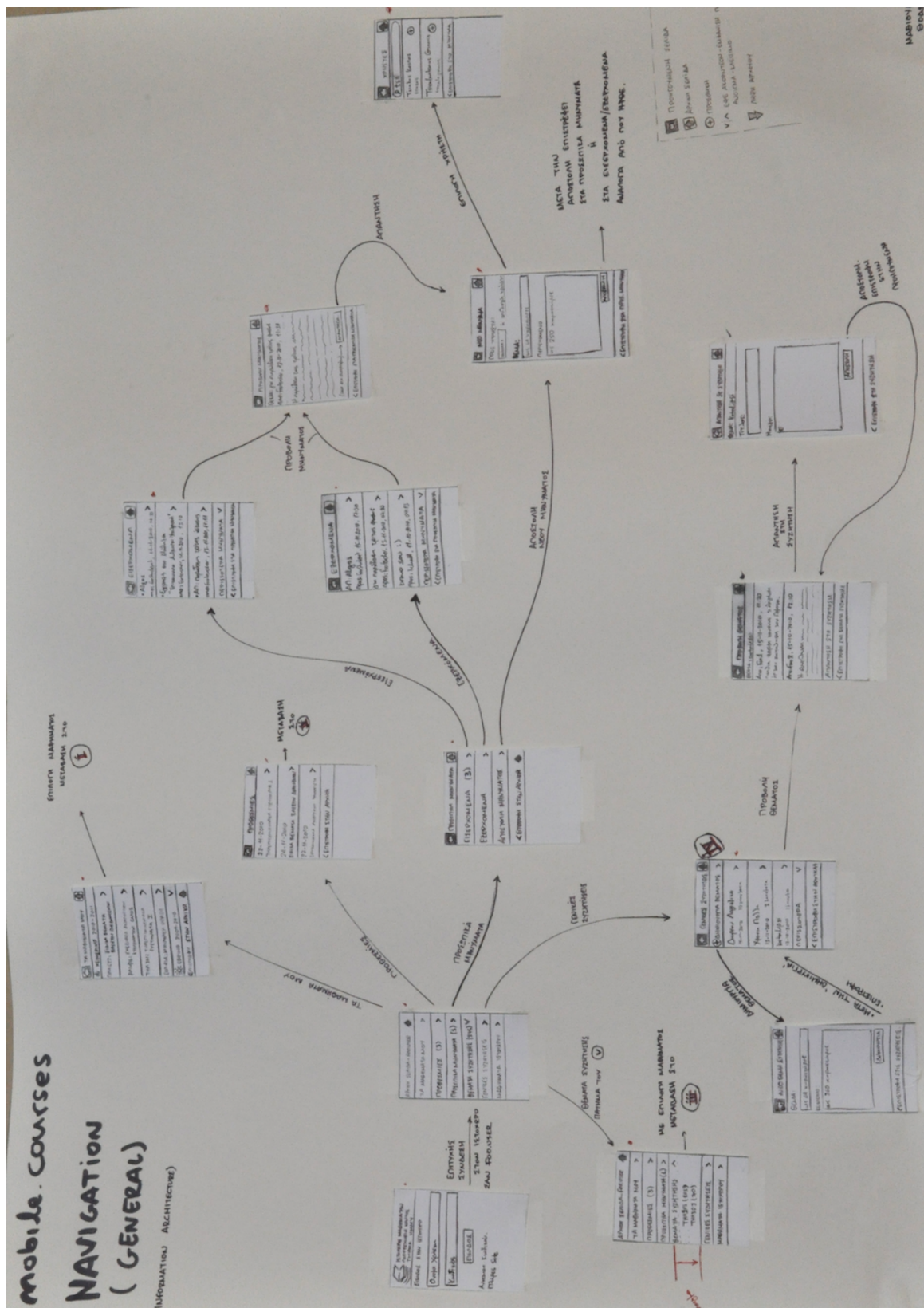
### 5.2.2 Αρχιτεκτονική της πληροφορίας (Information Architecture)

Η αρχιτεκτονική της πληροφορίας ορίζει τόσο τη δομή της πληροφορίας, όσο και τον τρόπο με τον οποίο οι χρήστες αλληλεπιδρούν με αυτή. Πρέπει να έχουμε υπόψη μας ότι όταν οι χρήστες χρησιμοποιούν την εφαρμογή μας είναι πολύ πιθανόν να χρεώνονται, είτε με τον βάση τη διάρκεια που διαρκεί η σύνδεση τους (σε κάποιο wifi που πληρώνουν με την ώρα) είτε με βάση τα δεδομένα που χρησιμοποιούν (σύνδεση μέσω δικτύου τηλεφωνίας και 3G). Αυτό σημαίνει ότι πρέπει να προσπαθήσουμε να μειώσουμε όσο γίνεται τον όγκο της πληροφορίας που μεταφέρεται και να έχουν μικρό βάθος διαφορετικών σελίδων οι ενέργειες του χρήστη ('click investment'). Γίνεται σαφές ότι τα λάθη του χρήστη “πληρώνονται” κυριολεκτικά. Ο χρήστης πληρώνει όχι μόνο για τις λάθος σελίδες που επισκέπτεται αλλά και όσες πρέπει να ξαναεπισκεφθεί για να εκτελέσει σωστά την ενέργεια που ήθελε εξ'αρχής. Ωστόσο με τη χρήση της HTML5 και το caching που εφαρμόζεται, τουλάχιστον γλιτώνουμε την φόρτωση των σελίδων που έχουμε ήδη επισκεφτεί, με αποτέλεσμα να χρεωνόμαστε μόνο τις επιπλέον στις οποίες βρεθήκαμε από λάθος. Όσο πιο 'καθαρός' και εύχρηστος είναι ο σχεδιασμός μας τόσο θα μειώνεται και η πιθανότητα να κάνει λάθη ο χρήστης. Επίσης πρέπει να έχουμε πολύ μικρής

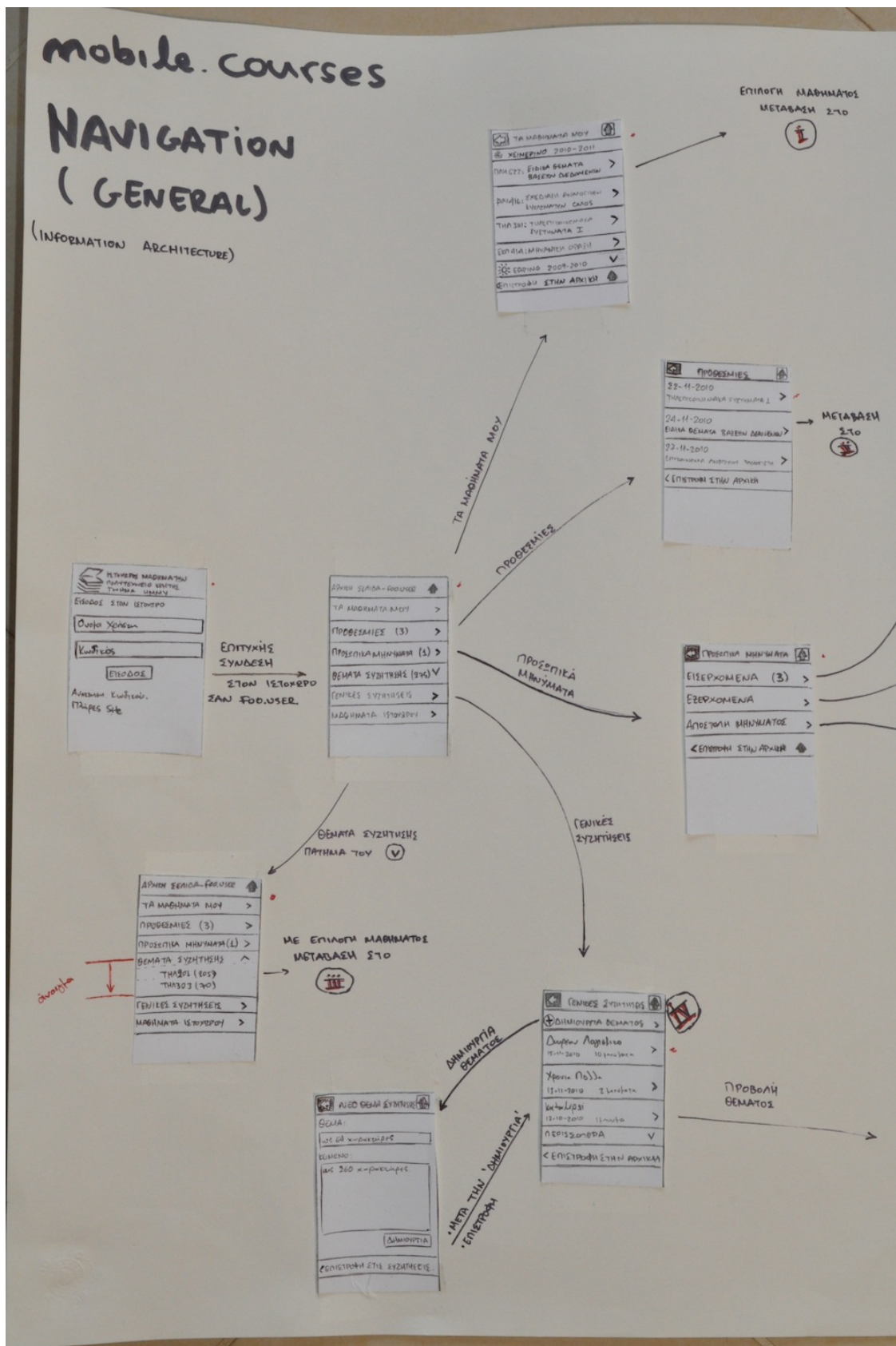


διάρκειας tasks τα οποία εκτελούνται σε περιορισμένο χρόνο. Για να μπορέσουμε να δούμε συνολικά τα tasks που εκτελούνταν και το βάθος τους στην εφαρμογή, καθώς και όλα τα μονοπάτια συνδυασμένα φτιάξαμε ένα clickstream με όλες τις σελίδες της εφαρμογής μας. Η ομαδοποίηση έγινε σε δύο επίπεδα. Στο ένα επίπεδο υπάρχουν οι σελίδες που αφορούν τον χρήστη συγκεκριμένα και στο άλλο οι σελίδες που σχετίζονται με τα μαθήματα και τις υπηρεσίες τους.

Ακολουθεί το clickstream για τις σελίδες που αφορούν τον χρήστη. Αρχικά ένα γενικό πλάνο για να γίνει κατανοητό το μοντέλο και στη συνέχεια σπάμε σε μικρότερα τμήματα τη σελίδα για να είναι ευανάγνωστη.

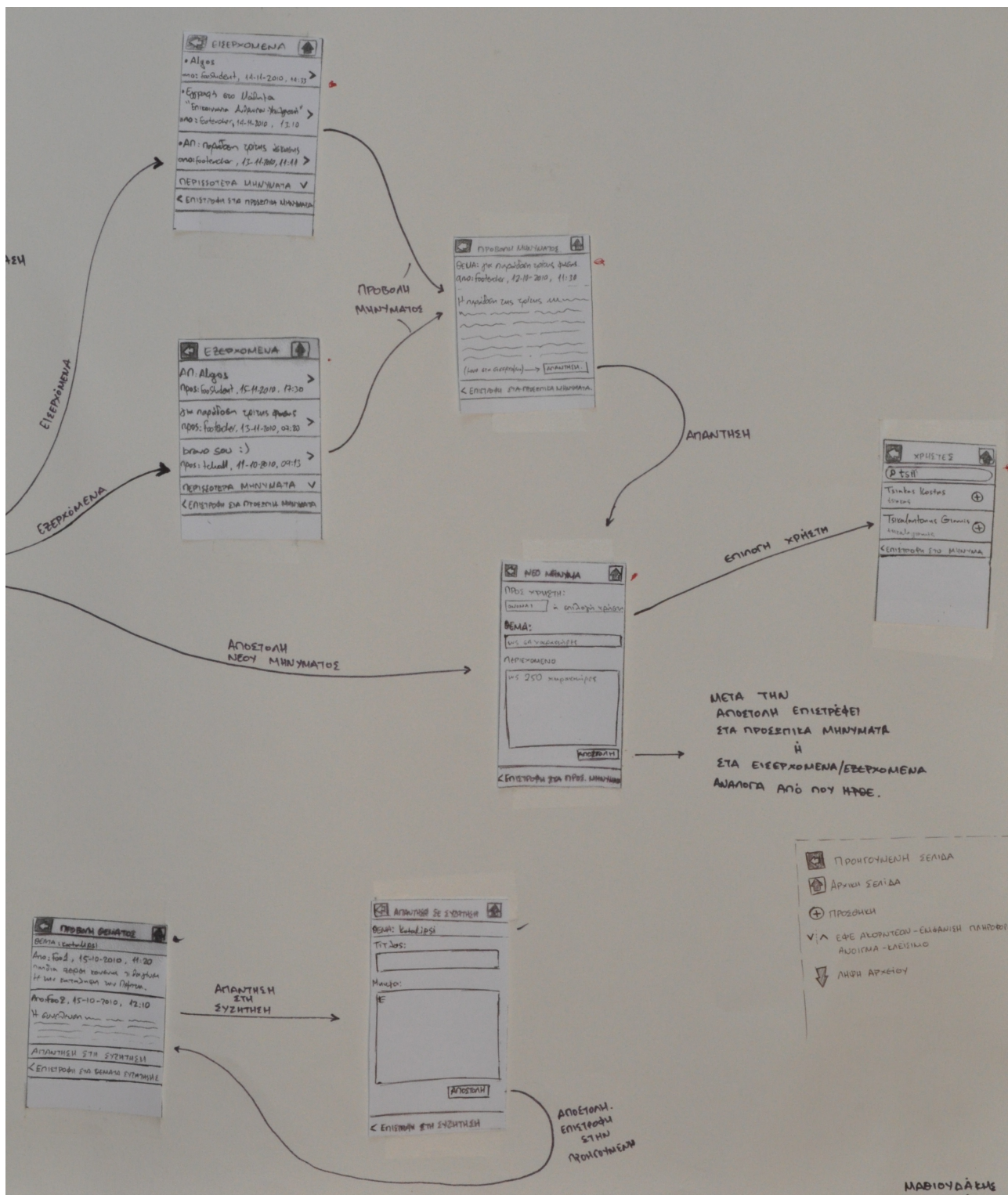


Εικόνα 5.2 - Σελίδες που σχετίζονται με τις ενέργειες του χρήστη. Το clickstream ξεκινάει από τη σελίδα εισόδου στο σύστημα και καταλήγει στην ολοκλήρωση των διαδικασιών που σχετίζονται με ενέργειες του χρήστη.



Εικόνα 5.3 - Σελίδες που σχετίζονται με το χρήστη. Πρώτο τμήμα από αριστερά.



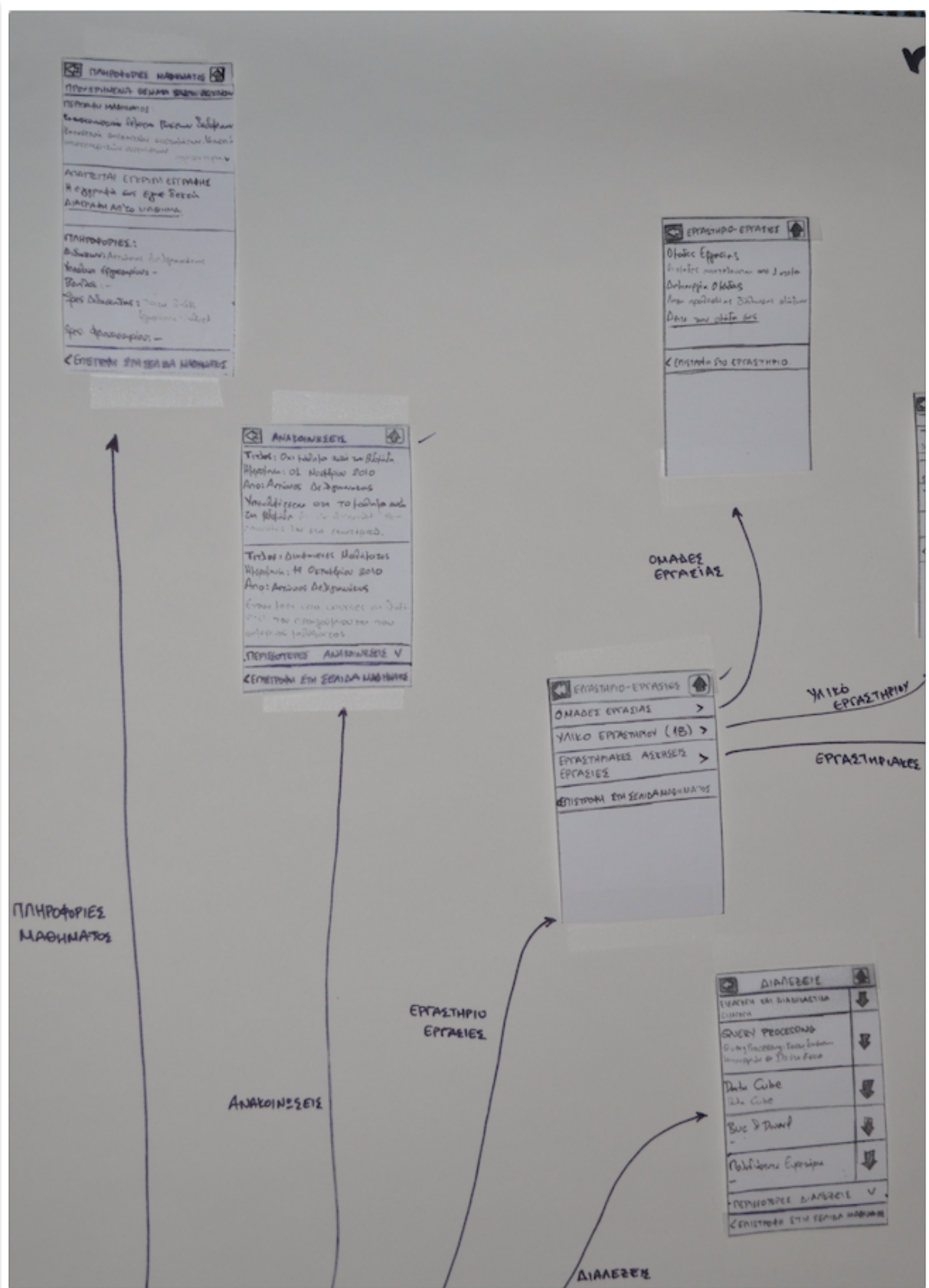


Εικόνα 5.4 - Σελίδες που σχετίζονται με το χρήστη. Δεύτερο τμήμα από αριστερά.

Ακολουθεί το clickstream για τις σελίδες που αφορούν τα μαθήματα. Αρχικά ένα γενικό πλάνο και στη συνέχεια σπάσαμε τη σελίδα για να είναι ευανάγνωστη όπως και προηγουμένως.





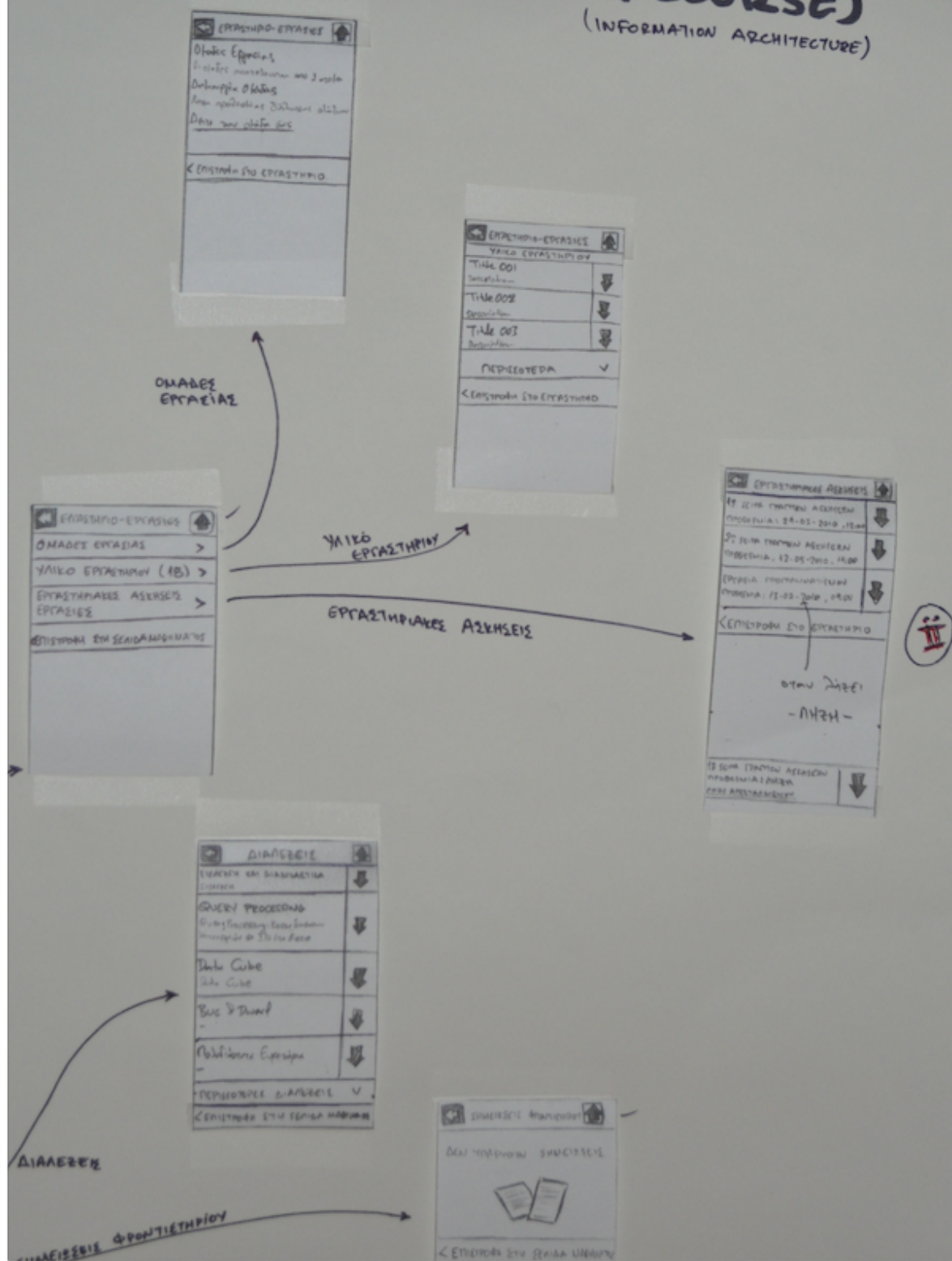


Εικόνα 5.7 - Σελίδες που σχετίζονται με τα μαθήματα. Πάνω αριστερό τμήμα.

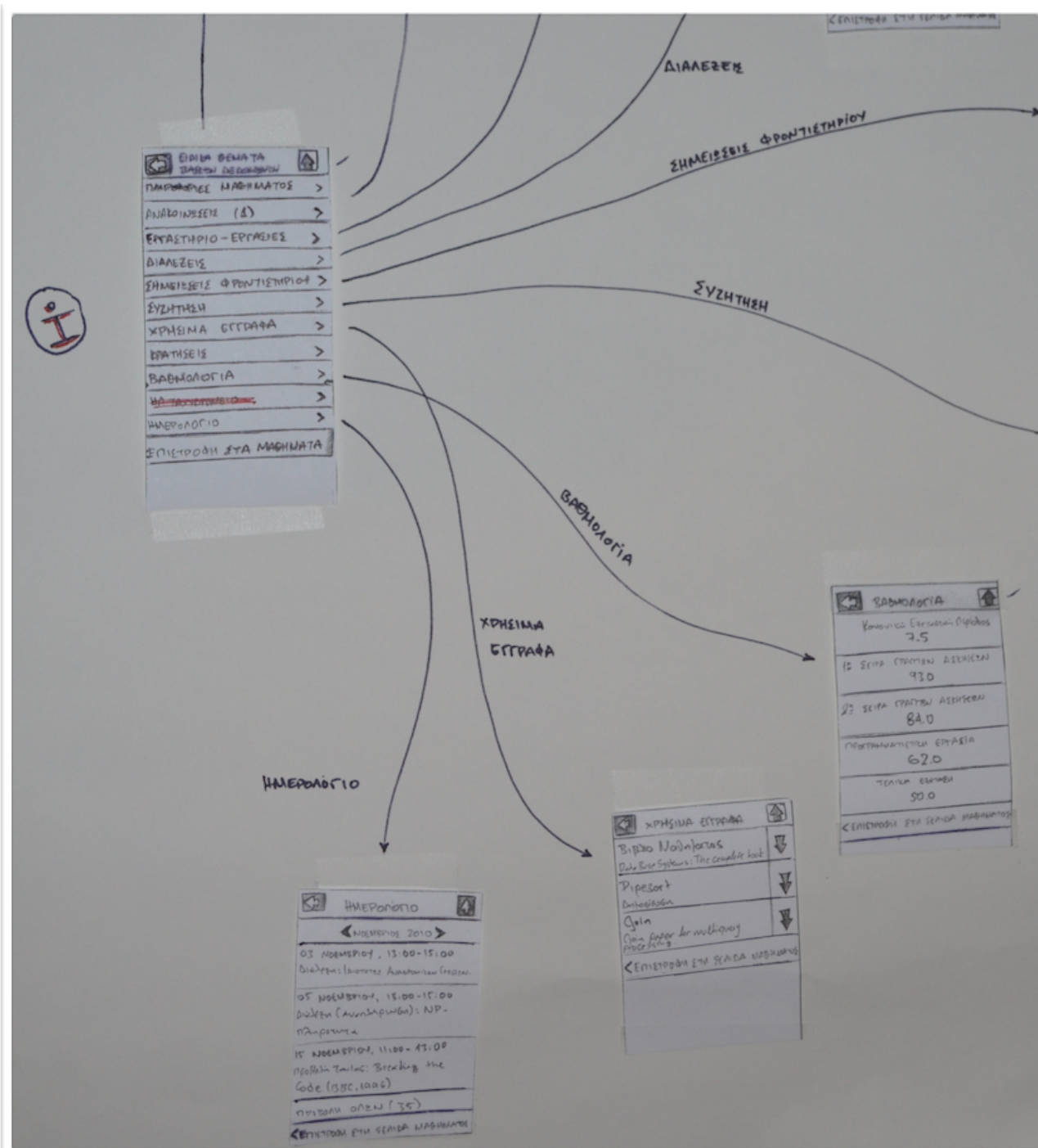
# mobile. Courses

## NAVIGATION (COURSE)

(INFORMATION ARCHITECTURE)

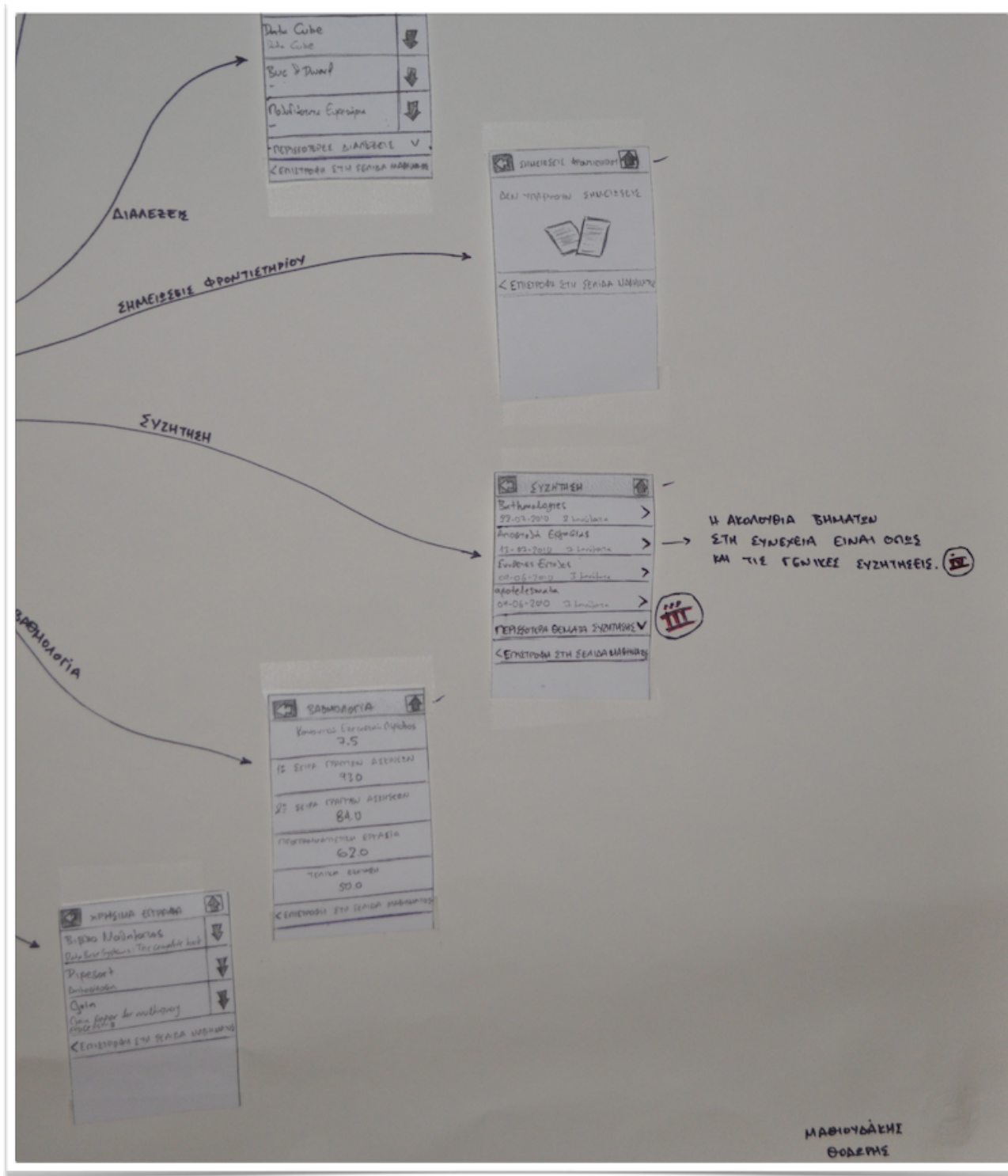


Εικόνα 5.8 - Σελίδες που σχετίζονται με τα μαθήματα. Πάνω δεξιό τμήμα.



Εικόνα 5.9 - Σελίδες που σχετίζονται με τα μαθήματα. Κάτω αριστερό τμήμα..





Εικόνα 5.10 - Σελίδες που σχετίζονται με τα μαθήματα. Κάτω δεξιό τμήμα.

Σημείωση: στην πορεία έγιναν κάποιες αλλαγές γιατί έπρεπε να προσαρμόσουμε τη σχεδίαση μας στο MOLE αντί για το Courses που είχαμε ξεκινήσει. Οι αλλαγές ήταν μικρές, και αναφέρονταν στην ομαδοποίηση 1-2 πραγμάτων καθώς και στη μετονομασία κάποιων άλλων, και δεν κρίναμε απαραίτητο να ξαναφτιάξουμε εξ'αρχής ένα sitemap, καθώς ουσιαστικά προστέθηκαν 2 νέες σελίδες με σαφείς επιλογές που δεν δημιουργούσαν πρόβλημα στο χρήστη.

### 5.2.3 Οπτική Σχεδίαση (Visual Design)

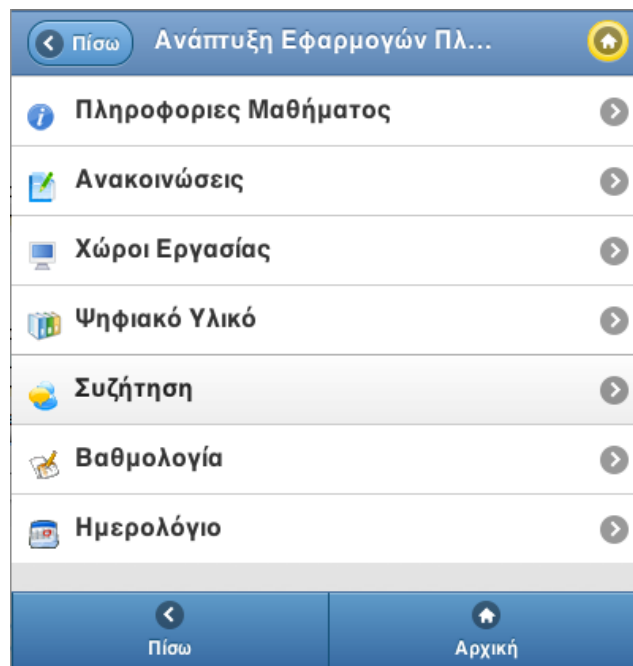
Υπάρχουν πολλές μεταβλητές που πρέπει να ελέγχουμε για να μπορούμε να ελέγχουμε συνολικά την σχεδίαση. Το μέγεθος της οθόνης, ο web browser, οι ρυθμίσεις του κειμένου, η ταχύτητα της σύνδεσης κ.α. Το να δημιουργήσουμε ένα ακριβές design και να υπάρχει συνέπεια σε όλες τις συσκευές θα ήταν ιδιαίτερα δύσκολο να υλοποιηθεί. Ωστόσο το ότι χρησιμοποιήσαμε τεχνολογίες που χρησιμοποιούνται απ'όλους τους σύγχρονους web browsers (CSS, Javascript, jQuery) μας βοήθησε πολύ στο να επιτύχουμε σε ένα μεγάλο βαθμό τη συνέπεια αυτή.

Τα χαρακτηριστικά στα οποία έπρεπε να δοθεί έμφαση εδώ ήταν:

1. Layout : πως θα εμφανίζεται η σελίδα στον χρήστη. Πως έπρεπε να δομηθεί έτσι ώστε να μην έχει πρόβλημα στις διαφορετικές οθόνες και στις διαφορετικές συσκευές. Λεπτομερής ανάλυση στην ενότητα "6.2 - Σχεδίαση για διαφορετικές οθόνες."
2. Χρώματα : λεπτομερής ανάλυση στην ενότητα "6.7 - Επιλογή χρωμάτων."
3. Τυπογραφία (Typography) : οι γραμματοσειρές που είναι διαθέσιμες σε όλες τις πλατφόρμες είναι οι Serif και Sans-Serif. Οπότε δεν πειράξαμε τις γραμματοσειρές που είχε εξ'αρχής το jQuery Mobile και οι οποίες είναι : "Helvetica, Arial, sans-serif". Οι γραμματοσειρές παρέχουν αναγνωσιμότητα εξαιτίας της απλότητας και της "καθαρότητας" τους. Επίσης το jQuery Mobile πας παρείχε όλα τα χαρακτηριστικά που απαιτούσαμε από το interface, δηλαδή καλή αντίθεση χρωμάτων (στην επιλογή του θέματος που επιλέξαμε εμείς), κενά δεξιά και αριστερά των κειμένων.
4. Γραφικά (Graphics) : Αποφύγαμε τη χρήση εικόνων στην εφαρμογή μας για δύο λόγους. Πρώτον για να μειώσουμε το μέγεθος της πληροφορίας που μεταφέρεται και δεύτερον για να μην βαρύνουμε την εμφάνιση της εφαρμογής μας. Έγινε περιορισμένη χρήση εικονιδίων τα οποία προσπαθήσαμε να είναι ευδιάκριτα και ξεκάθαρα ως προς τον σκοπό που εξυπηρετούν, καθώς και να είναι οικεία στους χρήστες. Το δεύτερο το επιτύχαμε κάνοντας χρήση όπου ήταν δυνατόν των εικονιδίων που υπάρχουν ήδη στο courses (και στο MOLE).



Εικόνα 5.11 - Σελίδες με εικονίδια - Αρχική Σελίδα Χρήστη

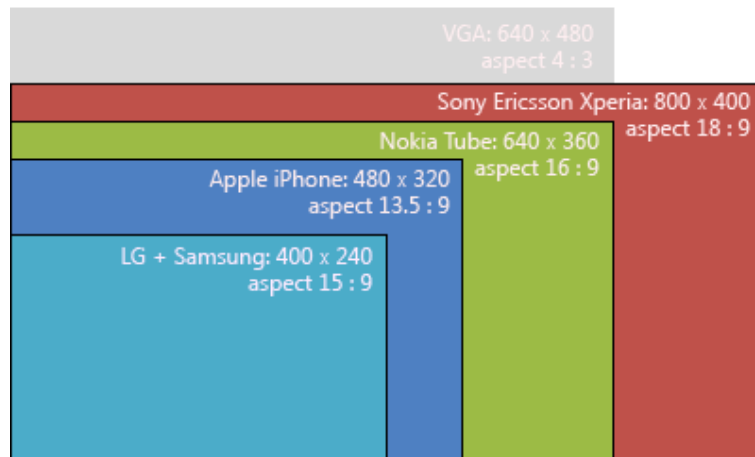


Εικόνα 5.12 - Σελίδες με εικονίδια - Σελίδα Μαθήματος

### 5.3 Σχεδίαση για διαφορετικές οθόνες

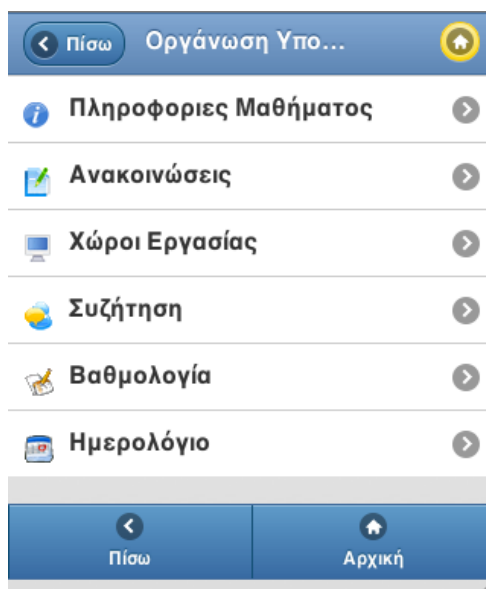
Ένα από τα κύρια προβλήματα που αντιμετωπίζει όποιος σχεδιάζει web applications για φορητές συσκευές είναι η ποικιλία διαφορετικών οθονών που έχει να υποστηρίξει. Σε αντίθεση με τους υπολογιστές που οι διαστάσεις παραμένουν σχεδόν σταθερές και οι

αναλογίες είναι πάνω κάτω ίδιες, στις φορητές συσκευές αυτό δεν ισχύει. Εκτός από διαφορετικές αναλύσεις έχουμε και κατά ένα μεγάλο ποσοστό από διαφορετικές αναλογίες στις οθόνες, παρόλο που όλα τα smartphones έχουν στενές και μακριές παραλληλόγραμμες οθόνες.

