



Πανεπιστήμιο Πειραιώς - Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

<<Πληροφορική>>

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	ΕΚΠΑΙΔΕΥΤΙΚΟ ΛΟΓΙΣΜΙΚΟ ΓΙΑ ΤΗΝ ΑΡΧΑΙΑ ΕΛΛΗΝΙΚΗ ΓΛΩΣΣΑ Α' ΓΥΜΝΑΣΙΟΥ EDUCATIONAL SOFTWARE FOR ANCIENT GREEK LANGUAGE
Όνοματεπώνυμο Φοιτητή	Νικολέτα Μεντεσίδου
Πατρώνυμο	Χαράλαμπος
Αριθμός Μητρώου	ΜΠΠΛ / 11032
Επιβλέπων	Ευθύμιος Αλέπης , Επίκουρος καθηγητής

Ημερομηνία Παράδοσης **Ιούνιος 2015**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Ευθύμιος Αλέπης
Επίκουρος καθηγητής

(υπογραφή)

Γεώργιος Τσιχριντζής
Καθηγητής

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

Πίνακας περιεχομένων

1. Εισαγωγή	3
2. Σχεδιασμός εφαρμογής	4
2.1. Περιγραφή Απαιτήσεων	4
2.1.1. Βασικές Απαιτήσεις	4
2.1.2. Ειδικές Απαιτήσεις	5
2.2. Μοντελοποίηση απαιτήσεων	7
2.2.1. Διάγραμμα περιπτώσεων χρήσης	7
2.2.2. Διαγράμματα δραστηριοτήτων	8
2.3. Σχεδιασμός βάσης δεδομένων	12
2.3.1. Εννοιολογικό μοντέλο	12
2.3.2. Σχεσιακό μοντέλο	15
2.3.3. Φυσικό μοντέλο	17
3. Υλοποίηση εφαρμογής	19
3.1. Αρχιτεκτονική 3-επιπέδων	19
3.1.1. Επίπεδο δεδομένων	20
3.1.2. Επίπεδο επιχειρησιακής λογικής	20
3.1.3. Επίπεδο παρουσίασης	20
3.2. Τεχνολογίες υλοποίησης	21
3.3. Περιγραφή υλοποίησης	25
4. Λειτουργικότητα συστήματος	29
4.1. Εφαρμογή διαχείρισης εκπαιδευτικού μοντέλου	29
4.1.1. Εφαρμογή εκπαιδευτικού	29
4.1.2. Εφαρμογή μαθητή	30
5. Μελλοντικές επεκτάσεις	34
Παράρτημα Α – Κώδικας εφαρμογής	35
Βιβλιογραφία	60

Πίνακας εικόνων

Εικόνα 1. Άσκηση απλής επιλογής	6
Εικόνα 2. Άσκηση πολλαπλής επιλογής.....	6
Εικόνα 3. Άσκηση συμπλήρωσης κενών.....	6
Εικόνα 4. Η αρχιτεκτονική τριών επιπέδων του προτεινόμενου συστήματος.....	19
Εικόνα 5.Επεξεργασία tags της HTML μέσω κανόνων CSS	21
Εικόνα 6. Οθόνη επιλογής τεστ για επισκόπηση.....	29
Εικόνα 7. Οθόνη επιλογής τεστ για επισκόπηση	30
Εικόνα 8. Εμφάνιση αποτελεσμάτων τεστ.....	30
Εικόνα 9 –Περιβάλλον εμφάνισης και ανάγνωσης σελίδων θεωρίας	31
Εικόνα 10 –Επιλογές μαθημάτων ανά ενότητα	32
Εικόνα 11. Ενδεικτική οθόνη διεξαγωγής μιας άσκησης	33

Πίνακας διαγραμμάτων

Διάγραμμα 1. Διάγραμμα περίπτωσης χρήσης του συνολικού συστήματος	8
Διάγραμμα 2. Διάγραμμα δραστηριοτήτων διαμόρφωσης θεωρίας και τεστ	9
Διάγραμμα 3. Διάγραμμα δραστηριοτήτων μελέτης θεωρίας	10
Διάγραμμα 4. Διάγραμμα δραστηριοτήτων εκτέλεσης τεστ	11
Διάγραμμα 5. Διάγραμμα δραστηριοτήτων επισκόπησης αποτελεσμάτων	12
Διάγραμμα 6. Σχεσιακό μοντέλο	16
Διάγραμμα 7. Η θέση της PHP μεταξύ των γλωσσών ανάπτυξης εφαρμογών διακομιστή (server-side technologies) (Κάτοχος εικόνας: W3Techs – Web Technology Surveys).....	22
Διάγραμμα 8. Επιδόσεις δοσοληψιών μόνο ανάγνωσης (read-only) του MySQL Server 5.6 (Κάτοχος εικόνας: MySQL)	24
Διάγραμμα 9. Δυνατότητες διεκπεραίωσης δοσοληψιών ανάγνωσης – εγγραφής για το MySQL Server 5.6 (Κάτοχος εικόνας: MySQL).....	24
Διάγραμμα 10. Δομή του διαδικτυακού τόπου του εκπαιδευτικού λογισμικού	25

Ευρετήριο πινάκων

Πίνακας 1. Η οντότητα “User”	12
Πίνακας 2. Η οντότητα “Exercise”	13
Πίνακας 3. Η οντότητα “question_text”	14
Πίνακας 4. Η οντότητα “question_options”	14
Πίνακας 5. Η οντότητα “results”	14
Πίνακας 6. Η οντότητα “theory”	15

Ευρετήριο κώδικα προγραμματισμού

Κώδικας 1. Πρόταση SQL για δημιουργία του πίνακα “Users”	17
Κώδικας 2. Πρόταση SQL για δημιουργία του πίνακα “exercises”	17
Κώδικας 3. Πρόταση SQL για δημιουργία του πίνακα “question_text”	17
Κώδικας 4. Πρόταση SQL για δημιουργία του πίνακα “question_options”	17
Κώδικας 5. Πρόταση SQL για δημιουργία του πίνακα “theory”	18
Κώδικας 6. Πρόταση SQL για δημιουργία του πίνακα “results”	18
Κώδικας 7: Κώδικας config.php	26
Κώδικας 8. Κώδικας header.php	27
Κώδικας 9. theory.php.....	35
Κώδικας 10. exercise.php.....	36
Κώδικας 11. manageusers.php	43
Κώδικας12. Ενδεικτικός κώδικας σελίδας θεωρίας	52

Περίληψη

Στο πλαίσιο της μεταπτυχιακής διατριβής επιλέχθηκε η ανάπτυξη ενός λογισμικού, το οποίο συνδυάζει την εφαρμογή των σύγχρονων τεχνολογιών διαδικτύου με τη διδασκαλία και εκμάθηση της αρχαίας ελληνικής γλώσσας στο επίπεδο της Α' Γυμνασίου.

Μελετήθηκε και δημιουργήθηκε ένα καινοτόμο περιβάλλον για τη διάχυση της γνώσης που αφορά στην αρχαία ελληνική γλώσσα σε περιβάλλον διαδικτύου, ελεύθερα σε οποιονδήποτε έχει πρόσβαση στο διαδίκτυο μέσα από έναν απλό φυλλομετρητή. Η θεωρία δομήθηκε σε ενότητες και για την παρουσίασή της χρησιμοποιήθηκε η γλώσσα HTML σε συνδυασμό με CSS3.

Προκειμένου το λογισμικό να αποκτήσει διαδραστικότητα και να ενισχύει τη μαθησιακή διαδικασία, υλοποιήθηκε ένα περιβάλλον διεξαγωγής ασκήσεων μέσα από το διαδικτυακό περιβάλλον. Υλοποιήθηκαν τρεις κλάσεις ασκήσεων: απλής επιλογής, πολλαπλής επιλογής και συμπλήρωσης κενών. Η πρόσβαση στο περιβάλλον των ασκήσεων γίνεται μόνο σε εγγεγραμμένους χρήστες, ώστε να καταγράφεται η επίδοσή τους και ο επιβλέπων να παρακολουθεί την εξέλιξη της αφομοίωσης των θεωρητικών θεμάτων απ' τους μαθητευόμενους.

Τέλος το λογισμικό προσφέρει ένα περιβάλλον γραφικής απεικόνισης της επίδοσης των εξεταζόμενων ώστε να δημιουργεί μια άμεση και εποπτική εικόνα της προόδου του εκπαιδευόμενου.

Το λογισμικό βασίστηκε σε ανοικτές πρότυπες τεχνολογίες διαδικτύου και μπορεί να αποτελέσει τη βάση για ανάπτυξη μιας ολοκληρωμένης πλατφόρμας για την ενσωμάτωση επιπλέον θεωρητικών ενοτήτων και τον εμπλουτισμό της με νέες ασκήσεις και δοκιμασίες. Επίσης η υιοθέτηση του σχεσιακού μοντέλου για την αποθήκευση του περιεχομένου αλλά και της βαθμολόγησης των εκπαιδευόμενων επιτρέπει την επέκταση του λογισμικού με ενότητες στατιστικής επεξεργασίας των βαθμολογικών δεδομένων και την εξαγωγή ιδιαίτερα χρήσιμων συμπερασμάτων για την εκπαιδευτική διαδικασία.

Abstract

The scope of this postgraduate thesis was to develop innovative software which exploits Internet technologies in order to provide teaching and learning functionalities for the ancient Greek language of the First high school class.

As a result a web application was developed to offer a uniform environment for the dissemination of knowledge concerning the ancient Greek language. The software offers free to access theoretical lessons to everyone who has Internet access using only a web browser. The theory lessons are structured into sections and published using the HTML language in combination with CSS3.

In order for the software to obtain interactivity and enhance the learning process, a practical examination environment was also developed. Each exam may consist of a combination of exercises. Each exercise may be a single choice, multiple choice or fill-the-blanks one. Only registered users may gain access to the exams section, in order to record their performance. The system administrator may monitor the progress of each individual learner.

Finally, a separate section a graphic representation of the exam results in order to create a comprehensive direct overview of the progress of the learner.

The software is based on open web technologies and can provide the basis for developing a comprehensive platform to integrate additional theoretical modules and enrich it with new exercises and tests. Also, the adoption of the relational model to store content and scores allow for further extension with statistical analysis modules for learners' rating and subsequent weaknesses of the educational process.

1. Εισαγωγή

Η παρούσα εργασία έχει ως βασικό της θέμα τη δημιουργία και παρουσίαση ενός εκπαιδευτικού λογισμικού με διαδραστικό χαρακτήρα, που βασικό στόχο του έχει να παρουσιάσει τη γραμματική της αρχαιοελληνικής διαλέκτου στους μαθητές της Α΄ Γυμνασίου.

Η εισαγωγή της τεχνολογίας στην εκπαίδευση κρίνεται πλέον απαραίτητη σε μια κοινωνία όπου η πληροφορική αποτελεί στοιχείο της καθημερινότητας και απαραίτητο εργαλείο με πολλές χρήσεις. Κατά τις δύο τελευταίες δεκαετίες, η ελληνική πολιτεία καταβάλλει προσπάθειες, με την υποστήριξη κοινοτικών πόρων, να εισάγει τις ΤΠΕ (Ε. Λιακοπούλου, 2010)(Τεχνολογίες της Πληροφορίας και των Επικοινωνιών) στη σχολική εκπαίδευση. Επομένως, μία από αυτές αφορά και στον τομέα της εκπαίδευσης όπου ο ρόλος που επιτελεί μπορεί να είναι καθοριστικός για την αποτελεσματικότητα της μάθησης.