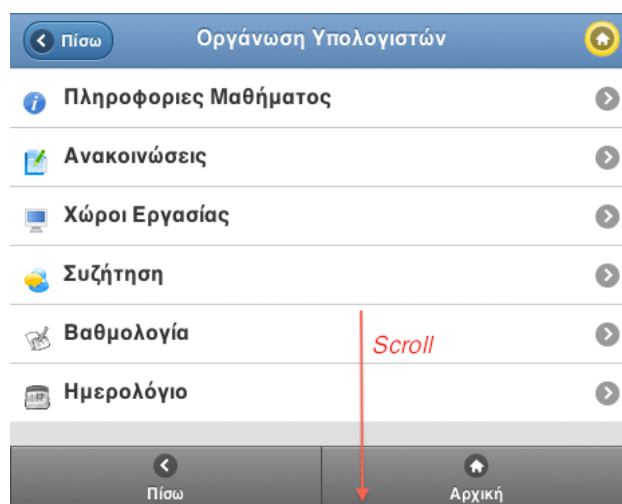


Εικόνα 5.13 - Ενδεικτικές αναλογίες οθονών smartphones

Και βέβαια το μικρό μέγεθος δεν μας αφήνει περιθώρια να φτιάξουμε μια εφαρμογή που θα αφήνει ανεκμετάλλευτο μέρος της οθόνης. Δηλαδή πρέπει να εκμεταλλευόμαστε κάθε φορά την πλήρη οθόνη ανεξάρτητα της συσκευής ή του πως κρατάει τη συσκευή αυτή ο χρήστης. Αυτό το καταφέραμε με τη χρήση του jQuery Mobile το οποίο κάνει χρήση CSS και Javascript για να μπορέσει να προσαρμόσει όλα τα στοιχεία της σελίδας στην εκάστοτε οθόνη.



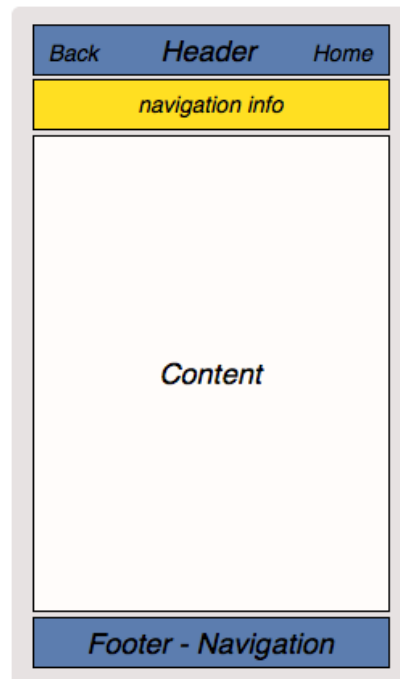
Εικόνα 5.14 - Προβολή εφαρμογής σε οθόνη iPhone 3GS σε portrait mode. (320\*480 px)



Εικόνα 5.15 - Προβολή εφαρμογής σε οθόνη iPhone 3GS σε landscape mode. (480\* 320 px)

Στα smartphones οι χρήστες χρησιμοποιούν τις συσκευές κατά κύριο λόγο σε portrait mode. Αυτό σημαίνει ότι όσο εξελίσσονται οι συσκευές αυτές οι οθόνες γίνονται ψηλότερες, και όχι φαρδύτερες όπως των υπολογιστών. Για να μπορέσουμε να έχουμε μια σχεδίαση που θα είναι συνεπής σε όλα τα είδη οθονών ακολουθήσαμε την στρατηγική της Κατακόρυφης Σχεδίασης (Vertical Design). Αυτή είναι η στρατηγική που χρησιμοποιείται κατά κύριο λόγο στις mobile συσκευές. Η ιδέα είναι ότι η πληροφορία εμφανίζεται από πάνω προς τα κάτω σε μορφή στοίβας, αποφεύγοντας να έχουμε

περισσότερα από ένα στοιχεία στον οριζόντιο άξονα. Στην κορυφή υπάρχει η πληροφορία που μας δείχνει που είμαστε καθώς και κάποιες επιλογές πλοήγησης. Στη μέση υπάρχει το κυρίως περιεχόμενο της σελίδας και στο κάτω μέρος υπάρχουν επιλογές πλοήγησης και εξόδου.



Εικόνα 5.16 - Βασική δομή της σελίδας μας

## 5.4 Ευκολία πλοήγησης

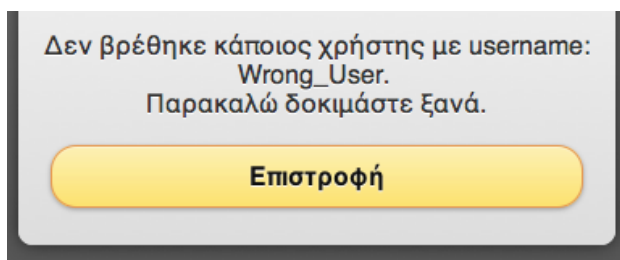
Άλλο ένα κρίσιμο σημείο ήταν η ευκολία πλοήγησης μέσα στην εφαρμογή. Θέλαμε ο χρήστης να μπορεί να φτάνει στους στόχους του με λίγα μόνο clicks, χωρίς να μπερδεύεται. Αυτό σημαίνει δύο πράγματα. Πρώτον έπρεπε να μην χρειάζεται να κάνει πολλά clicks για να φτάσει στην ολοκλήρωση μίας ενέργειας. Δεύτερον έπρεπε να του δίνεται η δυνατότητα να έχει άμεση πρόσβαση σε όλες τις βασικές λειτουργίες. Αυτό επιτεύχθηκε βάζοντας στην αρχική σελίδα όλες τις λειτουργίες που επικεντρώνονται γύρω από το χρήστη, και στη σελίδα του μαθήματος όλες αυτές που επικεντρώνονται γύρω από το συγκεκριμένο μάθημα. Σε κάθε σελίδα υπάρχει, στην κορυφή, επιστροφή στην αρχική πηγαίνοντας ουσιαστικά το χρήστη στη σελίδα με τις λειτουργίες που τον αφορούν άμεσα, και σε κάθε σελίδα που σχετίζεται με μάθημα του δίνουμε επιπλέον την δυνατότητα να μεταβεί στην σελίδα του μαθήματος. Σε κάθε σελίδα έχουμε επιλογή για μετάβαση στην προηγούμενη στην κορυφή. Επίσης υπάρχει και ένα navigation-bar στο

κάτω μέρος της οθόνης που δίνει τις ίδιες επιλογές στο χρήστη, δηλαδή “Πίσω” και “Αρχική Σελίδα”. Η ύπαρξη του navigation-bar στο κάτω μέρος της σελίδας κρίνεται απαραίτητη, γιατί έτσι δεν αναγκάζεται ο χρήστης, όταν έχει κάνει scroll για να φτάσει στο τέλος της σελίδας, να ξαναγυρίσει στην κορυφή για να εκτελέσει τις ενέργειες αυτές που είναι βασικές.

## 5.5 Χειρισμός Λαθών

Γενικότερα προσπαθήσαμε να φτιάξουμε μια εφαρμογή στην οποία η πιθανότητα να κάνει κάποιος χρήστης λάθος να είναι πολύ μικρή. Περιορίσαμε τις επιλογές που έχει σε κάθε σελίδα έτσι ώστε να είναι ευδιάκριτες και ξεκάθαρες ως προς το τι κάνουν, χωρίς να τον μπερδεύουν. Παρ’όλ’αυτά το να κάνει λάθος ο χρήστης είναι αναπόφευκτο. Ειδικά υπό τις συνθήκες κάτω από τις οποίες ενδέχεται να χρησιμοποιηθεί η εφαρμογή μας.

Σε περίπτωση λάθους εμφανίζουμε στο χρήστη σελίδες διαλόγου οι οποίες περιέχουν ένα περιγραφικό και ξεκάθαρο μήνυμα σχετικά με το πρόβλημα που έχει παρουσιαστεί, δίνοντας του την δυνατότητα να επιστρέψει στην σελίδα που έκανε λάθος και να το διορθώσει. Ακολουθούν μερικά παραδείγματα :



Εικόνα 5.17 - Παράδειγμα διαλόγου λαθών.

## 5.6 Συνέπεια

Η συνεπής εμφάνιση είναι πολύ σημαντική στην υλοποίηση μιας εφαρμογής. Οι λόγοι είναι ότι καταρχήν ότι δημιουργεί την αίσθηση της σταθερότητας εμπνέοντας εμπιστοσύνη στο χρήστη. Επίσης βοηθάει τους χρήστες να απομνημονεύσουν καλύτερα τις πιθανές επιλογές που έχουν και να δημιουργήσουν στο μυαλό τους το νοητικό μοντέλο που θέλουμε. Αυτό συμβαίνει γιατί όταν υπάρχει συνέπεια οι χρήστες συνδυάζουν στο μυαλό τους χρώματα με διαδικασίες, καθώς και θέσεις με πιθανές επιλογές. Για παράδειγμα όταν, ο χρήστης έχει συνηθίσει πάνω δεξιά στην οθόνη να υπάρχει επιλογή για μετάβαση στην αρχική σελίδα, το πρώτο πράγμα που θα κοιτάξει



όταν θέλει να εκτελέσει αυτή την ενέργεια θα είναι το πάνω αριστερό μέρος της οθόνης. Η εφαρμογή των αντίστοιχων κανόνων που έχουμε χρησιμοποιείται περιγράφεται αναλυτικά στην περιγραφή των δομικών στοιχείων της διεπαφής του χρήστη που ακολουθεί.

## 5.7 Επιλογή χρωμάτων

Τα χρώματα που χρησιμοποιήσαμε είναι κυρίως το μπλε, το άσπρο και το κίτρινο. Η επιλογή μας αυτή, έγινε για πολλούς λόγους. Καταρχήν το άσπρο είναι ουδέτερο χρώμα και προσδίδει μια καθαρότητα και μια σταθερότητα. Για το λόγο αυτό ήταν και το χρώμα του background μας. Το μπλε με το κίτρινο είναι συμπληρωματικά χρώματα. Αυτό σημαίνει ότι είναι ακριβώς απέναντι στο χρωματικό τροχό. Οι συνδυασμοί αυτοί των χρωμάτων δημιουργούν καλές αντιθέσεις, και κάνουν τα στοιχεία να φαίνονται σαν να μην είναι στο ίδιο επίπεδο. Ο λόγος που επιλέξαμε αυτόν τον συνδυασμό συμπληρωματικών χρωμάτων είναι ότι τα χρώματα αυτά εκφράζουν τα χαρακτηριστικά που θέλουμε να δώσουμε στην εφαρμογή μας. Το κίτρινο εκφράζει τον θετικισμό και τη χαρά. Επίσης διεγείρει τις πνευματικές διεργασίες και ενεργοποιεί την μνήμη. Για το λόγο αυτό και έχει επιλεγεί να υπάρχει πάνω σε κουμπιά που σχετίζονται με ενέργειες του χρήστη, καθώς και στις ειδοποιήσεις. Το μπλε από τη μεριά του είναι ξεκούραστο για το μάτι και εκφράζει την ηρεμία και την σταθερότητα. Για το λόγο αυτό το επιλέξαμε να υπάρχει στις οριζόντιες μπάρες της πλοήγησης και του τίτλου. Επίσης χρησιμοποιήσαμε κόκκινο όταν έπρεπε να δείξουμε πράγματα που ήθελαν προσοχή. Για παράδειγμα ημερομηνίες λήξης παράδοσης εργασιών κ.α. Η χρήση του κάθε χρώματος στην εφαρμογή περιγράφεται αναλυτικά στην περιγραφή των δομικών στοιχείων της διεπαφής του χρήστη που ακολουθεί.

## 5.8 Διεπαφή του χρήστη

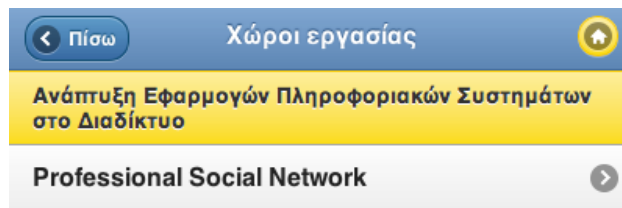
### 5.8.1 Βασικά στοιχεία της διεπαφής

Ακολουθούν τα βασικά γραφικά στοιχεία της διεπαφής μας :

Header : είναι μπλε χρώμα και από τη μία άκρη της σελίδας στην άλλη. Στα αριστερά υπάρχει επιλογή “Πίσω” για μετάβαση στην προηγούμενη σελίδα , στο κέντρο ο τίτλος που περιγράφει το που βρισκόμαστε και στα δεξιά ένα κουμπί για μετάβαση στην αρχική σελίδα. Διατηρείται αυτή η εμφάνιση σε όλες τις σελίδες με μοναδική εξαίρεση,



προφανώς, την αρχική σελίδα στην οποία δεν υπάρχει το “Home button” και που το πίσω έχει αντικατασταθεί από το “Έξοδος”.



Εικόνα 5.18 - Το Header στη σελίδα των Χώρων εργασίας του μαθήματος Ανάπτυξη Εφαρμογών.



Εικόνα 5.19 - Το Header στη αρχική σελίδα.

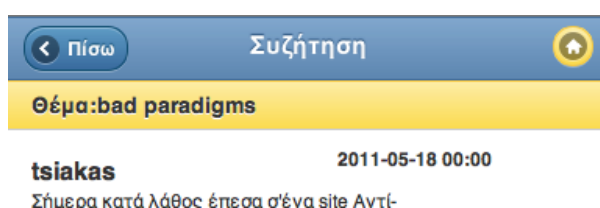
Μπάρα πλοήγησης (navigation-bar) : Βρίσκεται στο κάτω μέρος της οθόνης. Είναι μπλε χρώμα και από τη μία άκρη της σελίδας στην άλλη. Περιέχει δύο κουμπιά. Το αριστερό είναι για μετάβαση στην προηγούμενη σελίδα, και το δεξί για μετάβαση στην αρχική σελίδα. Έχουμε χρησιμοποιήσει τα ίδια εικονίδια που υπάρχουν και στο header για να μην προκαλείται σύγχυση.



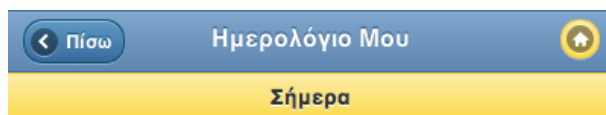
Εικόνα 5.20 - Navigation bar στο κάτω μέρος της σελίδας.

Μπάρα πρόσθετης πληροφορίας πλοήγησης : είναι κίτρινη και βρίσκεται ακριβώς κάτω από το header. Επειδή αναγκαστικά σε μερικές ενέργειες έπρεπε να διασχίσουμε περισσότερες από 1 σελίδες, και αυτό θα έκανε δύσκολο στο χρήστη να θυμάται που

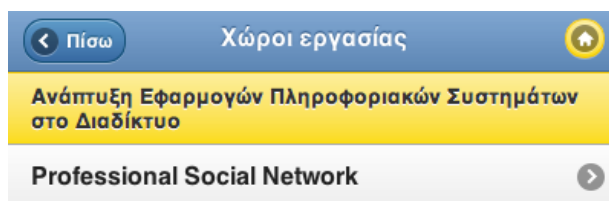
βρίσκεται, έπρεπε να βρούμε ένα τρόπο να δίνουμε μια επιπλέον πληροφορία στο χρήστη για το που είναι τη δεδομένη στιγμή. Επίσης αυτό το έκανε επιτακτική ανάγκη και το γεγονός ότι τα smartphones έχουν multitasking. Αυτό σημαίνει ότι αν ο χρήστης κλείσει τον browser, χωρίς να αποσυνδεθεί, την επόμενη φορά που θα μπει θα δει κατευθείαν την ίδια σελίδα στην οποία ήταν -αν δεν έχει λήξει το session. Συνήθως σε μια τέτοια περίπτωση ο χρήστης δεν θυμάται που ήταν. Για να λύσουμε τα παραπάνω προβλήματα βάλαμε αυτή την μπάρα. Περιέχει πληροφορία, για παράδειγμα, σε ποιο μάθημα είμαστε, για το ποια συζήτηση διαβάζουμε ή σε ποια συζήτηση απαντάμε, σε ποιά ημερομηνία βρισκόμαστε στο ημερολόγιο κ.α. Ακολουθούν μερικά ενδεικτικά παραδείγματα.



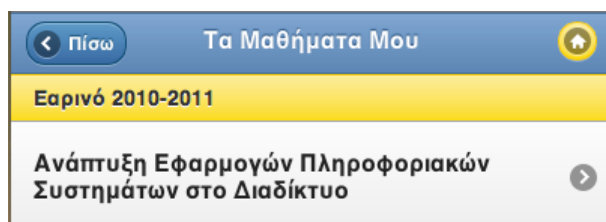
Εικόνα 5.21 - Μπάρα πρόσθετης πλοήγησης στη συζήτηση. Περιέχει το θέμα.



Εικόνα 5.22 - Μπάρα πρόσθετης πλοήγησης στο προσωπικό ημερολόγιο. Περιέχει την ημερομηνία.



Εικόνα 5.23 - Μπάρα πρόσθετης πλοήγησης στους Χώρους Εργασίας. Περιέχει το μάθημα στο οποίο βρισκόμαστε.



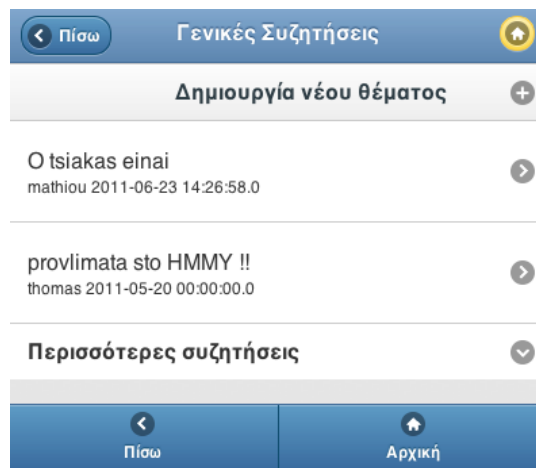
Εικόνα 5.24 - Μπάρα πρόσθετης πλοήγησης στη σελίδα των μαθημάτων του χρήστη. Χρησιμοποιείται για να διαχωρίσει τα εξάμηνα.

Κυρίως σώμα : είναι λευκό και βρίσκεται ανάμεσα στο header και την κάτω μπάρα πλοήγησης (όπου αυτή υπάρχει). Το περιεχόμενο του είναι σε μορφή στοίβας. Εικονίδια έχει στην αρχική σελίδα και την σελίδα μαθήματος μόνο (δες ενότητα 6.1.3 -Visual Design, 4. Γραφικά) . Σε όλες τις άλλες το περιεχόμενο είναι κείμενο το οποίο εμπεριέχεται μέσα σε λίστα. Στα δεξιά εμφανίζονται οι ειδοποιήσεις για νέα γεγονότα , με κίτρινο χρώμα.



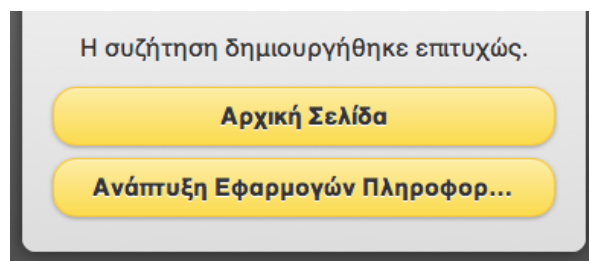
Εικόνα 5.25 - Ειδοποίηση για δύο (2) νέα εισερχόμενα στον χρήστη.

Επίσης στα δεξιά εμφανίζονται με γκρι χρώμα 3 διαφορετικά εικονίδια σε περίπτωση που το στοιχείο εκτελεί μια ενέργεια. Απουσία του εικονιδίου αυτού σημαίνει ότι το στοιχείο περιέχει απλά κείμενο. Τα τρία εικονίδια είναι βελάκι δεξιά (>) το οποίο δείχνει ότι το στοιχείο είναι link σε κάποια άλλη σελίδα, το σύμβολο της πρόσθεσης (+) το οποίο δείχνει ότι το στοιχείο μας στέλνει σε μια σελίδα για να προσθέσουμε ή να δημιουργήσουμε κάτι (π.χ. νέα συζήτηση) και βελάκι προς τα κάτω (v) το οποίο δείχνει ότι το στοιχείο θα φορτώσει περισσότερα στοιχεία θα οποία θα προστεθούν κάτω από τη στοίβα. Ακολουθεί ένα παράδειγμα που περιέχει και τα τρία αυτά στοιχεία.

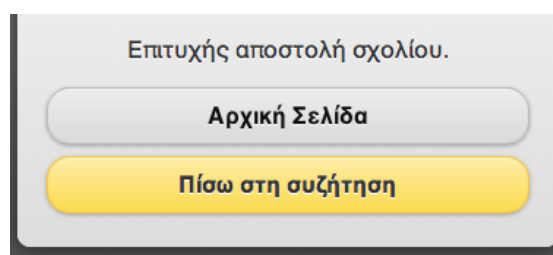


Εικόνα 5.26 - Εικονίδια που υποδηλώνουν συγκεκριμένη λειτουργικότητα στο κυρίως σώμα.

Κουμπιά : είναι κίτρινα και πιάνουν όλο το πλάτος της σελίδας. Είναι μεγάλα και ευδιάκριτα με σκοπό να αποφεύγονται βιαστικά λάθη στην προσπάθεια να πατήσει κάπου αλλού ο χρήστης (slip errors). Όπου υπάρχουν περισσότερα από ένα κίτρινο είναι αυτό με την πιο σημαντική λειτουργία ή αυτό που οδηγεί στην ολοκλήρωση μια ενέργειας, ενώ τα άλλα είναι γκρί-μεταλλικο. Όταν δεν έχει σημασία ο διαχωρισμός παραμένουν και τα δύο κίτρινα. Την ίδια φιλοσοφία ακολουθούμε και στα παράθυρα διαλόγου. Ακολουθούν μερικά ενδεικτικά παραδείγματα.



Εικόνα 5.27 - Δύο κουμπιά με ίδια βαρύτητα σε παράθυρο διαλόγου.



Εικόνα 5.28 - Δύο κουμπιά με διαφορετική βαρύτητα σε παράθυρο διαλόγου.

### 5.8.2 Είσοδος στο σύστημα.

Σχετίζεται με την Περίπτωση Χρήσης #1

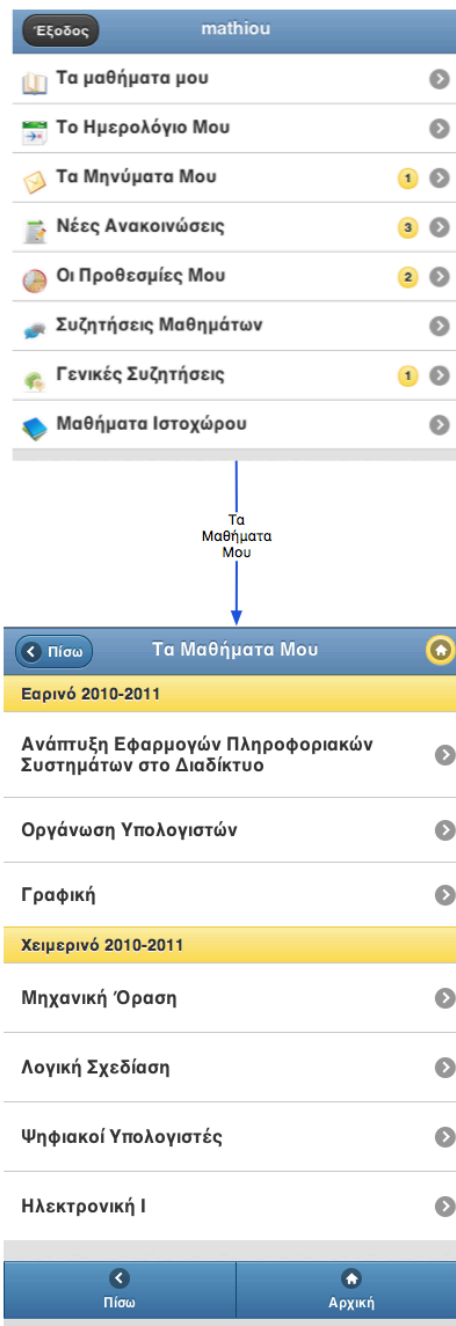


Εικόνα 5.29 - Η ακολουθία σελίδων για την είσοδο του χρήστη στο σύστημα.

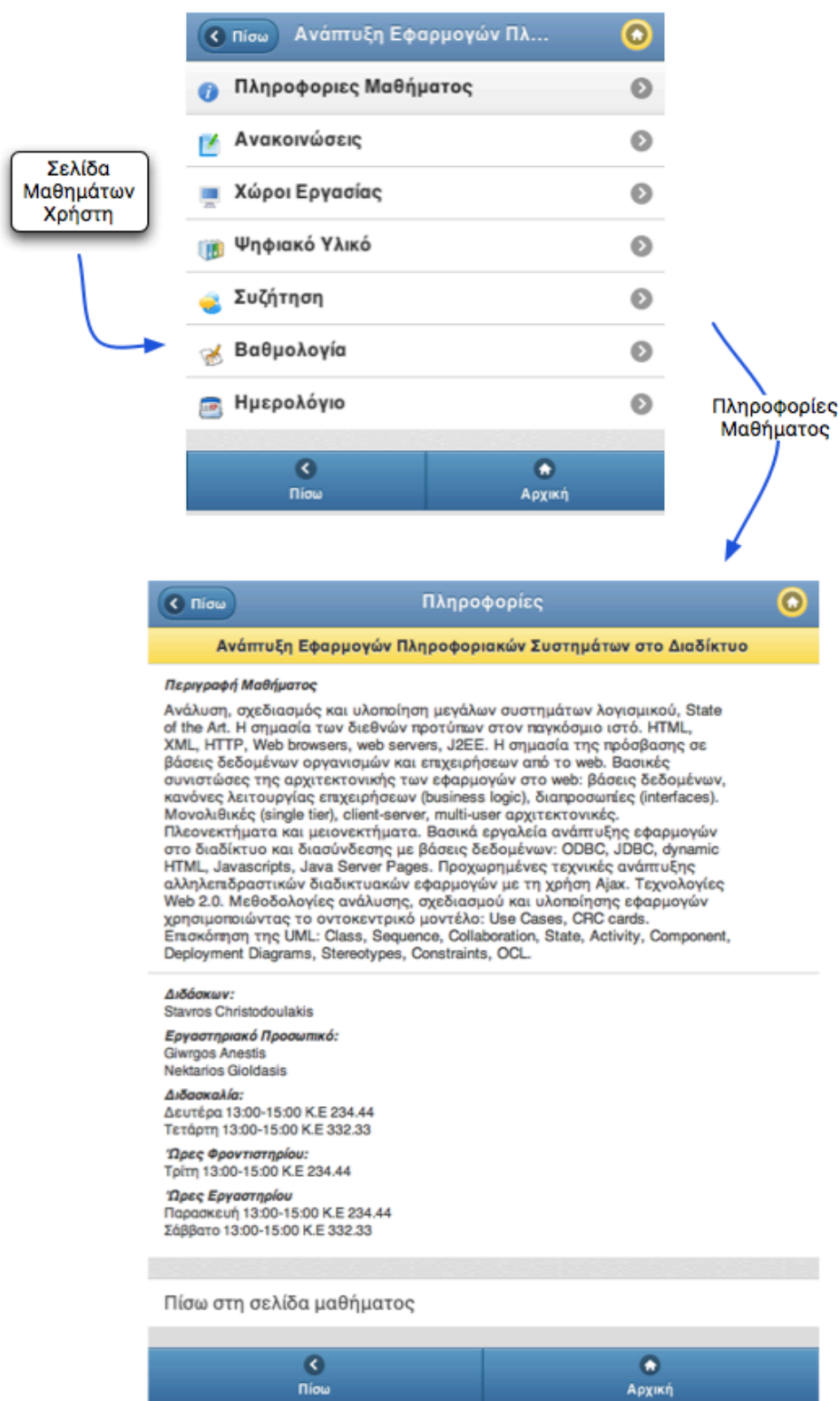
### 5.8.3 Προβολή πληροφοριών μαθήματος

Σχετίζεται με την Περίπτωση Χρήσης #14.

Η παρακάτω ακολουθία σελίδων μας πάει στη σελίδα του μαθήματος απ'όπου ξεκινάνε όλες οι ενέργειες που σχετίζονται με κάποιο συγκεκριμένο μάθημα. Οι ενέργειες που ακολουθούν θα ξεκινάνε από τη σελίδα του συγκεκριμένου μαθήματος.



Εικόνα 5.30 - Η ακολουθία σελίδων για να φτάσει ο χρήστης από την αρχική στη μαθήματα του.



Εικόνα 5.31 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στις πληροφορίες.

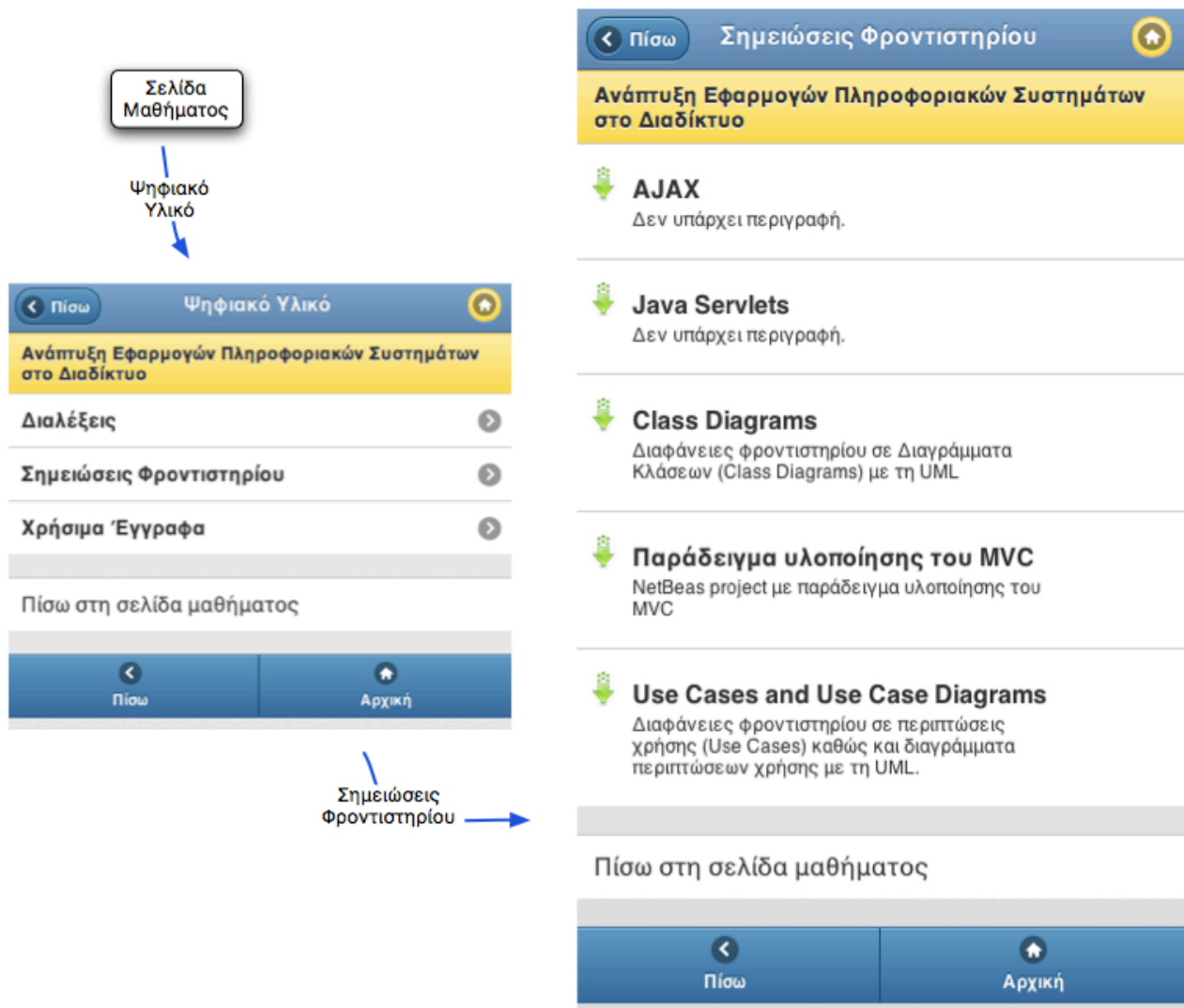
Σχετίζεται με την Περίπτωση Χρήσης #3





## 5.8.5 Πρόσβαση στο ψηφιακό υλικό του μαθήματος

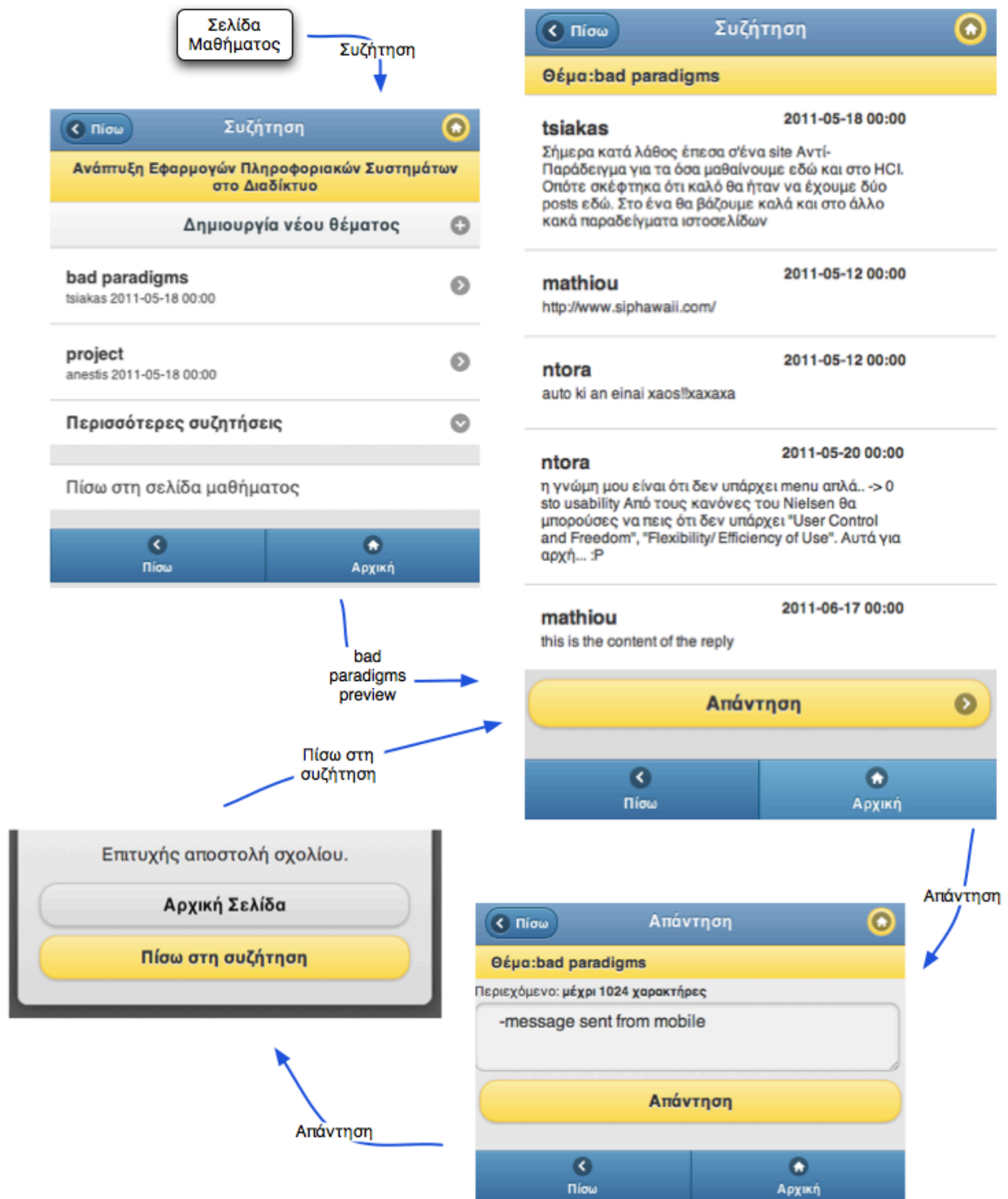
Σχετίζεται με την Περίπτωση Χρήσης #4



Εικόνα 5.33 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στο ψηφιακό υλικό.

## 5.8.6 Προβολή και σχολιασμός σε συζήτηση μαθήματος

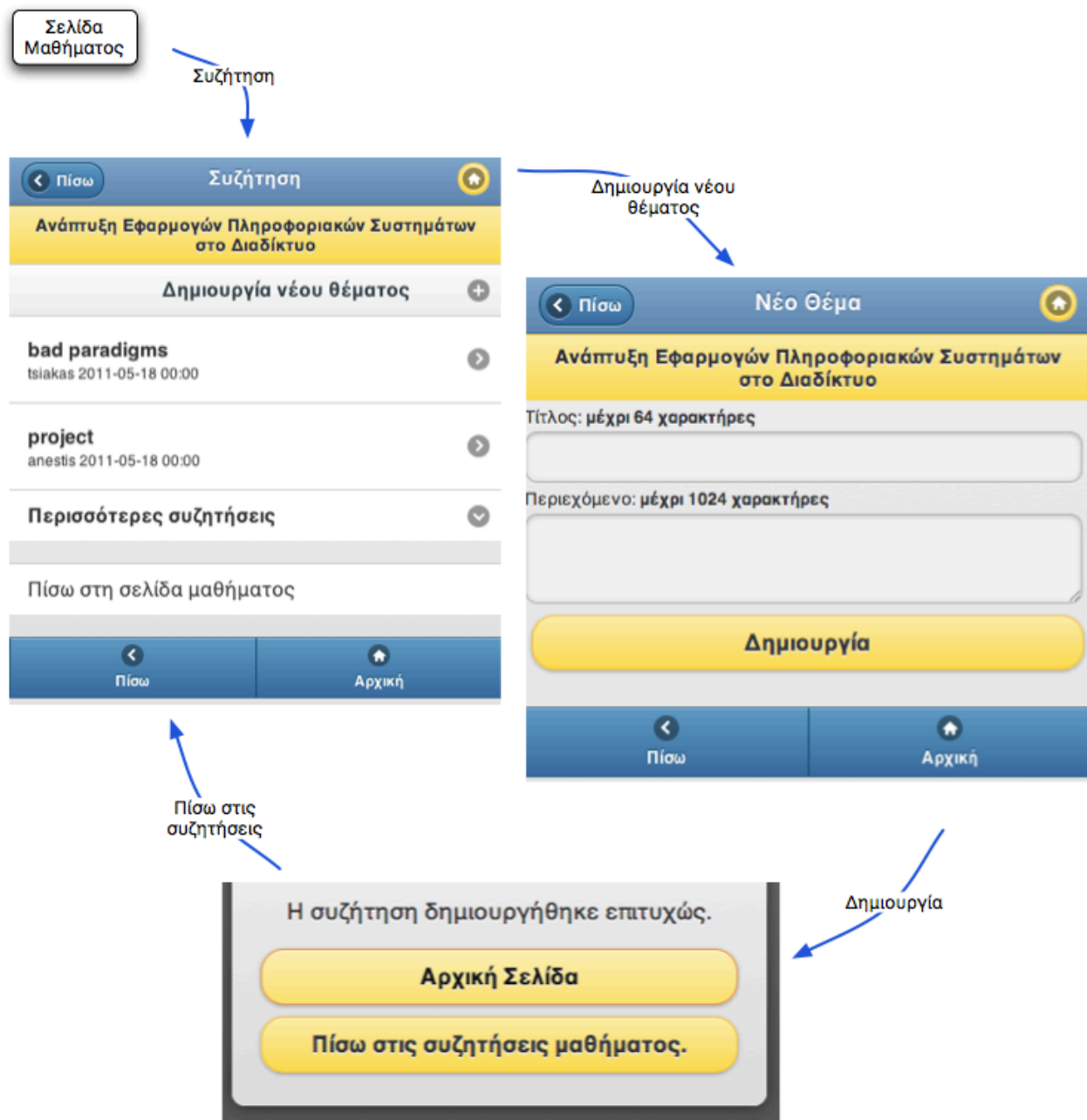
Σχετίζεται με τις Περιπτώσεις Χρήσης #5 & #6.