Πιο συγκεκριμένα, ένα εκπαιδευτικό λογισμικό δημιουργείται προκειμένου να συμπληρώσει ή ακόμη και να αντικαταστήσει την παραδοσιακή εκπαιδευτική διαδικασία και να ενισχύσει ή ακόμη και να εκπληρώσει εξ ολοκλήρου συγκεκριμένους μαθησιακούς στόχους. Με λίγα λόγια, μπορεί να ευνοήσει την επιδίωξη προωθημένων μαθησιακών στόχων, την εφαρμογή σημαντικών παιδαγωγικών αρχών και την υιοθέτηση καινοτόμων μεθόδων διδασκαλίας (Νικολουδάκης, 2010).

Στην προκειμένη περίπτωση και με το πνεύμα της λογικής αυτής, δημιουργήθηκε το παρόν εκπαιδευτικό λογισμικό με τον χαρακτήρα ενός ΕΔΣ (Εξυπνου Διδακτικού Συστήματος) που σε καμία περίπτωση δεν επιδιώκει την αντικατάσταση του εκπαιδευτικού-καθηγητή στη διαδικασία της μάθησης αλλά περισσότερο υποβοηθάει με το εικονικό του περιβάλλον σε μία πιο εύπεπτη και διασκεδαστική μετάδοση της γνώσης σε ένα αντικείμενο κατεξοχήν βαρύ και σύνθετο όπως είναι η γραμματική της αρχαιοελληνικής διαλέκτου για τον μαθητή της Α΄ Γυμνασίου.

2. Σχεδιασμός εφαρμογής

Στο κεφάλαιο αυτό γίνεται παρουσίαση του σχεδιασμού της εφαρμογής. Περιγράφονται οι βασικές και ειδικές απαιτήσεις του συστήματος και περιγράφεται αναλυτικά ο σχεδιασμός της Βάσης Δεδομένων.

2.1. Περιγραφή Απαιτήσεων

Οι απαιτήσεις της εφαρμογής καθορίζονται από τις μαθησιακές ανάγκες των μαθητών της Α' Γυμνασίου για την εκμάθηση της Αρχαίας Ελληνικής γλώσσας. Αυτό οδήγησε στη δημιουργία ενός περιβάλλοντος το οποίο είναι διαθέσιμο σε μαθητές και καθηγητές και προσφέρει τόσο θεωρητική θεμελίωση των εννοιών των αρχαίων ελληνικών που διδάσκονται στην Α' Γυμνασίου αλλά και αντίστοιχες ασκήσεις.

Η συγκεκριμένη γνώση αλλά και οι μέθοδοι πρακτικής εξάσκησης θα πρέπει να είναι προσβάσιμες από οποιοδήποτε σημείο, ώστε να είναι η δυνατή η μετέπειτα ενσωμάτωσή τους και σε άλλα εκπαιδευτικά εργαλεία και πλατφόρμες.

2.1.1. Βασικές Απαιτήσεις

Οι βασικές απαιτήσεις του εκπαιδευτικού λογισμικού, όπως προέκυψαν από την ανάλυση του προβλήματος για την παρουσίαση των θεωρητικών εννοιών των αρχαίων ελληνικών Α' Γυμνασίου αλλά και την πρακτική εφαρμογή τους μέσω διαδραστικών ασκήσεων, παρουσιάζονται παρακάτω (Pallud & Straub, 2014):

- **Φιλικό περιβάλλον.**
Στόχος η ύπαρξη μιας φιλικής προς το χρήστη διεπαφής η οποία να έλκει το χρήστη του συστήματος και να απαιτεί την ελάχιστη αλληλεπίδραση για την αναζήτηση ή και ανάγνωση των θεματικών εννοιών.
- **Έλλειψη ανάγκης για ειδικό λογισμικό σύνδεσης.**
Απαιτείται η ελάχιστη συμμετοχή χρηστών στην εγκατάσταση συστημάτων για τη λειτουργία της εφαρμογής. Η έλλειψη ειδικού λογισμικού για τη χρήση της εφαρμογής ελαχιστοποιεί το κόστος αλλά και την εξειδίκευση του μαθητή ή του διδάσκοντος για τη χρήση του συστήματος. Προκειμένου να διευκολυνθούν οι χρήστες της εφαρμογής προς αυτή την κατεύθυνση, επιλέχθηκε σχεδιασμός και υλοποίηση του συστήματος ως εφαρμογή ιστού (web application).

Στην τεχνολογία λογισμικού, μια εφαρμογή ιστού είναι μια εφαρμογή που γίνεται διαθέσιμη μέσω ενός φυλλομετρητή ιστοσελίδων (web browser) μέσω του διαδικτύου. Ένας φυλλομετρητής ιστοσελίδων είναι το λογισμικό που επιτρέπει στον χρήστη του να προβάλλει και να αλληλεπιδρά με κείμενα, εικόνες, βίντεο, μουσική και άλλες πληροφορίες, συνήθως αναρτημένες σε μια ιστοσελίδα ενός ιστότοπου στον Παγκόσμιο Ιστό ή σε ένα τοπικό δίκτυο. Το κείμενο και οι εικόνες σε μια ιστοσελίδα μπορεί να περιέχουν συνδέσμους προς άλλες ιστοσελίδες του ίδιου ή διαφορετικού ιστότοπου.

Ο φυλλομετρητής επιτρέπει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω συνδέσμων (links). Η κύρια γλώσσα που χρησιμοποιείται από τις εφαρμογές ιστού και τους φυλλομετρητές είναι η γλώσσα μορφοποίησης HTML (HyperText Markup Language) για την προβολή των ιστοσελίδων.

Για την ανάπτυξη εφαρμογών που χρειάζονται περισσότερες διαδραστικά χαρακτηριστικά και δυνατότητες χρησιμοποιούνται και άλλες γλώσσες προγραμματισμού (εκτός από την γλώσσα μορφοποίησης HTML), όπως η JavaScript, η Java και η PHP (Pallud & Straub, 2014) (αρχικά σήμαινε Personal Home Page, τώρα αντιπροσωπεύει τη φράση Hypertext Preprocessor).

Οι εφαρμογές Ιστού είναι δημοφιλείς καθώς κάθε σύγχρονο λειτουργικό σύστημα που προορίζεται για υπολογιστές γραφείου έχει προ-εγκατεστημένο έναν ή περισσότερους φυλλομετρητές. Οι φυλλομετρητές ουσιαστικά αποτελούν λογισμικό πελάτη του δικτυακού πρωτοκόλλου επιπέδου εφαρμογών HTTP.

Για κάθε φυλλομετρητή διατίθενται επίσης, και αρκετά πρόσθετα στοιχεία (add-ons), με στόχο την επαύξηση των δυνατοτήτων τους, τη βελτίωση της χρηστικότητας τους και την προστασία του χρήστη σε θέματα ασφάλειας. Η δυνατότητα να ενημερωθούν και να διατηρηθούν οι εφαρμογές Ιστού χωρίς τη διανομή και εγκατάσταση του λογισμικού ενδεχομένως σε χιλιάδες υπολογιστές πελατών είναι ένας βασικός λόγος για τη δημοτικότητα τους.

- **Γρήγορη πρόσβαση στην εφαρμογή.**

Καθώς η εφαρμογή είναι δικτυακή επηρεάζεται από την απόδοση του δικτύου. Στόχος είναι η ελαχιστοποίηση των δεδομένων που ανταλλάσσονται μέσω δικτύου ώστε η επικοινωνία να είναι όσο το δυνατόν γρηγορότερη. Όταν μια διεπαφή είναι αργή επηρεάζει αρνητικά τους χρήστες με αποτέλεσμα να μειώνεται σημαντικά η χρηστικότητα της.

- **Ταυτοποίηση.**

Το εκπαιδευτικό λογισμικό που θα αναπτυχθεί κατά βάση θα αποτελεί μια ελεύθερη πλατφόρμα διάχυσης του εκπαιδευτικού υλικού της γραμματικής των αρχαίων ελληνικών Α' Γυμνασίου. Αυτό σημαίνει ότι επιλέχθηκε η ελεύθερη πρόσβαση στις θεωρητικές ενότητες σε οποιονδήποτε επισκέπτη του διαδικτυακού τόπου.

Για να αποκτήσει όμως ένας χρήστης πρόσβαση στις ενότητες των ασκήσεων θα πρέπει να έχει ένα λογαριασμό και να ταυτοποιείται μέσω ενός ζεύγους ονόματος χρήστη και συνθηματικού. Η τακτική αυτή εξασφαλίζει την καταχώρηση των επιδόσεων του κάθε εγγεγραμμένου χρήστη και τη μετέπειτα επισκόπησή τους. Κάθε χρήστης για να έχει πρόσβαση στο περιβάλλον εκτέλεσης ασκήσεων πρέπει πρωτίτερα να έχει ταυτοποιηθεί, δηλαδή να έχει λάβει από το διαχειριστή του συστήματος ένα όνομα (username) και έναν κωδικό (password), τα οποία θα είναι μοναδικά για κάθε χρήστη και τα οποία θα τα χρησιμοποιεί για την εισαγωγή του στο σύστημα διεξαγωγής των ασκήσεων.

2.1.2. Ειδικές Απαιτήσεις

Στόχος της παρούσας εργασίας είναι η δημιουργία ενός ολοκληρωμένου περιβάλλοντος εκμάθησης των θεμάτων της αρχαίας ελληνικής της Α' Γυμνασίου αλλά και της εξάσκησης μέσω ενός συνόλου ασκήσεων που ο μαθητής μπορεί να εκτελέσει βάσει των αντίστοιχων θεωρητικών γνώσεων που έχει λάβει. Στη συνέχεια αναπτύσσεται η απαιτούμενη λειτουργικότητα και οι ειδικές απαιτήσεις του προτεινόμενου συστήματος.

Θεωρία

- Περιβάλλον προβολής θεμάτων θεωρίας σε μορφή διαδικτυακών σελίδων με κανόνες και παραδείγματα του αντίστοιχου κεφαλαίου.
- Δυνατότητα σειριακής πρόσβασης στα κεφάλαια της θεωρίας.
- Δυνατότητα επιλεκτικής μετάβασης σε συγκεκριμένο κεφάλαιο.

Ασκήσεις & είδη ασκήσεων

- Δυνατότητα συμπερίληψης ασκήσεων σε κάθε κεφάλαιο θεωρίας.
- Μετάβαση στις ασκήσεις μέσα από το περιβάλλον μελέτης της θεωρίας.
- Υποστήριξη απεριόριστου αριθμού ασκήσεων ανά κεφάλαιο.
- Κάθε άσκηση θα πρέπει να υποστηρίζει μη προκαθορισμένο αριθμό ερωτήσεων.
- Κάθε ερώτηση θα προσφέρει μη προκαθορισμένο αριθμό επιλογών.
- Θα πρέπει να υποστηρίζονται οι εξής τύποι ασκήσεων:
 - Ασκήσεις απλής επιλογής. Η ερώτηση προσφέρει ένα σύνολο (μη προκαθορισμένου πλήθους) επιλογών και ο χρήστης μπορεί να επιλέξει μόνο μία απάντηση.

ΑΣΚΗΣΗ 2 : Να επισημάνετε τους σωστούς και τους λανθασμένους τύπους σημειώνοντας Χ στο οικείο τετράγωνο.