Εικόνα 5.34 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος σε μια συγκεκριμένη συζήτηση και συμμετοχή σε αυτή.

### 5.8.7 Δημιουργία νέας συζήτησης μαθήματος.

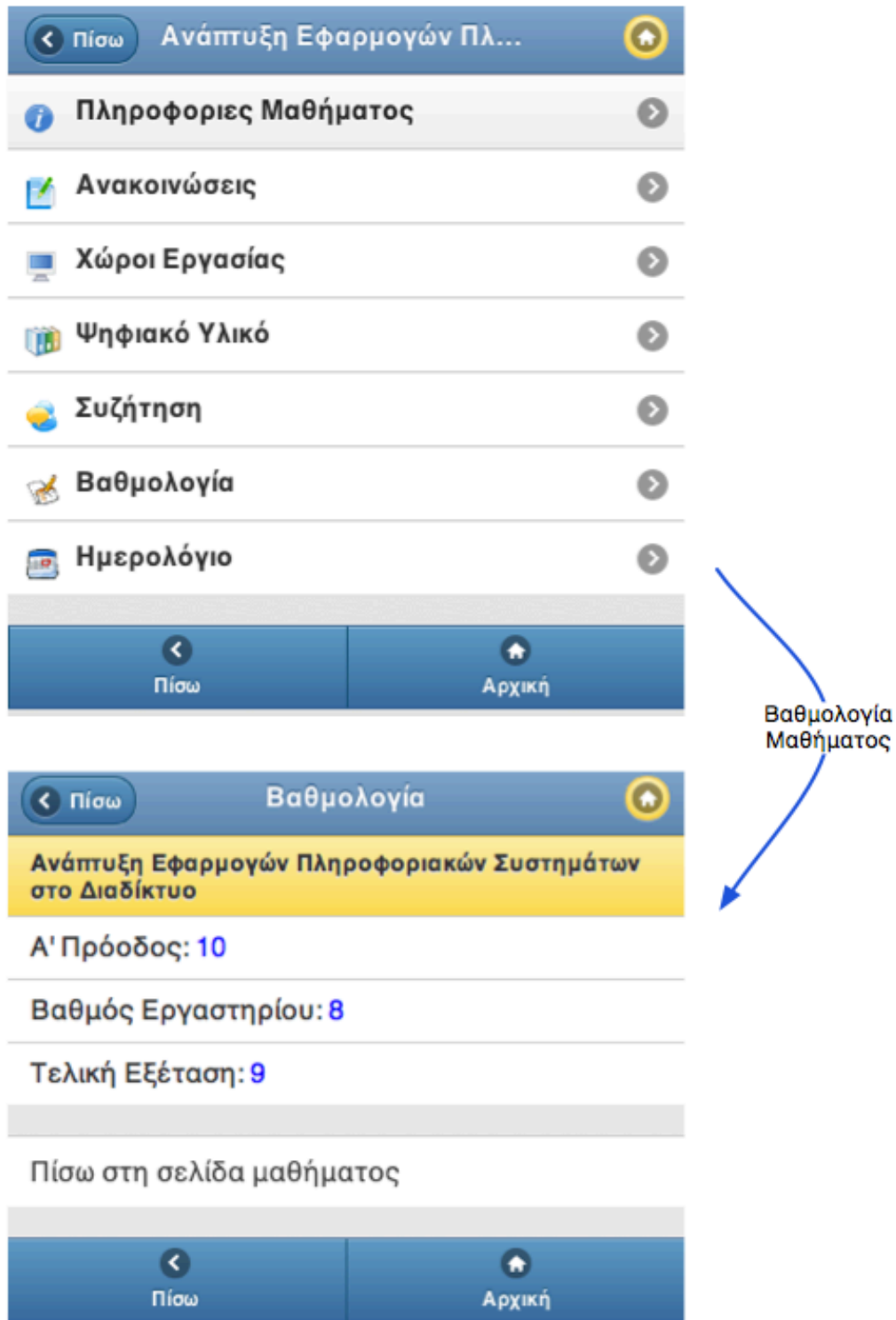
Σχετίζεται με την Περίπτωση Χρήσης #7



Εικόνα 5.35 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στις πληροφορίες.

### 5.8.8 Προβολή βαθμολογίας μαθήματος

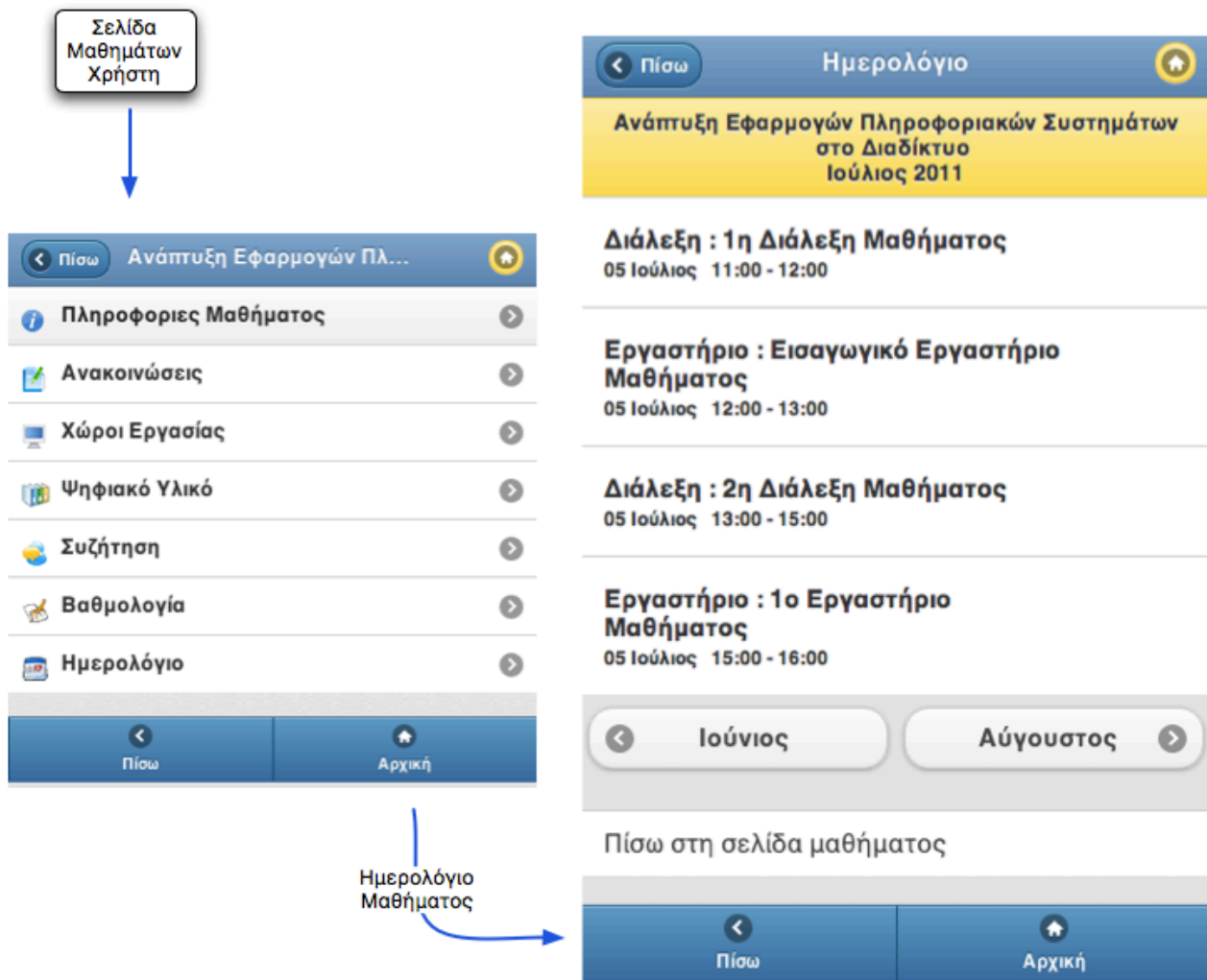
Σχετίζεται με την Περίπτωση Χρήσης #8



Εικόνα 5.34 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στη βαθμολογία.

### 5.8.9 Προβολή ημερολογίου μαθήματος

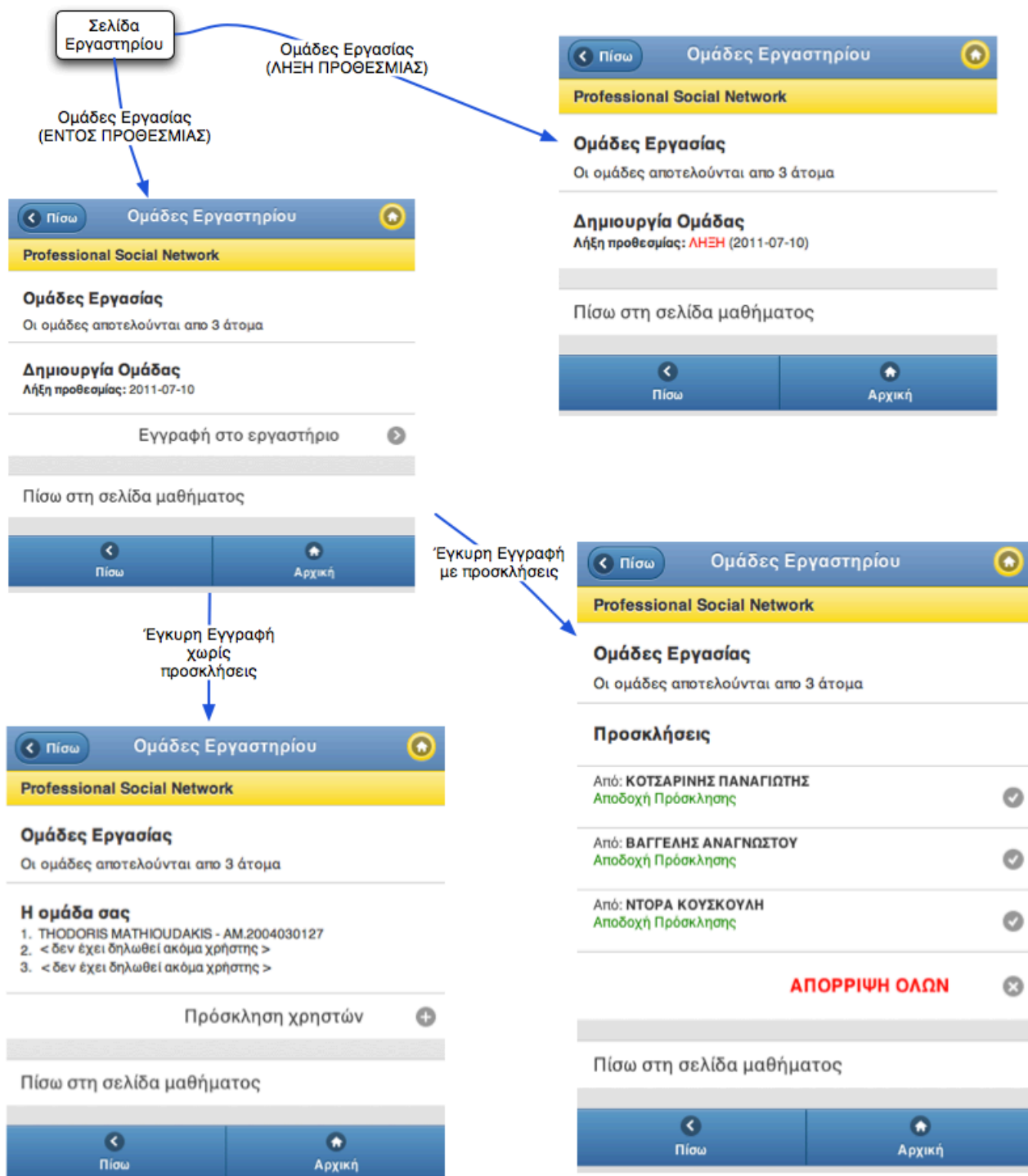
Σχετίζεται με την Περίπτωση Χρήσης #9



Εικόνα 5.35 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στο ημερολόγιο.

### 5.8.10 Εγγραφή σε εργαστήριο

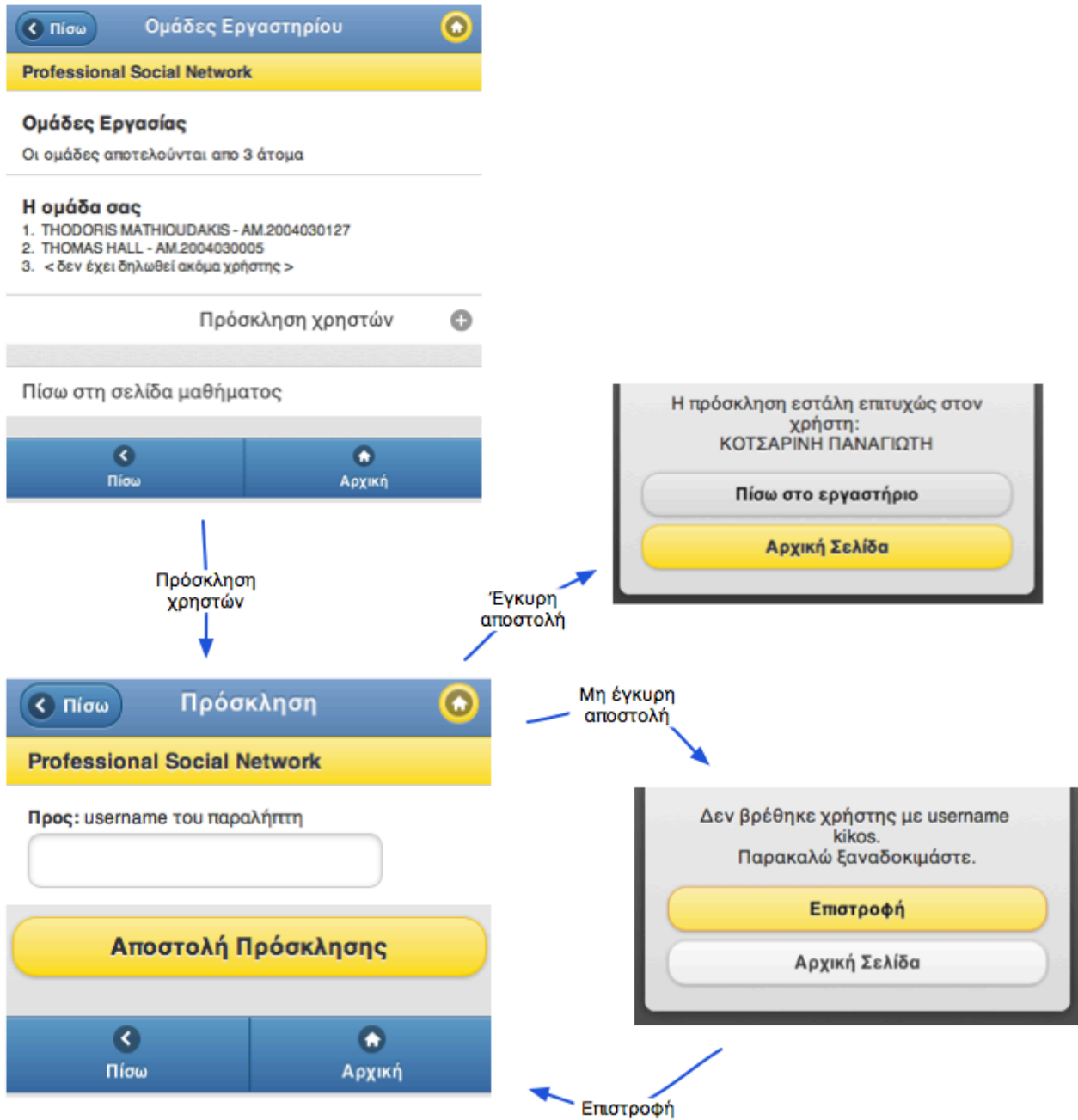
Σχετίζεται με την Περίπτωση Χρήσης #11



Εικόνα 5.36 - Η ακολουθία σελίδων από τη σελίδα ενός εργαστηρίου για την εγγραφή σε αυτό.

### 5.8.10 Πρόσκληση ατόμων σε ομάδα εργασίας

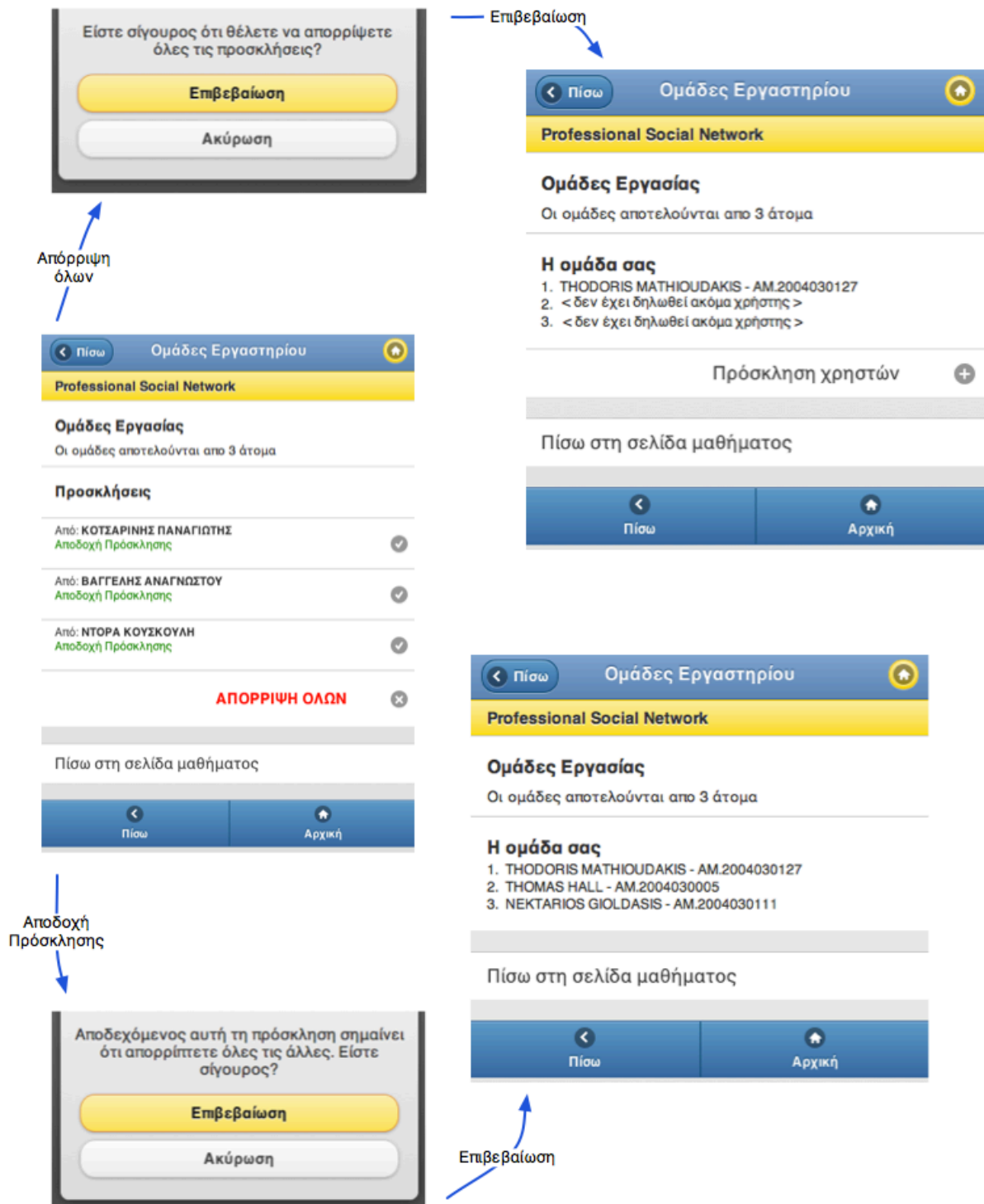
Σχετίζεται με την Περίπτωση Χρήσης #12



Εικόνα 5.37 - Η ακολουθία σελίδων για την πρόσκληση ατόμων στο εργαστήριο.

## 5.8.10 Αποδοχή ή απόρριψη πρόσκλησης σε ομάδα εργασίας

Σχετίζεται με την Περίπτωση Χρήσης #13

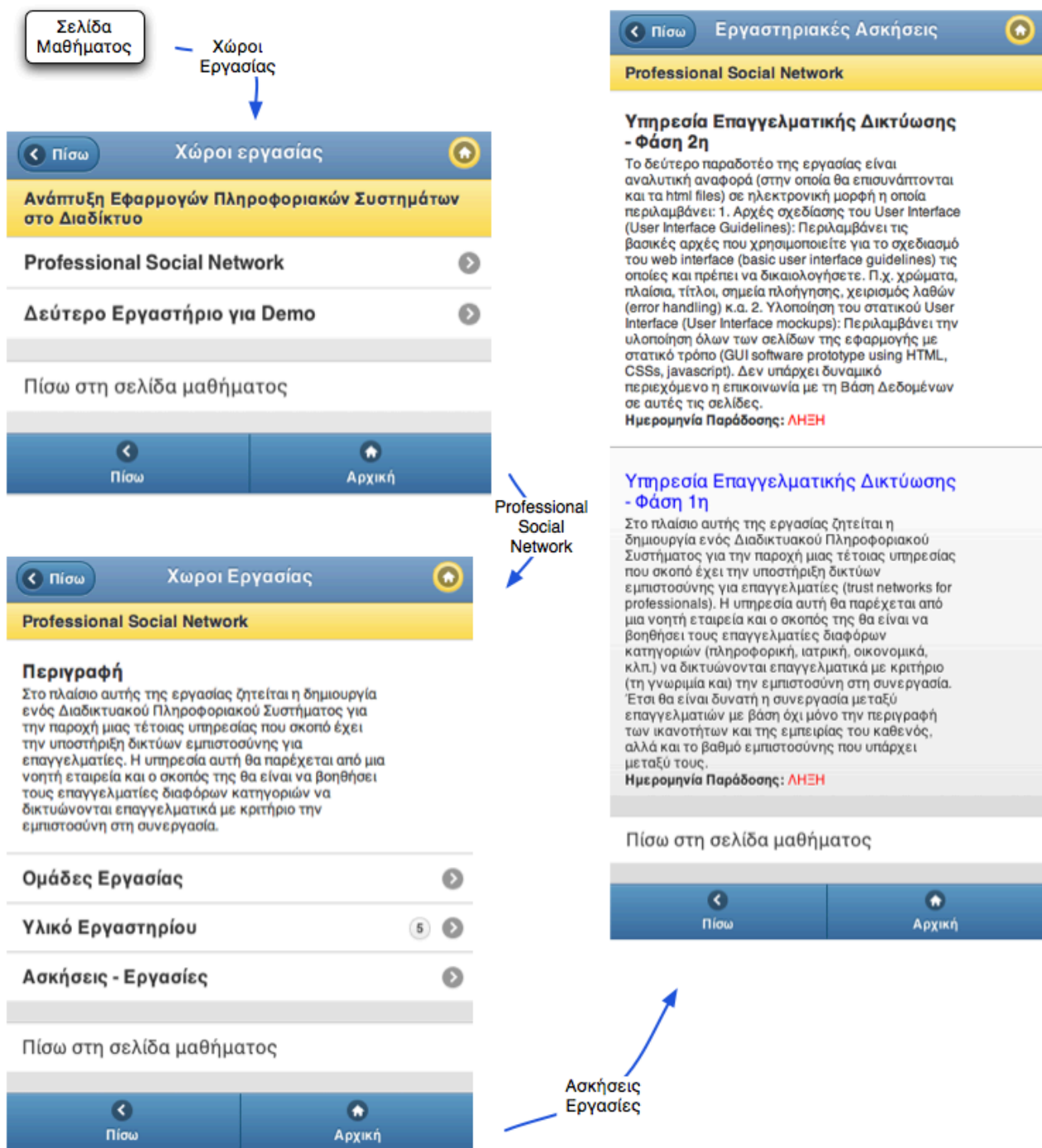


Εικόνα 5.38 - Η ακολουθία σελίδων για την αποδοχή ή απόρριψη προσκλήσεων χρηστών για κάποιο συγκεκριμένο εργαστήριο.



### 5.8.11 Πρόσβαση στις εργαστηριακές ασκήσεις

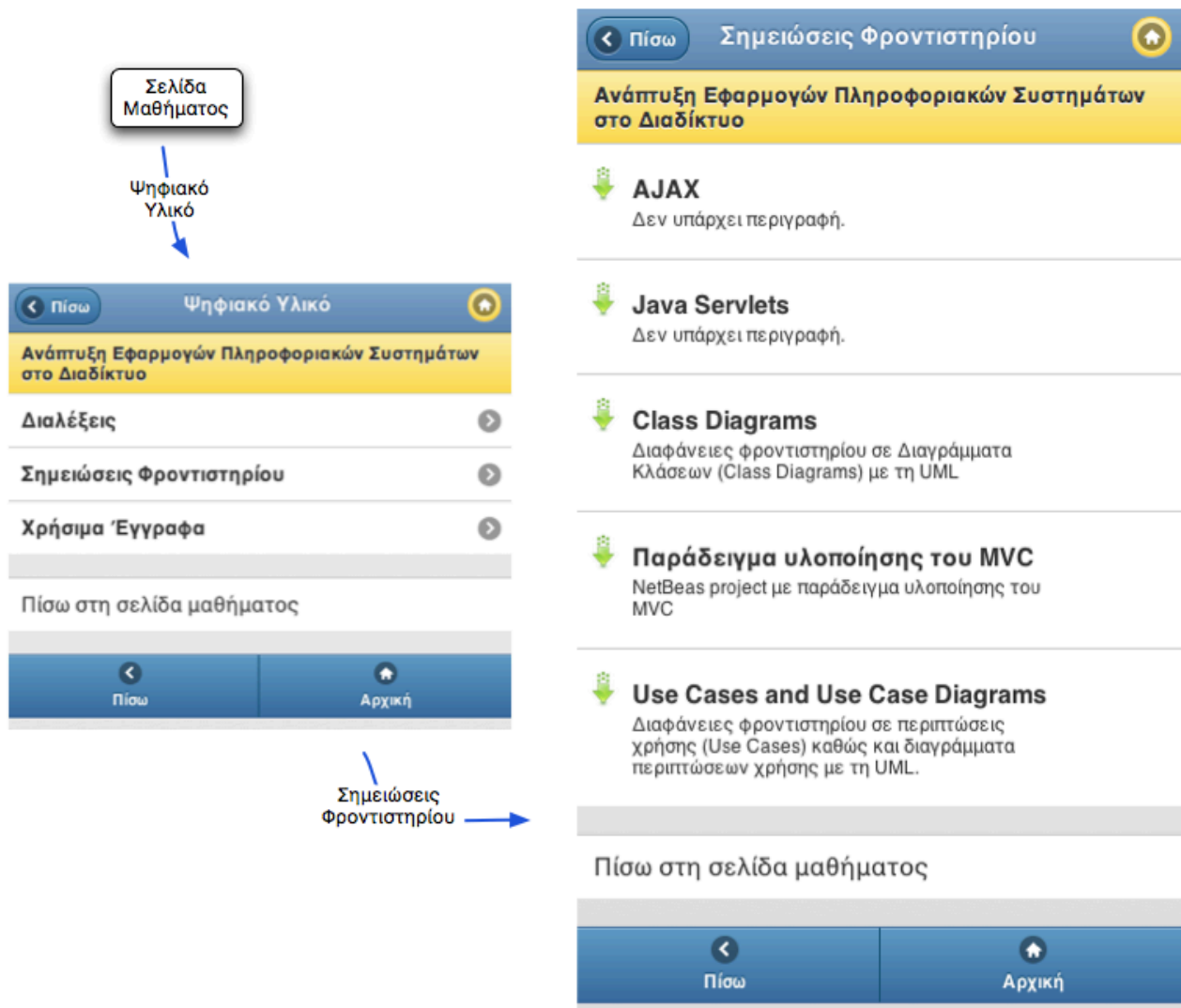
Σχετίζεται με την Περίπτωση Χρήσης #10



Εικόνα 5.39 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στις εργαστηριακές ασκήσεις.

## 5.8.12 Πρόσβαση στο ψηφιακό υλικό

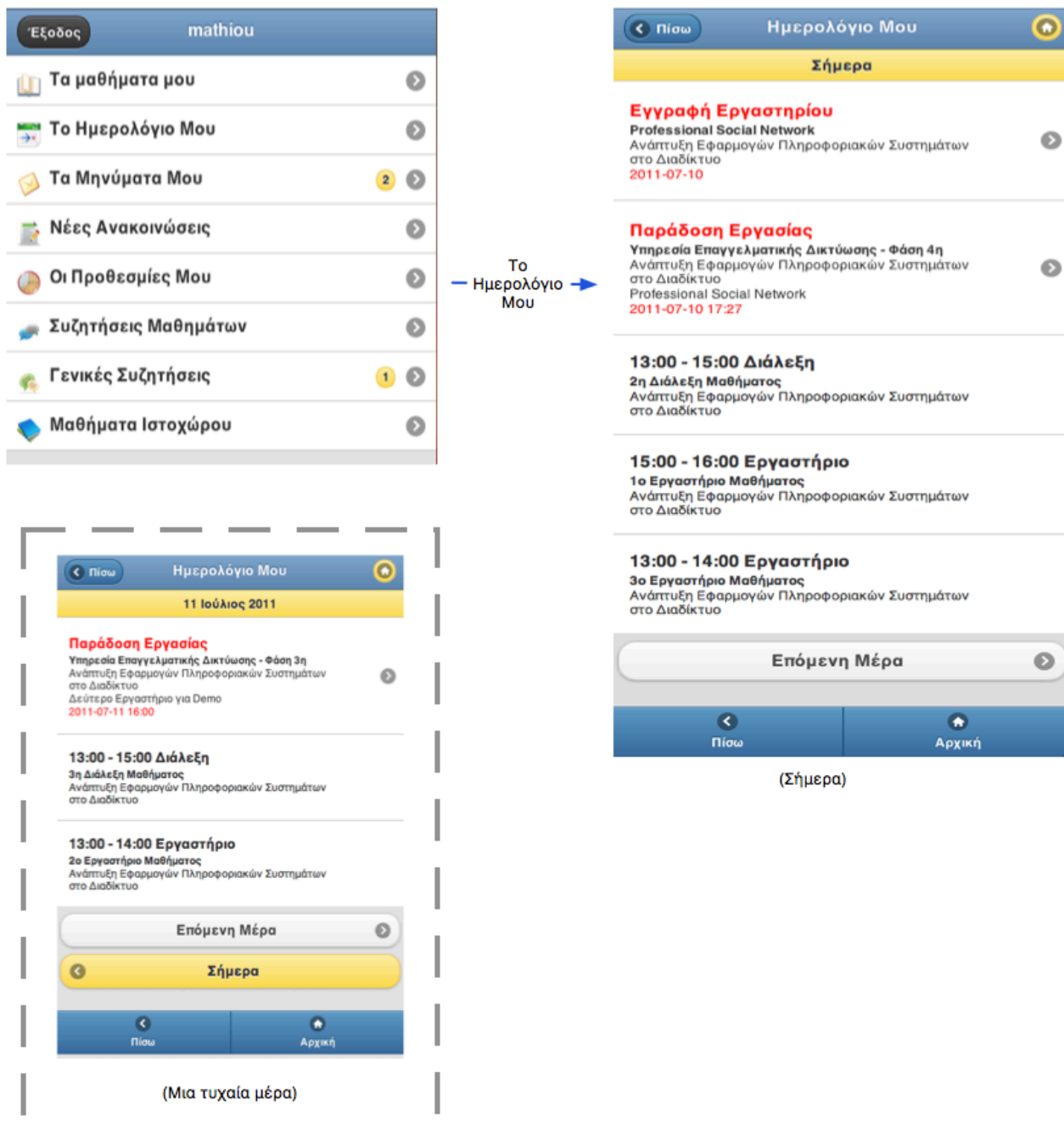
Σχετίζεται με την Περίπτωση Χρήσης #4



Εικόνα 5.40 - Η ακολουθία σελίδων από τη σελίδα ενός μαθήματος στο ψηφιακό υλικό του μαθήματος. Στο συγκριμένο παράδειγμα η μετάβαση γίνεται στις σημειώσεις του φροντιστηρίου.

## 5.8.13 Προβολή προσωπικού ημερολογίου του χρήστη και πλοήγηση

Σχετίζεται με την Περίπτωση Χρήσης #24

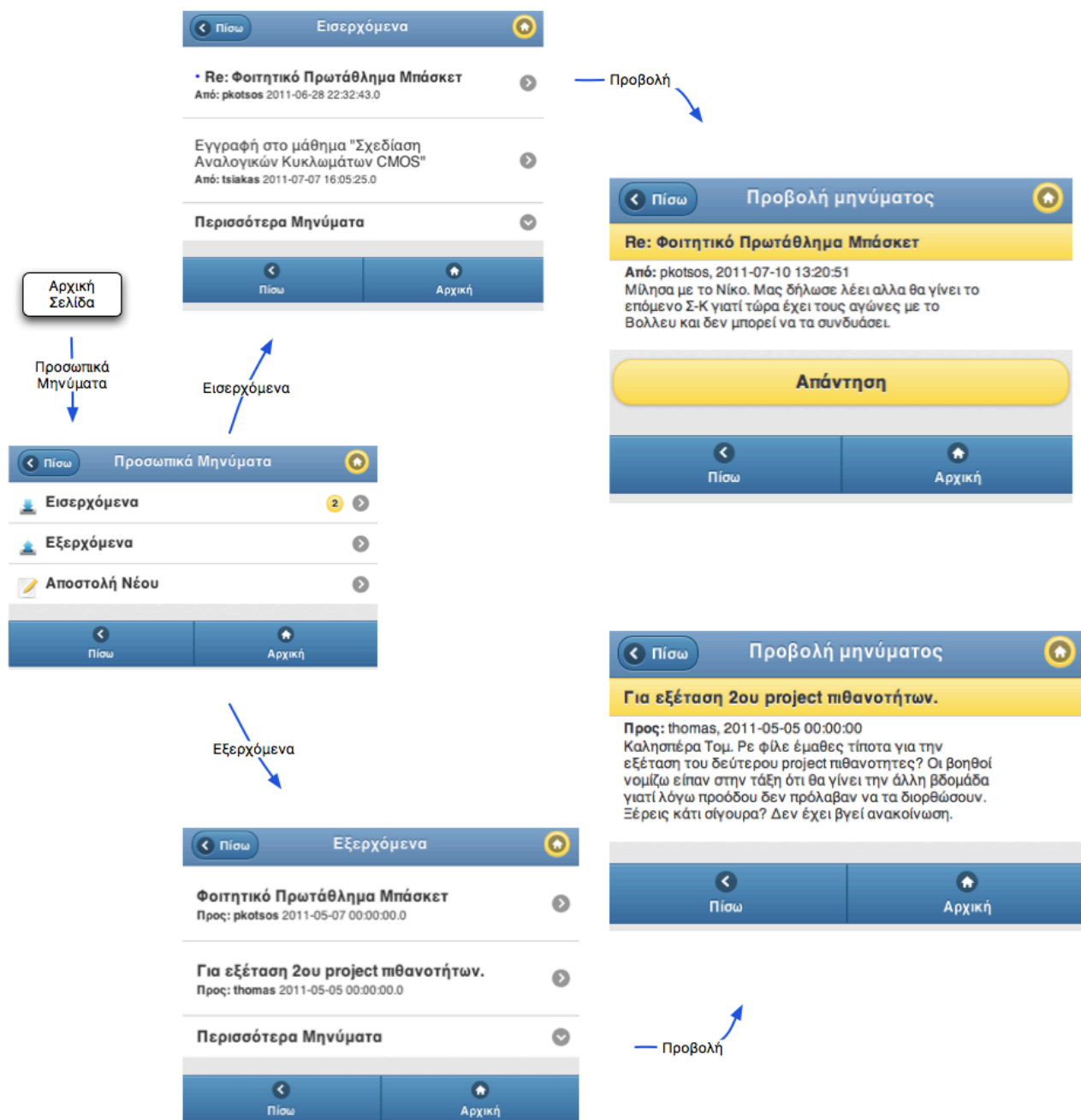


Εικόνα 5.41 - Η ακολουθία σελίδων από τη αρχική σελίδα του χρήστη στη σελίδα του προσωπικού ημερολογίου του.

### 5.8.14 Προβολή εισερχομένων και εξερχομένων μηνυμάτων

Σχετίζεται με την Περίπτωση Χρήσης #15 & #16

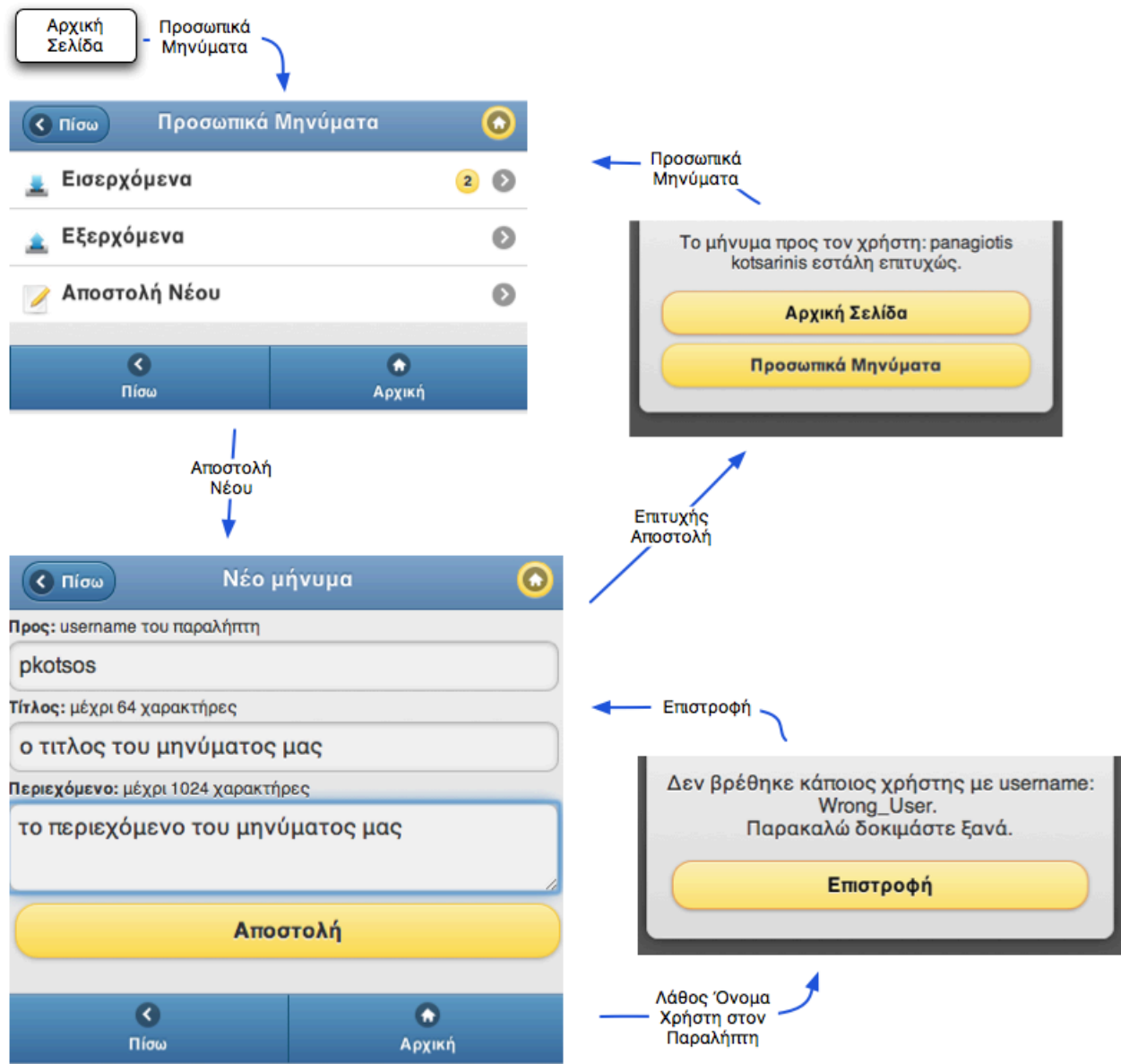
Η απάντηση σε εισερχόμενο μήνυμα έχει όμοια διεπαφή με την αποστολή νέου μηνύματος που περιγράφεται στην συνέχεια.



Εικόνα 5.43 - Η ακολουθία σελίδων από την αρχική σελίδα του χρήστη στα εισερχόμενα και τα εξερχόμενα μηνύματα, και στη συνέχεια στην προβολή ενός εισερχομένου και ενός εξερχόμενου μηνύματος αντίστοιχα.

### 5.8.15 Αποστολή νέου προσωπικού μηνύματος

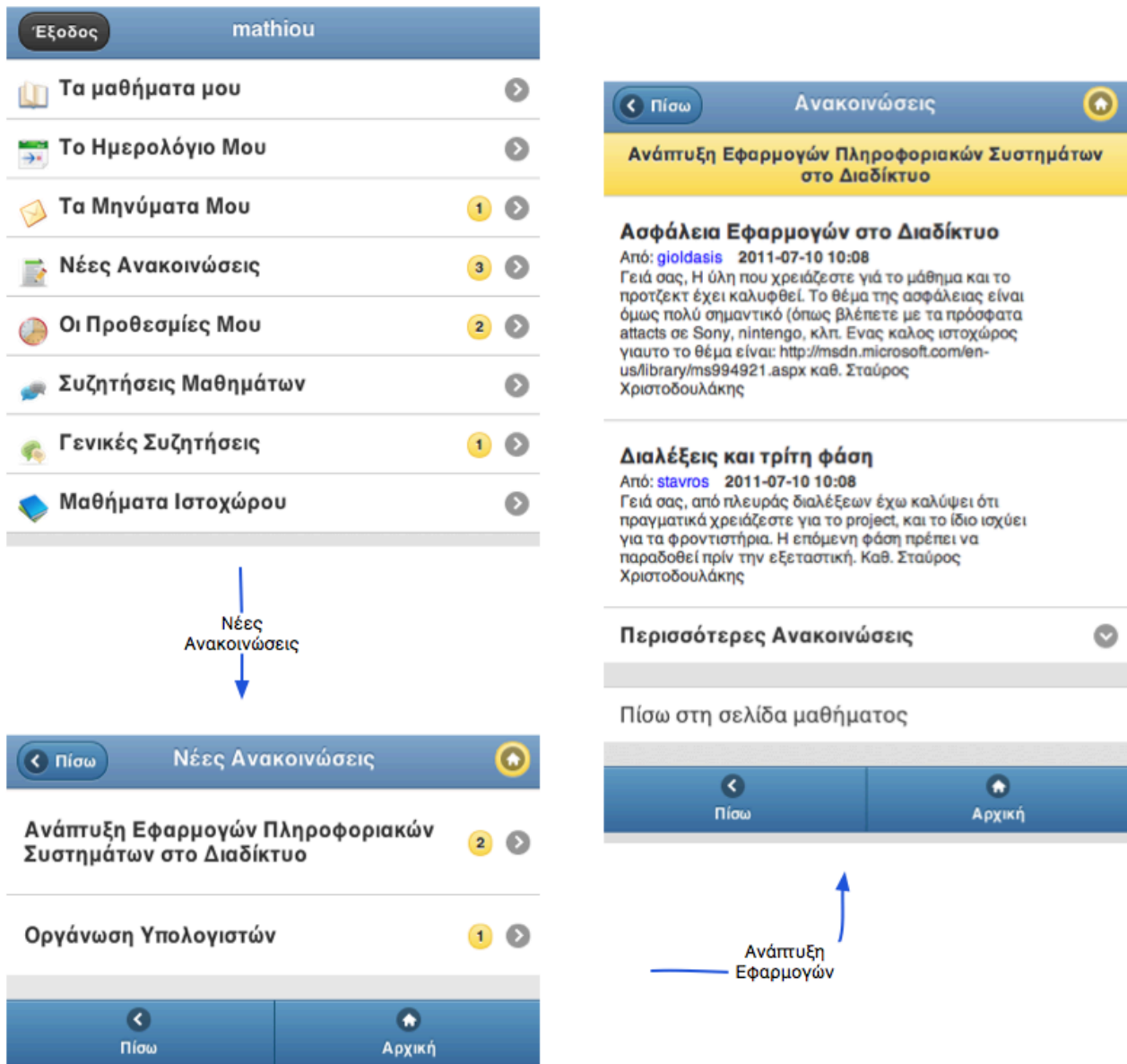
Σχετίζεται με την Περίπτωση Χρήσης #18



Εικόνα 5.44 - Η ακολουθία σελίδων από την αρχική σελίδα του χρήστη μέχρι την αποστολή ενός προσωπικού μηνύματος. Φαίνονται και τα μηνύματα σε περίπτωση επιτυχημένης αποστολής ή λάθος επιλογής χρήστη.

## 5.8.16 Προβολή νέων ανακοινώσεων στα μαθήματα του χρήστη

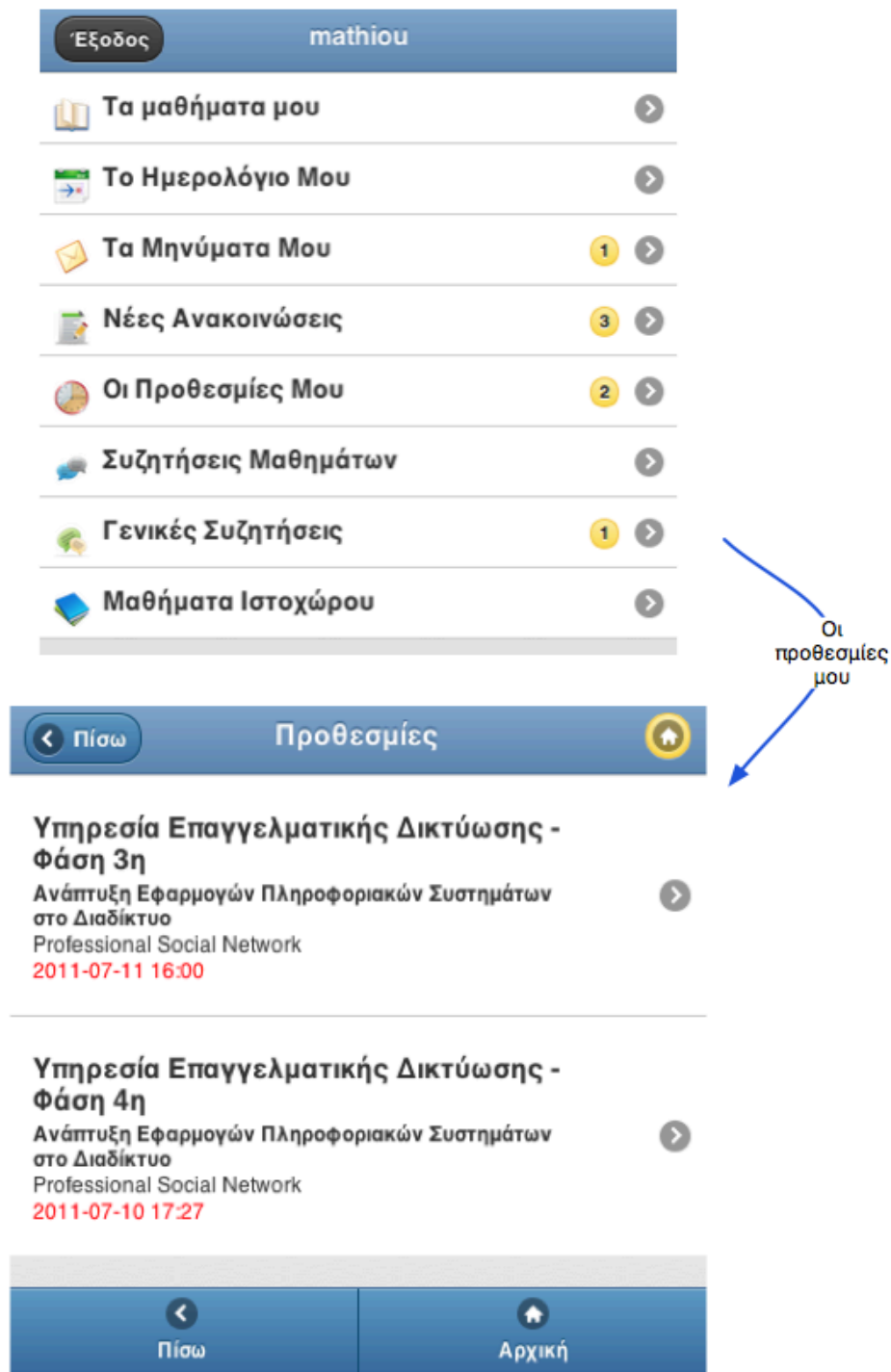
Σχετίζεται με την Περίπτωση Χρήσης #19



Εικόνα 5.45 - Η ακολουθία σελίδων από την αρχική σελίδα του χρήστη στη σελίδα με τις ανακοινώσεις των μαθημάτων που είναι εγγεγραμμένος.

### 5.8.17 Προβολή προθεσμιών παράδοσης εργασιών

Σχετίζεται με την Περίπτωση Χρήσης #20

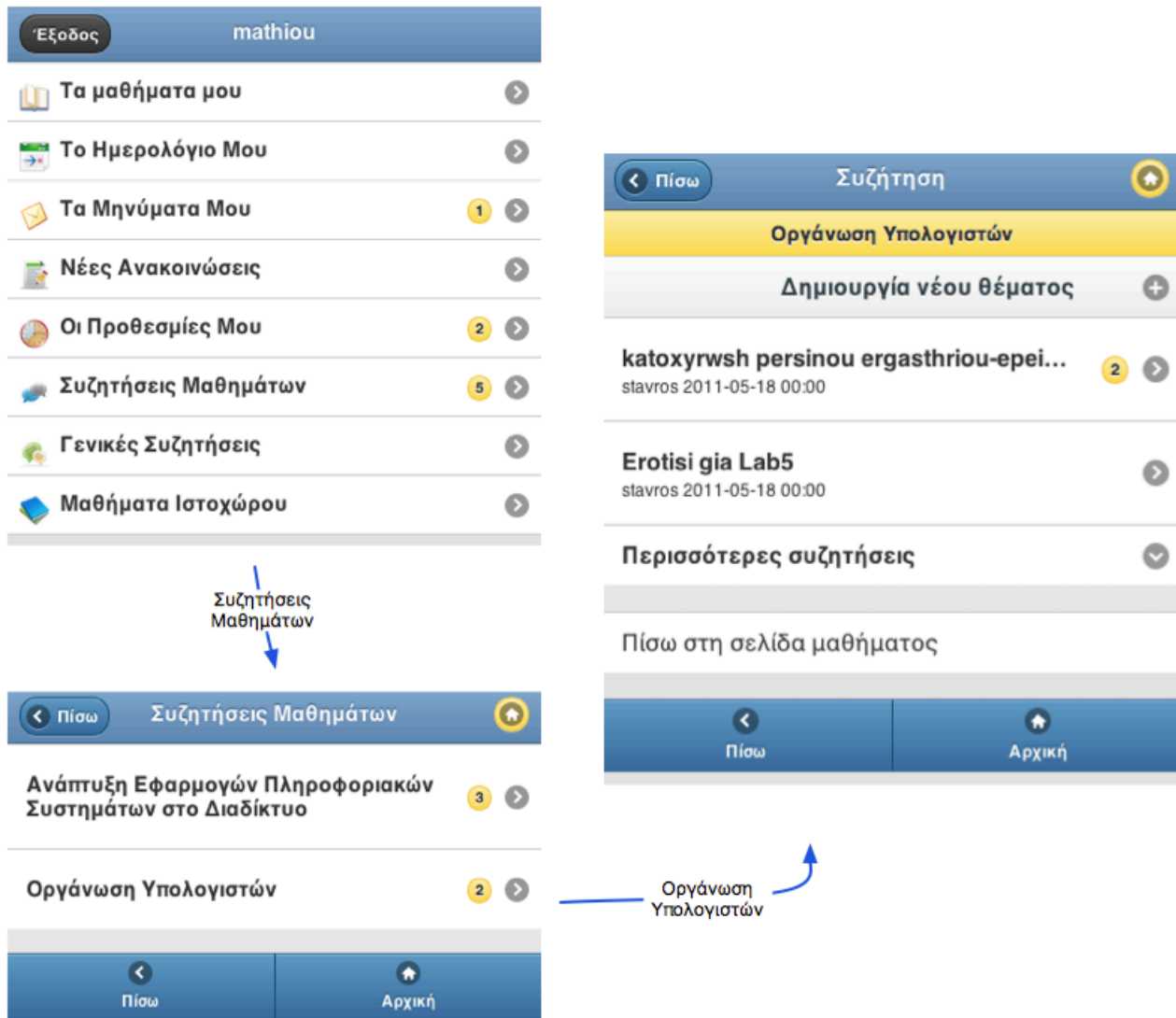


Εικόνα 5.46 - Η ακολουθία σελίδων από την αρχική σελίδα στη σελίδα με τις προθεσμίες παράδοσης μαθημάτων.



### 5.8.18 Προβολή νέων σχολίων σε συζητήσεις μαθημάτων του χρήστη.

Σχετίζεται με την Περίπτωση Χρήσης #5.3



Εικόνα 5.47 - Η ακολουθία σελίδων από την αρχική σελίδα του χρήστη στη σελίδα με τις συζητήσεις στα μαθήματα που είναι εγγεγραμμένος και εν τέλη σε κάποιο συγκεκριμένο μάθημα.



## 5.9 Παράδειγμα μεταφοράς σελίδων από το την Desktop έκδοση της εφαρμογής στην mobile έκδοση.

Για να μπορέσουμε να δείξουμε τις αλλαγές και την τροποποίηση που έχει γίνει, θα περιγράψουμε την εκτέλεση μια διαδικασίας σε βήματα και στις δύο εκδόσεις δείχνοντας πάνω στις οθόνες ποια τμήματα της μίας οθόνης ταιριάζουν με ποια στην άλλη. Η διαδικασία που επιλέξαμε είναι η προβολή ψηφιακού υλικού ενός μαθήματος. Την επιλέξαμε γιατί για να ολοκληρωθεί περνάει από όλες τις σημαντικές σελίδες και έτσι μπορούμε να δείξουμε εύκολα τις αλλαγές που έχουν προκύψει.

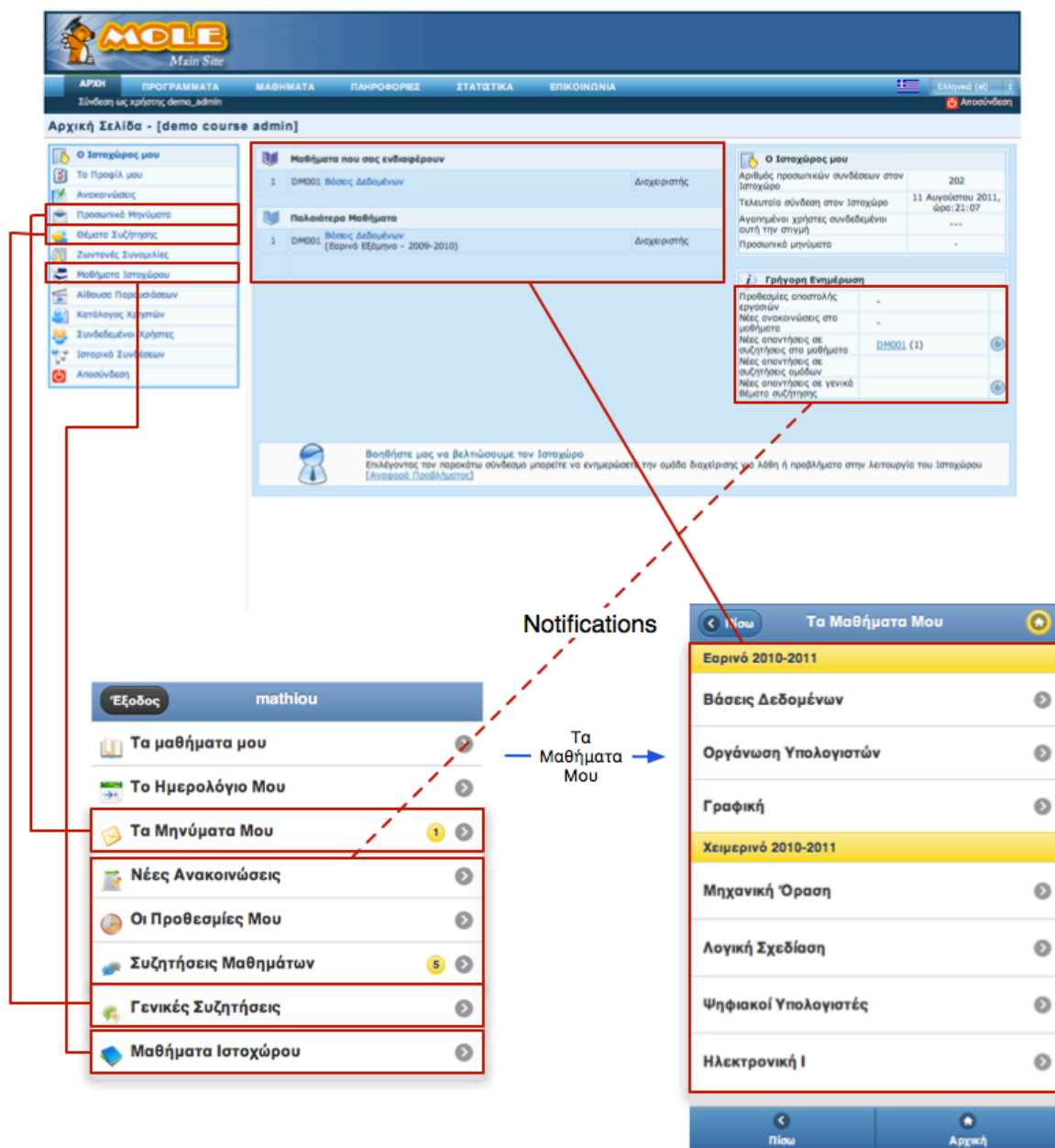
**Βήμα 1 :** φόρμα για είσοδο στο σύστημα. Στη σελίδα της mobile εφαρμογής βάλαμε μόνο τη φόρμα με την οποία χρήστης κάνει login και ένα link για το πλήρες site στο οποίο ο χρήστης μπορεί να κάνει και άλλες λειτουργίες. Είναι εύκολα φανερό ότι κρατάμε μόνο την χρήσιμη πληροφορία αφαιρώντας οτιδήποτε περιττό.



Εικόνα 6.41 - Οι σελίδες εισόδου στην desktop και mobile έκδοση αντίστοιχα.

Ο λόγος που δεν βάλαμε τις υπηρεσίες “εγγραφή χρήστη” και “είσοδο ως επισκέπτης” είναι ότι καταρχήν οι υπηρεσίες που προσφέρει η mobile εφαρμογή είναι μόνο για εγγεγραμμένα μέλη και ότι η εγγραφή δεν γίνεται άμεσα δεκτή αλλά μόνο μετά από αποδοχή των διαχειριστών του συστήματος.

**Βήμα 2 :** μετά από επιτυχή σύνδεση στο προηγούμενο βήμα βρισκόμαστε στην αρχική σελίδα του χρήστη. Η αρχική του χρήστη στην mobile έκδοση διαφέρει σε σχέση με την desktop ως προς την πληροφορία που περιέχει.



Εικόνα 6.42 - Η αρχική σελίδα του χρήστη στις δύο εκδόσεις. Φαίνεται πως έχουμε κρατήσει όλη τη πληροφορία που σχετίζεται με το χρήστη στην αρχική σελίδα της mobile έκδοσης και έχουμε μεταφέρει όλα τα μαθήματα σε μία δεύτερη σελίδα.

Η διαφορά έγκειται στο ότι στην mobile έκδοση στοχεύουμε στις υπηρεσίες που περιστρέφονται γύρω από το χρήστη. Αυτό μας οδήγησε στο να φτιάξουμε μια άλλη σελίδα για τα μαθήματα απομονώνοντας σε διαφορετικό μονοπάτι όλες τις ενέργειες που σχετίζονται με αυτά.

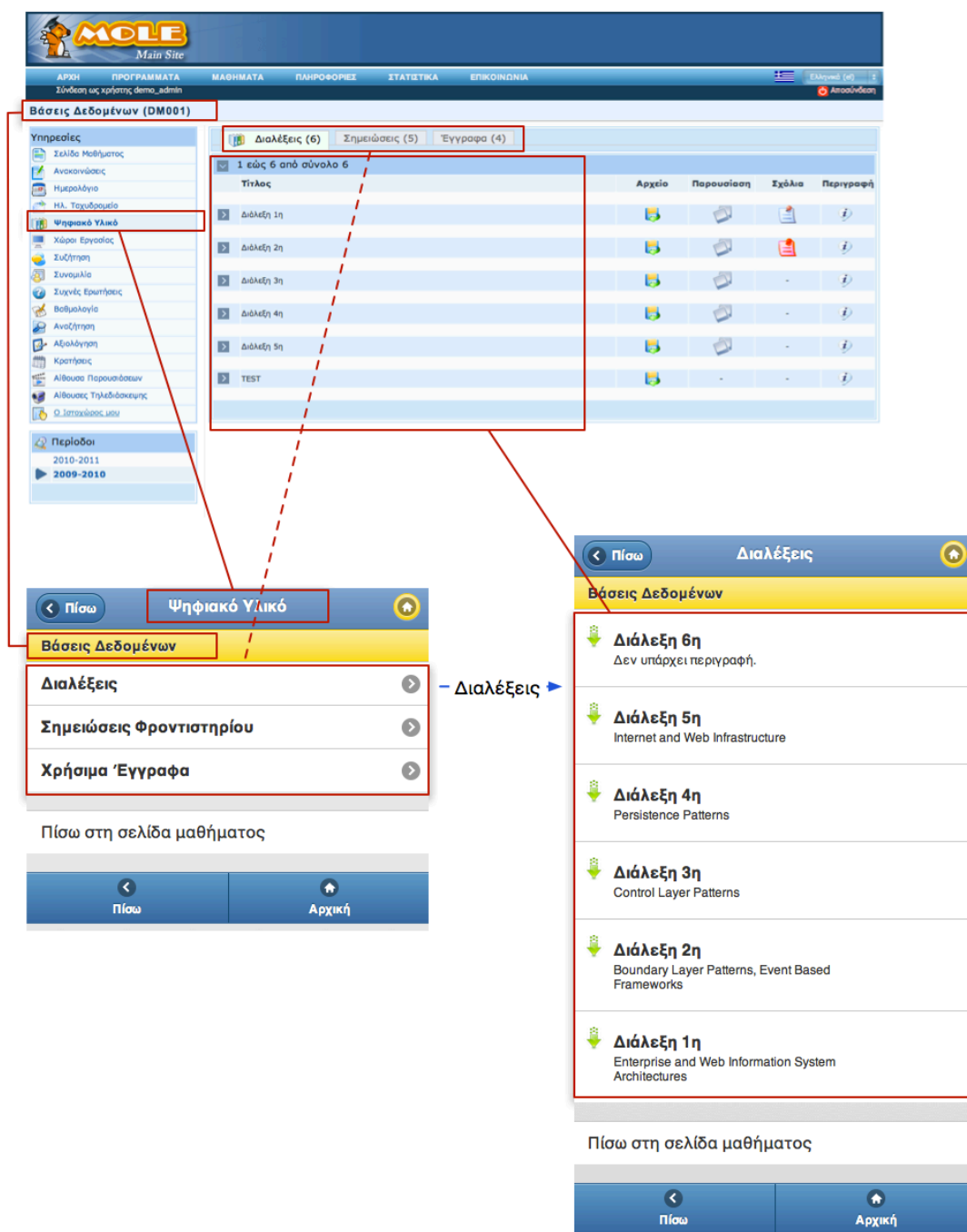
Βήμα 3: μπήκαμε στο μάθημα “Βάσεις Δεδομένων” στο βήμα 2. Εδώ αναγκαστήκαμε να



Εικόνα 6.43 - Η σελίδα ενός συγκεκριμένου μαθήματος και στις δύο εκδόσεις της εφαρμογής μας. Για να μπορέσουμε να εκμεταλλευτούμε σωστά την οθόνη βάλαμε τις πληροφορίες του μαθήματος σε άλλη σελίδα, δίνοντας χώρο σε όλες τις υπόλοιπες υπηρεσίες.

βάλουμε την πληροφορία του μαθήματος συγκεντρωμένη σε άλλη σελίδα. Έτσι αφήνουμε καθαρή τη σελίδα του μαθήματος από πληροφορία που δεν είναι χρήσιμη μέσω των συγκεκριμένων συσκευών, και παράλληλα επιτρέπουμε εύκολα στο χρήστη αν την χρειαστεί να μπορεί να την βρει.

**Βήμα 4:** πατήσαμε στο “Ψηφιακό Υλικό” στο βήμα 3. Εδώ αναγκαστήκαμε να σπάσουμε την πληροφορία σε δύο σελίδες.



Εικόνα 6.44 - Το ψηφιακό υλικό ενός συγκεκριμένου μαθήματος στις δύο διαφορετικές εκδόσεις της εφαρμογής. Η μεταφορά στην mobile έκδοση μας ανάγκασε να μεταφέρουμε τα tabs με τις επιλογές του υλικού σε μία σελίδα και το αντίστοιχα περιεχόμενο τους σε μια επόμενη.

Στη μία επιλέγουμε τι θέλουμε να δούμε από το ψηφιακό υλικό και στην άλλη βλέπουμε τα αντίστοιχα στοιχεία. Αυτό όμως δεν δημιουργήσε πρόβλημα γιατί πρώτον ο χρήστης μπορεί και βλέπει σε ποιο μάθημα είναι και με ένα μόνο tap να πάει στη σελίδα του μαθήματος (“πίσω στη σελίδα του μαθήματος”) ή με το πίσω μπορεί να πάει πίσω στο ψηφιακό υλικό.



# *Κεφάλαιο 6*

*Αξιολόγηση της Εφαρμογής στα στάδια της υλοποίησης*

## 6.1 Εισαγωγή

Στο κεφάλαιο αυτό περιγράφονται όλα τα βήματα που έγιναν κατά τη διάρκεια της υλοποίησης της εφαρμογής. Τόσο κατά το στάδιο της σχεδίασης (Ευριστική αξιολόγηση σε paper prototypes, walkthrough και pluralistic walkthrough) όσο και μετά από την ολοκλήρωση της υλοποίησης (think aloud evaluation). Παρουσιάζονται οι διαδικασίες που ακολουθήθηκαν, και ο τρόπος με τον οποίο συλλέξαμε όλες τις πληροφορίες που χρειαζόμασταν για να διορθώσουμε τα λάθη που υπήρχαν.

## 6.2 Ευριστική Αξιολόγηση (Heuristic Evaluation)

Η Ευριστική αξιολόγηση είναι μια μέθοδος εξέτασης ενός συστήματος από ειδικούς ευχρηστίας (user interface experts). Εξετάζει κατά πόσο τηρούνται οι διάφοροι κανόνες και αρχές σχεδιασμού διαδραστικών συστημάτων γενικά αλλά και ειδικότερα όσον αφορά την συγκεκριμένη εφαρμογή.

Μπορεί να εφαρμοστεί σε : Πρωτότυπα χαρτιού, Πρωτότυπα ενδιάμεσης ποιότητας, τελικό σύστημα. Εμείς, δεδομένου ότι χτίσαμε το σύστημα από την αρχή, είχαμε την πολυτέλεια να το εφαρμόσουμε στα χάρτινα πρωτότυπα κάνοντας εύκολα αλλαγές και διορθώσεις. Η εμπειρία από τη χρήση της μεθόδου υποδεικνύει ότι 3-5 αξιολογητές μπορούν να εντοπίσουν κατά μέσο όρο το 80% των προβλημάτων ευχρηστίας που υπάρχουν στο σύστημα, εξαλείφοντας όλα τα σημαντικά προβλήματα. Στην περίπτωση μας, επειδή η διπλωματική εργασία είναι ατομική, και επειδή ήταν πολύ δύσκολο να βρω διαθέσιμους φοιτητές με την ανάλογη εμπειρία, αυτής της μορφής η αξιολόγηση έγινε μόνο από εμένα. Αυτός ήταν ένας από τους λόγους για τον οποίου δεν την θεώρησα επαρκή και εκτέλεσα και αξιολόγηση με αληθινούς χρήστες, η οποία θα περιγραφεί στην συνέχεια του κεφαλαίου. Ωστόσο, η ιδιαιτερότητα μου, να είμαι και εγώ ο ίδιος χρήστης του συστήματος, μου έδωσε ένα πλεονέκτημα κατανόησης σε σχέση με την αξιολόγηση που θα έκανε κάποιος ανεξάρτητος αξιολογητής.

Η μέθοδος της ευριστικής αξιολόγησης αναπτύχθηκε από τον Jacob Nielsen στις αρχές της δεκαετίας του 1990. Ο Nielsen πρότεινε δέκα (10) γενικούς κανόνες για την αξιολόγηση διαδραστικών συστημάτων, οι οποίοι προέκυψαν από την ανάλυση 249 προβλημάτων ευχρηστίας.

Οι δέκα κανόνες είναι οι ακόλουθοι:

1. Ορατότητα της κατάστασης του συστήματος (Visibility of System Status)
2. Συσχέτιση συστήματος και πραγματικού κόσμου (Match between system and the real world)



3. Έλεγχος του συστήματος από το χρήστη (User Control & Freedom)
4. Συνέπεια και τήρηση προτύπων (Consistency & Standards)
5. Πρόληψη λαθών (Error Prevention)
6. Ελαχιστοποίηση του μνημονικού φορτίου (Recognition rather than recall)
7. Ευελιξία και αποδοτικότητα χρήσης (Flexibility & efficiency of use)
8. Αποφυγή περιττών στοιχείων (Aesthetic & minimalistic design)
9. Υποβοήθηση χρηστών στην αναγνώριση, διάγνωση και ανάνηψη από σφάλματα (Help users recognize, diagnose, and recover from errors)
10. Επαρκής υποστήριξη (Help & Documentation)

Προφανώς οι κανόνες του Nielsen δεν αποτελούν πανάκεια. Για ακριβή αξιολόγηση οι κανόνες χρειάζεται να τροποποιηθούν ώστε να μπορούν ανταποκρίνονται στη λογική της εφαρμογής που αξιολογείται. Οι κανόνες του Nielsen δεν αναφέρονται σε web applications συγκεκριμένα, κάτι που είναι ξεκάθαρο δεδομένου της χρονικής περιόδου που προτάθηκαν, πόσο μάλλον για web applications σε φορητές συσκευές (mobile devices). Αυτός είναι ένας δεύτερος λόγος ο οποίος έκανε αναγκαία την αξιολόγηση με αληθινούς χρήστες.

Ωστόσο η τήρηση και εφαρμογή αυτών των κανόνων στο σύστημα μας, είδαμε ότι μας οδήγησε στην δημιουργία ενός εύχρηστου συστήματος το οποίο στη συνέχεια έγινε εύκολα αποδεκτό και κατανοητό από τους χρήστες μας, γλιτώνοντας έτσι τις ριζικές αλλαγές στη δομή και την εμφάνιση της εφαρμογής.

#### Περιγραφή διαδικασίας:

Αφού σχεδίασα το σύστημα σε χάρτινα πρωτότυπα, άφησα μια βδομάδα χωρίς να ασχοληθώ καθόλου με την εφαρμογή, έτσι ώστε να μη είναι στο μυαλό μου η ακολουθία των σελίδων και η δομή της εφαρμογής. Στη συνέχεια εξέτασα το σύστημα δύο φορές σαν αξιολογητής προσπαθώντας να είμαι όσο το δυνατόν πιο “ανεξάρτητος” και αντικειμενικός, ακολουθώντας τους κανόνες του Nielsen.

Στο πρώτο πέρασμα : αναζήτηση προβλημάτων που ενδέχεται να αντιμετωπίσουν αρχάριοι χρήστες, καθώς και προβλημάτων σε σχέση με το νοητικό μοντέλο του συστήματος που ως σχεδιαστής προσπαθούσα να δημιουργήσω στους χρήστες. Το αρνητικό εδώ είναι, προφανώς, ότι έχω το ρόλο τόσο του σχεδιαστή όσο και του αξιολογητή. Το θετικό είναι ότι, και εγώ ως χρήστης του συστήματος, έχω ίδιο νοητικό μοντέλο στο μυαλό μου με τους αληθινούς χρήστες που στοχεύουμε.

Στο δεύτερο πέραςμα : έλεγξα τους πρωτεύοντες στόχους των χρηστών, τους στόχους ευχρηστίας που έχουν τεθεί καθώς και τις περιβαλλοντικές παραμέτρους. Για να γίνει το τελευταίο, μετέφερα στο κινητό μου φωτογραφίες από τα paper prototypes σε φακέλους έτσι ώστε να μπορώ να ελέγχω τη ροή και εκτέλεσα τη διαδικασία σε μια καφετέρια, όπου υπήρχε κόσμος και γενικότερη φασαρία. Σε ρεαλιστικές συνθήκες χρήσης μιας φορητής συσκευής δηλαδή.

### 6.3 Πρωτότυπα χαρτιού (Paper prototypes) & Walkthrough

Το δεύτερο στάδιο της αξιολόγησης ήταν να βάλω αληθινούς χρήστες να κάνουν περάσματα (walkthrough) της εφαρμογή σε χάρτινα πρωτότυπα (paper prototypes). Αφού έκανα τις αλλαγές που προέκυψαν από το πρώτο στάδιο και ξαναδημιούργησα νέες διεπαφές, έβαλα 3 χρήστες να εκτελέσουν κάποιες επιλεγμένες ενέργειες. Οι χρήστες αυτοί περιγράφονται από τα personas (1), (2) και (3) που περιγράφηκαν στο Κεφ.4

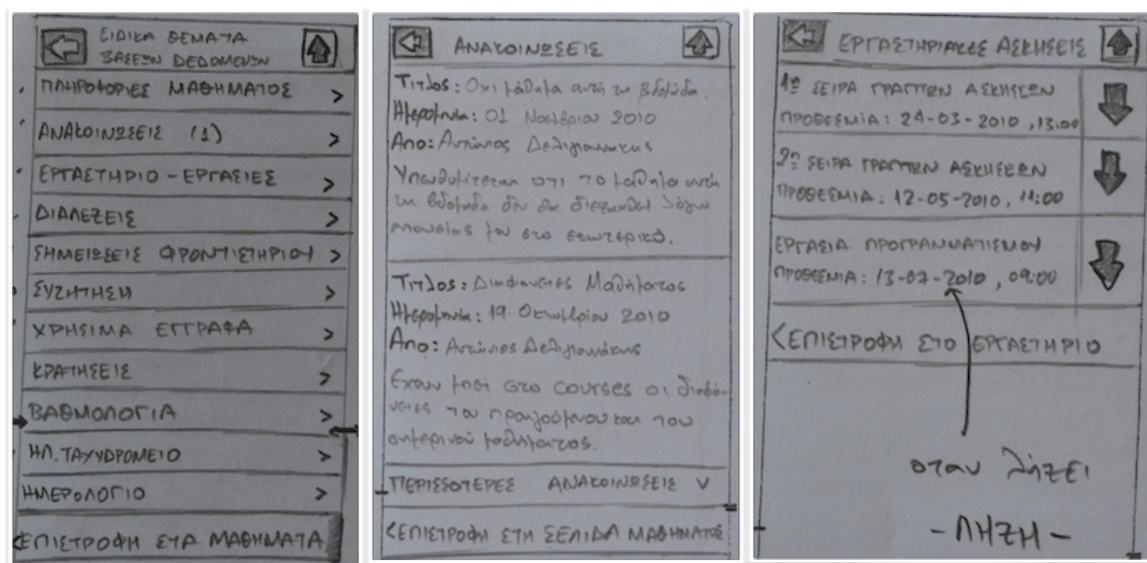
Οι ενέργειες που επέλεξα ήταν :

1. Προβολή πληροφοριών μαθήματος
2. Προβολή ανακοινώσεων μαθήματος
3. Πρόσβαση στις Διαλέξεις ενός μαθήματος. Δεν εκτέλεσα την αντίστοιχη λειτουργία για τα χρήσιμα έγγραφα και τις σημειώσεις φροντιστηρίου καθώς είναι η ίδια διαδικασία και θα κούραζε άσκοπα τους χρήστες μου.
4. Προβολή και απάντηση σε κάποια συζήτηση ενός μαθήματος. Δεν εκτέλεσα την αντίστοιχη λειτουργία στις γενικές συζητήσεις αφού η εμφάνιση και η λειτουργικότητα είναι ίδια.
5. Προβολή βαθμολογίας μαθήματος
6. Προβολή Ημερολογίου
7. Εγγραφή σε κάποιο εργαστήριο, και προβολή διαθέσιμου υλικού.
8. Προβολή εργασιών.
9. Προβολή εισερχομένων μηνυμάτων και αποστολή νέου μηνύματος.
10. Προβολή νέων ανακοινώσεων σε μαθήματα που είναι εγγεγραμμένος ο χρήστης.
11. Προβολή προθεσμιών παράδοσης εργασιών στα μαθήματα του χρήστη.

Οι αλλαγές και τα λάθη των χρηστών που προέκυψαν είχαν να κάνουν κυρίως με το διαφορετικό νοητικό μοντέλο που είχαν στο μυαλό τους οι χρήστες. Παρατήρησα ότι τα λάθη προέκυψαν στη λειτουργικότητα την οποία εγώ ως χρήστης του ιστοχώρου δεν χρησιμοποιώ συχνά, με αποτέλεσμα ως σχεδιαστής να το σκεφτώ διαφορετικά.