τοῦ νομοῦ	<input type="radio"/> Σωστό <input type="radio"/> Λάθος
τοῖς νόμοις	<input type="radio"/> Σωστό <input type="radio"/> Λάθος
ταῖς ἄμπελοις	<input type="radio"/> Σωστό <input type="radio"/> Λάθος
οἱ ταύροι	<input type="radio"/> Σωστό <input type="radio"/> Λάθος
τῶν ταυρῶν	<input type="radio"/> Σωστό <input type="radio"/> Λάθος
τοῖς ποτηρίοις	<input type="radio"/> Σωστό <input type="radio"/> Λάθος

Εικόνα 1. Άσκηση απλής επιλογής

- Ασκήσεις πολλαπλής επιλογής. Η ερώτηση προσφέρει ένα σύνολο (μη προκαθορισμένου πλήθους) επιλογών και ο χρήστης μπορεί να επιλέξει όσες απαντήσεις θεωρεί ότι ανταποκρίνονται

ΑΣΚΗΣΗ 1η : Να βάλετε Χ στο τετράγωνο που αντιστοιχεί στον σωστό γραμματικό τύπο:

α) αιτ. ενικού του ονόμ. γλῶσσα : τὴν γλῶσσαν ■

β) γεν. πληθ. του ονόμ. ἀθλητής: τῶν ἀθλητῶν ■

γ) γεν. ενικού του ονόμ. θάλασσα : τῆς θάλασσας ■

δ) δοτ. πληθ. του ονόμ. βοήθεια : ταῖς βοηθείαις ■

ε) αιτ. πληθ. του ονόμ. μῦθα : τὰς μῦθας ■

Εικόνα 2. Άσκηση πολλαπλής επιλογής

- Συμπλήρωση κενού. Το ερώτημα έχει κενές θέσεις σε μια πρόταση, οι οποίες συμπληρώνονται από μία λίστα πιθανών τιμών.

ΑΣΚΗΣΗ 1η: Να συμπληρώσετε τα κενά των παρακάτω προτάσεων επιλέγοντας τον κατάλληλο τύπο του εἰμί από τον πίνακα δεξιά.

α) Θεῖος περι τούς λόγους, ὃ Φαῖδρε.

β) ὑμεῖς ἐκ τοῦ κόσμου τούτου .

γ) ἡμεῖς πάντες φίλοι καὶ σύμμαχοι.

δ) σὺ ὁ βασιλεὺς τῶν Ἰουδαίων.

ε) οὗτός ὁ υἱός μου ὁ ἀγαπητός.

Εικόνα 3. Άσκηση συμπλήρωσης κενών

Γραφική απεικόνιση αποτελεσμάτων

- Τα αποτελέσματα θα πρέπει να αναπαρίστανται και με αριθμητικές τιμές (λάθη / σωστά).
- Για την πιο εποπτική παρουσίασή τους θα πρέπει να προσφέρεται γραφική αναπαράστασή τους με χρήση ραβδογραμμάτων.

2.2. Μοντελοποίηση απαιτήσεων

Τα παραδοτέα της φάσης της ανάλυσης του συστήματος περιγράφουν τις ιδιότητες που πρέπει να έχει αυτό για να ικανοποιεί τις πληροφοριακές ανάγκες των δυνητικών χρηστών του. Η φάση του σχεδιασμού που ακολουθεί επεξεργάζεται τον τρόπο που τεχνολογικά το σύστημα θα εκπληρώσει τους στόχους που τέθηκαν νωρίτερα.

Στη φάση της σχεδίασης συστήματος περιγράφονται με λεπτομέρεια οι προδιαγραφές του συστήματος, με τις οποίες θα εξασφαλίζεται η λειτουργικότητα που προσδιορίστηκε κατά την ανάλυσή του.

Για τη συστηματική καταγραφή των σχεδιαστικών παραμέτρων του συστήματος θα χρησιμοποιηθεί η γλώσσα μοντελοποίησης UML (Rumbaugh, Jacobson, & Booch, 1999). Με την γλώσσα μοντελοποίησης UML αναλύεται και σχεδιάζεται το παρόν εκπαιδευτικό λογισμικό. Από τα 9 βασικά διαγράμματα επιλέγονται τα use case και τα activity διαγράμματα, αφού πρώτα περιγραφούν οι ρόλοι των βασικών ενεργοποιών/actors.

Στο διάγραμμα use case (Διάγραμμα 1) απεικονίζεται το συνολικό σύστημα, οι λειτουργικές ενότητες του καθώς και οι διεπαφές των διαφορετικών κατηγοριών χρηστών με την καθεμιά ενότητα.

2.2.1. Διάγραμμα περιπτώσεων χρήσης

Περιγραφή Ενεργοποιών/actors



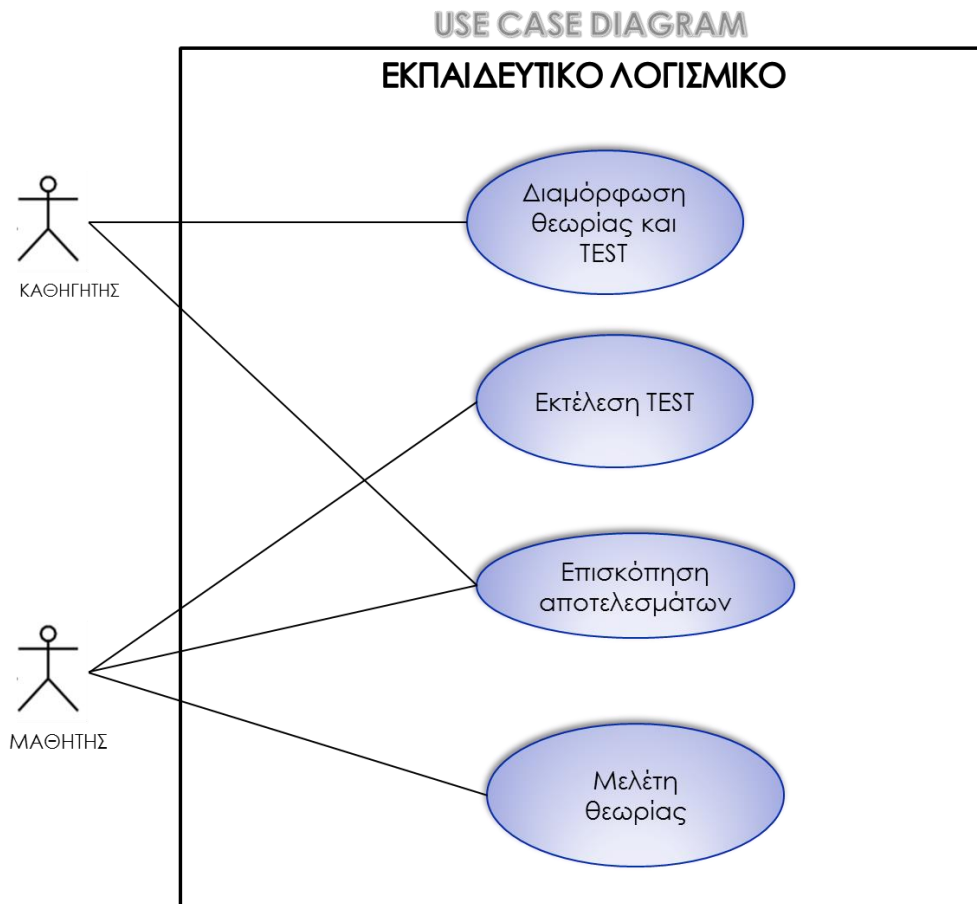
ΚΑΘΗΓΗΤΗΣ

Ο καθηγητής είναι ο διαχειριστής του συστήματος δηλαδή αυτός που διαμορφώνει κάθε φορά το θεωρητικό υλικό που θα προβάλλεται στον μαθητή καθώς και τα Test. Τέλος, έχει τη δυνατότητα επισκόπησης των αποτελεσμάτων τόσο σε ατομικό όσο και σε στατιστικό επίπεδο για να παρακολουθεί την πρόοδο των μαθητών.



ΜΑΘΗΤΗΣ

Ο μαθητής είναι ο εκάστοτε χρήστης του συστήματος που μπαίνει για να διαβάσει το θεωρητικό υλικό της γραμματικής και αν κάνει login ενεργοποιεί τη δυνατότητά του να κάνει τα Test κάθε ενότητας. Τέλος, μπορεί να παρακολουθήσει τα αποτελέσματα των Test που θα κάνει και έτσι να εξάγει συμπεράσματα για την πρόδο του.



Διάγραμμα 1. Διάγραμμα περίπτωσης χρήσης του συνολικού συστήματος

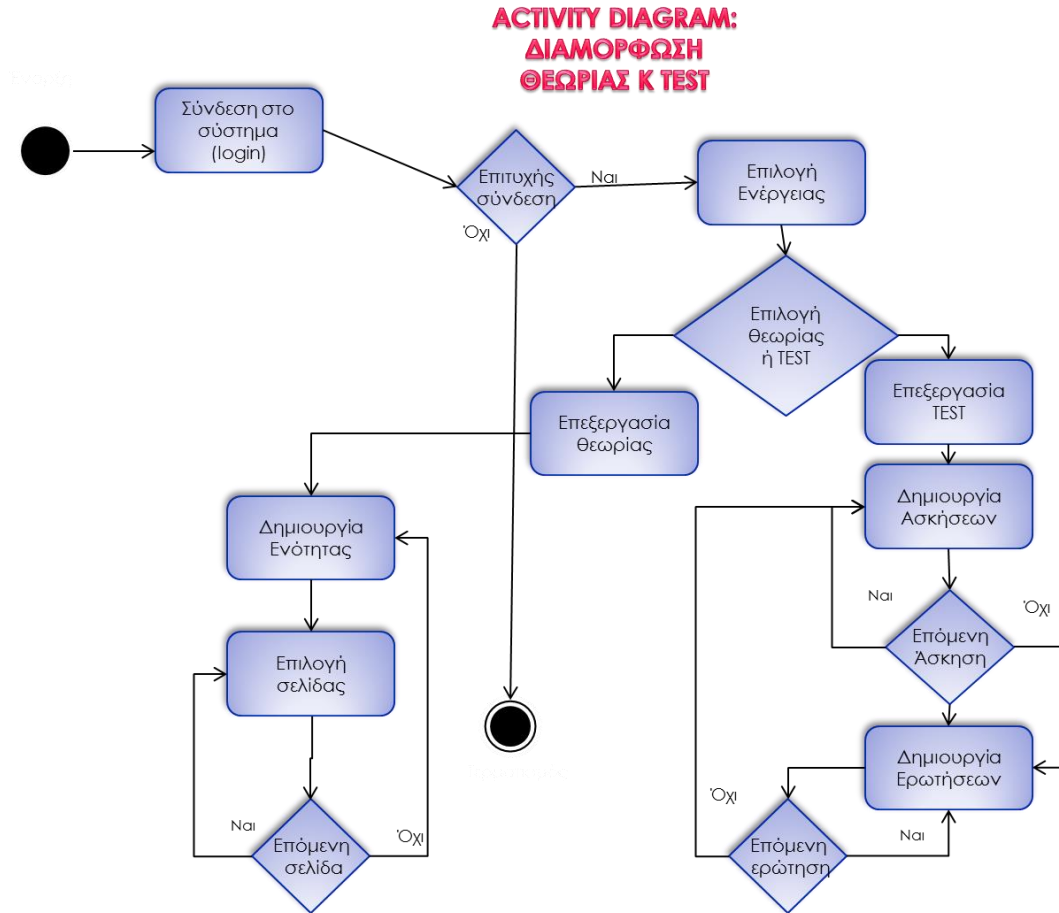
2.2.2. Διαγράμματα δραστηριοτήτων

Πρόκειται για τα διαγράμματα δραστηριοτήτων στα οποία αναπαριστάται η συμπεριφορά μιας λειτουργίας ως ένα σύνολο από διαδοχικές ενέργειες. Εδώ, οι βασικές δραστηριότητες του συστήματος αναπαρίστανται με στρογγυλεμένο ορθογώνιο ενώ οι αποφάσεις με ρόμβο.

Οι κύριες δραστηριότητες που μοντελοποιούνται είναι:

1. Διαμόρφωση θεωρίας και τεστ
2. Μελέτη θεωρίας
3. Εκτέλεση τεστ
4. Επισκόπηση αποτελεσμάτων

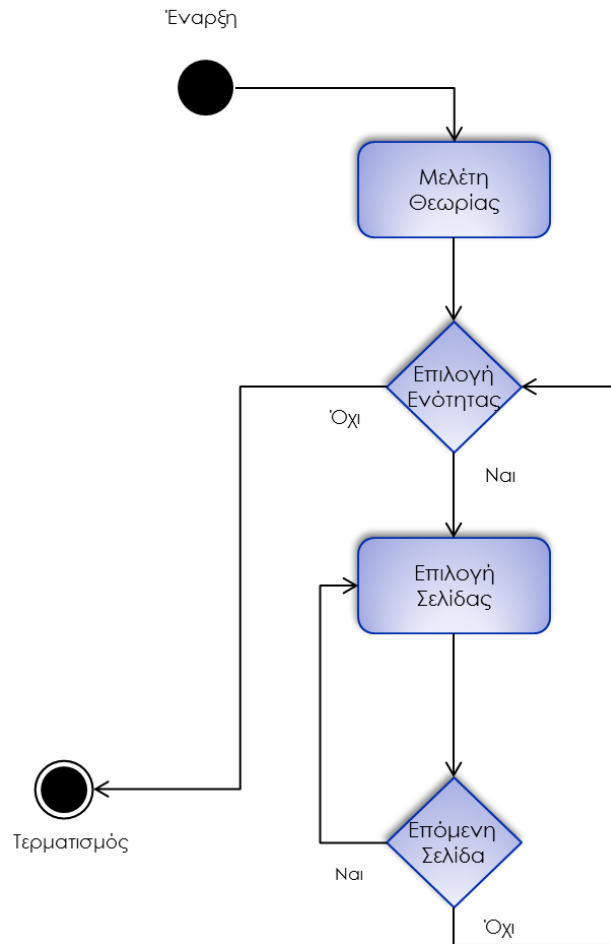
1. Διαμόρφωση θεωρίας και τεστ



Διάγραμμα 2. Διάγραμμα δραστηριοτήτων διαμόρφωσης θεωρίας και τεστ

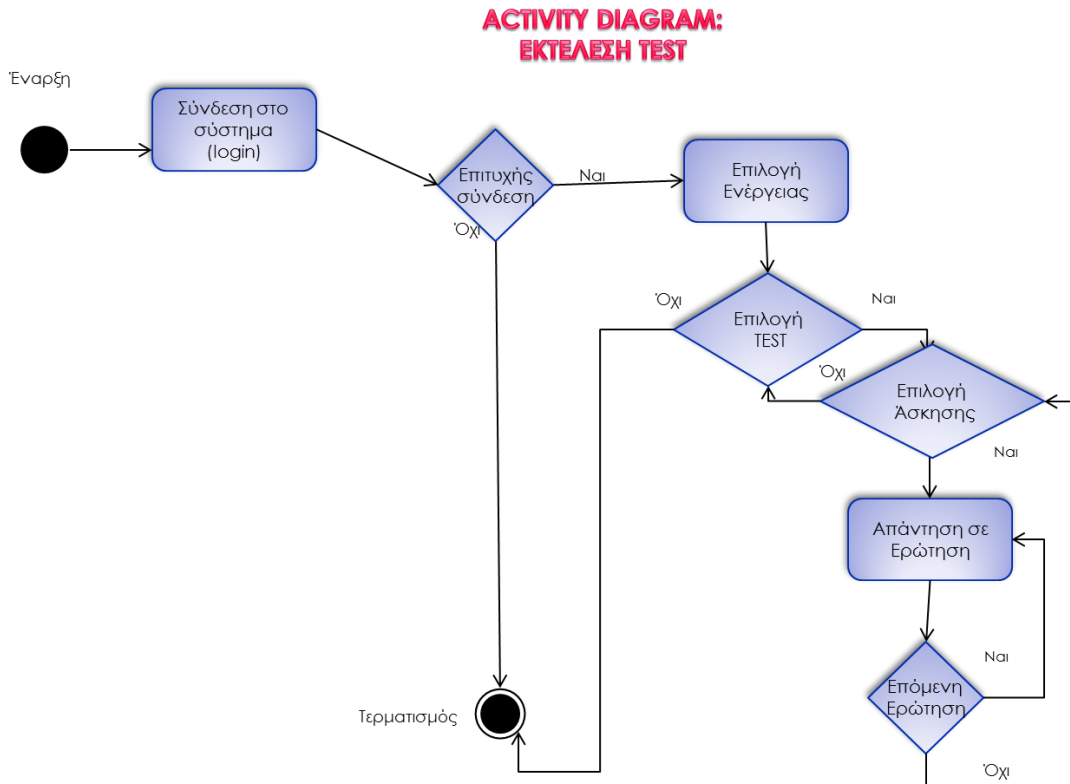
2. Μελέτη θεωρίας

ACTIVITY DIAGRAM: ΜΕΛΕΤΗ ΘΕΩΡΙΑΣ



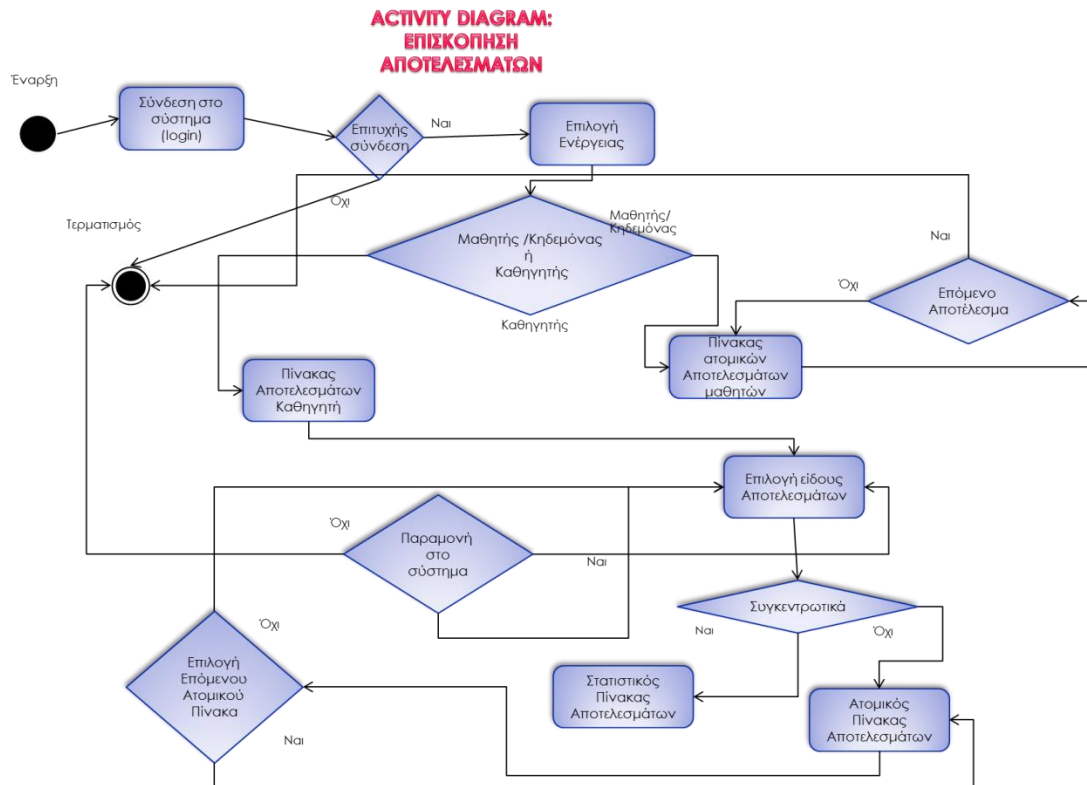
Διάγραμμα 3. Διάγραμμα δραστηριοτήτων μελέτης θεωρίας

3. Εκτέλεση τεστ



Διάγραμμα 4. Διάγραμμα δραστηριοτήτων εκτέλεσης τεστ

4. Επισκόπηση αποτελεσμάτων



Διάγραμμα 5. Διάγραμμα δραστηριοτήτων επισκόπησης αποτελεσμάτων

2.3. Σχεδιασμός βάσης δεδομένων

Για τον σχεδιασμό της βάσης δεδομένων χρησιμοποιήθηκε αρχικά το μοντέλο οντοτήτων-συσχετίσεων (Entity-Relationship - ER) για τη μοντελοποίηση του προβλήματος και την περιγραφή της δομής των δεδομένων του συστήματος με ένα απλό και σαφή τρόπο. Στη συνέχεια το μοντέλο οντοτήτων-συσχετίσεων μετατρέπεται στο σχεσιακό μοντέλο που αντιστοιχεί.

2.3.1. Εννοιολογικό μοντέλο

Οι οντότητες της εφαρμογής μαζί με τα χαρακτηριστικά τους, όπως προκύπτουν από τις λειτουργικές απαιτήσεις του συστήματος, παρουσιάζονται στους Πίνακες 1 έως 6 που ακολουθούν.

Πίνακας 1. Η οντότητα “User”

Όνομα οντότητας	User
Περιγραφή	Διατηρεί τα στοιχεία των εγγεγραμμένων χρηστών
Σχεσιακός πίνακας	Users
Ιδιότητες	
Usr_ID	Πρωτεύον κλειδί. Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)
Usr_Username	Μοναδικό αναγνωριστικό όνομα χρήστη. Χαρακτηρίζεται ως UNIQUE στη βάση δεδομένων και δεν

	επιτρέπει κενές τιμές (Not Null)
Usr_Password	Συνθηματικό. Δεν επιτρέπει κενές τιμές (Not Null)
Usr_Lastname	Επώνυμο χρήστη
Usr_Firstname	Όνομα χρήστη
Usr_Email	Διεύθυνση ηλεκτρονικού ταχυδρομείου. Για μελλοντική χρήση και αυτοματοποιημένη αποστολή ειδοποιήσεων, ασκήσεων και αποτελεσμάτων
Usr_Phone	Τηλέφωνο χρήστη
Usr_Mobile	Κινητό τηλέφωνο χρήστη. Για μελλοντική χρήση και αυτοματοποιημένη αποστολή SMS ειδοποιήσεων, ασκήσεων και αποτελεσμάτων
Usr_Role_Admin	Παίρνει τιμή True εάν ο χρήστης είναι εκπαιδευτικός
Usr_Role_Contributor	Παίρνει τιμή True εάν ο χρήστης είναι μαθητής
Usr_Status	Για μελλοντική χρήση. Δηλώνει εάν ένας χρήστης είναι ενεργός ή ανενεργός.