#### Περιγραφή διαδικασίας:

Περιέγραψα στους χρήστες τι ενέργειες είχαν να κάνουν και στη συνέχεια για κάθε ένα task που είχαν να εκτελέσουν τους εμφάνιζα την πρώτη σελίδα. Ανάλογα με τις επιλογές που έκαναν προχωρούσα στην επόμενη σελίδα. Κατά τη διάρκεια της εκτέλεσης κρατούσα σημειώσεις για τυχόν λάθη που συνέβαιναν. Άφηνα τους χρήστες να φτάσουν μέχρι το τέλος, ακόμα και αν έκαναν λάθος επιλογές ή “κύκλους”, ή μέχρι το σημείο να μην μπορούν να προχωρήσουν (κάτι που δεν συνέβη ποτέ). Αφού ολοκλήρωναν το task τους έλεγα πως είναι η σωστή διαδικασία με βάση το πως δουλεύει το σύστημα. Στην αξιολόγηση αυτή δεν υπήρχε συζήτηση με τον χρήστη. Δεν τον ρωτούσα τη γνώμη του, ούτε τον άφηνα να κρίνει αν συμφωνεί ή όχι με τα σωστά βήματα. Τα λάθη, αφού συγκεντρώθηκαν από όλους τους χρήστες, αξιολογούνταν στην συνέχεια χωρίς την παρουσία τους κάνοντας τις αλλαγές που κρίνονταν απαραίτητες με βάση τα λάθη αυτά. Στην ενότητα 6.1.2- “Αρχιτεκτονική της πληροφορίας” φαίνονται όλες οι σελίδες που χρησιμοποιήθηκαν σε αυτή τη φάση, ομαδοποιημένες με βάση το χρήστη και τα μαθήματα. Ακολουθούν κάποιες ενδεικτικές εικόνες από τα στοιχεία που χρησιμοποιήθηκαν στο paper prototype walkthrough.



Εικόνα 6.1 - Ενδεικτικές σελίδες από paper-prototypes-walkthrough.

## 6.4 Pluralistic walkthrough

Στο τρίτο στάδιο της αξιολόγησης ακολουθήσαμε τη μέθοδο του “Πλουραλιστικού περάσματος” - Pluralistic Walkthrough. Στη μέθοδο αυτή και πάλι είχα πολλαπλούς ρόλους εξαιτίας της έλλειψης ατόμων με την κατάλληλη γνώση. Ωστόσο αυτό δεν επηρέασε την ποιότητα της αξιολόγησης καθώς οι χρήστες ακολούθησαν την μέθοδο όπως έπρεπε και βγήκαν ασφαλή συμπεράσματα.

Στη μέθοδο αυτή θέλουμε να αποκτήσουμε πληροφορία για την ευχρηστία της εφαρμογής από δύο διαφορετικούς πληθυσμούς:

1. Τελικούς χρήστες - όπου στη περίπτωση μας περιγράφονται από τους χρήστες (2) και (3) που περιγράφηκαν στο Κεφ.4
2. Ειδικούς σε θέματα ευχρηστίας.

Η μέθοδος είναι παρόμοια με το παραδοσιακό walkthrough με την διαφορά ότι εδώ έχουν οριστεί κάποια συγκεκριμένα χαρακτηριστικά :

1. Έχουμε τρία (3) είδη συμμετεχόντων.
  - i. αντιπροσωπευτικοί χρήστες
  - ii. προγραμματιστές της εφαρμογής (developers)
  - iii. ειδικούς σε θέματα ανθρώπινης κατανόησης (human factor professionals)
2. Τα αντίγραφα παρουσιάζονται στους χρήστες με την σειρά που θα εμφανιστούν και ενώ είναι συνδεδεμένοι στο σύστημα (όπως και στο παραδοσιακό walkthrough).
3. Ζητάμε από τους συμμετέχοντες να υποθέσουν το ρόλο των χρηστών. Θεωρητικά στο σημείο αυτό οι developers και οι HF-professional έπρεπε να λειτουργήσουν σαν χρήστες. -Δεδομένου ότι αυτούς τους ρόλους τους είχα αναλάβει εγώ δεν έγινε κάτι τέτοιο.
4. Οι χρήστες πρέπει να γράφουν πάνω σε κάθε διαφάνεια την ενέργεια που θα έκαναν για να φτάσουν στο στόχο τους κάθε φορά, με όση περισσότερη λεπτομέρεια γινόταν.

5. Στη συνέχεια ξεκινάει η συζήτηση όπου οι χρήστες μιλάνε μόνοι τους. Αφού ολοκληρωθεί ο πρώτος κύκλος σχολίων, οι developers και οι HF-professional λένε τη γνώμη τους.

Περιγραφή διαδικασίας:

Η διαδικασία που ακολουθήθηκε βήμα προς βήμα ήταν η εξής:

1. Έδωσα στους χρήστες ένα φυλλάδιο με οδηγίες και τους έδωσα χρόνο να το διαβάσουν και να μου κάνουν τυχόν ερωτήσεις που είχαν πάνω σε αυτό.

## PLURALISTIC WALKTHROUGH

### ΕΝΕΡΓΕΙΕΣ ΠΟΥ ΘΑ ΑΞΙΟΛΟΓΗΘΟΥΝ:

1. ΠΡΟΒΟΛΗ ΠΛΗΡΟΦΟΡΙΩΝ ΜΑΘΗΜΑΤΟΣ
2. ΠΡΟΒΟΛΗ ΑΝΑΚΟΙΝΩΣΕΩΝ ΜΑΘΗΜΑΤΟΣ
3. ΠΡΟΣΒΑΣΗ ΣΤΙΣ ΔΙΑΛΕΞΕΙΣ ΤΟΥ ΜΑΘΗΜΑΤΟΣ
4. ΠΡΟΒΟΛΗ ΣΥΖΗΤΗΣΗΣ ΚΑΙ ΑΠΑΝΤΗΣΗ ΣΕ ΚΑΠΟΙΑ ΣΥΖΗΤΗΣΗ.
5. ΠΡΟΒΟΛΗ ΒΑΘΜΟΛΟΓΙΑΣ ΜΑΘΗΜΑΤΟΣ
6. ΠΡΟΒΟΛΗ ΗΜΕΡΟΛΟΓΙΟΥ ΜΑΘΗΜΑΤΟΣ
7. ΕΓΓΡΑΦΗ ΣΕ ΚΑΠΟΙΟ ΕΡΓΑΣΤΗΡΙΟ ΚΑΙ ΠΡΟΒΟΛΗ ΔΙΑΘΕΣΙΜΟΥ ΥΛΙΚΟΥ.
8. ΠΡΟΒΟΛΗ ΕΡΓΑΣΙΩΝ
9. ΠΡΟΒΟΛΗ ΕΙΣΕΡΧΟΜΕΝΩΝ ΜΗΝΥΜΑΤΩΝ ΚΑΙ ΑΠΟΣΤΟΛΗ ΝΕΩΝ ΜΗΝΥΜΑΤΩΝ.
10. ΠΡΟΒΟΛΗ ΠΡΟΒΕΣΙΩΝ ΠΑΡΑΔΟΣΗΣ ΕΡΓΑΣΙΩΝ
11. ΠΡΟΒΟΛΗ ΝΕΩΝ ΑΝΑΚΟΙΝΩΣΕΩΝ ΤΩΝ ΜΑΘΗΜΑΤΩΝ ΣΑΣ.

### ΒΗΜΑΤΑ ΠΟΥ ΘΑ ΑΚΟΛΟΥΘΗΣΟΥΝ:

- Α) ΠΑΙΡΝΟΥΜΕ ΤΗ ΔΙΑΦΑΝΕΙΑ ΠΟΥ ΛΕΕΙ Ο ΔΙΑΧΕΙΡΙΣΤΗΣ ΚΑΙ Η ΟΠΟΙΑ ΑΝΤΑΠΟΚΡΙΝΕΤΑΙ ΣΤΟ ΒΗΜΑ ΠΟΥ ΚΑΝΟΥΜΕ.
- Β) ΣΥΜΠΛΗΡΩΝΟΥΜΕ ΜΕ ΟΣΗ ΠΕΡΙΣΣΟΤΕΡΗ ΛΕΠΤΟΜΕΡΕΙΑ ΜΠΟΡΟΥΜΕ ΤΙΣ ~~ΕΡΩΤΗΣΕΙΣ~~ ΕΡΩΤΗΣΕΙΣ ΣΤΗ ΔΙΑΦΑΝΕΙΑ. (ΤΙ ΚΑΝΩ; ΓΙΑΤΙ ΤΟ ΚΑΝΩ; ΤΙ ΜΕ ΠΡΟΒΛΗΜΑΤΙΖΕΙ; ΤΙ ΜΟΥ ΑΡΕΣΕΙ; Κ.Τ.Λ.).
- ❗ ΔΕΝ ΒΛΕΠΟΥΜΕ ΕΠΟΜΕΝΕΣ Ή ΠΡΟΗΓΟΥΜΕΝΕΣ ΔΙΑΦΑΝΙΕΣ ΑΝ ΔΕΝ ΤΟ ΛΕΙ Ο ΔΙΑΧΕΙΡΙΣΤΗΣ.
- Γ) ΔΕΝ ΜΙΛΑΜΕ ΜΕΧΡΙ ΝΑ ΟΠΟΙΛΗΡΕΣΟΥΝ ΟΛΟΙ ΟΙ ΧΡΗΣΤΕΣ ΤΟ ΒΗΜΑ -Γ3-. ΑΦΟΥ ΤΕΛΕΙΩΣΟΥΝ ΟΛΟΙ ΣΥΖΗΤΑΙ ΜΕΤΑΞΥ ΣΑΣ ΤΙ ΕΠΙΛΕΞΑΤΕ, ΓΙΑΤΙ, ΤΙ ΣΑΣ ΠΡΟΒΛΗΜΑΤΙΖΕ Κ.Τ.Λ.
- Δ) ΜΠΑΙΝΩ ΚΑΙ ΕΓΩ ΣΤΗ ΣΥΖΗΤΗΣΗ. ΣΑΣ ΥΠΟΔΕΙΧΝΩ ΤΗ ΣΩΣΤΗ ΕΛΛΟΓΗ ΚΑΙ ΣΥΖΗΤΑΜΕ.

❗ ΝΑ ΕΙΣΤΕ ΟΣΟ ΤΟ ΔΥΝΑΤΟΝ ΠΙΟ ΕΠΙΣΤΙΜΩΙ - ΟΧΙ ΠΑΡΑΛΟΓΟΙ - ΓΙΑΤΙ ΣΚΟΠΟΣ ΕΙΝΑΙ ΝΑ ΑΞΙΟΛΟΓΗΘΕΙ ΤΟ Ο.Ι. ΚΑΙ ΝΑ ΕΝΤΟΠΙΣΤΟΥΝ ΠΡΟΒΛΗΜΑΤΑ.

Εικόνα 6.2 - Φυλλάδιο οδηγιών που δόθηκε στους χρήστες

2. Στη συνέχεια τους έκανα μια σύντομη περιγραφή γύρω από την εφαρμογή και το περιβάλλον της. Τους τόνισα επίσης τις ιδιαιτερότητες της χρήσης μια τέτοιας εφαρμογής σε πραγματικές συνθήκες, τις οποίες θα έπρεπε να έχουν στο μυαλό τους κατά τη διάρκεια της αξιολόγησης.
3. Στη συνέχεια ξεκινήσαμε την εκτέλεση των tasks. Εδώ οι χρήστες έγραφαν τι θα έκαναν για να συνεχίσουν την ενέργεια σε κάθε μία διαφάνεια. Κάθε χρήστης έγραφε πάνω σε κάθε διαφάνεια τι σκεφτόταν, τι ενέργεια θα έκανε και τι περίμενε να δει στη συνέχεια. Στη συνέχεια γινόταν συζήτηση πάνω στη συγκεκριμένη διαφάνεια. Δεν προχωρούσε κανένας σε επόμενη διαφάνεια αν δεν είχαν ολοκληρώσει όλοι. Το πότε θα άλλαζαν διαφάνεια το όριζα εγώ, ως διαχειριστής της διαδικασίας. Αρχικά μιλούσαν μόνο οι χρήστες μεταξύ τους, λέγοντας τι τους δυσκόλεψε ή τι δεν αντιλήφθηκαν πως έπρεπε να γίνει. Για να είναι ιδιαίτερα αυστηροί και επικριτικοί προσπαθούσαν να δίνω εγώ το ρυθμό για κάτι τέτοιο, τονίζοντας διαρκώς πως όλη αυτή η διαδικασία γίνεται για να αξιολογηθεί το σύστημα και ότι δεν πρέπει να είναι ελαστικοί. Όσο μιλούσαν εγώ κρατούσα σημειώσεις σχετικά με το τι σκεφτόταν και τι τους προβλημάτιζε, παίρνοντας το ρόλο του HF-professional. Στη συνέχεια έμπαινα και εγώ στη συζήτηση αυτή τη φορά με το ρόλο του developer κάνοντας προτάσεις για τα προβλήματα που είχαν προκύψει και πως συζητώντας με τους χρήστες εναλλακτικές επιλογές όπου ήταν αυτό δυνατό. Ακολουθούν 7 hard-copies των διαφανειών μετά την ολοκλήρωση της διαδικασίας. Σε καθένα από αυτά φαίνονται δύο από τους χρήστες μας καθώς και οι δικές μου σημειώσεις. Επέλεξα τις συγκεκριμένες καθώς ήταν από αυτές οι οποίες με οδήγησαν σε σημαντικές αλλαγές και σε κανόνες που ακολουθήθηκαν μέχρι την τελική μορφή της εφαρμογής.

ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ...	
ΠΑΡΡΟΦΟΡΕΣ ΜΑΘΗΜΑΤΟΣ	>
ΑΝΑΚΩΙΝΩΣΕΙΣ (1)	>
ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ	>
ΔΙΑΜΕΣΕΙΣ	>
ΣΗΜΕΙΩΣΕΙΣ ΦΡΟΝΤΙΣΤΗΡΙΟΥ	>
ΣΥΖΗΤΗΣΗ	>
ΧΡΗΣΙΜΑ ΕΓΓΡΑΦΑ	>
ΒΑΘΜΟΛΟΓΙΑ	>
ΗΜΕΡΟΛΟΓΙΟ	>
< ΕΠΙΣΤΡΟΦΗ ΣΤΑ ΜΑΘΗΜΑΤΑ	

Τι θέλει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να βρω τις υπηρεσίες του μαθήματος,  
να ειδοποιηθώ στην αρχική ή τη σελίδα μαθημάτων.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Μοδάλινο στο εργαστήριο-εργασίες.

Χρήστης : Θωμάς

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Είναι πολλά! Λίγο κουραστικό. Αν  
βωρούς με κάδοιο τρόπο να τα  
μειώσεις ή να βάλεις εικονίδια.  
Περίμενα να έχει περισσότερα  
notifications. Είμαι εγγεγραμμένος;  
Πως εθύνονται;

Εικόνα 6.3 - 1ο Παράδειγμα: Κεντρική σελίδα ενός συγκεκριμένου μαθήματος - Χρήστης Α.



ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΣΤΕΦΕΝ...	
ΠΑΡΡΟΦΟΡΙΕΣ ΜΑΘΗΜΑΤΟΣ	>
ΑΝΑΚΩΝΣΕΙΣ (1)	>
ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ	>
ΔΙΑΜΕΡΕΙΣ	>
ΣΗΜΕΙΩΣΕΙΣ ΦΡΟΝΤΙΣΤΗΡΙΟΥ	>
ΣΥΖΗΤΗΣΗ	>
ΧΡΗΣΙΜΑ ΕΓΓΡΑΦΑ	>
ΒΑΘΜΟΛΟΓΙΑ	>
ΗΜΕΡΟΛΟΓΙΟ	>
< ΕΠΙΣΤΡΟΦΗ ΣΤΑ ΜΑΘΗΜΑΤΑ	

Τι θέλει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να μπω στα: 1) Παρροφορίες μαθήματος, 2) Ανακοινώσεις,  
3) Εργαστήριο-Εργασίες, 4) Διαλέξεις, 5) Σημειώσεις φροντιστηρίου,  
6) Συζήτηση, 7) Χρήσιμα έγγραφα, 8) Βαθμολογία,  
9) Ημερολόγιο & να επιβεβαιώσω ότι αρχίζω  
ή ότι τελειώω μαθημάτων.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Μπορώ στο Εργαστήριο-Εργασίες.

Χρήστης: Ντόρα

Σχόλια - Παρατηρήσεις

Εικόνα 6.4 - 1ο Παράδειγμα: Κεντρική σελίδα ενός συγκεκριμένου μαθήματος - Χρήστης Β

ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ...	
ΠΛΗΡΟΦΟΡΙΕΣ ΜΑΘΗΜΑΤΟΣ	>
ΑΝΑΚΩΙΝΥΣΕΙΣ (1)	>
ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ	>
ΔΙΑΜΕΡΙΣΕΙΣ (N)	>
ΣΗΜΕΙΩΣΕΙΣ ΦΡΟΝΤΙΣΤΗΡΙΟΥ (N)	>
ΣΥΖΗΤΗΣΗ (N)	<
ΧΡΗΣΙΜΑ ΕΓΓΡΑΦΑ (N)	>
ΒΑΘΜΟΛΟΓΙΑ	>
ΗΜΕΡΟΛΟΓΙΟ	>
< ΕΠΙΣΤΡΟΦΗ ΣΤΑ ΜΑΘΗΜΑΤΑ	

⊗ ΕΓΓΡΑΦΗ ΣΤΗ ΚΟΡΥΦΗ ΑΝ ΔΕΝ ΕΙΝΑΙ ΓΡΑΜΜΕΝΟΣ.

Τι θέλει ή μπορεί να κάνει ο χρήστης;

← ΕΙΔΟΝΙΔΙΑ!

↗ ΤΑ ΧΡΗΣΙΜΑ ΕΓΓΡΑΦΑ ΚΟΝΤΑ ΣΕ ΔΙΑΜΕΡΙΣΕΙΣ + ΣΗΜΕΙΩΣΕΙΣ.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:  
NOTIFICATIONS

ΕΓΓΡΑΦΗ / ΔΙΑΓΡΑΦΗ ΑΠ' ΤΟ ΜΑΘΗΜΑ? ⊗

ΕΜΦΑΝΙΣ ΟΤΑΝ ΜΠΑΙΝΕΙ ΣΤΟ ΜΑΘΗΜΑ!

ΔΙΑΓΡΑΦΗ ΟΧΙ ΣΤΟ MOBILE. ΠΕΡΙΤΤΟ.

ΧΡΗΣΤΗΣ : —

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Εικόνα 6.5 - 1ο Παράδειγμα : Κεντρική σελίδα ενός συγκεκριμένου μαθήματος - Δικές μου σημειώσεις πάνω στη συγκεκριμένη διαφάνεια.

ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ	
ΥΛΙΚΟ ΕΡΓΑΣΤΗΡΙΟΥ	
TITLE 01 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 02 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 03 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 04 ΠΕΡΙΓΡΑΦΗ	↓
ΠΕΡΙΣΣΟΤΕΡΑ ΑΡΧΕΙΑ ▼	
← ΕΠΙΣΤΡΟΦΗ ΣΤΟ ΕΡΓΑΣΤΗΡΙΟ	

Τι θέλει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να εισγράψω στις  
σελίδες: αρχική κ' εργαστήριο,  
να φορτώσω περισσότερα  
υλικά και να κατεβάσω τα  
αρχεία αυτά.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Πατώνω πάνω στον τίτλο ή  
στο βελόκι.

ΧΡΗΣΤΗΣ: Θωφιάς

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Όλα δίνω κατεβαίνουν ή ανόιγουν  
πάντα; Έχει ποτέ σκέτη περιγραφή  
και όχι λήψη στο courses;

Εικόνα 6.6 - 2ο Παράδειγμα: Σελίδα με το υλικό εργαστηρίου. - Χρήστης Α

ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ	
ΥΛΙΚΟ ΕΡΓΑΣΤΗΡΙΟΥ	
TITLE 01 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 02 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 03 ΠΕΡΙΓΡΑΦΗ	↓
TITLE 04 ΠΕΡΙΓΡΑΦΗ	↓
ΠΕΡΙΣΣΟΤΕΡΑ ΑΡΧΕΙΑ ▼	
Σ ΕΠΙΣΤΡΟΦΗ ΣΤΟ ΕΡΓΑΣΤΗΡΙΟ	

Τι θέλει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να θυρίσω την αρχική  
σελίδα ή εργαστήριο, να "φορτώσω"  
περισσότερο υλικό ή σ' να υατέβα-  
λω ή να δω τα αρχία αυτά.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Πατών πάνω στο βελού.








ΧΡΗΣΤΗΣ : Ντόρα

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Σε ποιο μορφή ή εργαστήριο ήγαι???

Εικόνα 6.7 - 2ο Παράδειγμα : Σελίδα με το υλικό εργαστηρίου. - Χρήστης Β



 ΔΙΑΛΕΞΕΙΣ 
ΕΙΣΑΓΩΓΗ ΚΑΙ ΔΙΑΔΙΚΑΣΤΙΚΑ ΕΙΣΑΓΩΓΗ 
QUERY PROCESSING Query Processing, κόστος συζητήσεων 
Data Cubes Data Cubes 
Ποιοδοτικότητα Ευρεσιμίας 
ΠΕΡΙΣΣΟΤΕΡΕΣ ΔΙΑΛΕΞΕΙΣ 
< ΕΠΙΣΤΡΟΦΗ ΣΤΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ

ΧΡΗΣΤΗΣ : Θωμάς

Τι θέλει ή μπορεί να κάνει ο χρήστης; Βλέπω γρήγορα τις ώρες 5 διαλέξεις και μπορώ αν το επιλέξω να δω και άλλα 5. Ανεπαρκώς δάνω σε μία διάλεξη μπορώ να την κατεβάσω. Μπορώ να εδωστρέψω στη σελίδα μάθηματος και στην αρχική σελίδα.








ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Για να δω την διάλεξη θα δώσω δάνω στο βελάκι για να την κατεβάσω και να τη δω σαν αρχείο.

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Η (-) πάντα σηφαίνει ότι δεν υπάρχει περιγραφή σωστά; Σε ποιο μάθημα είναι?

Εικόνα 6.9 - 3ο Παράδειγμα : Σελίδα με διαλέξεις μαθήματος - Χρήστης Α

 ΔΙΑΛΕΞΕΙΣ 
ΕΙΣΑΓΕΓΗ ΚΑΙ ΔΙΑΔΙΚΑΣΤΙΚΑ ΕΙΣΑΓΕΓΗ 
QUERY PROCESSING Query Processing, κομμάτι διαδικασίας 
Data Cubes Data Cubes 
Πολυδιάστατα Ερωτήματα <del>Ερωτήματα</del> 
ΠΕΡΙΣΣΟΤΕΡΕΣ ΔΙΑΛΕΞΕΙΣ 
< ΕΠΙΣΤΡΟΦΗ ΣΤΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ

Τι βλέπει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να βλέπω δηλαδή τις πέντε  
πρώτες διαλέξεις ή αν το επιλέξω μπορώ  
να δω όλη την. Αν επιλέξω, πατήσω σε μια  
διάλεξη μπορώ να την κατεβάσω.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

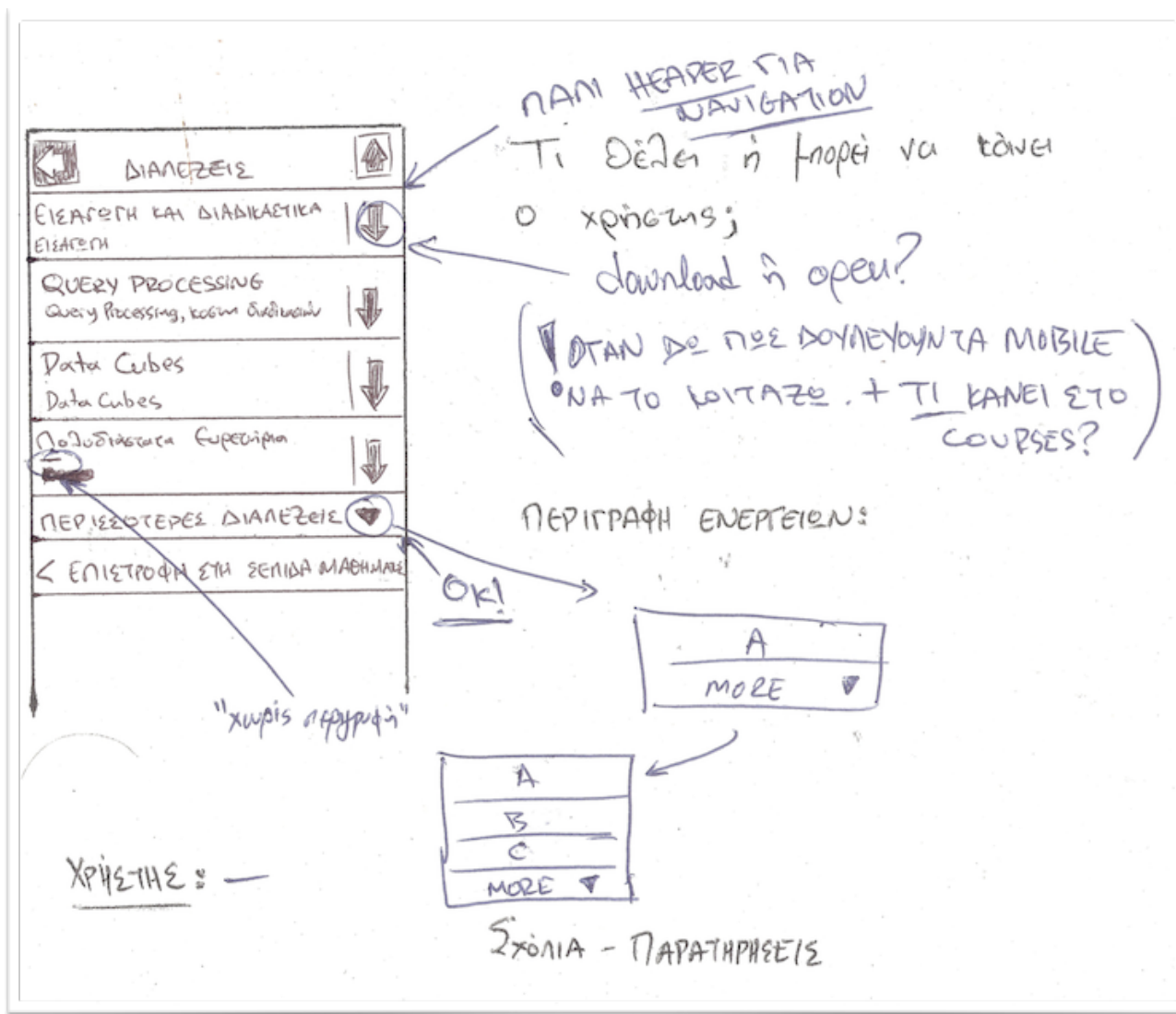
Πατώνω πάνω στο όνομα της διαλέξης  
ή στο βελάκι.

Χρήστης: Ντόρα

Σχόλια - Παρατηρήσεις



Εικόνα 6.10 - 3ο Παράδειγμα: Σελίδα με διαλέξεις μαθήματος - Χρήστης Β





Εικόνα 6.11 - 3ο Παράδειγμα : Σελίδα με διαλέξεις μαθήματος - Δικές μου σημειώσεις πάνω στη συγκεκριμένη διαφάνεια.



 ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ 
<b>Ομάδες Εργασίας</b> Οι ομάδες αποτελούνται από 2 άτομα <b>Αντικείμενο Ομάδας</b> Δείτε την ομάδα σας
<b>Υλικο Εργαστηρίου</b> Εργαστηριακές Στοιχειώδεις Υλικοί Υλικοί (12 στοιχεία)
<b>ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ</b> <b>ΕΡΓΑΣΙΕΣ</b> (3) >
<b>ΕΠΙΣΤΡΟΦΗ ΣΤΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ</b>

ΧΡΗΣΤΗΣ : Θωμάς

Τι βλέπει ή μπορεί να κάνει ο χρήστης;

Βρίσκει στη σελίδα εργαστηρίου. Μπορώ να δω το project, το εργαστηριακό υλικό και ότι έχει σχέση με την ομάδα μου.



**ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:**

Για να δω το project θα πατήσω στο "εργαστηριακές ασκήσεις-εργασίες".

**ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ**

Το υλικό εργαστηρίου δεν είναι link? Σε ποιο μάθημα ή εργαστήριο είναι?

Εικόνα 6.12 - 4ο Παράδειγμα : Σελίδα εργαστηρίου - Χρήστης Α

 ΕΡΓΑΣΤΗΡΙΟ - ΕΡΓΑΣΙΕΣ 
Ομάδες Εργασίας Οι ομάδες αποτελούνται από 2 άτομα Ανήκουν στην Ομάδα Δείτε την ομάδα σας
ΥΛΙΚΟ ΕΡΓΑΣΤΗΡΙΟΥ Εργαστηριακές Σημειώσεις Υλικά Υλικά (10 ερωτήσεις)
ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ ΕΡΓΑΣΙΕΣ (3) >
ΕΠΙΣΤΡΟΦΗ ΣΤΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ

ΧΡΗΣΤΗΣ : Ντορα

Τι θέλει ή μπορεί να κάνει  
ο χρήστης;

Μπορώ να μπω & να δω το project μου,  
να δω ποσα αρχια υπάρχουν στο  
"νέιο εργαστήριο" & να δω την  
ομάδα μου.

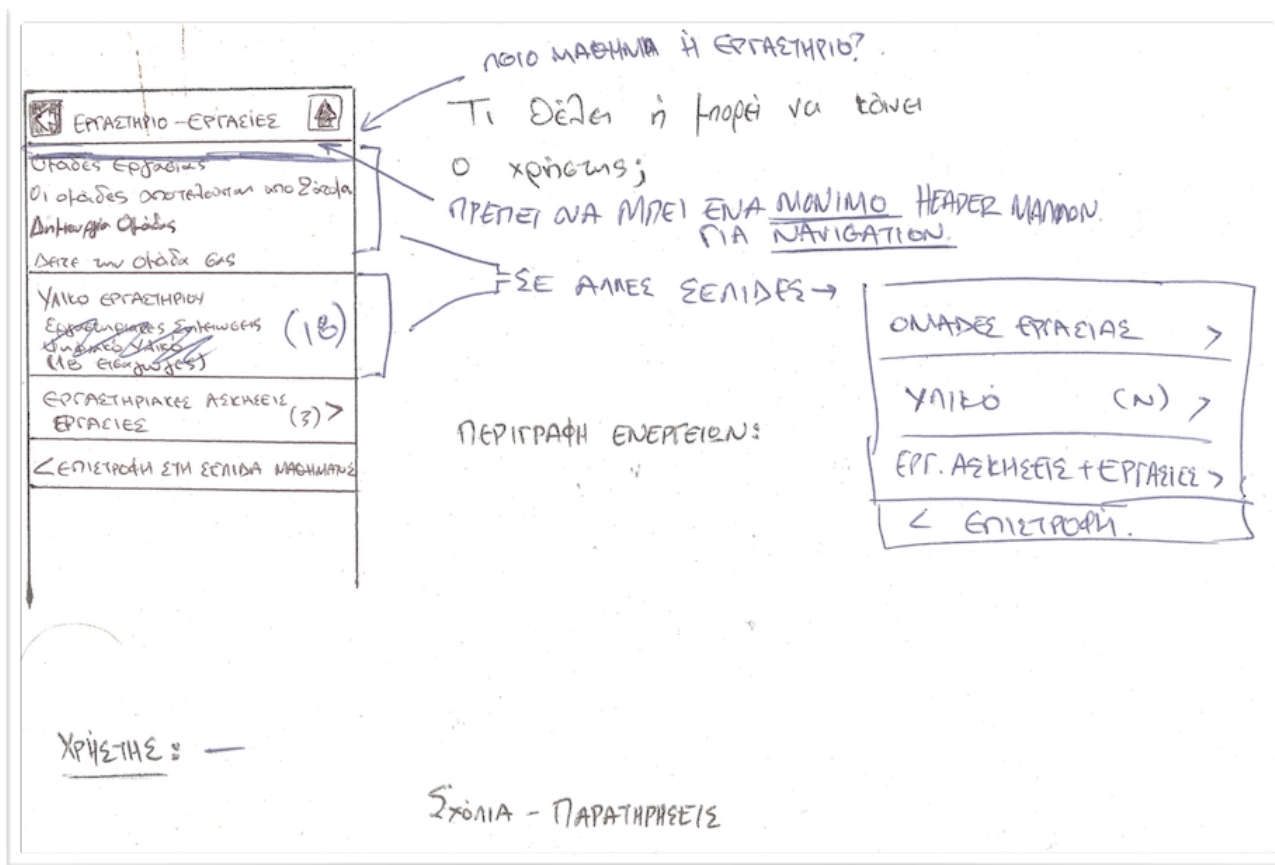
ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Αν θέλω να δω το project μου  
θα πατησω επάνω στο "Εργαστη-  
ριακή Αγωγή - Εργασίες".






ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Μπορώ να δω το υλικό εργαστηρίου;  
Ποσα ατυρήα υπάρχουν??

Εικόνα 6.13 - 4ο Παράδειγμα : Σελίδα εργαστηρίου - Χρήστης Β



Εικόνα 6.14 - 4ο Παράδειγμα : Σελίδα εργαστηρίου - Δικές μου σημειώσεις πάνω στη συγκεκριμένη

 ΤΑ ΜΑΘΗΜΑΤΑ ΜΟΥ 	
 ΧΕΙΜΕΡΙΝΟ 2010-2011	
ΠΑΝ692: ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	>
ΗΡΥ416: ΣΧΕΔΙΑΣΗ ΑΝΑΠΟΤΙΚΩΝ ΕΥΚΛΕΩΜΑΤΩΝ CMOS	>
ΤΗΛ301: ΤΗΛΕΠΙΚΩΝΗΜΕΝΑ ΣΥΣΤΗΜΑΤΑ I	>
 ΕΑΡΙΝΟ 2009-2010	
 ΕΠΙΣΤΡΟΦΗ ΣΤΗΝ ΑΡΧΙΚΗ	

Τι θέλει ή μπορεί να κάνει ο χρήστης; Μπορώ να βρω για μαθήματα αυτού του εξαμήνου και εγιστρέσω στην αρχική. Βλέπω επίσης και τα μαθήματα του προηγούμενου εξαμήνου.



ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:  
Εγιστρώ το μάθημα που θέλω να βρω.

ΧΡΗΣΤΗΣ: Θωφίας

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Στο προηγούμενο εξάμηνο υποθέτω ότι βλέπω τα δικά μου μαθήματα καλά, ε?

Εικόνα 6.15 - 5ο Παράδειγμα : Σελίδα μαθημάτων του χρήστη - Χρήστης Α

 ΤΑ ΜΑΘΗΜΑΤΑ ΜΟΥ 	
ΧΩΜΕΡΙΝΟ 2010-2011	
ΠΑΛ682: ΕΙΔΙΚΑ ΘΕΜΑΤΑ ΒΑΣΕΩΝ ΔΕΔΑΓΜΕΝΩΝ	>
ΠΡΥ416: ΣΧΕΔΙΑΣΗ ΑΝΑΠΟΤΙΚΩΝ ΕΥΚΛΕΝΑΤΩΝ CMOS	>
ΤΗΛ301: ΤΗΛΕΠΙΚΩΜΜΕΝΙΑ ΚΑΙ ΣΥΣΤΗΜΑΤΑ Ι	>
ΣΦΙ ΕΑΡΙΝΟ 2009-2010	
< ΕΠΙΣΤΡΟΦΗ ΣΤΗΝ ΑΡΧΙΚΗ	

Τι θέλει ή μπορεί να κάνει ο χρήστης;

Μπαίνω στα μαθήματα μου (αυτού του εξαμήνου) & επιστρέφω στην αρχική σελίδα. Επίσης, μπορώ να δω & τα μαθήματα του προηγούμενου εξαμήνου

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:

Επιλέγω το μάθημα στο οποίο θέλω να μπω.

Χρήστης: Ντορά

Σχόλια - Παρατηρήσεις

Εικόνα 6.16 - 5ο Παράδειγμα : Σελίδα μαθημάτων του χρήστη - Χρήστης Β

ΤΑ ΜΑΘΗΜΑΤΑ ΜΟΥ
🏠

---

ΧΘΕΡΙΝΟ 2010-2011

ΠΛΗ692: ΕΙΔΙΚΑ ΘΕΜΑΤΑ  
ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ >

ΗΡΥ416: ΣΧΕΔΙΑΣΗ ΑΝΑΛΟΓΙΚΩΝ  
ΚΥΚΛΩΜΑΤΩΝ CMOS >

ΤΗΛ301: ΣΤΗΛΑΠΕΡΙΩΜΕΝΙΑΚΑ  
ΣΥΣΤΗΜΑΤΑ Ι >

ΕΡΕ ΕΑΡΙΝΟ 2009-2010 ▼

< ΕΠΙΣΤΡΟΦΗ ΣΤΗΝ ΑΡΧΙΚΗ

Τι δίνει ή πορεί να κάνει  
ο χρήστης;

παιχτό παιχνίδι. Αν δα βάλει χαρτιά  
να φύγει.

ΠΕΡΙΓΡΑΦΗ ΕΝΕΡΓΕΙΩΝ:  
ΜΟΝΙΜΑ ΑΝΟΙΧΤΟ. ΔΕΝ ΕΠΙΒΑΡΥΝΕΙ.

(Το expand να είναι μόνο στο "more items")  
παιχνίδι.

ΧΡΗΣΤΗΣ: \_\_\_\_\_

ΣΧΟΛΙΑ - ΠΑΡΑΤΗΡΗΣΕΙΣ

Εικόνα 6.17 - 5ο Παράδειγμα : Σελίδα μαθημάτων του χρήστη - Δικές μου σημειώσεις πάνω στη συγκεκριμένη διαφάνεια.

4. Αφού ολοκληρώθηκε η αξιολόγηση δόθηκε ένα σύντομο ερωτηματολόγιο στους χρήστες για να αξιολογήσουν το interface και γενικότερα την ευχρηστία της εφαρμογής. Τα σχόλια ήταν γενικά θετικά λέγοντας πως είναι εύχρηστη αν διορθωθούν τα προβλήματα που εντοπίστηκαν και ότι παρέχει στο χρήστη όλη την αναγκαία λειτουργικότητα. Οι χρήστες τόνισαν ότι η έλλειψη χρωμάτων στο interface δεν τους βοήθησε, και ότι με την προσθήκη τους η διεπαφή θα γινόταν πολύ καλύτερη και ακόμα πιο εύχρηστη.

Περιορισμοί της μεθόδου :

1. Η διαδικασία είναι τόσο αργή όσο ο πιο αργός συμμετέχων.
2. Δεν μπορούν να προσομοιωθούν όλες οι πιθανές ενέργειες σε hard-copy.
3. Πρέπει να επιλέξουμε ένα “σωστό μονοπάτι” το οποίο θα ακολουθηθεί. Αυτό σημαίνει ότι οι χρήστες που ακολούθησαν λάθος μονοπάτια ή εναλλακτικά σωστά θα πρέπει να αναθεωρήσουν και να προσαρμόσουν την σκέψη τους στο δεδομένο σωστό.

## 6.5 Think Aloud Evaluation

Αφού υλοποιήσαμε την εφαρμογή, ακολουθώντας τους κανόνες και τα αποτελέσματα που προέκυψαν από τις προηγούμενες φάσεις της αξιολόγησης, βάλαμε χρήστες να εκτελέσουν τις ίδιες ενέργειες πάνω στο πραγματικό σύστημα. Σκοπός ήταν να δούμε πως αντιδρούν οι χρήστες σε πραγματικές συνθήκες και όχι πάνω από ένα χαρτί χωρίς τη διαδραστικότητα που έχει η ηλεκτρική συσκευή.