Πίνακας 2. Η οντότητα “Exercise”

Όνομα οντότητας	Exercise
Περιγραφή	Διατηρεί τις εκφωνήσεις των ασκήσεων
Σχεσιακός πίνακας	Exercises
Ιδιότητες	
exe_id	Πρωτεύον κλειδί. Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)
exe_text	Λεκτικό της εκφώνησης
exe_image	Εικόνα συνδεδεμένη με την άσκηση.
exe_unit	Εκπαιδευτική ενότητα στην οποία ανήκει η άσκηση
exe_class	Κλάση της άσκησης. Παίρνει τις τιμές: <ul style="list-style-type: none"> • 1: Απλής επιλογής • 2: Πολλαπλής επιλογής • 3: Συμπλήρωσης κενών
exe_sorter	Σειρά εμφάνισης της άσκησης στο τεστ

Πίνακας 3. Η οντότητα “question_text”

Όνομα οντότητας	question_text
Περιγραφή	Διατηρεί τα ερωτήματα μιας άσκησης
Σχεσιακός πίνακας	question_text
Ιδιότητες	
qt_id	Πρωτεύον κλειδί. Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)
qt_exercise	ID της άσκησης στην οποία ανήκει το ερώτημα. Διατηρεί τη σχέση exercise -> question_text
qt_text	Λεκτικό της ερώτησης
qt_sorter	Σειρά εμφάνισης της ερώτησης μέσα στην άσκηση.

Πίνακας 4. Η οντότητα “question_options”

Όνομα οντότητας	question_options
Περιγραφή	Διατηρεί τις επιλογές απάντησης κάθε ερωτήματος της άσκησης
Σχεσιακός πίνακας	question_options
Ιδιότητες	
qo_id	Πρωτεύον κλειδί. Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)
qo_qt_id	ID του ερωτήματος στο οποίο αντιστοιχεί η επιλογή
qo_text	Λεκτικό της επιλογής (π.χ. Σωστό)
qo_correct_answer	Τιμή της σωστής απάντησης για την επιλογή
qo_optionsfile	Αρχείο κειμένου, το οποίο σε διαδοχικές γραμμές περιέχει τις επιλογές για συμπλήρωση κενού σε ασκήσεις κλάσης 3.
qo_shorter	Σειρά εμφάνισης της επιλογής

Πίνακας 5. Η οντότητα “results”

Όνομα οντότητας	results	
Περιγραφή	Διατηρεί τα αποτελέσματα κάθε ερώτησης των ασκήσεων ξεχωριστά	
Σχεσιακός πίνακας	Results	
Ιδιότητες		
res_id	Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)	
res_session_id	Μοναδικός κωδικός του τεστ (απαρτίζεται από την ημερομηνία σύνδεσης, ώρα σύνδεσης και id του χρήστη)	Σύνθετο πρωτεύον κλειδί. Δεν επιτρέπεται η καταχώρηση δύο απαντήσεων για το ίδιο τεστ και την ίδια ερώτηση

res_question_option	Id της επιλογής που απαντάται
res_user_id	Id του χρήστη που εκτελεί το τεστ
res_timestamp	Ημερομηνία και ώρα σύνδεσης του χρήστη στο σύστημα
res_answer	Τιμή της απάντησης του χρήστη
res_exe_class	Κλάση της ερώτησης
res_exe_id	Id της άσκησης στην οποία ανήκει η ερώτηση
res_exe_unit	Ενότητα στην οποία ανήκει η ερώτηση

Πίνακας 6. Η οντότητα “theory”

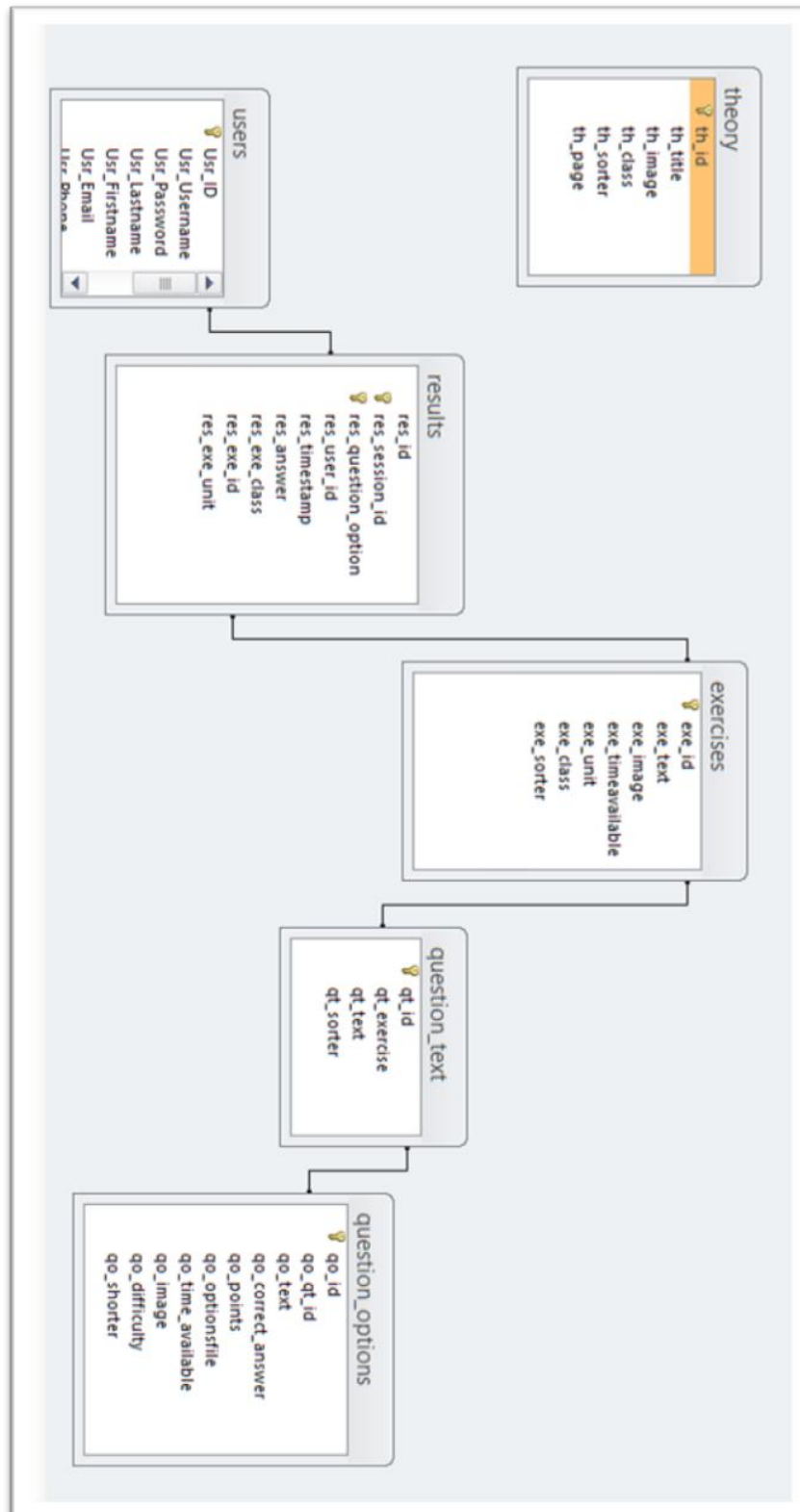
Όνομα οντότητας	theory
Περιγραφή	Διατηρεί τις πληροφορίες για τα κεφάλαια της θεωρίας και τις σείδες που αναπτύσσουν το θεωρητικό μέρος
Σχεσιακός πίνακας	Theory
Ιδιότητες	
th_id	Πρωτεύον κλειδί. Μοναδικός κωδικός ο οποίος παράγεται αυτόματα από τη βάση δεδομένων (auto increment)
th_title	Λεκτικό του τίτλου του κεφαλαίου της θεωρίας
th_image	Αρχείο εικόνας που συνδέεται με το κεφάλαιο θεωρίας
th_class	Ενότητα της θεωρίας. Έχει υλοποιηθεί η ενότητα 1
th_sorter	Σειρά εμφάνισης των κεφαλαίων στο περιβάλλον ανάγνωσης θεωρίας
th_page	Αρχείο html στο οποίο έχει αναπτυχθεί η θεωρία

2.3.2. Σχεσιακό μοντέλο

Στο Διάγραμμα 6, φαίνονται οι πίνακες της βάσης δεδομένων (που αντιστοιχούν στο διάγραμμα Ο/Σ που δόθηκε παραπάνω). Κάθε οντότητα είναι ένας πίνακας και τα χαρακτηριστικά της είναι τα πεδία του πίνακα.

Σε κάθε πίνακα, με το εικονίδιο κλειδιού υποδηλώνεται το πρωτεύον κλειδί. Πρωτεύον κλειδί είναι ένα πεδίο (ή σύνολο πεδίων) στον πίνακα που παρέχει ένα μοναδικό αναγνωριστικό σε κάθε εγγραφή του πίνακα (δηλαδή δεν είναι δυνατό να υπάρχουν δύο εγγραφές με το ίδιο κλειδί). Κάθε βέλος αντιστοιχεί σε συσχέτιση ανάμεσα σε δυο πίνακες.

Αναλυτικότερα δημιουργήθηκαν οι εξής πίνακες:



Διάγραμμα 6. Σχεσιακό μοντέλο

2.3.3. Φυσικό μοντέλο

Για το φυσικό σχεδιασμό, δηλαδή τη δήλωση του σχήματος και τον προσδιορισμό των δεδομένων της βάσης, χρησιμοποιήθηκε η γλώσσα SQL (Structured Query Language).

Για τη δημιουργία των πινάκων εκτελέστηκαν οι αντίστοιχες SQL δηλώσεις, οι οποίες παρουσιάζονται στις παρακάτω ενότητες κώδικα.

Κώδικας 1. Πρόταση SQL για δημιουργία του πίνακα “Users”

```
CREATE TABLE `users` (  
  `Usr_ID` int(10) NOT NULL AUTO_INCREMENT,  
  `Usr_Username` varchar(20) NOT NULL,  
  `Usr_Password` varchar(20) NOT NULL,  
  `Usr_Lastname` varchar(20) DEFAULT NULL,  
  `Usr_Firstname` varchar(20) DEFAULT NULL,  
  `Usr_Email` varchar(30) DEFAULT NULL,  
  `Usr_Phone` varchar(20) DEFAULT NULL,  
  `Usr_Mobile` varchar(20) DEFAULT NULL,  
  `Usr_Role_Admin` varchar(5) DEFAULT NULL,  
  `Usr_Role_Contributor` varchar(5) DEFAULT NULL,  
  `Usr_Status` varchar(10) DEFAULT NULL,  
  PRIMARY KEY (`Usr_ID`),  
  UNIQUE KEY `Usr_Username_UNIQUE` (`Usr_Username`)  
) ENGINE=InnoDB AUTO_INCREMENT=22 DEFAULT CHARSET=utf8;
```

Κώδικας 2. Πρόταση SQL για δημιουργία του πίνακα “exercises”

```
CREATE TABLE `exercises` (  
  `exe_id` int(10) NOT NULL AUTO_INCREMENT,  
  `exe_text` varchar(255) NOT NULL,  
  `exe_image` text,  
  `exe_unit` int(11) DEFAULT NULL,  
  `exe_class` int(11) DEFAULT NULL,  
  `exe_sorter` varchar(2) DEFAULT NULL,  
  PRIMARY KEY (`exe_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=29 DEFAULT CHARSET=utf8;
```

Κώδικας 3. Πρόταση SQL για δημιουργία του πίνακα “question_text”

```
CREATE TABLE `question_text` (  
  `qt_id` int(11) NOT NULL AUTO_INCREMENT,  
  `qt_exercise` int(11) NOT NULL,  
  `qt_text` varchar(255) DEFAULT NULL,  
  `qt_sorter` varchar(2) DEFAULT NULL,  
  PRIMARY KEY (`qt_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=175 DEFAULT CHARSET=utf8;
```

Κώδικας 4. Πρόταση SQL για δημιουργία του πίνακα “question_options”

```
CREATE TABLE `question_options` (  
  `qo_id` int(10) NOT NULL AUTO_INCREMENT,  
  `qo_qt_id` int(10) DEFAULT NULL,  
  `qo_text` varchar(100) DEFAULT NULL,  
  `qo_correct_answer` varchar(20) DEFAULT NULL,  
  `qo_optionsfile` varchar(45) DEFAULT NULL,  
  `qo_shorter` varchar(2) DEFAULT NULL,  
  PRIMARY KEY (`qo_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=358 DEFAULT CHARSET=utf8;
```


Κώδικας 5. Πρόταση SQL για δημιουργία του πίνακα “theory”

```
CREATE TABLE `theory` (  
  `th_id` int(10) NOT NULL AUTO_INCREMENT,  
  `th_title` varchar(200) NOT NULL,  
  `th_image` varchar(200) NOT NULL,  
  `th_class` int(10) NOT NULL,  
  `th_sorter` int(11) DEFAULT NULL,  
  `th_page` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`th_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8;
```

Κώδικας 6. Πρόταση SQL για δημιουργία του πίνακα “results”

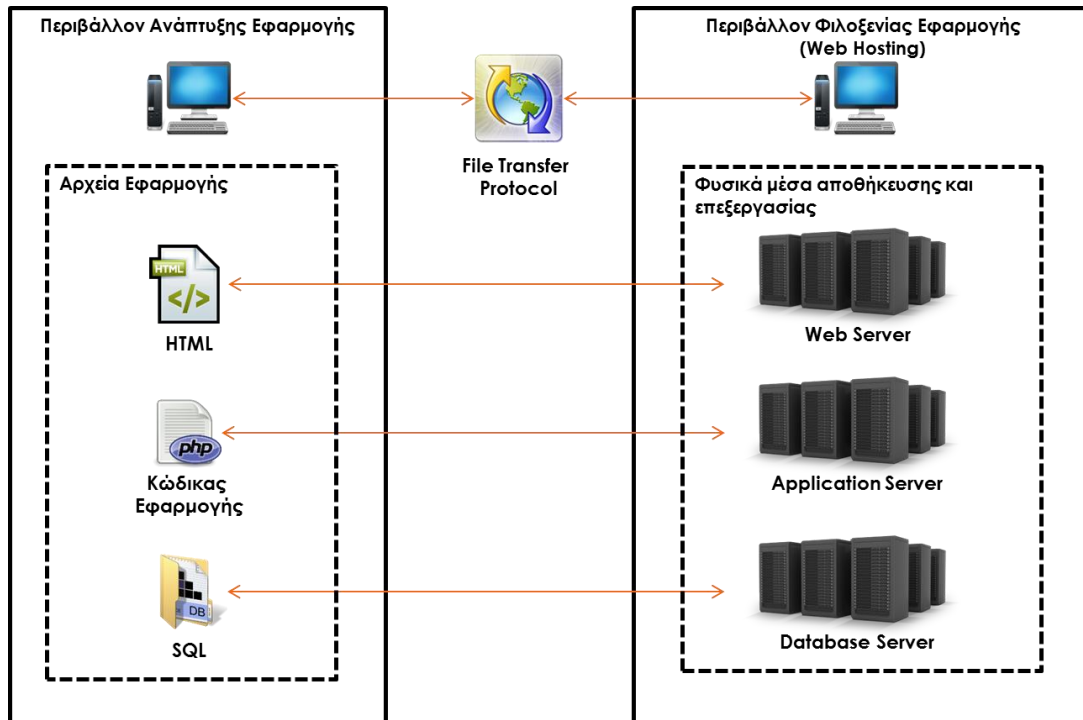
```
CREATE TABLE `results` (  
  `res_id` int(11) NOT NULL AUTO_INCREMENT,  
  `res_session_id` varchar(25) NOT NULL,  
  `res_question_option` int(11) NOT NULL,  
  `res_user_id` int(11) DEFAULT NULL,  
  `res_timestamp` datetime DEFAULT NULL,  
  `res_answer` varchar(45) DEFAULT NULL,  
  `res_exe_class` int(11) DEFAULT NULL,  
  `res_exe_id` int(11) DEFAULT NULL,  
  `res_exe_unit` int(11) DEFAULT NULL,  
  PRIMARY KEY (`res_session_id`, `res_question_option`),  
  UNIQUE KEY `res_id_UNIQUE` (`res_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=299 DEFAULT CHARSET=utf8;
```

3. Υλοποίηση εφαρμογής

Στο κεφάλαιο αυτό γίνεται αναλυτική παρουσίαση της υλοποίησης της εφαρμογής.

3.1. Αρχιτεκτονική 3-επιπέδων

Για την υλοποίηση του συστήματος θα εφαρμοστεί η αρχιτεκτονική 3 επιπέδων (3-tier architecture) (Liu, Heo, & Sha, 2009). Το αρχιτεκτονικό μοντέλο περιγράφεται αναλυτικά στην εικόνα που ακολουθεί.



Εικόνα 4. Η αρχιτεκτονική τριών επιπέδων του προτεινόμενου συστήματος

Δεδομένων των προοπτικών εφαρμογής ενός αντίστοιχου συστήματος στο διαδίκτυο, η προτεινόμενη αρχιτεκτονική προσφέρει ένα σύνολο πλεονεκτημάτων:

1. Η διαχείριση των δεδομένων είναι ανεξάρτητη από τον φυσικό τρόπο αποθήκευσης των δεδομένων (συστήματα βάσεων δεδομένων).
2. Ανεξαρτησία από το γραφικό περιβάλλον διεπαφής. Είναι σχετικά απλή η διαδικασία μετάβασης σε επιπλέον πλατφόρμες παρουσίασης των εφαρμογών (κινητά τηλέφωνα, tablets, σύγχρονες τηλεοπτικές συσκευές, κ.λπ.) Επιπρόσθετα η χρήση cascading stylesheets (CSS) επιτρέπει την εύκολη τροποποίηση της εμφάνισης των σελίδων του λογισμικού, χωρίς να απαιτείται η παραμικρή αλλαγή στον κώδικα των σελίδων.
3. Ευελιξία στην τροποποίηση του επιπέδου παρουσίασης των πληροφοριών, ενώ τα επίπεδα επιχειρησιακής λογικής και διαχείρισης δεδομένων παραμένουν αμετάβλητα.
4. Είναι δυνατή η απασχόληση ομάδων εργασίας με τελείως διαφορετικές εξειδικεύσεις κατά τη διάρκεια της ανάπτυξης του συστήματος (για παράδειγμα ειδικοί στις βάσεις δεδομένων αναπτύσσουν το σχήμα της βάσης, ενώ ταυτόχρονα σχεδιαστές εφαρμογών διαδικτύου σχεδιάζουν και υλοποιούν τις σελίδες των διεπαφών).
5. Τα αποθηκευμένα δεδομένα είναι πιο ασφαλή, δεδομένου ότι η πρόσβαση σε αυτά γίνεται έμμεσα μέσω συνόδων HTTP (οι οποίες έχουν την ιδιότητα να είναι connectionless).

6. Κλιμάκωση. Κάθε επίπεδο της προτεινόμενης αρχιτεκτονικής μπορεί να αναβαθμίζεται ανεξάρτητα από τα υπόλοιπα ανάλογα με το μοντέλο χρήσης των εφαρμογών και τον αντίστοιχο φόρτο που προκαλείται.
7. Βελτιωμένες επιδόσεις. Το επίπεδο παρουσίασης (presentation layer) προσφέρει μηχανισμούς caching για την ελαχιστοποίηση χρήσης του δικτύου για μεταφορά δεδομένων που χρησιμοποιούνται επαναλαμβανόμενα. Επίσης παρέχεται η δυνατότητα ανεξάρτητης αναβάθμισης των άλλων δύο επιπέδων της αρχιτεκτονικής με περισσότερους διακομιστές και εφαρμογή πολιτικών κατανομής υπολογιστικού φορτίου (load balancing).
8. Εύκολη εγκατάσταση των εφαρμογών στις εγκαταστάσεις του παρόχου.
9. Αξιοποίηση των πολιτικών ασφάλειας που προσφέρει ο πάροχος των υπηρεσιών φιλοξενίας των εφαρμογών.

Στη συνέχεια περιγράφονται τα τεχνολογικά επίπεδα που απαρτίζουν το εκπαιδευτικό λογισμικό.

3.1.1. Επίπεδο δεδομένων

Το επίπεδο δεδομένων περιλαμβάνει τον διακομιστή βάσεων δεδομένων όπου αποθηκεύονται οι πληροφορίες του εκπαιδευτικού λογισμικού και τα δεδομένα των χρηστών. Τα δεδομένα αποθηκεύονται και διατηρούνται στους διακομιστές του επιπέδου δεδομένων ανεξάρτητα από τους διακομιστές εφαρμογών ή τις εφαρμογές υλοποίησης της επιχειρησιακής λογικής. Το επίπεδο δεδομένων τυπικά αποθηκεύει τα δεδομένα της εφαρμογής με ασφάλεια και διεκπεραιώνει τις δοσοληψίες αναζήτησης και διάθεσης των πληροφοριών στα άλλα επίπεδα του μοντέλου τριών επιπέδων που χρησιμοποιείται.

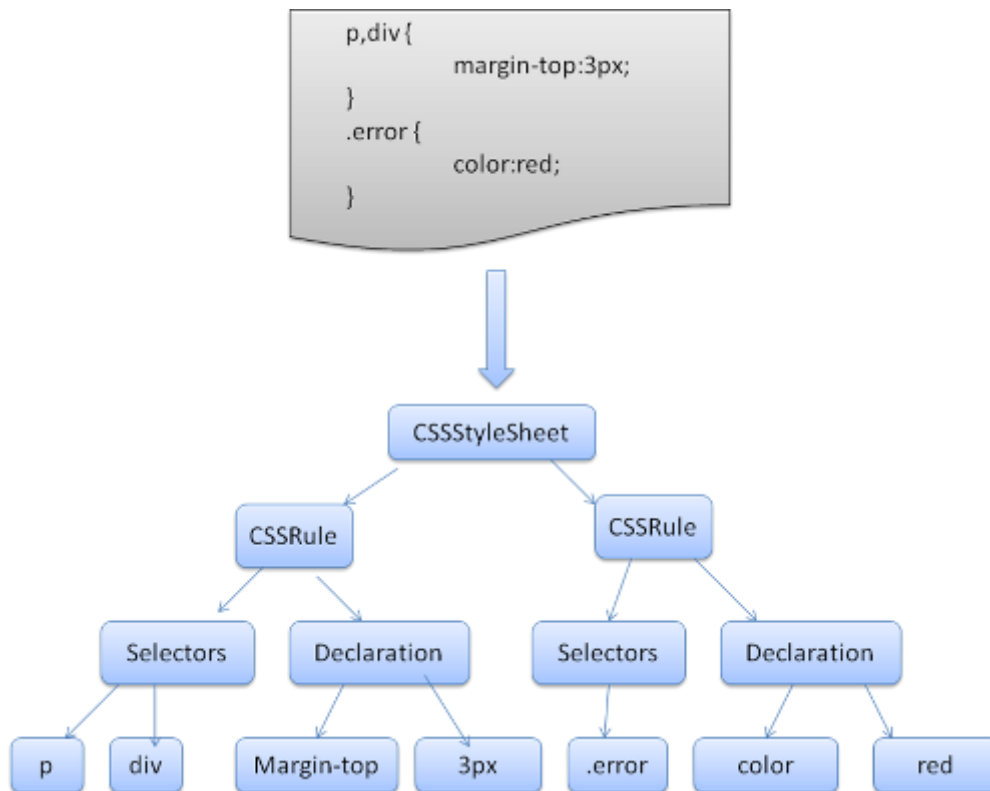
3.1.2. Επίπεδο επιχειρησιακής λογικής

Το επίπεδο επιχειρησιακής λογικής υλοποιήθηκε μέσω ενός συνόλου σελίδων PHP, οι οποίες εκτελούνται στον application server του συστήματος. Ο μηχανισμός της τεχνολογίας PHP προσφέρει την εκτέλεση του κώδικα στο διακομιστή εφαρμογών και τη δυναμική διαμόρφωση των HTML σελίδων που διατίθενται στον φυλλομετρητή του χρήστη της εφαρμογής. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι η PHP είναι τεχνολογία διακομιστή, με αποτέλεσμα οι επεξεργαστικές διεργασίες να διεκπεραιώνονται στην πλευρά του διακομιστή. Επίσης η πλευρά του χρήστη δεν απαιτεί κανένα πρόσθετο λογισμικό (π.χ. plugin ή runtime (όπως η java) για να εκτελεστεί η εφαρμογή.