Στη μέθοδο αυτή έβαλα τους χρήστες μου να εκτελέσουν τις ενέργειες :

1. Προβολή πληροφοριών μαθήματος
2. Προβολή ανακοινώσεων μαθήματος
3. Πρόσβαση στις Διαλέξεις ενός μαθήματος. Δεν εκτέλεσα την αντίστοιχη λειτουργία για τα χρήσιμα έγγραφα και τις σημειώσεις φροντιστηρίου καθώς είναι η ίδια διαδικασία και θα κούραζε άσκοπα τους χρήστες μου.
4. Προβολή και απάντηση σε κάποια συζήτηση ενός μαθήματος. Δεν εκτέλεσα την αντίστοιχη λειτουργία στις γενικές συζητήσεις αφού η εμφάνιση και η λειτουργικότητα είναι ίδια.

5. Προβολή βαθμολογίας μαθήματος
6. Προβολή Ημερολογίου
7. Εγγραφή σε κάποιο εργαστήριο, και προβολή διαθέσιμου υλικού.
8. Προβολή εργασιών.
9. Προβολή εισερχομένων μηνυμάτων και αποστολή νέου μηνύματος.
10. Προβολή νέων ανακοινώσεων σε μαθήματα που είναι εγγεγραμμένος ο χρήστης.
11. Προβολή προθεσμιών παράδοσης εργασιών στα μαθήματα του χρήστη.

Βάλαμε δύο χρήστες οι οποίοι περιγράφονται από τα personas (1) και (4) που περιγράφηκαν στο Κεφ.4 . Επέλεξα αυτούς τους δύο γιατί ο (1) δεν είχε ιδιαίτερη εμπειρία με touch συσκευές ενώ η (4) αντίθετα ήταν advanced user σε τέτοιου είδους συσκευές και παράλληλα είναι και οι δύο ήταν εξοικειωμένοι με τις ορολογίες της σχολής. Στην μεθοδολογία αυτή αφήνουμε το χρήστη χωρίς βοήθεια να ακολουθήσει όλα τα βήματα που χρειάζονται για να φτάσει στην ολοκλήρωση μιας ενέργειας. Εμείς καθόμαστε δίπλα του και παρατηρούμε που κάνει λάθη, που μπερδεύεται και πως προσπαθεί να διορθώσει τα λάθη του. Επίσης βασική προϋπόθεση είναι ο χρήστης να μας λέει τι σκέφτεται, και όταν δεν το κάνει, προσπαθούμε να τον κάνουμε εμείς να μιλήσει σχετικά με αυτό. Στο think aloud evaluation που γίνεται σε εφαρμογές γραφείου υπάρχει η δυνατότητα καταγραφής των όσων συμβαίνουν στην οθόνη, κάτι που μας δίνει τη δυνατότητα να ξαναμελετήσουμε τα αντίστοιχα βίντεο και να δούμε πιο προσεκτικά τι λάθη έχουν γίνει. Δυστυχώς δεν είχαμε αυτή η δυνατότητα στο κινητό οπότε έπρεπε να καθυστερούμε λίγο το χρήστη κάθε φορά που θέλαμε να καταγράψουμε κάτι. Το θετικό στην περίπτωση μας είναι ότι γενικά οι χρήστες μας ήταν ομιλητικοί και δεν έκαναν πολλά λάθη, οπότε και το δικό μου έργο γινόταν πιο εύκολο.

Τα αποτελέσματα ήταν ενθαρρυντικά και αποδείχτηκε και στην πράξη ότι όλη η διαδικασία της αξιολόγησης, που είχε γίνει μέχρι τότε, μας γλίτωσε από πολύ κόπο, σε σχέση με το να είχαμε να κάνουμε όλες τις αλλαγές αφού είχε γραφτεί κώδικας. Ωστόσο και πάλι προέκυψαν προβλήματα τα οποία όμως δεν είχαν σχέση με την δομή και την αρχιτεκτονική της πληροφορίας. Η μόνη περίπτωση που υπήρξε πρόβλημα στην δομή και στην αρχιτεκτονική ήταν στην εγγραφή και στην αποδοχή/απόρριψη προσκλήσεων σε χώρο εργασίας. Αυτή την ενέργεια αφού την τροποποίησα στην εφαρμογή ξαναέβαλα τους χρήστες να την εκτελέσουν και η νέα δομή τους βόλεψε, οπότε και οι ίδιοι



συμφώνησαν ότι οι αλλαγές ήταν θετικές. Γενικότερα τα προβλήματα που προέκυψαν περιοριζόταν στο μέγεθος των κουμπιών το οποίο δυσκόλεψε τους χρήστες που δεν είχαν εμπειρία με touch συσκευές. Αυτό έγινε στα κουμπιά φόρτωσης περισσότερων στοιχείων (π.χ. ανακοινώσεων, μηνυμάτων κ.τ.λ.) και στο κουμπί πάνω δεξιά που μας επέστρεφε στην αρχική. Το κουμπί που μας επιστρέφει στην αρχική δεν το αλλάξαμε γιατί θα δημιουργούσε slip errors με αντικείμενα που βρισκόταν στην κορυφή της οθόνης. Οπότε αφού το συζητήσαμε και με τους χρήστες θεωρήσαμε καλύτερο να μείνει ως έχει, καθώς είναι από τα πράγματα που δεν μπερδεύουν γενικά το χρήστη και ότι η δυσκολία ήταν καθαρά θέμα συνήθειας, η οποία ξεπερνιέται εύκολα. Όπως είπαν και οι χρήστες “καλύτερα να προσπαθήσω δύο φορές για να το πετύχω, παρά να πατηθεί αμέσως και να πάω αλλού από εκεί που περιμένω”. Για το κουμπί του “Φόρτωση περισσότερων” είχαμε φτιάξει δική μας κλάση στο CSS αρχείο για να είναι στενότερο από τα άλλα με σκοπό να ξεχωρίζει. Ωστόσο είδαμε ότι η αλλαγή του ύψους του κουμπιού το έκανε πιο δύσκολο για τους χρήστες να το ‘πετύχουν’ με αποτέλεσμα να μπαίνουν συνήθως στο τελευταίο στοιχείο της λίστας και να πρέπει να επιστρέψουν για να ξαναδοκιμάσουν. Οπότε ορίσαμε και σε αυτό το ίδιο ύψος με τα άλλα για να αποφευχθεί αυτό το λάθος. Οι χρήστες μας είπαν ότι εξ’αρχής δεν δημιουργεί πρόβλημα γιατί είναι στο σημείο που περιμένεις να το δεις και δεδομένου ότι το εικονίδιο είναι ξεκάθαρο δεν τίθεται κανένα θέμα δυσκολίας ή σύγχυσης.



## *Κεφάλαιο 7*

*Επανασχεδιασμός της δομής και της γραφικής διεπαφής του  
Μ.Ο.Λ.Ε. για την υποστήριξη απαιτήσεων προσωποποίησης  
πρόσβασης.*

## 7.1 Εισαγωγή.

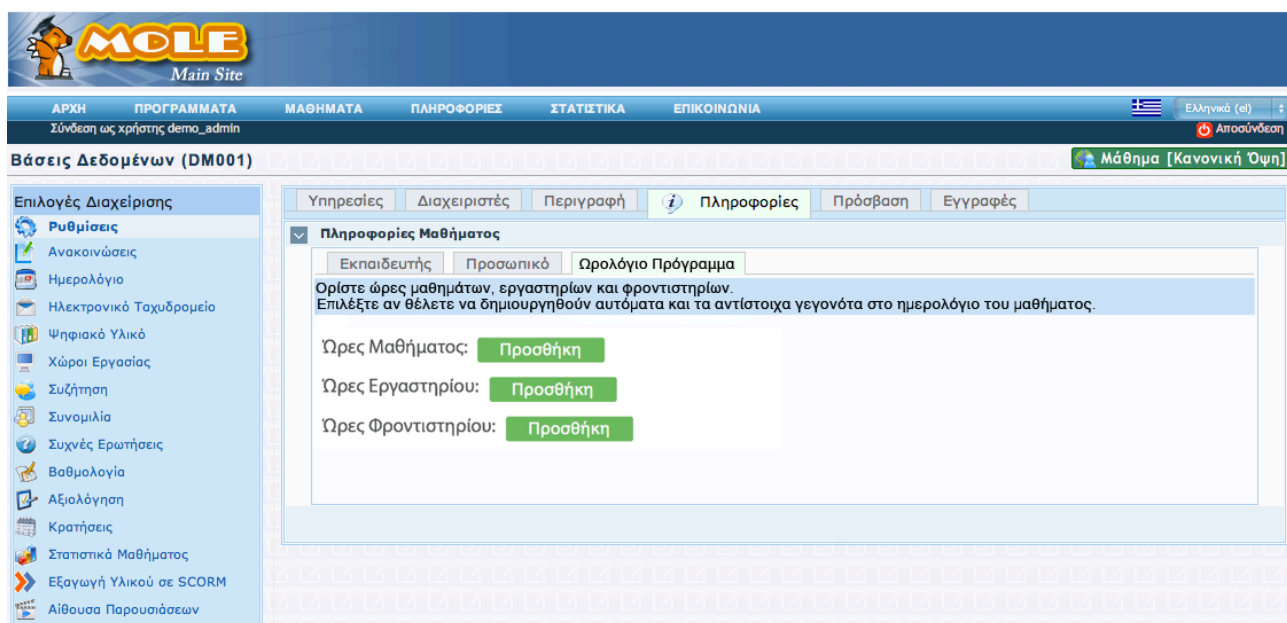
Στο κεφάλαιο αυτό παρουσιάζονται οι αλλαγές που πρέπει να γίνουν στη δομή και στη γραφική διεπαφή του MOLE για να μπορέσουν να προστεθεί η λειτουργικότητα που σχετίζεται με τις προσωποκεντρικές λειτουργίες.

## 7.2 Αλλαγές που απαιτούνται στη δομή του MOLE.

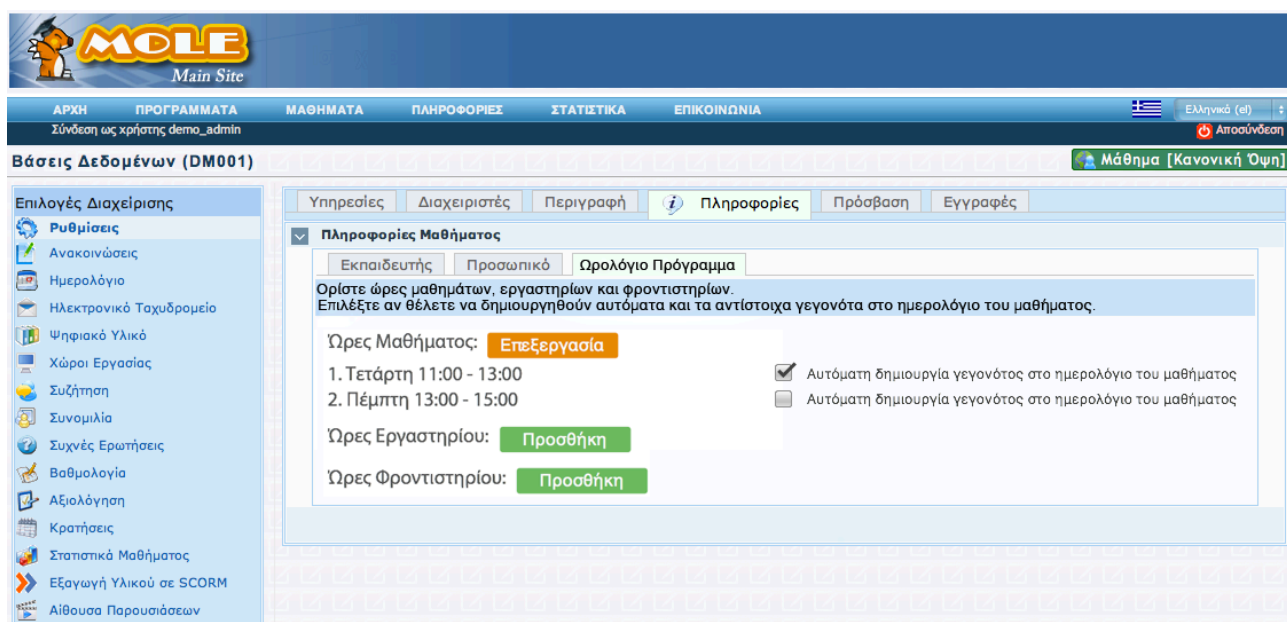
Γενικότερα προσπαθήσαμε να μην χαλάσουμε το μοντέλο του ήδη υπάρχοντος MOLE και να προσαρμόσουμε την εφαρμογή μας πάνω στο μοντέλο αυτό. Αυτό έγινε κατά ένα πολύ μεγάλο μέρος. Ωστόσο για να μπορέσουμε να παρέχουμε τις προσωποκεντρικές λειτουργίες που έχουν προστεθεί στην mobile έκδοση έπρεπε να προσθέσουμε μια επιπλέον δομή η οποία θα μας επέτρεπε να αντιμετωπίζουμε τα επιμέρους τμήματα κάθε μαθήματος -εργαστήρια, διαλέξεις, φροντιστήρια- σαν ξεχωριστά γεγονότα. Προσπαθήσαμε η μετατροπή να γίνει όσο το δυνατόν πιο απλά και με την μικρότερη παρέμβαση στο ήδη υπάρχον σύστημα και εννοιολογικό μοντέλο.

Αυτό που κάναμε ήταν να σπάσουμε τον πίνακα που περιείχε την πληροφορία του μαθήματος (περιγραφή, ώρες μαθήματος, ώρες εργαστηρίου ,ώρες φροντιστηρίου) σε 4 επι μέρους πίνακες. Έτσι έχουμε τον πίνακα με τη περιγραφή του μαθήματος και τρεις πίνακες που συνδέονται με αυτόν, οι οποίοι διατηρούν την πληροφορία για τις ώρες μαθημάτων. Οι πίνακες αυτοί είναι "course\_class\_hours", "course\_lab\_hours", "course\_institution\_hours". Με τον τρόπο αυτό μπορούμε άμεσα μόλις κάποιος καθηγητής δημιουργήσει ένα μάθημα, να δημιουργούνται αυτόματα και στον πίνακα με τα γεγονότα του ημερολογίου όλες οι διαλέξεις, τα εργαστήρια και τα φροντιστήρια σαν ανεξάρτητα γεγονότα τα οποία μπορούν να τροποποιηθούν από τους διαχειριστές (καθηγητές, βοηθούς κ.τ.λ.).

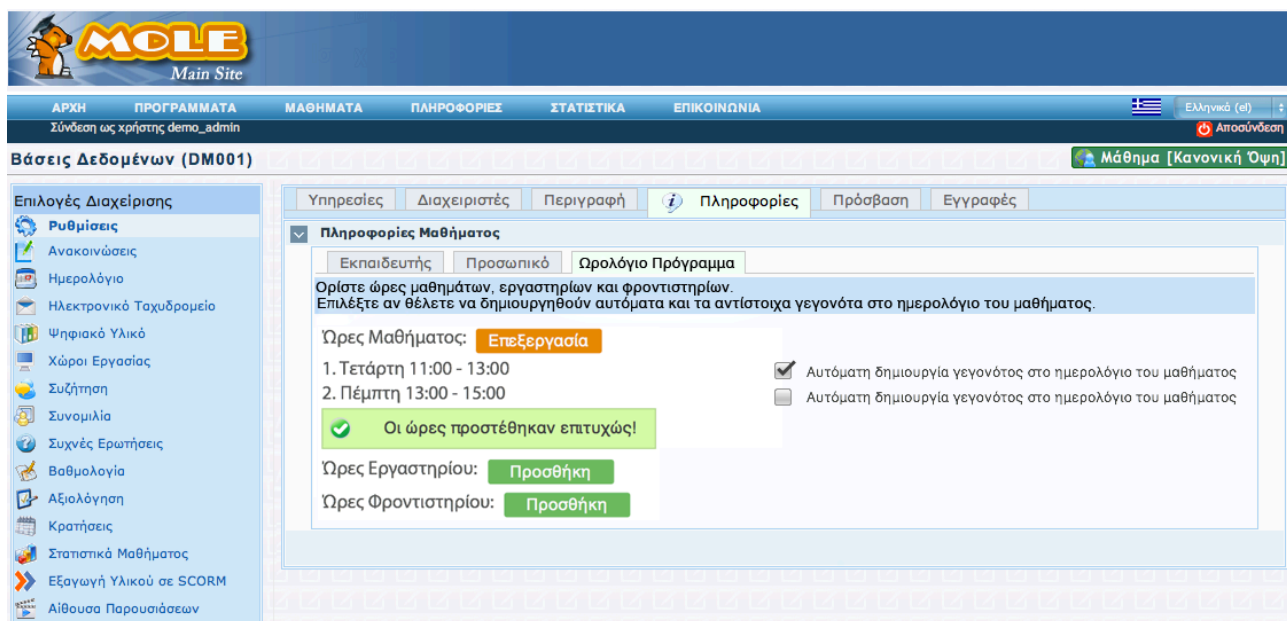
Επίσης μας ζητήθηκε αφού έγιναν αυτές οι αλλαγές, να δημιουργήσουμε και σε mock-ups τις διεπαφές εκείνες που θα έπρεπε να εισαχθούν στο MOLE για να μπορεί να υποστηριχτεί αυτή η λειτουργία. Ακολουθούν τα prototypes των διεπαφών αυτών :



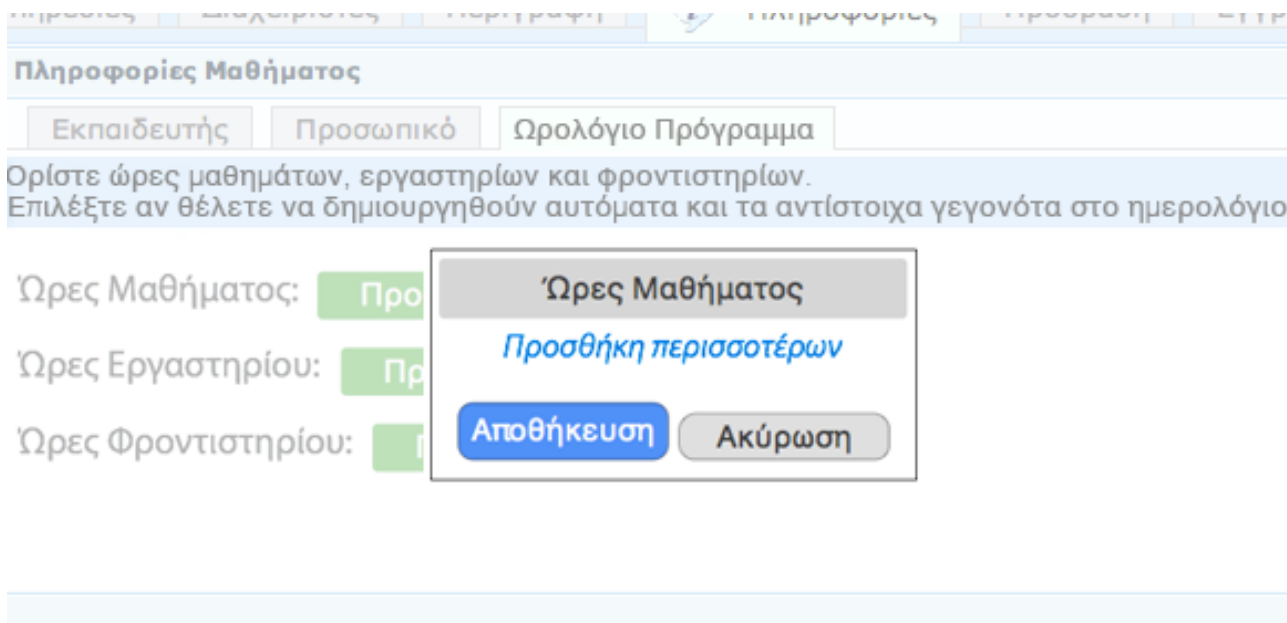
Εικόνα 8.3 - Σελίδα δημιουργίας μαθήματος. Στο μάθημα δεν έχει οριστεί ακόμα καμία ώρα.



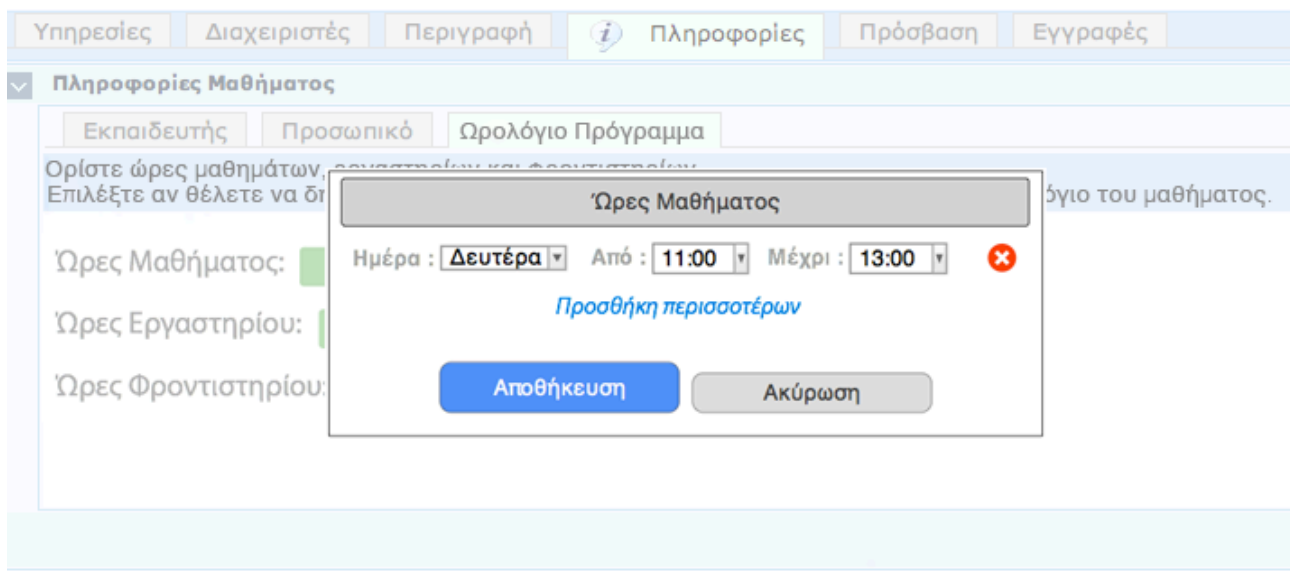
Εικόνα 8.4 - Σελίδα δημιουργίας μαθήματος. Στο μάθημα έχουν οριστεί ώρες μαθήματος μόνο.



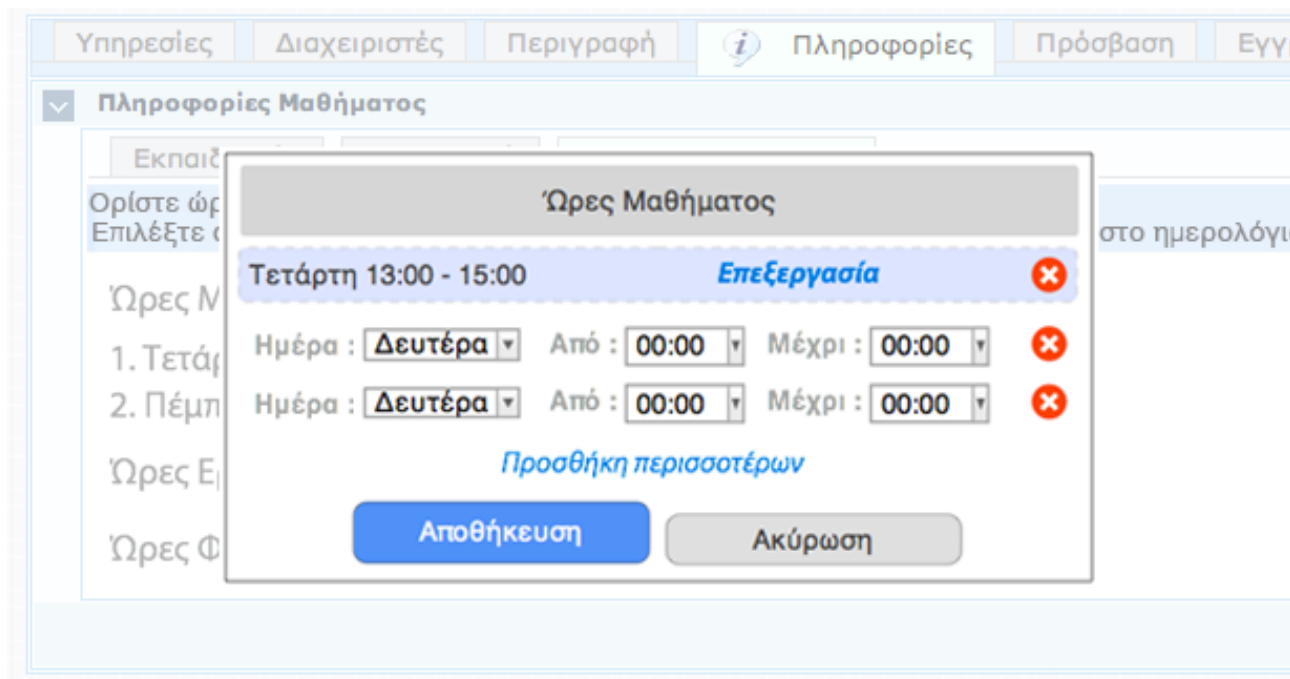
Εικόνα 8.5 - Σελίδα δημιουργίας μαθήματος. Αφού επιστρέφουμε από προσθήκη ή επεξεργασία των ωρών του μαθήματος.



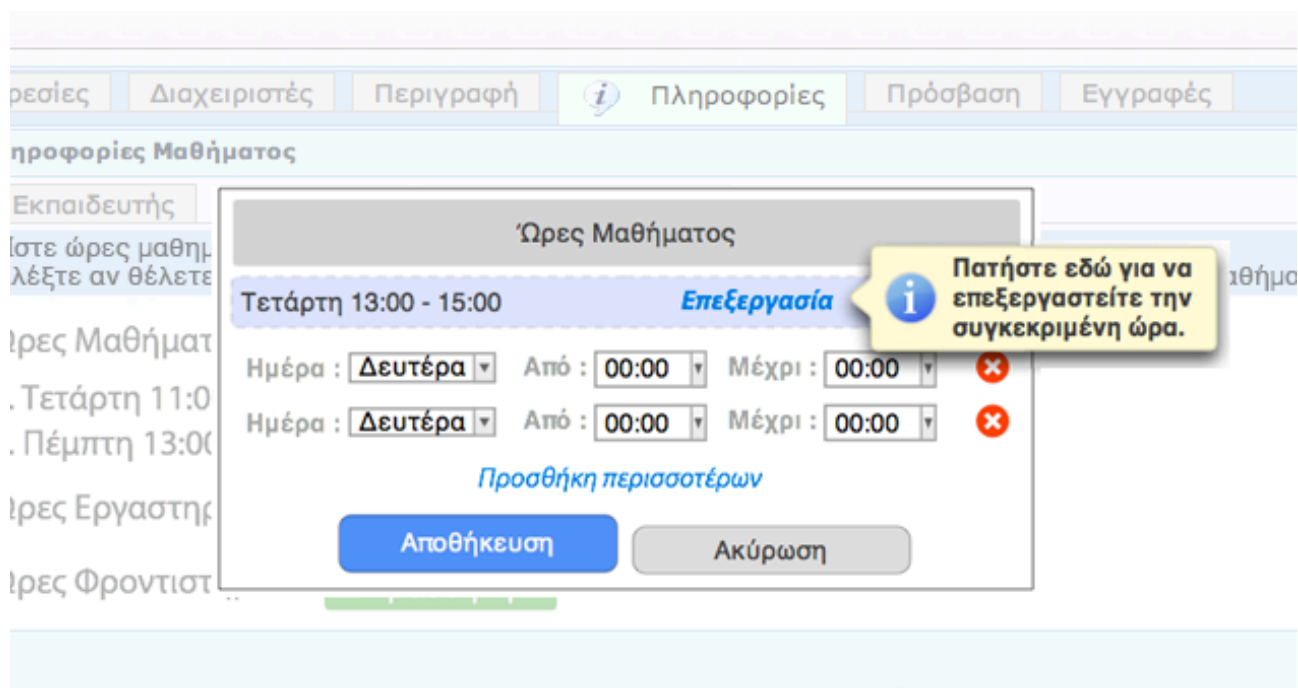
Εικόνα 8.6 - Pop-up παράθυρο που εμφανίζεται όταν πάμε να προσθέσουμε κάποια ώρα (Πατώντας στο "Προσθήκη") και δεν υπάρχει ήδη καμία ώρα μαθήματος στο συγκεκριμένο μάθημα. Ο χρήστης μπορεί να προσθέσει περισσότερες από μία ώρες.



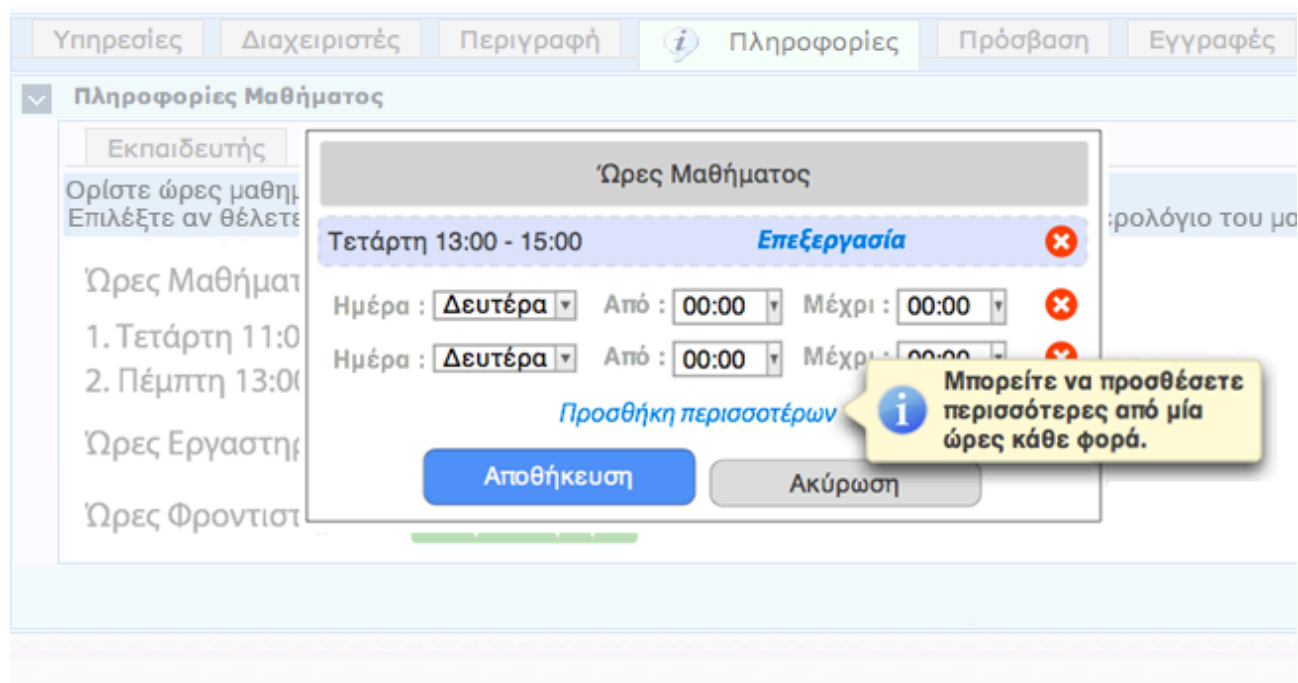
Εικόνα 8.7 - Το ίδιο με το προηγούμενο pop-up παράθυρο αφού ο χρήστης έχει προσθέσει κάποια ώρα και δεν το έχει αποθηκεύσει ακόμα. Ο χρήστης μπορεί να προσθέσει περισσότερες από μία ώρες.



Εικόνα 8.8 - Pop-up παράθυρο που εμφανίζεται όταν πάμε να προσθέσουμε κάποια ώρα (Πατώντας στο "Επεξεργασία") και υπάρχει ήδη ώρα μαθήματος στο συγκεκριμένο μάθημα. Ο χρήστης μπορεί να προσθέσει περισσότερες από μία ώρες.



Εικόνα 8.9 - Το ίδιο με την εικόνα 8.8 με hover του ποντικιού στο "Επεξεργασία". Εμφάνιση περιγραφικού μηνύματος.



Εικόνα 8.10 - Το ίδιο με την εικόνα 8.8 με hover του ποντικιού στο "Προσθήκη Περισσοτέρων". Εμφάνιση επεξηγηματικού μηνύματος.





The screenshot shows the MOLE Main Site interface. The top navigation bar includes links for Αρχή, Προγράμματα, Μαθήματα, Πληροφορίες, Στατιστικά, and Επικοινωνία. The user is logged in as demo\_admin. The main content area is titled "Βάσεις Δεδομένων (DM001)" and features a "Ημερολόγιο" (Calendar) section. The calendar displays a list of events with columns for Ημερομηνία (Date), Διάρκεια (Duration), Τύπος - Θέμα (Type - Topic), Περιγραφή (Description), Αλλαγή (Change), and Διαγραφή (Delete). The events listed are:

Ημερομηνία	Διάρκεια	Τύπος - Θέμα	Περιγραφή	Αλλαγή	Διαγραφή
20 Αυγούστου 2010, ώρα: 09:00	03:00:00	Προβολή Ταινίας	Documentary: The true story of Internet		
20 Αυγούστου 2010, ώρα: 09:00	02:00:00	Φροντιστήριο - 1ο Φροντιστήριο	Οντότητες, Συσχετίσεις, Χαρακτηριστικά και τύποι τους, περιορισμοί πληθικότητας, κλειδιά, σχέση ανάμεσα σε οντότητες-συσχετίσεων, σχέσεις γενίκευσης-εξειδίκευσης		
16 Αυγούστου 2010, ώρα: 09:00	03:00:00	Διάλεξη - 3η Διάλεξη	Μοντέλο Οντοτήτων-Συσχετίσεων, UML Διάγραμμα Οντοτήτων - Κλάσεων		
09 Αυγούστου 2010, ώρα: 09:00	03:00:00	Διάλεξη - 2η Διάλεξη	Λογικά Μοντέλα Δεδομένων		
02 Αυγούστου 2010, ώρα: 09:00	03:00:00	Διάλεξη - 1η Διάλεξη	Εισαγωγή, Διαχείριση Δεδομένων και Συστήματα Διαχείρισης Βάσεων Δεδομένων		

On the left side, there is a sidebar with "Επιλογές Διαχείρισης" (Management Options) including links for Ρυθμίσεις, Ανακοινώσεις, Ημερολόγιο, Ηλεκτρονικό Ταχυδρομείο, Ψηφιακό Υλικό, Χώροι Εργασίας, Συζήτηση, Συνομιλία, Συχνές Ερωτήσεις, Βαθμολογία, Αξιολόγηση, Κρατήσεις, Στατιστικά Μαθήματος, Εξαγωγή Υλικού σε SCORM, and Αίθουσα Παρουσιάσεων.

Εικόνα 8.12 - Σελίδα Ημερολογίου με δυνατότητες επεξεργασίας γεγονότων και προσθήκης νέου καθώς και προβολής των υποχρεώσεων των φοιτητών.

The screenshot shows the MOLE Main Site interface with the "Νέο Γεγονός" (New Event) form open. The form includes fields for Θέμα (Topic), Περίοδος (Period), and Περιγραφή (Description). The "Περίοδος" section has dropdowns for "Ώρα" (Time) and "Μέχρι" (Until), and a date selector for "Ημερομηνία" (Date) with fields for "Ημέρα" (Day), "Μήνας" (Month), and "Έτος" (Year). The "Περιγραφή" section has a rich text editor with various formatting options. Below the form, there is a "Είδος Γεγονότος" (Event Type) dropdown set to "Διάλεξη" (Lecture) and a checkbox for "Εισαγωγή συνδέσμου σε ψηφιακό υλικό" (Link introduction to digital material). At the bottom, there are buttons for "Δημιουργία" (Create) and "Ακύρωση" (Cancel).

Εικόνα 8.13 - Σελίδα για δημιουργία νέου γεγονότος πατώντας στο "Προσθήκη Νέου Γεγονότος" στην σελίδα του ημερολογίου ( Εικόνα 8.12).

**MOLE Main Site**

ΑΡΧΗ ΠΡΟΓΡΑΜΜΑΤΑ ΜΑΘΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΕΣ ΣΤΑΤΙΣΤΙΚΑ ΕΠΙΚΟΙΝΩΝΙΑ

Σύνδεση ως χρήστης demo\_admin

Ελληνικά (el) Αποσύνδεση

Βάσεις Δεδομένων (DM001) **Ημερολόγιο [Κανονική Όψη]**

**Επιλογές Διαχείρισης**

- Ρυθμίσεις
- Ανακοινώσεις
- Ημερολόγιο**
- Ηλεκτρονικό Ταχυδρομείο
- Ψηφιακό Υλικό
- Χώροι Εργασίας
- Συζήτηση
- Συνομιλία
- Συχνές Ερωτήσεις
- Βαθμολογία
- Αξιολόγηση
- Κρατήσεις
- Στατιστικά Μαθήματος
- Εξαγωγή Υλικού σε SCORM
- Αίθουσα Παρουσιάσεων

**Ημερολόγιο** **Ενημέρωση** **Προβολή Υποχρεώσεων Φοιτητών**

**Ενημέρωση Γεγονότος**

Θέμα: 1ο Φροντιστήριο

Περίοδος: Ώρα: Από: 09:00 Μέχρι: 11:00

Ημερομηνία: Ημέρα: 20 Μήνας: 08 Έτος: 2011

Περιγραφή: **Οντότητες, Συσχετίσεις, Χαρακτηριστικά και τύποι τους, περιορισμοί πληθικότητας, κλειδιά, αδύναμα σύνολα οντοτήτων-συσχετίσεων, σχέσεις γενίκευσης-εξειδίκευσης**

Είσοδων συνδέσμου σε ψηφιακό υλικό

Είδος Γεγονότος: Φροντιστήριο

Σημαντική Υποχρέωση Φοιτητών

**Ενημέρωση** **Ακύρωση**

Εικόνα 8.14 - Σελίδα για την επεξεργασία γεγονότος πατώντας στο “αλλαγή” στο αντίστοιχο συμβάν στην σελίδα του ημερολογίου (Εικόνα 8.12).

**MOLE Main Site**

ΑΡΧΗ ΠΡΟΓΡΑΜΜΑΤΑ ΜΑΘΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΕΣ ΣΤΑΤΙΣΤΙΚΑ ΕΠΙΚΟΙΝΩΝΙΑ

Σύνδεση ως χρήστης demo\_admin

Ελληνικά (el) Αποσύνδεση

Βάσεις Δεδομένων (DM001) **Ημερολόγιο [Κανονική Όψη]**

**Επιλογές Διαχείρισης**

- Ρυθμίσεις
- Ανακοινώσεις
- Ημερολόγιο**
- Ηλεκτρονικό Ταχυδρομείο
- Ψηφιακό Υλικό
- Χώροι Εργασίας
- Συζήτηση
- Συνομιλία
- Συχνές Ερωτήσεις
- Βαθμολογία
- Αξιολόγηση
- Κρατήσεις
- Στατιστικά Μαθήματος
- Εξαγωγή Υλικού σε SCORM
- Αίθουσα Παρουσιάσεων

**Ημερολόγιο** **Ενημέρωση** **Προβολή Υποχρεώσεων Φοιτητών**

**Ενημέρωση Γεγονότος**

Θέμα: 1ο Φροντιστήριο

Περίοδος: Ώρα: Από: 09:00 Μέχρι: 11:00

Ημερομηνία: Ημέρα: 20 Μήνας: 08 Έτος: 2011

Περιγραφή: **Οντότητες, Συσχετίσεις, Χαρακτηριστικά και τύποι τους, περιορισμοί πληθικότητας, κλειδιά, αδύναμα σύνολα οντοτήτων-συσχετίσεων, σχέσεις γενίκευσης-εξειδίκευσης**

Είσοδων συνδέσμου σε ψηφιακό υλικό

Είδος Γεγονότος: Φροντιστήριο

Διάλεξη

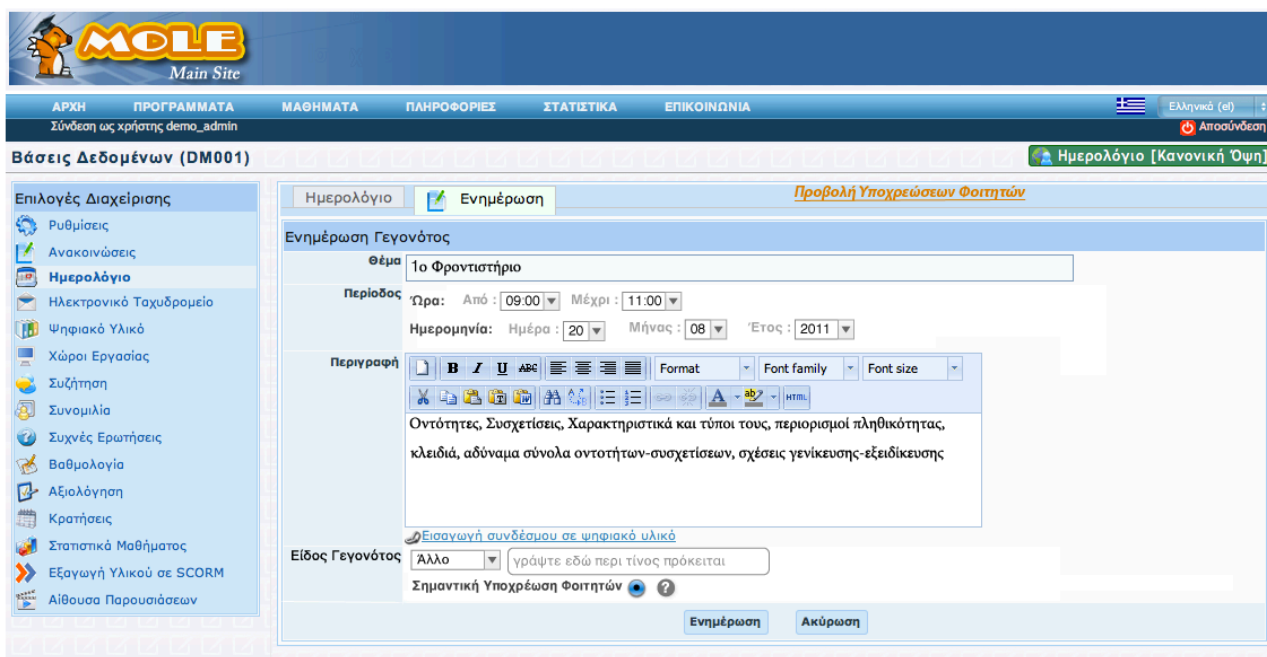
Εργαστήριο

Φροντιστήριο

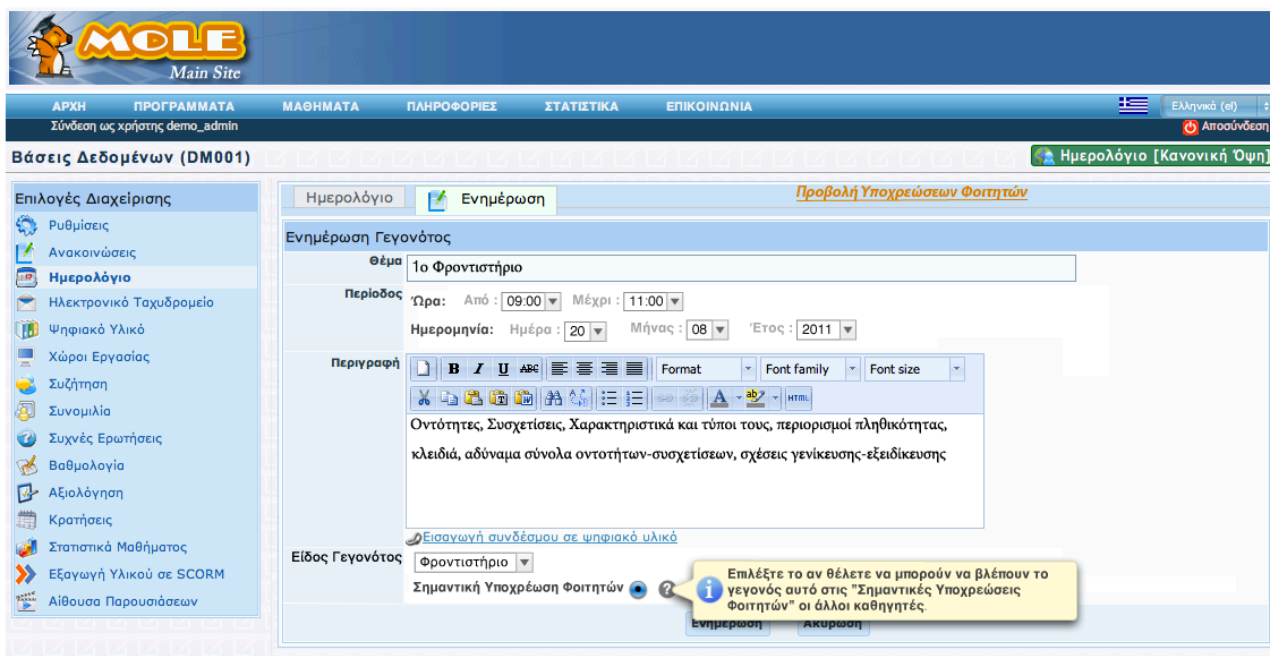
Άλλο

**Ενημέρωση** **Ακύρωση**

Εικόνα 8.15 - Pop-up παράθυρο για δημιουργία-επεξεργασία νέου γεγονότος. Φαίνονται οι επιλογές που έχει ο χρήστης για τύπους γεγονότος.



Εικόνα 8.16 - Pop-up παράθυρο για δημιουργία-επεξεργασία νέου γεγονότος. Φαίνεται η τροποποίηση που γίνεται για να μπορεί να εισάγει ο χρήστης έναν τύπο εκτός των δεδομένων που έχουμε εμείς ορίσει.



Εικόνα 8.17 - Pop-up παράθυρο για δημιουργία-επεξεργασία νέου γεγονότος. Ενημερωτικό μήνυμα για το τι σημαίνει "Σημαντική Υποχρέωση Φοιτητών" και το τι κάνει η επιλογή της.

Έτη Σπουδών

Μαθήματα Εξαμήνου

Επιλέξτε Έτος Σπουδών

3ο Έτος

Επιλέξτε τύπο γεγονότων

Παραδόσεις Εργασιών

3ο έτος σπουδών

22 Μαΐου 2011 - Παράδοση Εργασίας  
Τηλεπικοινωνιακά Συστήματα Ι  
Άσκηση 1  
Καθηγητής : Αθανάσιος Λιάβας

27 Μαΐου 2011 - Παράδοση Εργασίας  
Θεωρία Υπολογισμού  
Εργασία Προγραμματισμού  
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης

16 Μαΐου 2011 - Παράδοση Εργασίας  
Θεωρία Υπολογισμού  
2η Σειρά Γραπτών Ασκήσεων  
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης

Κλείσιμο

Εικόνα 8.18 - Pop-up παράθυρο για την προβολή των υποχρεώσεων των φοιτητών που τρέχουν την παρούσα ημερομηνία. Ο καθηγητής έχει επιλέξει να δει τις παραδόσεις εργασιών για τα μαθήματα του τρίτου έτους σπουδών.

Έτη Σπουδών

Μαθήματα Εξαμήνου

Επιλέξτε Έτος Σπουδών

3ο Έτος

Επιλέξτε τύπο γεγονότων

Παραδόσεις Εργασιών

Παραδόσεις Εργασιών

Εργαστήρια

Διαλέξεις

Φροντιστήρια

Άλλα γεγονότα

Όλα τα γεγονότα

3ο έτος σπουδών	
22 Μαΐου 2011 - Παράδοση Εργασίας	
Τηλεπικοινωνιακά Συστήματα Ι	
Άσκηση 1	
Καθηγητής : Αθανάσιος Λιάβας	
27 Μαΐου 2011 - Παράδοση Εργασίας	
Θεωρία Υπολογισμού	
Εργασία Προγραμματισμού	
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης	
16 Μαΐου 2011 - Παράδοση Εργασίας	
Θεωρία Υπολογισμού	
2η Σειρά Γραπτών Ασκήσεων	
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης	

Κλείσιμο

Εικόνα 8.19 - Pop-up παράθυρο για την προβολή των υποχρεώσεων των φοιτητών που τρέχουν την παρούσα ημερομηνία. Έχει επιλεγεί το τρίτο έτος σπουδών και εδώ φαίνονται οι διάφοροι τύποι γεγονότων βάση των οποίων μπορεί να φιλτράρει ο καθηγητής τις υποχρεώσεις των φοιτητών.

Έτη Σπουδών

Μαθήματα Εξαμήνου

Επιλέξτε Έτος Σπουδών

3ο Έτος

Επιλέξτε τύπο γεγονότων

Παραδόσεις Εργασιών

3ο έτος σπουδών

22 Μαΐου 2011 - Παράδοση Εργασίας	
Τηλεπικοινωνιακά Συστήματα Ι	
Άσκηση 1	
Καθηγητής : Αθανάσιος Λιάβας	
27 Μαΐου 2011 - Παράδοση Εργασίας	
Θεωρία Υπολογισμού	
Εργασία Προγραμματισμού	
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης	
16 Μαΐου 2011 - Παράδοση Εργασίας	
Θεωρία Υπολογισμού	
2η Σειρά Γραπτών Ασκήσεων	
Καθηγητής : Μιχαήλ Γ. Λαγουδάκης	

Κλείσιμο

Αποστολή μηνύματος στον καθηγητή

σχετίσεις, Χαρακτηριστικά  
ηθικότητας, κλειδιά, αδύνα  
σχετίσεων, σχέσεις γενίκευ  
τήτων-Συσχετίσεων, UML Δ  
λόμενων

α Δεδομένων

χείριση Δεδομένων και Συσ  
σεων Δεδομένων

Εικόνα 8.20 - Pop-up παράθυρο για την προβολή των υποχρεώσεων των φοιτητών που τρέχουν την παρούσα ημερομηνία. Εδώ φαίνεται το ενημερωτικό μήνυμα όταν ο καθηγητής κάνει hover πάνω από το link για αποστολή μηνύματος σε κάποιον από τους καθηγητές που έχουν ορίσει υποχρεώσεις φοιτητών.

Έτη Σπουδών

Μαθήματα Εξαμήνου

Επιλέξτε Μάθημα

Τηλεπικοινωνιακά Συστή...

Επιλέξτε τύπο γεγονότων

Όλα

Τηλεπικοινωνιακά Συστήματα Ι

Καθηγητής : Αθανάσιος Λιάβας

11 Μαΐου 2011 - Διάλεξη

Τίτλος Εργασίας

14 Μαΐου 2011 - Φροντιστήριο

Τίτλος Εργασίας

17 Μαΐου 2011 - Διάλεξη

Τίτλος Εργασίας

20 Μαΐου 2011- Φροντιστήριο

Τίτλος Εργασίας

22 Μαΐου - Παράδοση Εργασίας

Άσκηση 1

Κλείσιμο

Εικόνα 8.21 - Pop-up παράθυρο για την προβολή των υποχρεώσεων των φοιτητών που τρέχουν την παρούσα ημερομηνία. Ο καθηγητής έχει επιλέξει να δει τις υποχρεώσεις των φοιτητών για κάποιο συγκεκριμένο μάθημα του εξαμήνου που τρέχει και στο οποίο προφανώς ανήκει και το μάθημα του.







## Κεφάλαιο 8

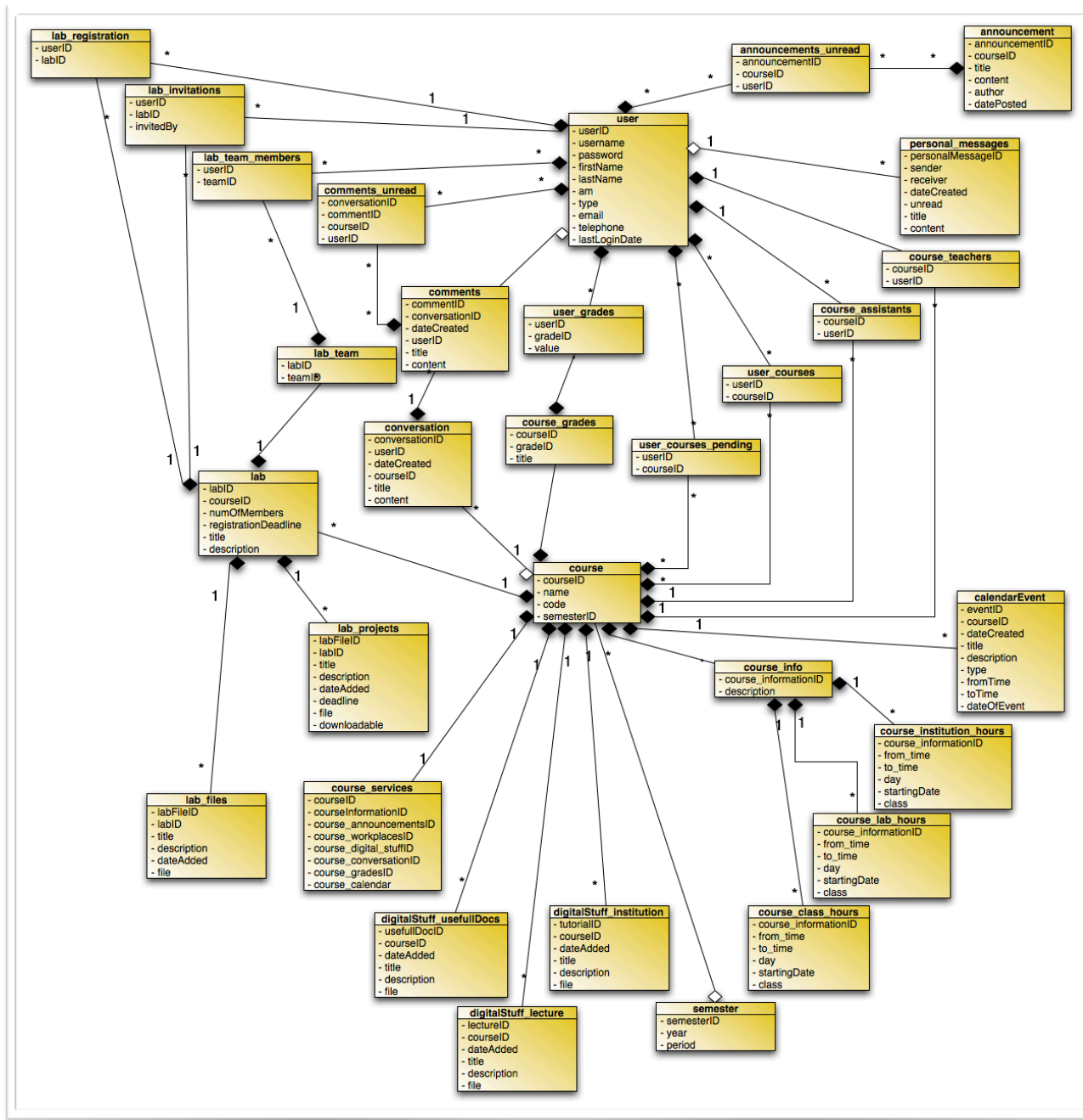
*Αναλυτική περιγραφή Βάσης Δεδομένων, Data Access Objects και Entity Model για την υποστήριξη της λειτουργικότητας των προσωποκεντρικών εφαρμογών στο M.O.L.E.*

## 8.1 Εισαγωγή

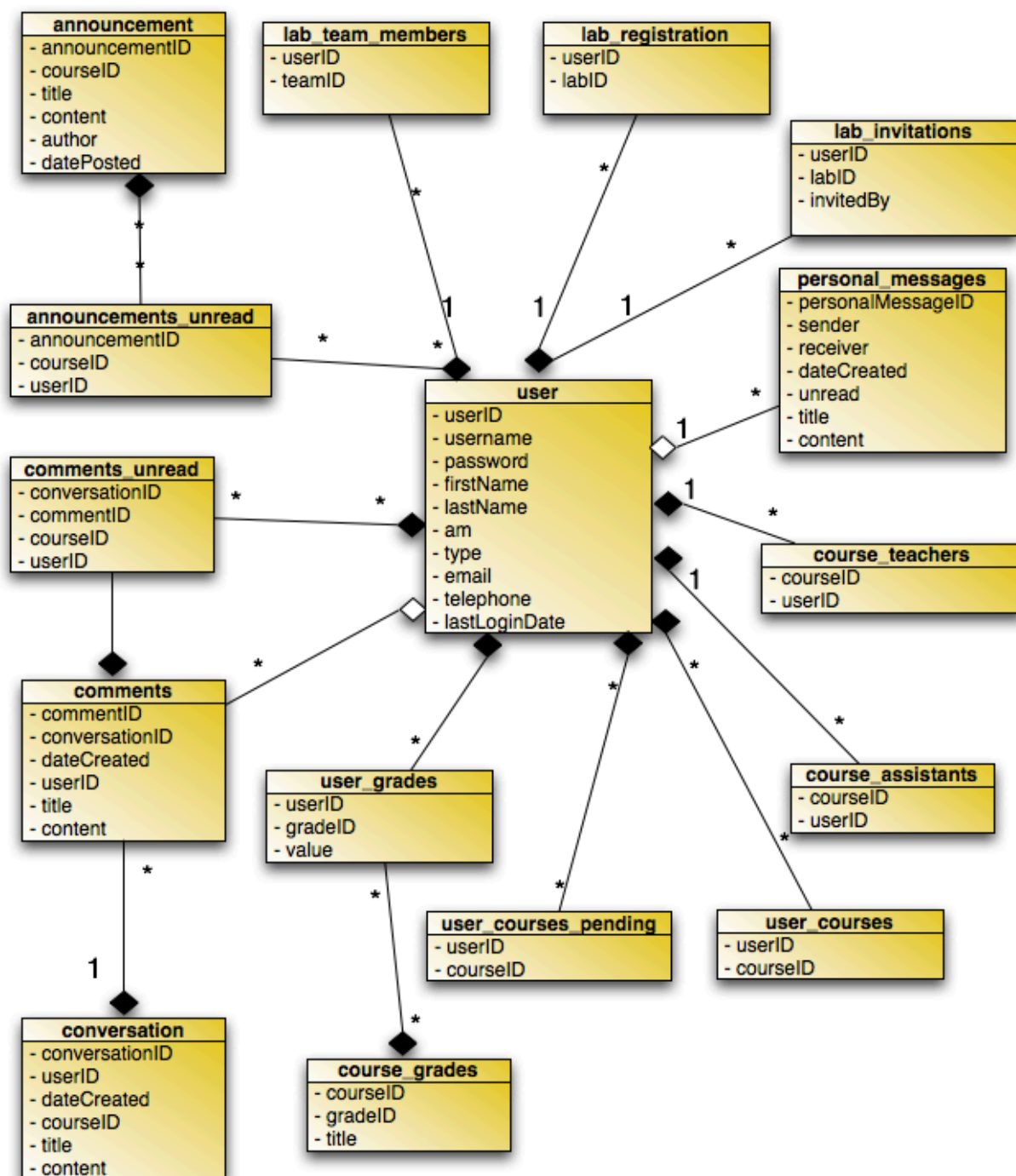
Στο κεφάλαιο αυτό αναλύεται η βάση δεδομένων που δημιουργήσαμε, τα Data Access Objects, το Μοντέλο Οντοτήτων και παρουσιάζεται και μια διαγραμματική αναπαράσταση του συστήματος μας. Φαίνονται τα διαγράμματα UML της βάσης και των κλάσεων οντοτήτων. Περιγράφονται αναλυτικά όλα τα DAO καθώς και όλοι οι πίνακες της εφαρμογής μας.

## 8.2 Ανάλυση της Βάσης Δεδομένων του συστήματος μας.

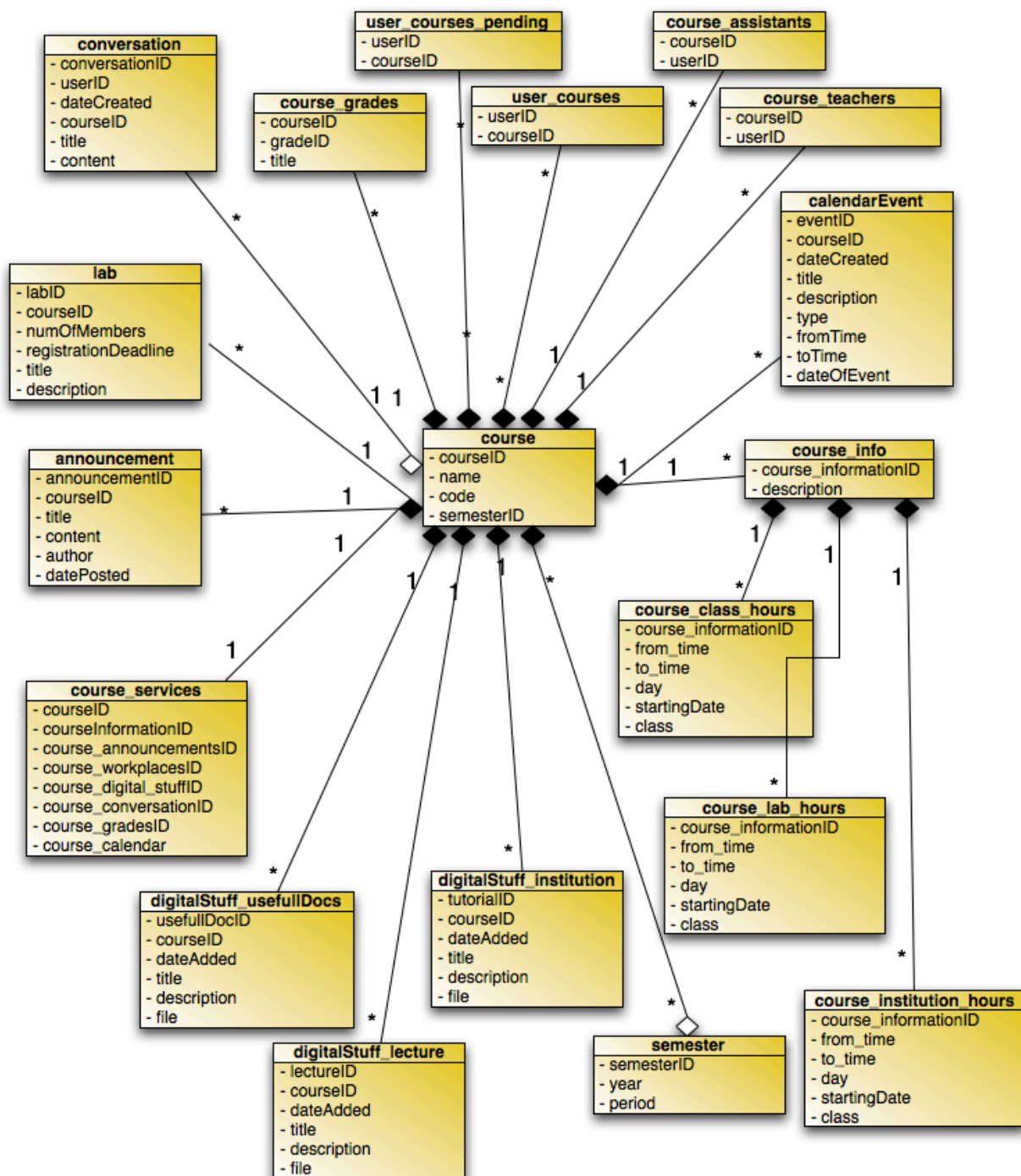
Στην ενότητα αυτή φαίνονται τα διαγράμματα της βάσης δεδομένων που χρησιμοποιήσαμε σε UML. Αρχικά φαίνεται η πλήρης βάση (εικόνα 8.1) για να γίνει κατανοητό το πλήρες μοντέλο και στη συνέχεια (εικόνες 8.2 έως 8.4) την σπάμε σε μικρότερα τμήματα, με βάση τις έννοιες της εφαρμογής, για να μπορεί να είναι ευανάγνωστη και εύκολα κατανοητή. Στη συνέχεια ακολουθεί το UML diagram με τις κλάσεις οντοτήτων της εφαρμογής (εικόνα 8.5).



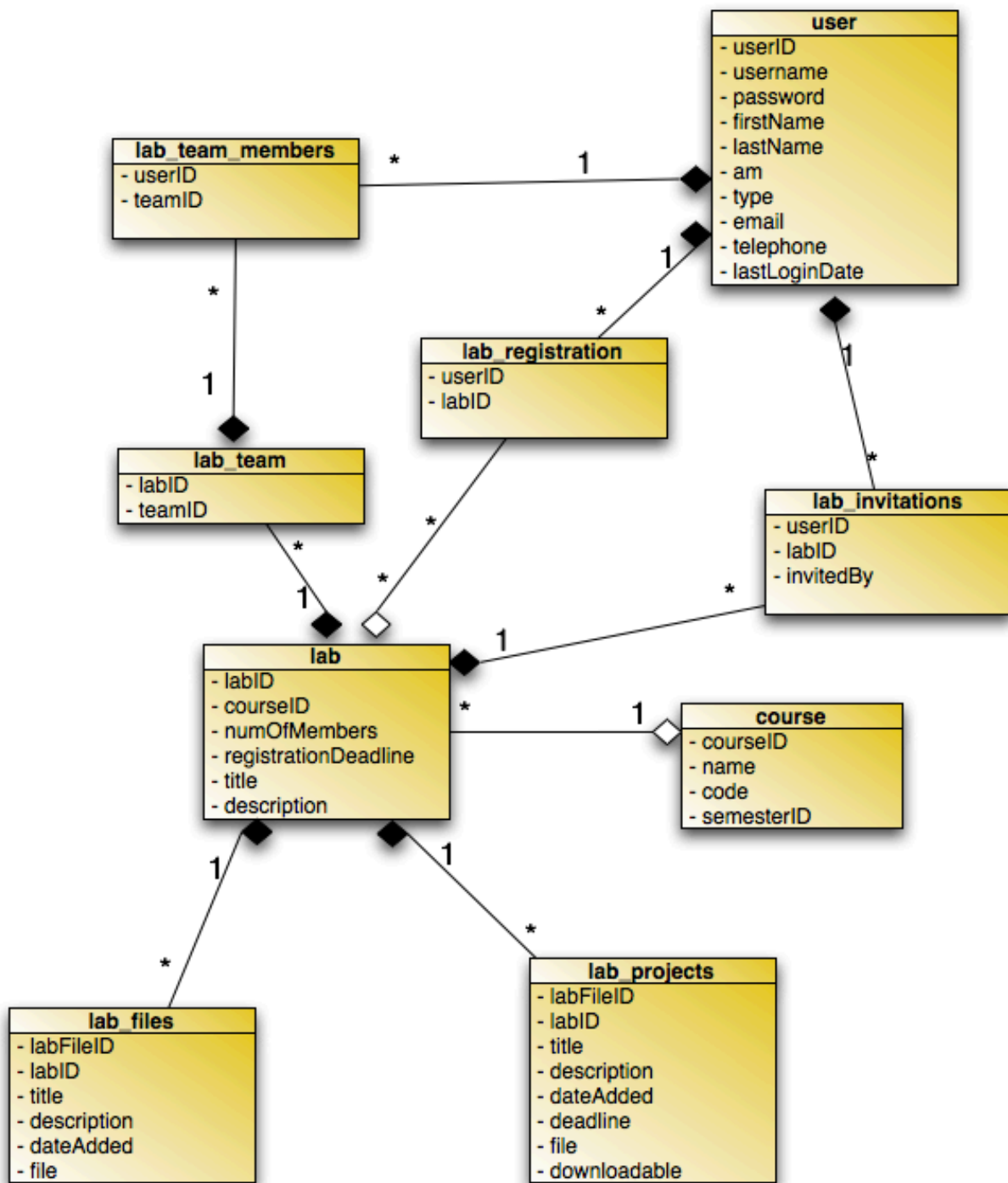
Εικόνα 8.1 - UML diagram της πλήρης Βάσης Δεδομένων της εφαρμογής.



Εικόνα 8.2 - UML διάγραμμα των πινάκων, του συστήματος μας, που σχετίζονται με την έννοια του χρήστη.



Εικόνα 8.3 - UML διάγραμμα των πινάκων, του συστήματός μας, που σχετίζονται με την έννοια του μαθήματος.



Εικόνα 8.4-UML διάγραμμα των πινάκων, του συστήματος μας, που σχετίζονται με την έννοια του εργαστηρίου.



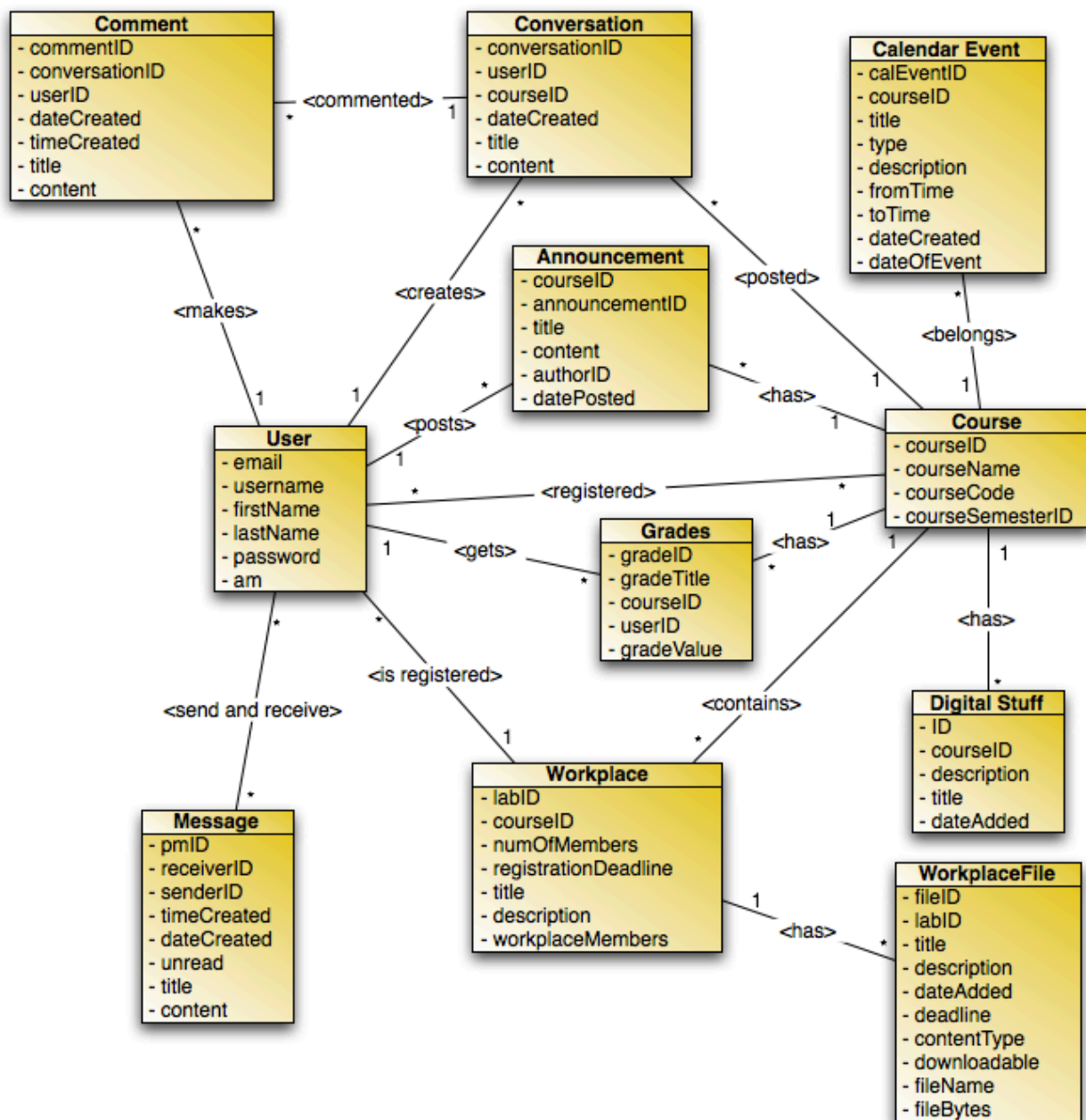
### 8.3 Αναλυτική περιγραφή των κλάσεων της βάσης δεδομένων.

1. Πίνακας announcement: περιέχει την πληροφορία για όλες τις ανακοινώσεις των μαθημάτων.
2. Πίνακας announcements\_unread: περιέχει την πληροφορία για τις ανακοινώσεις που δεν έχουν διαβαστεί από συγκεκριμένους χρήστες.
3. Πίνακας calendarEvent: περιέχει όλα τα γεγονότα του ημερολογίου των μαθημάτων. Εδώ αποθηκεύονται και τα γεγονότα που δημιουργούνται αυτόματα από τη δημιουργία μαθημάτων (φροντιστήρια, διαλέξεις, εργαστήρια) και άλλα γεγονότα που δημιουργούν οι καθηγητές.
4. Πίνακας comments: περιέχει όλα τα σχόλια των συζητήσεων.
5. Πίνακας comments\_unread: περιέχει συνδυασμούς χρηστών-σχολίων για να μπορούμε να βλέπουμε ποια σχόλια δεν έχουν διαβαστεί από κάποιον συγκεκριμένο χρήστη.
6. Πίνακας conversation: περιέχει όλες τις συζητήσεις γενικές και μαθημάτων
7. Πίνακας course: περιέχει όλα τα μαθήματα που υποστηρίζονται από τον ιστοχώρο μας.
8. Πίνακας course\_assistants: περιέχει συνδυασμούς χρηστών-μαθήματος για να έχουμε την πληροφορία για τους βοηθούς μαθημάτων.
9. Πίνακας course\_class\_hours: περιέχει τις ώρες διαλέξεων για κάθε μάθημα.
10. Πίνακας course\_grades: περιέχει τους τίτλους των βαθμών για τα μαθήματα.
11. Πίνακας course\_info: περιέχει το κείμενο που περιγράφει το κάθε μάθημα.
12. Πίνακας course\_institution\_hours: περιέχει τις ώρες φροντιστηρίου για κάθε μάθημα.

13. Πίνακας `course_lab_hours`: περιέχει τις ώρες εργαστηρίου για κάθε μάθημα.
14. Πίνακας `course_services`: περιέχει τις υπηρεσίες που υποστηρίζονται από κάθε μάθημα.
15. Πίνακας `course_teachers`: περιέχει τους καθηγητές κάθε μαθήματος.
16. Πίνακας `digitalStuff_institution`: περιέχει τα αρχεία του φροντιστηρίου των μαθημάτων.
17. Πίνακας `digitalStuff_lecture`: περιέχει τα αρχεία των διαλέξεων των μαθημάτων.
18. Πίνακας `digitalStuff_usefullDocs`: περιέχει τα αρχεία των χρήσιμων εγγράφων των μαθημάτων.
19. Πίνακας `lab`: περιέχει τα εργαστήρια των μαθημάτων.
20. Πίνακας `lab_files`: περιέχει το εργαστηριακό υλικό των μαθημάτων.
21. Πίνακας `lab_invitations`: περιέχει τις προσκλήσεις για τις εργαστηριακές ομάδες από εγγεγραμμένους χρήστες σε νέους.
22. Πίνακας `lab_projects`: περιέχει την πληροφορία για τις εργαστηριακές ασκήσεις.
23. Πίνακας `lab_registration`: περιέχει τους χρήστες και τα εργαστήρια στα οποία είναι εγγεγραμμένοι.
24. Πίνακας `lab_team`: περιέχει τις εργαστηριακές ομάδες.
25. Πίνακας `lab_team_members`: περιέχει τα μέλη που ανήκουν σε κάθε εργαστηριακή ομάδα.
26. Πίνακας `personal_messages`: περιέχει όλα τα προσωπικά μηνύματα.
27. Πίνακας `semester`: περιέχει τα εξάμηνα.
28. Πίνακας `user`: περιέχει τους χρήστες του συστήματος
29. Πίνακας `user_courses`: περιέχει συνδυασμό χρηστών - μαθημάτων για να ξέρουμε που είναι εγγεγραμμένος κάθε φοιτητής.

30. Πίνακας user\_courses\_pending: περιέχει τις αιτήσεις εγγραφής φοιτητών, σε μαθήματα, που εκκρεμούν ακόμα.
31. Πίνακας user\_grades: περιέχει τους βαθμούς κάθε φοιτητή.

Ακολουθεί το UML diagram των entity classes:



Εικόνα 8.5 - Το UML diagram των entity classes της εφαρμογής μας.

## 9.4 Αναλυτική περιγραφή των DAO

Θέλοντας να παρέχουμε τη δυνατότητα εύκολης σύνδεσης της εφαρμογής με τη ήδη υπάρχουσα βάση του MOLE δημιουργήσαμε την abstract class DAOFactory και τις αντίστοιχες abstract classes για τα απαραίτητα DAO objects, έτσι ώστε να υπάρχει η δυνατότητα να μπορεί εύκολα να υλοποιηθεί το αντίστοιχο μοντέλο για διαφορετικές βάσεις δεδομένων - καθώς και για διαφορετικού τύπου βάσεις δεδομένων και όχι μόνο για mysql που έχουμε χρησιμοποιήσει εμείς. Ακολουθεί η περιγραφή της κλάσης και η αναλυτική περιγραφή των αντίστοιχων Data Access Objects καθώς και η περιγραφή των μεθόδων που έχουμε υλοποιήσει στις αντίστοιχες κλάσεις για την χρήση της MySQL. Οι abstract κλάσεις ακολουθούν την ονοματολογία *NameOfClassDAO* και οι υλοποιήσεις τους για MySQL την *MySQLNameOfClassDAO*.