3.1.3. Επίπεδο παρουσίασης

Για την παρουσίαση των τελικών HTML σελίδων που δημιουργούνται μέσω του κώδικα PHP θα χρησιμοποιηθεί ο HTTP server Microsoft Internet Information Server 8. Πρόκειται για έναν διαδομένο και ευρύτατα αποδεκτό διακομιστή HTTP ο οποίος χρησιμοποιείται από όλους σχεδόν τους παρόχους διαδικτυακής φιλοξενίας εφαρμογών. Για την επιλογή του έπαιξε σημαντικό ρόλο η μεγάλη γνωσιακή βάση υποστήριξης του συγκεκριμένου διακομιστή, όπως επίσης και η ολοκλήρωσή του στο περιβάλλον ανάπτυξης.

Η εφαρμογή χρησιμοποιεί HTML5 για την περιγραφή του περιεχομένου, ενώ για τη μορφοποίηση των σελίδων που διαβιβάζονται στο φυλλομετρητή για εμφάνιση χρησιμοποιείται τεχνολογία Cascading Stylesheets. Χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML. Η CSS είναι μια γλώσσα προορισμένη να προσδιορίζει τα χαρακτηριστικά εμφάνισης μιας ιστοσελίδας, διαμορφώνοντας χαρακτηριστικά όπως γραμματοσειρές, χρώματα, στοίχιση, τοποθέτηση στοιχείων στη σελίδα, κλπ.



Εικόνα 5.Επεξεργασία tags της HTML μέσω κανόνων CSS

3.2. Τεχνολογίες υλοποίησης

Η ανάπτυξη της εφαρμογής βασίστηκε σε σύγχρονες τεχνολογίες. Στόχος ήταν η δημιουργία μιας απλής και εύχρηστης εφαρμογής, η οποία να προσφέρει ένα απολύτως εποπτικό και εργονομικό περιβάλλον χρήσης και να απαιτεί την ελάχιστη αλληλεπίδραση για την εκτέλεση των κοινών εργασιών.

Οι τεχνολογίες που χρησιμοποιήθηκαν παρουσιάζονται αναλυτικά στη συνέχεια:

HTML

Η HTML είναι το ακρωνύμιο για το HyperText Markup Language (γλώσσα μορφοποίησης υπερκειμένου) και είναι η βασική γλώσσα δόμησης σελίδων του World Wide Web (ή απλά ιστού, Web) (Pallud & Straub, 2014).

Η HTML έχει σχεδιαστεί με σκοπό:

- Την στοιχειοθέτηση (μορφοποίηση - formatting) ηλεκτρονικών κειμένων.
- Την αναμετάδοση τους στο Διαδίκτυο.
- Την αναπαράστασή τους σε διαφορετικού είδους οθόνες.

Η περιγραφή της στοιχειοθέτησης του κειμένου γίνεται με την εισαγωγή ετικετών της HTML στο σώμα του εγγράφου. Οι περισσότερες ετικέτες της HTML χρησιμοποιούνται για την οργάνωση του περιεχομένου σε μια λογική - ιεραρχική δομή. Υπάρχουν ωστόσο και καθαρά μορφολογικοί χαρακτήρες.

Η HTML επιτρέπει την ενσωμάτωση υπερσυνδέσμων (links, δεικτών) προς άλλα κείμενα αλλά και προς αρχεία ήχου, εικόνων, κινούμενων σχεδίων, κλπ. Έτσι, μπορεί να χρησιμοποιηθεί για τον σχεδιασμό και την ανάπτυξη εφαρμογών υπερμέσων. Ένα άλλο χαρακτηριστικό της HTML είναι ότι η γλώσσα αυτή είναι επεκτάσιμη, υπό την έννοια ότι σε αυτή μπορούν να προστεθούν νέες ιδιότητες και λειτουργίες χωρίς να ακυρώνονται υπερκείμενα ανεπτυγμένα σε παλαιότερες εκδόσεις της.

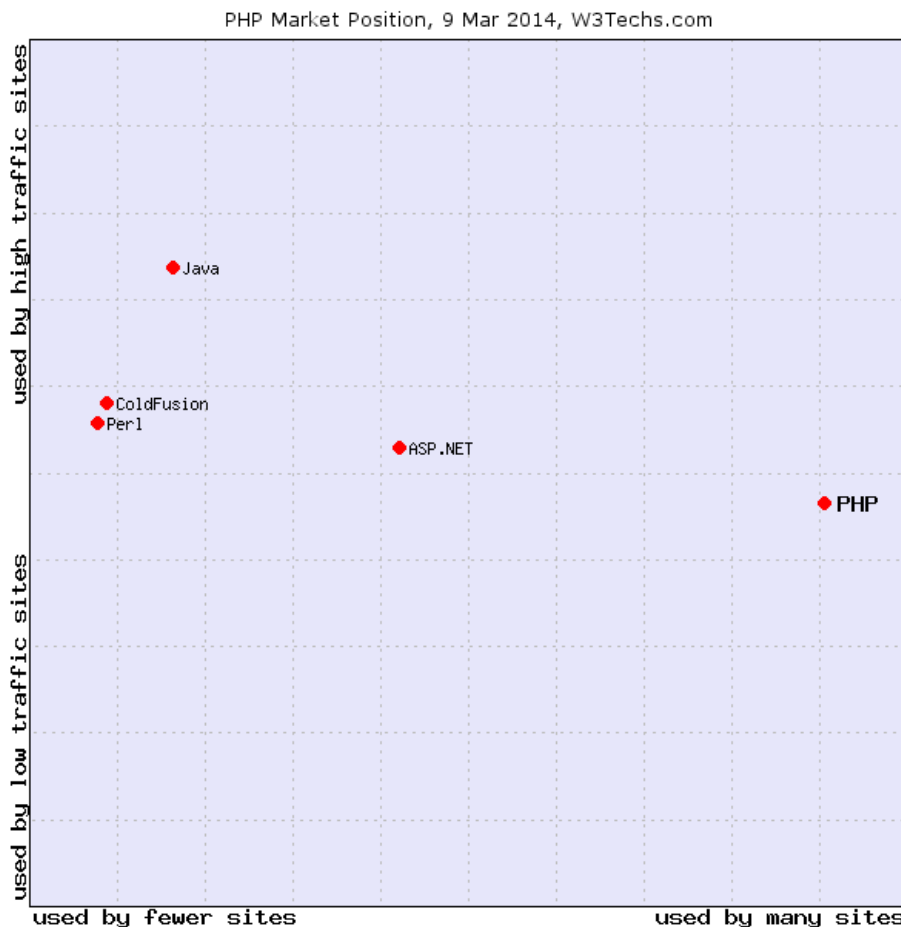
PHP

Το επίπεδο επιχειρησιακής λογικής θα υλοποιηθεί μέσω ενός συνόλου σελίδων php, οι οποίες εκτελούνται στον application server του συνολικού συστήματος. Ο μηχανισμός της τεχνολογίας php προσφέρει την εκτέλεση του κώδικα στο διακομιστή εφαρμογών και τη δυναμική διαμόρφωση των HTML σελίδων που διατίθενται στον φυλλομετρητή του χρήστη της εφαρμογής. Το πλεονέκτημα αυτής της προσέγγισης είναι ότι η php είναι τεχνολογία διακομιστή, με αποτέλεσμα οι επεξεργαστικές διεργασίες να διεκπεραιώνονται στην πλευρά του διακομιστή. Επίσης η πλευρά του χρήστη δεν απαιτεί κανένα πρόσθετο λογισμικό (π.χ. plugin ή runtime (όπως η java) για να εκτελεστεί η εφαρμογή.

Επίσης η php είναι μια απλοποιημένη γλώσσα με πολύ σύντομο κύκλο εκμάθησης, γεγονός που διευκολύνει και επιταχύνει την ανάπτυξη της εφαρμογής. Επίσης προσφέρει ολοκληρωμένο σύνολο εντολών διεπαφής με διακομιστές βάσεων δεδομένων, γεγονός που διευκολύνει ιδιαίτερα στην περίπτωση του εκπαιδευτικού λογισμικού, όπου θα υπάρχει έντονη συναλλαγή με το σύστημα διαχείρισης βάσεων δεδομένων.

Επίσης η php είναι ανεξάρτητη από το λειτουργικό σύστημα που χρησιμοποιείται στο διακομιστή της εγκατάστασης: λειτουργεί σε UNIX, Mac και Windows servers.

Η php δεν απαιτεί αυξημένους υπολογιστικούς πόρους του διακομιστή. Αυτό την καθιστά μια γρήγορη γλώσσα στην εκτέλεση του προγράμματος χωρίς να επιβραδύνει τις υπόλοιπες διεργασίες του διακομιστή. Επίσης η php είναι ευρύτατα αποδεκτή ως σταθερή γλώσσα η οποία χρησιμοποιείται κατά κόρον για ανάπτυξη εφαρμογών βάσεων δεδομένων. Το Διάγραμμα 7 αποτυπώνει τη συγκριτική θέση προτίμησης της php στο σύνολο των προγραμματιστικών περιβαλλόντων διακομιστή που διατίθενται σήμερα.



Διάγραμμα 7. Η θέση της PHP μεταξύ των γλωσσών ανάπτυξης εφαρμογών διακομιστή (server-side technologies) (Κάτοχος εικόνας: W3Techs – Web Technology Surveys)

Ένα σημαντικό χαρακτηριστικό της ρηρ είναι ότι υλοποιεί πολλαπλά επίπεδα ασφάλειας για την προστασία των εφαρμογών από κακόβουλες επιθέσεις.

Επίσης η ρηρ προσφέρει σημαντικές δυνατότητες διασύνδεσης εφαρμογών. Χρησιμοποιεί ένα αρθρωτό σύστημα επεκτάσεων μέσω βιβλιοθηκών για το χειρισμό γραφικών, δεδομένων XML, κρυπτογραφίας, κ.ο.κ.