### DAOFactory

```
public abstract class DAOFactory
```

```
{
```

```
    public static final String mysql="mysql";
```

```
    public abstract UserDAO getUserDAO();
```

```
    public abstract CourseDAO getCourseDAO();
```

```
    public abstract MessageDAO getMessageDAO();
```

```
    public abstract ConversationDAO getConversationDAO();
```

```
    public abstract CommentDAO getCommentDAO();
```

```
    public abstract GradesDAO getGradesDAO();
```

```
    public abstract AnnouncementDAO getAnnouncementDAO();
```

```
    public abstract WorkplaceDAO getWorkplaceDAO();
```

```
    public abstract WorkplaceFileDAO getWorkplaceFileDAO();
```

```
    public abstract PersonalCalendarDAO getPersonalCalendarDAO();
```

```
    public abstract DigitalStuffDAO getDigitalStuffDAO();
```

```
    public static DAOFactory getDAOfactory(String select)
```

```
{
```

```
        If (select.equals("mysql"))
```

```
        {
```

```
            return new MySQLDAOFactory();
```

```
        }else{
```

```
System.out.println("Uncompatible type of database implementation.Please check  
the manual");
```

```
        return null;
```

```
    }
```

### WorkplaceFileDAO

Το DAO αυτό περιέχει τις κλάσεις που σχετίζονται με την διαχείριση των αρχείων των χώρων εργασίας.

1. **public** LinkedList<WorkplaceFileTransfer> **getAllWorkplaceProjects**(int workplaceID) : επιστρέφει μια λίστα με όλες τις εργασίες για ένα συγκεκριμένο χώρο εργασίας. Παίρνει πληροφορία από τον πίνακα 'lab\_projects' με βάση κάποιο συγκεκριμένο id χώρου εργασίας. Ο πίνακας αυτός περιέχει όλες τις εργασίες των χώρων εργασίας.
2. **public** LinkedList<WorkplaceFileTransfer> **getAllWorkplaceStuff**(int workplaceID) : επιστρέφει μια λίστα με όλα τα έγγραφα ενός χώρου εργασίας. Παίρνει πληροφορία από τον πίνακα 'lab\_files' με βάση το id του χώρου εργασίας. Ο πίνακας αυτός περιέχει όλο το υλικό που υπάρχει στους χώρους εργασίας.
3. **public** WorkplaceFileTransfer **getProject**(int fileID) : δεδομένου του fileID επιστρέφει όλη τη πληροφορία ενός συγκεκριμένου project. Παίρνει πληροφορία από τον πίνακα 'lab\_projects' με βάση το id του αρχείου.
4. **public** LinkedList<WorkplaceFileTransfer> **getProjectDeadlines**(int userID) : επιστρέφει μια λίστα με όλες τις προθεσμίες εργασιών ενός συγκεκριμένου χρήστη. Χρησιμοποιείται στην αρχική σελίδα του χρήστη. Για να τραβήξει την πληροφορία που χρειάζεται κάνει join τους πίνακες 'lab\_projects', 'lab' και 'user\_courses' με βάση το id του χρήστη. Επίσης κάνει χρήση της συνάρτησης NOW() της mysql για να φιλτράρει αυτές που εκκρεμούν μόνο.

### WorkplaceDAO

1. **public** LinkedList<WorkplaceTransfer> **getCourseWorkplaces**(int courseID) : επιστρέφει μια λίστα με τους χώρους εργασίας ενός συγκεκριμένου μαθήματος. Παίρνει πληροφορία από τον πίνακα 'lab' με βάση το id του μαθήματος. Ο πίνακας περιέχει την πληροφορία για όλους τους χώρους εργασίας.
2. **public** WorkplaceTransfer **getWorkplace**(int workplaceID) : επιστρέφει την πληροφορία ενός συγκεκριμένου χώρου εργασίας. Παίρνει πληροφορία από τον πίνακα 'lab' με βάση το id του χώρου εργασίας.
3. **public int** **getNumOfFiles**(int workplaceID) : επιστρέφει τον αριθμό των αρχείων του εργαστηρίου. Χρησιμοποιείται στο notification. Μετράει το

πλήθος των tuples, στον πίνακα 'lab\_files', που έχουν id χώρου εργασίας ίδιο με αυτό που αναζητάμε και επιστρέφει τον αριθμό αυτό.

4. **public boolean userIsRegisteredInWorkplace(int userID, int workplaceID)** : επιστρέφει true αν ένας συγκεκριμένος χρήστης είναι εγγεγραμμένος σε κάποιο χώρο εργασίας. Χρησιμοποιείται για να κάνουμε έλεγχο όταν μπαίνει ο χρήστης στη σελίδα του εργαστηρίου. Ψάχνει τον χρήστη στον πίνακα 'lab\_registration', ο οποίος διατηρεί την πληροφορία για τις εγγραφές των χρηστών στους χώρους εργασίας.
5. **public boolean registerUser2Lab(int userID, int workplaceID)** : κάνει την εγγραφή ενός χρήστη και επιστρέφει true αν όλα έγιναν σωστά. Για να γίνει η εγγραφή προστίθεται πληροφορία στους εξής πίνακες: 'lab\_registration' για την εγγραφή του συγκεκριμένου χρήστη, 'lab\_team' για την δημιουργία ομάδας και 'lab\_team\_members' για τη δημιουργία ομάδας εργαστηρίου. Αν η εγγραφή ακολουθηθεί από αποδοχή κάποιας πρόσκλησης τότε ο χρήστης μπαίνει στην ομάδα του ατόμου που τον προσκάλεσε. Αν όχι, τότε δημιουργείται νέα ομάδα.
6. **public LinkedList getInvitationsForUser(int userID, int workplaceID)** : επιστρέφει μια λίστα με τις προσκλήσεις που έχουν γίνει σε κάποιο συγκεκριμένο χρήστη για κάποιο συγκεκριμένο εργαστήριο. Παίρνει πληροφορία από τον πίνακα 'lab\_invitations' με βάση το id του χρήστη και τον id του χώρου εργασίας.
7. **public LinkedList getMembersOfUserTeam(int userID, int workplaceID)** : επιστρέφει τη λίστα με τους χρήστες που συμμετέχουν στην ίδια ομάδα εργασίας με τον συγκεκριμένο χρήστη. Παίρνει πληροφορία από τη βάση κάνοντας join τους πίνακες 'lab\_teams' και 'lab\_team\_members' με βάση το id της ομάδας και τα id του χρήστη και του χώρου εργασίας.
8. **public boolean rejectInvitationsToUserForAWorkplace(int userID, int labID)** : απορρίπτει όλες τις προσκλήσεις προς ένα χρήστη σε κάποιο συγκεκριμένο χώρο εργασίας. Διαγράφει πληροφορία από τον πίνακα 'lab\_invitations'.
9. **public boolean acceptWorkplaceInvitation(int userID, int inviterID, int labID)** : αποδέχεται κάποια συγκεκριμένη πρόσκληση προς το χρήστη σε κάποιο συγκεκριμένο εργαστήριο και αυτόματα διαγράφει όλες τις άλλες προσκλήσεις για το αντίστοιχο εργαστήριο από άλλους χρήστες. Για να γίνει η αποδοχή διαγράφεται η ομάδα που είχε δημιουργηθεί με βάση το χρήστη όταν έκανε την εγγραφή και στην οποία ήταν μόνος του (η ομάδα αυτή δημιουργείται στον πίνακα 'lab\_team' μόλις ο χρήστης γραφτεί σε κάποιο χώρο εργασίας). Διαγράφει το χρήστη από τον πίνακα

‘lab\_team\_members’ αφού πια η παλιά του ομάδα δεν υπάρχει. Βρίσκει το id της ομάδας του χρήστη που τον έχει προσκαλέσει και εισάγει πληροφορία στο ‘lab\_team\_members’ βάζοντας το χρήστη μας στην ομάδα αυτού που τον προσκάλεσε.

10. **public boolean sendInvitationToUser(int senderID, int receiver, int labID) :** δημιουργεί μια πρόσκληση για κάποιο συγκεκριμένο χώρο εργασίας από το χρήστη μας προς κάποιον άλλον χρήστη. Δημιουργεί ένα tuple στον πίνακα ‘lab\_invitations’ με την πρόσκληση προς κάποιον συγκεκριμένο χρήστη.

### PersonalCalendarDAO

Στο όνομα των συναρτήσεων το ‘today’ χρησιμοποιήθηκε γιατί καλούνται πρώτα για να δει ο χρήστης τα γεγονότα της τρέχουσας ημέρας. Είναι παραμετροποιημένες και είναι οι ίδιες που δείχνουν και τα γεγονότα των άλλων ημερών.

1. **public LinkedList<CalendarEventTransfer>getTodayEventsFromCalendar(int userID, String day, String month, String year) :** επιστρέφει λίστα με τα γεγονότα για μια συγκεκριμένη μέρα στο προσωπικό ημερολόγιο του χρήστη. Κάνει join του πίνακες user\_courses (έχοντας επιλέξει μόνο τα μαθήματα του χρήστη πρώτα) και calendarEvent με βάση το id του μαθήματος και στη συνέχεια επιλέγει τα γεγονότα που πληρούν τα ημερολογιακά κριτήρια που έχουν τεθεί στην κλήση της συνάρτησης (έτος, μήνας, μέρα).
2. **public LinkedList<WorkplaceFileTransfer> getTodayDeadlines(int userID, String day, String month, String year) :** επιστρέφει μια λίστα με τις προθεσμίες παράδοσης στα μαθήματα του χρήστη για τη συγκεκριμένη μέρα. Κάνει join τους πίνακες lab\_projects, lab και user\_courses για να μπορέσει να πάρει τα πρότζεκτ που είναι στα μαθήματα του χρήστη. Στη συνέχεια φιλτράρει τα αποτελέσματα με βάση τα ημερολογιακά κριτήρια που έχουν τεθεί στην κλήση της συνάρτησης(έτος, μήνας, μέρα).
3. **public LinkedList<WorkplaceTransfer> getTodayRegistrationDeadlines(int userID, String day, String month, String year) :** επιστρέφει λίστα με τις προθεσμίες εγγραφής σε εργαστήρια που λήγουν την συγκεκριμένη μέρα. Κάνει join τους πίνακες lab και user\_courses με βάση το id του μαθήματος και επιλέγει με βάση το id του χρήστη και τα ημερολογιακά κριτήρια που έχουν τεθεί στην κλήση της συνάρτησης (ημέρα, μήνας, έτος).

### GradesDAO



Εδώ παρόλο που υπάρχει μία μόνο συνάρτηση και η οποία θα μπορούσε να ενταχθεί στο αντίστοιχο DAO του μαθήματος, θεωρήσαμε καλύτερο να φτιάξουμε διαφορετικό DAO για τις βαθμολογίες, καθώς είναι από τα πράγματα τα οποία μπορεί να μελλοντικά να χρειαστούν πιο σύνθετη λειτουργικότητα.

1. **public** LinkedList<GradesTransfer> **getCourseGrades**(int courseID, int userID) : επιστρέφει τη λίστα με τους βαθμούς ενός συγκεκριμένου χρήστη σε ένα συγκεκριμένο μάθημα. Κάνει join τους πίνακες course\_grades και user\_grades με βάση τα id του χρήστη και του συγκεκριμένου μαθήματος.

### DigitalStuffDAO

1. **public** LinkedList<DigitalStuffTransfer> **getAllLectures**(int courseID) : επιστρέφει μια λίστα με όλες τις διαλέξεις ενός συγκεκριμένου μαθήματος. Παίρνει πληροφορία από τον πίνακα digitalStuff\_lecture με βάση το id του μαθήματος.
2. **public** LinkedList<DigitalStuffTransfer> **getAllInstitutionNotes**(int courseID) : επιστρέφει μια λίστα με όλες τις σημειώσεις φροντιστηρίου ενός συγκεκριμένου μαθήματος. Παίρνει πληροφορία από τον πίνακα digitalStuff\_institution με βάση το id του μαθήματος.
3. **public** LinkedList<DigitalStuffTransfer> **getAllUsefullDocs**(int courseID) : επιστρέφει μια λίστα με όλα τα χρήσιμα έγγραφα ενός συγκεκριμένου μαθήματος. Παίρνει πληροφορία από τον πίνακα digitalStuff\_usefullDocs με βάση το id του μαθήματος.

### CourseDAO

1. **public** LinkedList<CourseTransfer> **getUserCourses**(int userID) : επιστρέφει μια λίστα με τα μαθήματα που είναι εγγεγραμμένος ο χρήστης. Κάνει join τους πίνακες user\_courses και course με βάση το id του χρήστη.
2. **public** LinkedList<CourseTransfer> **getAllCourses**(int semester) : επιστρέφει όλα τα μαθήματα για ένα συγκεκριμένο εξάμηνο. Παίρνει πληροφορία από τον πίνακα courses μόνο. Η επιλογή των μαθημάτων γίνεται με βάση το id του εξαμήνου το οποίο συμπεριλαμβάνεται στην πληροφορία του κάθε μαθήματος.
3. **public** LinkedList **getCourseServices**(int courseID) : επιστρέφει τη λίστα με τις υπηρεσίες που υποστηρίζονται σε κάθε μάθημα. Παίρνει πληροφορία



από τον πίνακα `course_services` ο οποίος περιέχει ουσιαστικά μόνο την πληροφορία για το ποιες υπηρεσίες υποστηρίζονται σε κάθε μάθημα, και όχι τις υπηρεσίες αυτές καθαυτές. Η ιδέα εδώ είναι ότι στον πίνακα αυτό κάθε tuple αντιστοιχεί σε ένα μάθημα. Και έχει την τιμή του `id` του μαθήματος στις υπηρεσίες που υποστηρίζονται και `null` σε αυτές που δεν υποστηρίζονται. Η επιλογή γίνεται με βάση το `id` του μαθήματος.

4. **public** `CourseTransfer` **getCourse**(`int` `courseID`) : επιστρέφει την πληροφορία (όνομα, κωδικό, εξάμηνο) για ένα συγκεκριμένο μάθημα. Παίρνει πληροφορία από τον πίνακα `course` με βάση το `id` του μαθήματος.
5. **public** `CourseInfoTransfer` **getCourseInfo**(`int` `courseID`) : επιστρέφει τις πληροφορίες μαθήματος για ένα συγκεκριμένο μάθημα. Παίρνει πληροφορία από πολλούς πίνακες καθώς με τη νέα δομή που έχει η βάση όλες οι πληροφορίες είναι δυναμικές και όχι στατικές όπως είναι τώρα στο `courses`. Παίρνει την περιγραφή του μαθήματος από τον πίνακα `course_info` με βάση το `course_informationID` το οποίο πρέπει να ταυτίζεται με το `id` του μαθήματος, (όπως εξηγήσαμε στο 3. `etCourseServices(courseID)`). Στη συνέχεια κάνει `join` τους πίνακες `user` και `course_teachers` με βάση το `id` του μαθήματος, για να μπορέσει να πάρει την πληροφορία για τους καθηγητές. Με την ίδια λογική στη συνέχεια κάνει `join` τους πίνακες `user` και `course_assistants` για να πάρει την πληροφορία για το βοηθητικό προσωπικό του εργαστηρίου. Στη συνέχεια παίρνει από τους πίνακες `course_lab_hours`, `course_class_hours`, `course_institution_hours`, με βάση το `id` του μαθήματος, τις ώρες του εργαστηρίου, των διαλέξεων και του φροντιστηρίου αντίστοιχα.
6. **public boolean** **userIsRegisteredInCourse**(`int` `userID`, `int` `courseID`) : επιστρέφει `true` αν ο χρήστης είναι εγγεγραμμένος στο συγκεκριμένο μάθημα. Ελέγχει στον πίνακα `user_courses` αν ο χρήστης είναι εγγεγραμμένος στο μάθημα. Ο έλεγχος γίνεται με βάση τα `id` του χρήστη και του μαθήματος.
7. **public boolean** **userIsRegisteredInCoursePending**(`int` `userID`, `int` `courseID`) : επιστρέφει `true` αν ο χρήστης έχει κάνει αίτηση εγγραφής σε κάποιο μάθημα και ο καθηγητής δεν έχει αποδεχτεί ή απορρίψει ακόμα το αίτημα του. Ελέγχει στον πίνακα `user_courses_pending` αν ο χρήστης έχει κάνει αίτηση εγγραφής στο μάθημα. Ο έλεγχος γίνεται με βάση τα `id` του χρήστη και του μαθήματος. Εδώ θα υπάρχει ο χρήστης μόνο αν έχει κάνει αίτηση εγγραφής και ο καθηγητής δεν τον έχει δεχτεί ακόμα. Μόλις τον δεχτεί ή τον απορρίψει θα διαγράφεται αυτόματα από αυτόν τον πίνακα.

8. **public boolean RegisterUser(int courseID, int userID)** : κάνει την εγγραφή του χρήστη και επιστρέφει true αν έχουν γίνει όλα σωστά. Εισάγει τον χρήστη στον πίνακα user\_courses\_pending μέχρι να τον δεχτεί ή να τον απορρίψει ο καθηγητής.
9. **public LinkedList<CalendarEventTransfer> getAllCourseEventsFromMonth(int courseID, String month, String year)** : επιστρέφει όλα τα γεγονότα του ημερολογίου ενός μαθημάτων έναν συγκεκριμένο μήνα. Παίρνει πληροφορία από τον πίνακα calendarEvent με βάση το id του μαθήματος και ημερολογιακά κριτήρια (έτος, μήνας).

### ConversationDAO

1. **public LinkedList<ConversationTransfer> getGeneralConversations()**: επιστρέφει μια λίστα με όλες τις γενικές συζητήσεις. Παίρνει πληροφορία από τον πίνακα conversation με βάση το id του μαθήματος. Έχουμε δεσμεύσει το courseID=0 για τις γενικές συζητήσεις. Με τον τρόπο αυτό χρησιμοποιούμε ένα μόνο πίνακα για όλες τις συζητήσεις, ανεξάρτητα αν είναι γενικές ή μαθήματος. Αυτό μας βοηθάει στο να γλιτώσουμε κόπο, χρόνο και πολυπλοκότητα από το να δημιουργήσουμε όλες τις κλάσεις που σχετίζονται με συζητήσεις δύο φορές, μία για τα μαθήματα και μία για τις γενικές συζητήσεις.
2. **public ConversationTransfer getConversation(int convID)** : επιστρέφει μια συγκεκριμένη συζήτηση και τα σχόλια της. Παίρνει πληροφορία από τους πίνακες comments και conversation (με διαφορετικά queries) με βάση το id της συζήτησης.
3. **public boolean convExists(int conversationID)** : επιστρέφει true αν μια συγκεκριμένη συζήτηση υπάρχει. Ελέγχει στον πίνακα conversation με βάση το id της συζήτησης για το αν υπάρχει μια συγκεκριμένη συζήτηση.
4. **public LinkedList<ConversationTransfer> getCourseConversations(int courseID)** : επιστρέφει τη λίστα με τις συζητήσεις ενός μαθήματος. Παίρνει πληροφορία από τον πίνακα conversation με βάση το id του μαθήματος.
5. **public LinkedList<CourseTransfer> getCoursesWithUnreadConv(int userID)** : επιστρέφει τη λίστα με τα μαθήματα που έχουν αδιάβαστα σχόλια σε συζητήσεις. Κάνει join στους πίνακες course και comments\_unread με βάση το id του μαθήματος και του χρήστη.

6. **public int getUnreadFromCourse(int courseId)** : επιστρέφει τον συνολικό αριθμό των αδιάβαστων σχολίων στις συζητήσεις ενός μαθήματος. Παίρνει πληροφορία από τον πίνακα `comments_unread` με βάση το `id` του μαθήματος και με χρήση της συνάρτησης `count()` της MySQL.
7. **public int getAllUnreadComments(int userID)** : επιστρέφει τον συνολικό αριθμό αδιάβαστων σχολίων που έχει ένας χρήστης στο σύνολο των συζητήσεων των μαθημάτων που τον ενδιαφέρουν. Παίρνει πληροφορία από τον πίνακα `comments_unread`, με βάση το `id` του χρήστη και τον περιορισμό το `id` του μαθήματος στο οποίο αντιστοιχεί η συζήτηση να μην είναι 0. Δηλαδή η συζήτηση να μην είναι γενική. Γίνεται χρήση της συνάρτησης `count()` της MySQL.
8. **public int getUnreadFromConv(int convID)** : επιστρέφει τον αριθμό των αδιάβαστων μηνυμάτων σε μία συζήτηση. Παίρνει πληροφορία από τον πίνακα `comments_unread` με βάση το `id` της συζήτησης και με χρήση της συνάρτησης `count()` της MySQL.
9. **public boolean createNewConversation(String title, String content, int courseId, int userID)** : δημιουργεί μια νέα συζήτηση. Επιστρέφει `true` αν όλα έγιναν σωστά. Για τη δημιουργία μια γενικής συζήτησης το `courseID=0`. Εισάγει πληροφορία στον πίνακα `conversation`. Χρησιμοποιεί τις παραμέτρους που εισάγονται όταν γίνεται η κλήση της και την συνάρτηση `now()` της MySQL. Το `id` του μαθήματος είναι 0 αν η συζήτηση που δημιουργείται ανήκει στις γενικές συζητήσεις.
10. **public boolean conversationReaded(int userID, int convID)** : καλείται όταν ένας χρήστης δει κάποια συζήτηση στην οποία είχε αδιάβαστα σχόλια και τα αφαιρεί από τη λίστα με τα αδιάβαστα. Διαγράφει από τον πίνακα `comments_unread` τα `tuples` που αντιστοιχούν στα σχόλια που ανήκουν στην συζήτηση που είδε ο χρήστης, με βάση το `id` του χρήστη και της συζήτησης.
11. **public int getAllUnreadGeneralComments(int userID)** : επιστρέφει το συνολικό αριθμό αδιάβαστων σχολίων που έχει ένας χρήστης στις γενικές συζητήσεις. Με χρήση της συνάρτησης `count()` της MySQL επιστρέφει την πληροφορία από τον πίνακα `comments_unread` με βάση το `id` του χρήστη και με `id` μαθήματος '0' που αντιστοιχεί στις γενικές συζητήσεις.
12. **public boolean generalConversationWithUnreadedCommentsByUser(int userID, int convID)** : επιστρέφει `true` αν μια γενική συζήτηση έχει αδιάβαστα σχόλια από ένα χρήστη. Παίρνει πληροφορία από τον πίνακα `comments_unread` με βάση το `id` της συζήτησης και του χρήστη.

### CommentDAO

1. **public boolean createNewComment**(String title, String content, **int** convID, **int** userID) : δημιουργεί ένα σχόλιο σε μία συζήτηση και επιστρέφει true αν όλα γίνουν σωστά. Εισάγει πληροφορία στον πίνακα comments. Χρησιμοποιεί την πληροφορία που εισάγεται στην κλήση της συνάρτησης και την συνάρτηση curdate() της MySQL για να εισάγει την ημερομηνία στην μορφή που θέλουμε.

### AnnouncementDAO

1. **public** LinkedList<AnnouncementTransfer> **getCourseAnnouncements**(**int** courseID) : επιστρέφει τη λίστα με όλες τις ανακοινώσεις των μαθημάτων. Παίρνει πληροφορία από τον πίνακα announcement με βάση το id του μαθήματος.
2. **public int countUnreadAnnouncementsFromCourse**(**int** courseID, **int** userID) : επιστρέφει τον συνολικό αριθμό αδιάβαστων ανακοινώσεων σε κάποιο μάθημα για έναν συγκεκριμένο χρήστη. Χρησιμοποιείται σε notification στην σελίδα με τις νέες ανακοινώσεις του χρήστη. Με χρήση της συνάρτησης count() της MySQL επιστρέφει την πληροφορία από τον πίνακα announcements\_unread με βάση το id του χρήστη και του μαθήματος.
3. **public int countAllUnreadAnnouncementsOfUser**(**int** userID) : επιστρέφει τον συνολικό αριθμό των αδιάβαστων ανακοινώσεων σε όλα τα μαθήματα που είναι εγγεγραμμένος ένας φοιτητής. Με χρήση της συνάρτησης count() υπολογίζει το πλήθος, και επιστρέφει την πληροφορία από τον πίνακα announcements\_unread με βάση το id του χρήστη.
4. **public** LinkedList<CourseTransfer> **getCoursesWithUnreadAnnouncements** (**int** userID) : επιστρέφει τα μαθήματα στα οποία ένα συγκεκριμένος φοιτητής έχει αδιάβαστες ανακοινώσεις. Κάνει join τους πίνακες course και announcements\_unread στο id του μαθήματος και παίρνει τα tuples που το attribute του id του χρήστη ταιριάζει με αυτό που δόθηκε στη κλήση της συνάρτησης.
5. **public boolean announcementsReaded**(**int** userID, **int** courseID) : καλείται όταν ο φοιτητής διαβάσει κάποιο νέο μήνυμα, και το αφαιρεί από τα αδιάβαστα μηνύματα. Διαγράφει από τον πίνακα announcements\_unread τις ανακοινώσεις εκείνες που μόλις διάβασε ο χρήστης.

### UserDAO

1. **public** User **existedUser**(String username, String password) : επιστρέφει username και password αν ο χρήστης υπάρχει ή null αν δεν υπάρχει. Χρησιμοποιείται για τον έλεγχο εισόδου του χρήστη στο σύστημα μας. Ελέγχει την ύπαρξη χρήστη μέσα στον πίνακα user με βάση το username και password του χρήστη.
2. **public** String **IDtoUsername**(int userID) : δεδομένου ενός id χρήστη επιστρέφει το username του. Ψάχνει στον πίνακα user με βάση το id και επιστρέφει το username. Τα username και id είναι μοναδικά για κάθε χρήστη.
3. **public** UserTransfer **getUser**(int userID) : επιστρέφει όλα τα δεδομένα ενός συγκεκριμένου χρήστη. Παίρνει πληροφορία από τον πίνακα user με βάση το id του χρήστη.
4. **public** UserTransfer **findUserFromUsername**(String receipt) : αναζήτηση χρήστη με βάση το username. Ψάχνει τον πίνακα user με βάση το username του χρήστη, το οποίο είναι μοναδικό.

### MessageDAO

1. **public** LinkedList<MessageTransfer> **getUserInbox**(int userID) : επιστρέφει τη λίστα με όλα τα εισερχόμενα μηνύματα ενός χρήστη. Παίρνει πληροφορία από τον πίνακα personal\_messages ταυτίζοντας το id του χρήστη, που δίνεται στην κλήση της συνάρτησης, με το “receiver” attribute παίρνοντας έτσι μόνο τα εισερχόμενα για κάποιον χρήστη.
2. **public** LinkedList<MessageTransfer> **getUserOutbox**(int userID) : επιστρέφει μια λίστα με όλα τα εξερχόμενα μηνύματα ενός χρήστη. Παίρνει πληροφορία από τον πίνακα personal\_messages ταυτίζοντας το id του χρήστη, που δίνεται στην κλήση της συνάρτησης, με το “sender” attribute παίρνοντας έτσι μόνο τα εξερχόμενα για κάποιον χρήστη.
3. **public** int **getUserInboxUnread**(int userID) : επιστρέφει τον αριθμό των αδιάβαστων εισερχομένων μηνυμάτων ενός συγκεκριμένου χρήστη. Παίρνει πληροφορία από τον πίνακα personal\_messages με βάση το id του παραλήπτη και με το attribute “unreaded” το οποίο παίρνει τιμές “yes” - “no”. Η συνάρτηση επιστρέφει μόνο τον αριθμό αυτών που έχουν “yes”.
4. **public** MessageTransfer **getMessage**(int messageId) : επιστρέφει την πληροφορία για ένα συγκεκριμένο μήνυμα. Ανεξάρτητα από το αν είναι

εξερχόμενο ή εισερχόμενο. Παίρνει πληροφορία από τον πίνακα `personal_messages` με βάση το `id` του μηνύματος.

5. `public boolean messageReaded(int messageId)` : αλλάζει την κατάσταση ενός μηνύματος από `unreaded` σε `readed` μόλις το διαβάσει ο χρήστης. Αυτό γίνεται ορίζοντας την τιμή του attribute “unread” από “yes” σε “no”.
6. `public boolean SendMessage(int senderID, int recipientID, String title, String content)` : αποστολή (δημιουργία στη βάση) νέου προσωπικού μηνύματος προς κάποιον χρήστη. Δημιουργεί ένα `tuple` στον πίνακα `personal_messages` με τα χαρακτηριστικά που δόθηκαν στην κλήση της συνάρτησης και με χρήση της συνάρτησης `now()` της MySQL, για τον ορισμό του ημερολογιακού στίγματος τη στιγμή που δημιουργήθηκε το μήνυμα.

# *Κεφάλαιο 9*

*Ανακεφαλαίωση - Μελλοντικές Επεκτάσεις*



## 10.1 Ανακεφαλαίωση

Σήμερα η εξέλιξη της τεχνολογίας και των υπολογιστών, τόσο από πλευράς υλικού όσο και λογισμικού, έχει οδηγήσει σε εντελώς διαφορετική αντίληψη, λειτουργία και συμπεριφορά, σχετικά με την διαχείριση της πληροφορίας και την πρόσβαση σε αυτήν. Οι υπολογιστές που ήταν, και παραμένουν, το παράθυρο μας στη παγκόσμιας κλίμακας πληροφορία έχουν αλλάξει. Έχουν γίνει πιο γρήγοροι, πιο αποδοτικοί και πολύ πιο μικροί. Τόσο μικροί που χωράνε στην τσέπη μας, δίνοντας μας την δυνατότητα να έχουμε πρόσβαση στα αρχεία μας, στα mail μας και στο διαδίκτυο όπου και αν βρισκόμαστε. Επίσης οι γρήγοροι ρυθμοί ζωής μας αναγκάζουν να κινούμαστε σε πιο έντονους ρυθμούς, κάνοντας την εύκολη πρόσβαση στην πληροφορία που μας ενδιαφέρει, από οπουδήποτε και οποιαδήποτε στιγμή, αναγκαία.

Στα πλαίσια αυτής της διπλωματικής εργασίας δημιουργήθηκε μια εφαρμογή η οποία παρέχει την λειτουργικότητα του Courses (MOLE) καθώς και κάποιες ακόμα πιο προσωπο-κεντρικές λειτουργίες μέσω smartphones και tablets. Η ιδιαιτερότητα της εφαρμογής μας είναι ότι ανεξαρτήτως λειτουργικού συστήματος και κατασκευαστή συσκευής παρέχονται στο χρήστη οι ίδιες δυνατότητες και η ίδια λειτουργικότητα μέσω του browser, μέσα από μια διεπαφή που είναι εύχρηστη σε όλες τις mobile συσκευές. Στα στάδια ανάπτυξης της εφαρμογής έγινε εντατική αξιολόγηση τόσο από μένα, ως usability expert με τη μέθοδο της Ευριστικής Αξιολόγησης όσο και με τη συμμετοχή χρηστών με τις μεθόδους του paper-prototype-walkthrough, Pluralistic Walkthrough και Think Aloud. Για την υλοποίηση της χρησιμοποιήθηκαν τεχνολογίες που είναι ελεύθερες προς χρήση, χωρίς κόστος, και που υποστηρίζονται από το σύνολο των browsers των σύγχρονων φορητών συσκευών, και που δεν εξαρτώνται από το λειτουργικό ή τη συσκευή που χρησιμοποιούμε. Με τη χρήση του jQuery Mobile καταφέραμε να έχουμε ένα συνεπές οπτικό αποτέλεσμα ανεξάρτητα από το μέγεθος της οθόνης, την ανάλυση και τον τρόπο που κρατάει ο χρήστης τη συσκευή. Επίσης είδαμε και στην πράξη όλα τα προβλήματα που εμφανίζονται στη μεταφορά μιας desktop εφαρμογής σε mobile έκδοση, και μέσα από ένα συστηματικό τρόπο αξιολόγησης τα ξεπεράσαμε υλοποιώντας μια εύχρηστη και λειτουργική εφαρμογή. Αναγνωρίζοντας την ιδιαιτερότητα των mobile συσκευών προσαρμόσαμε την λειτουργικότητα, παρέχοντας πιο προσωποκεντρικές λειτουργίες και κάναμε και τις αντίστοιχες αλλαγές στην desktop πλατφόρμα. Συνολικά μπορούμε να πούμε ότι είδαμε πως μπορεί να συνυπάρξει μια εφαρμογή γραφείου με την αντίστοιχη της σε mobile έκδοση και πως μπορούν να ξεπεραστούν τα προβλήματα του μέσου που εμφανίζονται.



## 10.2 Μελλοντικές Επεκτάσεις

Μελλοντικές επεκτάσεις που προτείνονται από την παρούσα διπλωματική είναι:

1. Εισαγωγή video μέσω της HTML5. Έτσι θα έχουν τη δυνατότητα οι χρήστες να παρακολουθούν video (π.χ Διαλέξεις μαθημάτων, Παρουσιάσεις), όπου και αν βρίσκονται, μέσα στον browser χωρίς να εξαρτώνται από plug-ins συγκεκριμένων κατασκευαστών (πχ flash player). Ο λόγος που δεν το υλοποιήσαμε αυτό ήταν ότι δεν έχουν καθιερωθεί ακόμα τα standards σχετικά με την κωδικοποίηση και η υποστήριξη από τους browsers ήταν ελλιπής.
2. Χρήση του <canvas> της HTML5. Εδώ οι ιδέες είναι πολλές και στοχεύουν κυρίως στα tablets. Μια ιδέα είναι η υλοποίηση ενός εικονικού-καμβά στον οποίο θα μπορούν να δουλεύει τόσο ο καθηγητής όσο και οι καθηγητές παράλληλα κάνοντας πιο διαδραστική την διδασκαλία. Επίσης θα μπορούσε να δημιουργηθεί μια διαδικτυακή εφαρμογή - μέρος του MOLE - η οποία θα βοηθούσε τους καθηγητές να αποθηκεύουν στη βάση συγκεκριμένα μοντέλα που σχετίζονται με κάποιο μάθημα, μέσα από ένα πιο ελεύθερο τρόπο. Να μπορεί να σχεδιάζει δηλαδή ο καθηγητής στον καμβά της HTML5 μια FSM για παράδειγμα, και αυτή να μπορεί να αποθηκευτεί σαν αντικείμενο μέσα στη βάση. Το αντικείμενο αυτό θα μπορεί να ξανακληθεί και να τροποποιηθεί τόσο από τον καθηγητή όσο και από τους μαθητές. Υπάρχουν πολλές ιδέες για το πως αυτές οι εφαρμογές θα μπορούν να βοηθήσουν στην εκπαίδευση και να κάνουν πιο πλούσιο το διδακτικό έργο. Και εδώ υπάρχουν αντίστοιχα θέματα καθώς η HTML5 δεν έχει γίνει standard και το canvas-element δεν υποστηρίζεται παντού όπως πρέπει, αλλά σε μερικά χρόνια η δημιουργία τέτοιου είδους εφαρμογών θα είναι εφικτή και εμείς θα έχουμε τη βάση πάνω στην οποία θα μπορούμε να αναπτύξουμε κάτι τέτοιο, σαν μέρος του MOLE.
3. Εισαγωγή χαρτών μέσω browser (google maps) για πλοήγηση στο τμήμα και για πιο διαδραστική ενημέρωση σχετικά με γεγονότα του Πολυτεχνείου ή συμβάντα που αφορούν συγκεκριμένο χρήστη.
4. Προσθήκη στοιχειώδους λειτουργίας εισαγωγής πληροφοριών από καθηγητές. Για παράδειγμα δημιουργία ανακοινώσεων, αποστολή μηνυμάτων κ.α

## Βιβλιογραφία

1. Mobile Design & Development by Brian Fling - O'Reilly Media
2. Core Servlets and Javasever pages: Core Technologies by Marty Hall and Larry Brown - Prentice Hall PTR/Sun Microsystems Press
3. Ajax for Web Application Developers by Kris Hadlock - Sams Publising
4. Nielsen Usability Inspection Methods by Jacob Nielsen and Robert L.Mack - John Wiley & Sons
5. Personalization-based Optimization of Web Interfaces for Mobile Devices by Michael Hinz, Zoltán Fiala, Frank Wehner - Dresden University of Technology
6. Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices by Pekka Parhi MediaTeam Oulu Department of Electrical and Information Engineering & Amy K. Karlson, Benjamin B. Bederson Human-Computer Interaction Lab Computer Science Department University of Maryland, College Park, MD
7. Design Patterns for User Interface for Mobile Applications by Erik G. Nilsson
8. Usability of Rich Internet Applications and Web-Based Tools DESIGN GUIDELINES BASED ON USER TESTING OF 46 FLASH TOOLS IN 2002 by Hoa Loranger, Amy Schade, and Jakob Nielsen
9. jQuery Mobile Documentation & Demos [<http://jquerymobile.com/demos/1.0a4.1>]
10. HTML5 Reference [ <http://dev.w3.org/html5/html-author>]
11. CSS Mobile Profile 2.0 [ <http://www.w3.org/TR/2008/CR-css-mobile-20081210> ]
12. Mobile Web Application Best Practices [ <http://www.w3.org/TR/2010/REC-mwabp-20101214> ]
13. Various aspects of mobile web development [ <http://www.quirksmode.org/mobile> ]
14. Mobile Usability Testing [ <http://www.slideshare.net/barbaraballard/mobile-usability-testing> ]
15. Dive Into HTML5 by Mark Pilgrim [ <http://diveintohtml5.org> ]

16. Mobile Usability in Educational Contexts: What have we learnt? Agnes Kukulska-Hulme The Open University, UK [ <http://www.irrodl.org/index.php/irrodl/article/view/356> ]
17. Designing Usable Pocket Rockets: A Three Step Guide to Usability on the Mobile Web [ <http://www.mobithinking.com/> ]
18. Top 6 Mistakes to Avoid in Mobile Usability [ <http://spyrestudios.com/top-6-mistakes-to-avoid-in-mobile-usability> ]
19. Titanium, PhoneGap, Sencha Touch, jQuery Mobile – Clearing up confusion [ <http://beautyindesign.com/blog/titanium-phonegap-sencha-touch-jquery-mobile-clearing-up-confusion> ]
20. Comparing Mobile Web (HTML5) Frameworks: Sencha Touch, jQuery Mobile, jQuery Touch, Titanium [ <http://interfacethis.com/2011/adventures-in-html5-part-one> ]
21. Browser Storage for HTML5 Apps [ <http://mobile.tutsplus.com/tutorials/mobile-web-apps/browser-storage> ]
22. Best practices to develop perfect websites for mobile devices [ <http://woorkup.com/2010/01/10/best-practices-to-develop-perfect-websites-for-iphone-and-mobile-devices> ]
23. Global Internet Stats [ <http://gs.statcounter.com> ]
24. Build a jQuery Mobile App: App Logic & Interface [ <http://mobile.tutsplus.com/tutorials/mobile-web-apps/build-a-jquery-mobile-survey-app-app-logic-interface> ]
25. Java Servlet [ [http://en.wikipedia.org/wiki/Java\\_Servlet](http://en.wikipedia.org/wiki/Java_Servlet) ]
26. JavaServer Pages [ [http://en.wikipedia.org/wiki/JavaServer\\_Pages](http://en.wikipedia.org/wiki/JavaServer_Pages) ]
27. Model–view–controller [ <http://en.wikipedia.org/wiki/Model%E2%80%93View%E2%80%93Controller> ]
28. MVC design pattern [ <http://www.techrepublic.com/article/mvc-design-pattern-brings-about-better-organization-and-code-reuse/1049862> ]
29. Combining Servlets, JSP, and JavaBeans [ <http://www.stardeveloper.com/articles/display.html?article=2001060501&page=1> ]

30. JSP : Intro and Overview [ <http://courses.coreservlets.com/Course-Materials/pdf/csajsp2/09-JSP-Intro.pdf> ]
31. Programming tutorials and source code examples [ <http://www.java2s.com> ]
32. jQuery Mobile : A Forms Tutorial [ <http://mobile.tutsplus.com/tutorials/mobile-web-apps/jquery-mobile-forms> ]