Ιδιαίτερα σημαντικό σε σχέση με το έργο του εκπαιδευτικού λογισμικού είναι το χαρακτηριστικό που δίνει η ρηρ για διασύνδεση με συστήματα βάσεων δεδομένων. Διατίθενται διεπαφές για συστήματα βάσεων δεδομένων MySQL, Microsoft SQL Server, Oracle, Informix, Postgres, κ.α. Επίσης η ρηρ είναι συμβατή και εκτελείται στην πλειονότητα των application servers (Microsoft Internet Information Server, Apache, THHTTPD, AOLServer, κ.α.).

Javascript

Η JavaScript είναι γλώσσα προγραμματισμού η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του πελάτη (client-side) σε ιστοσελίδες (Pallud & Straub, 2014). Η βασική ιδέα πίσω από την JavaScript ήταν η δημιουργία μιας γλώσσας, η οποία θα μπορούσε να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών που θα εκτελούνταν από την πλευρά του πελάτη, χωρίς να παρουσιάζει την ίδια πολυπλοκότητα με τη γλώσσα προγραμματισμού Java.

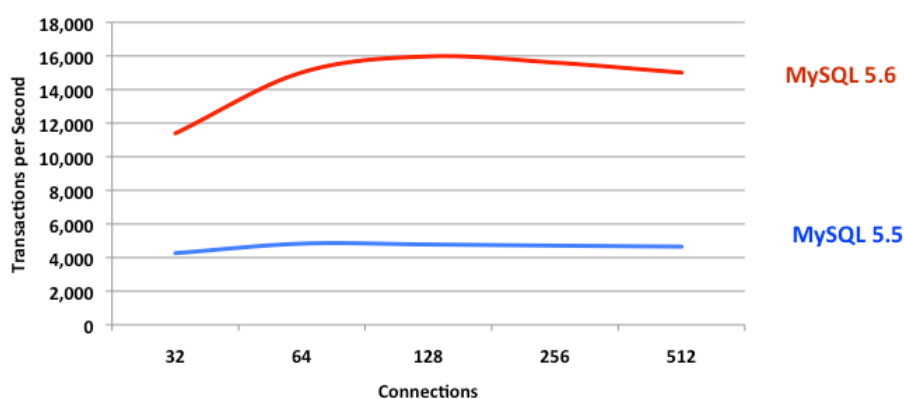
Η Javascript είναι μια σχετικά απλή γλώσσα κατάλληλη για την υλοποίηση απλών εργασιών, οι οποίες εκτελούνται για μικρό χρονικό διάστημα. Μερικές από τις εργασίες που μπορούν να υλοποιηθούν με τη χρήση της συγκεκριμένης γλώσσας είναι οι εξής:

- Η προσθήκη μηνυμάτων που κυλούν ή αλλάζουν στη γραμμή κατάστασης του φυλλομετρητή.
- Η επικύρωση των περιεχομένων μιας φόρμας πριν αυτά υποβληθούν για περαιτέρω επεξεργασία.
- Η αναγνώριση του είδους και της έκδοσης του φυλλομετρητή που χρησιμοποιείται ώστε το ίδιο περιεχόμενο να εμφανίζεται με διαφορετικό τρόπο ανάλογα με το φυλλομετρητή.
- Η εμφάνιση μηνυμάτων στο χρήστη είτε ως μέρος μιας ιστοσελίδας είτε σαν πλαίσια προειδοποίησης.
- Η ανίχνευση των εγκατεστημένων πρόσθετων στοιχείων (plug-ins) και η επισήμανση στο χρήστη σχετικά με ποιο plug-in απαιτείται για την εκτέλεση μιας εφαρμογής.

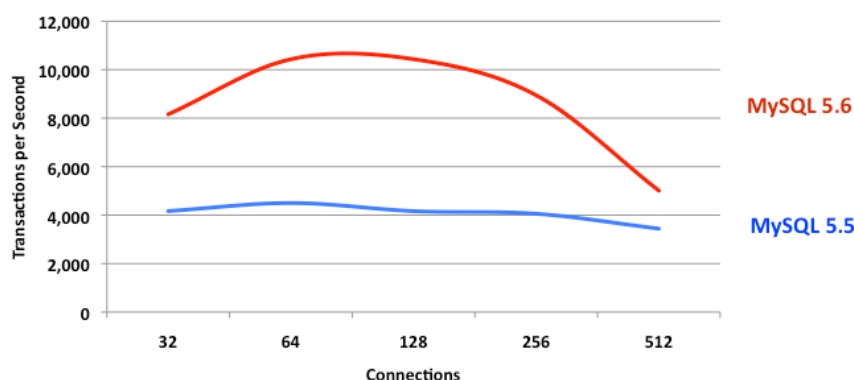
MySQL

Στην περίπτωση του εκπαιδευτικού λογισμικού, το επίπεδο δεδομένων θα υλοποιηθεί από το σύστημα MySQL RDBMS. Η απόφαση αυτή βασίστηκε στους παρακάτω λόγους:

1. Είναι διαθέσιμο χωρίς κόστος. Πρόκειται για ελεύθερο λογισμικό, το οποίο μας επιτρέπει να δημιουργήσουμε ένα περιβάλλον ανάπτυξης πανομοιότυπο με το πλήρως λειτουργικό περιβάλλον της εφαρμογής, όταν αυτή διατεθεί στο διαδίκτυο, χωρίς να απαιτείται κάποια δαπάνη.
1. Το σύστημα MySQL server προσφέρει την απαραίτητη επεξεργαστική ισχύ και τη δυνατότητα διεκπεραίωσης ταυτόχρονων δοσοληψιών, ώστε να ικανοποιεί τη λειτουργία της εφαρμογής σε πραγματικό περιβάλλον, προσβάσιμο από το διαδίκτυο. Τα διαγράμματα που ακολουθούν αναφέρουν τις δυνατότητες διεκπεραίωσης δοσοληψιών του MySQL Server.



Διάγραμμα 8. Επιδόσεις δοσοληψιών μόνο ανάγνωσης (read-only) του MySQL Server 5.6 (Κάτοχος εικόνας: MySQL)



Διάγραμμα 9. Δυνατότητες διεκπεραίωσης δοσοληψιών ανάγνωσης – εγγραφής για το MySQL Server 5.6 (Κάτοχος εικόνας: MySQL)

CSS

CSS: Το CSS (Cascading Style Sheets) είναι μια απλή γλώσσα που βοηθάει να ορίσει κανείς με σαφήνεια και ιδιαίτερη ευελιξία τον τρόπο με τον οποίο θα εμφανίζονται τα διάφορα στοιχεία σε μια ιστοσελίδα.

Τα πλεονεκτήματα που έχει η χρήση CSS έναντι της μορφοποίησης μέσω HTML χαρακτηριστικών (attributes) είναι τα εξής:

- Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασική HTML.
- Ευκολότερη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός ολόκληρου ιστοτόπου μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στον ιστότοπο.
- Μικρότερο μέγεθος αρχείου, δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.
- Καλύτερο SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση απλά στο περιεχόμενο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexing).
- Γρηγορότερες σελίδες. Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS ο φυλλομετρητής την πρώτη φορά που θα φορτώσει κάποια σελίδα του ιστότοπου το αποθηκεύει στην προσωρινή μνήμη, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που ο χρήστης του επισκέπτεται κάποια άλλη σελίδα του ιστότοπου.

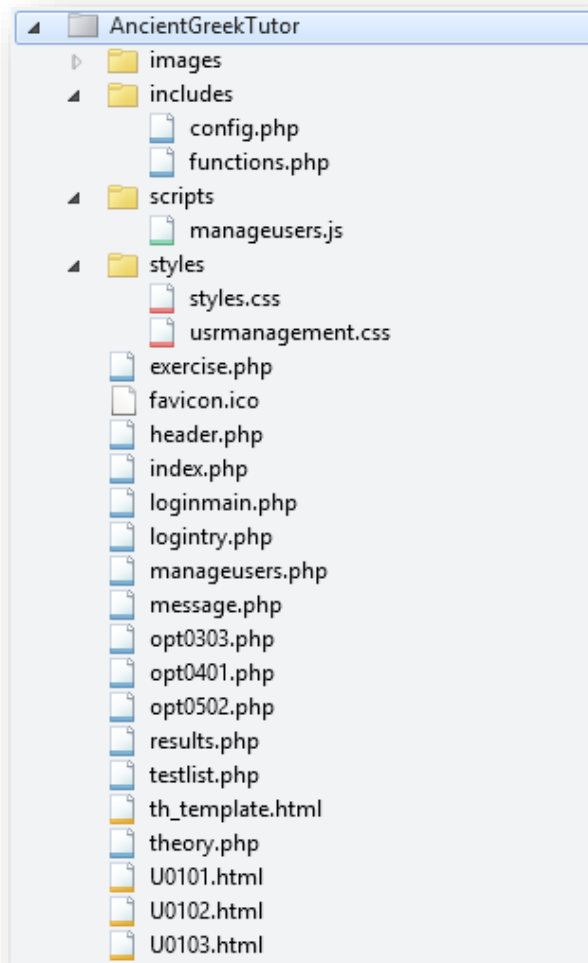
3.3. Περιγραφή υλοποίησης

Παρακάτω περιγράφουμε τις βασικές αρχές που έχουν τηρηθεί κατά την ανάπτυξη της εφαρμογής του εκπαιδευτικού λογισμικού.

Αρχικά, έγινε προσπάθεια διαχωρισμού των χαρακτηριστικών εμφάνισης του site από τη λειτουργικότητα του. Για το λόγο αυτό, έχει χρησιμοποιηθεί αρχείο css, το οποίο περιγράφει το πώς θα εμφανίζονται τα διάφορα στοιχεία της HTML.

Έπειτα, τα επαναχρησιμοποιούμενα στοιχεία του ιστότοπου (κεφαλίδα σελίδων) έχουν οριστεί σε ξεχωριστά αρχεία, ώστε να μην επαναλαμβάνεται συνεχώς ο ίδιος κώδικας και να είναι ευκολότερο να γίνουν αλλαγές.

Με βάση τα παραπάνω, ο ιστότοπος έχει την παρακάτω δομή (Διάγραμμα 10).



Διάγραμμα 10. Δομή του διαδικτυακού τόπου του εκπαιδευτικού λογισμικού

- images: πρόκειται για τον κατάλογο που περιλαμβάνονται οι εικόνες της εφαρμογής.
- styles: Είναι ο κατάλογος που περιέχει τα αρχεία css.
- includes: Είναι ο κατάλογος που περιέχει όλα τα βοηθητικά αρχεία, δηλαδή τα αρχεία:
 - config.php: περιέχει τις ρυθμίσεις της βάσης δεδομένων και τους κανόνες σύνδεσης της εφαρμογής σε αυτήν.

- `functions.php`: περιέχει τους ορισμούς όλων των συναρτήσεων που θα χρησιμοποιηθούν επαναληπτικά από την εφαρμογή.
- `scripts`: Είναι ο κατάλογος που περιέχει τα αρχεία javascript που χρησιμοποιούνται.
- `index.php`: Πρόκειται για την αρχική σελίδα της εφαρμογής η οποία προσφέρει τις βασικές επιλογές και προσαρμόζεται ανάλογα με το είδος του χρήστη (διαχειριστής ή μαθητής).

Αρχεία εφαρμογής

Τα αρχεία που υλοποιούν το εκπαιδευτικό λογισμικό μαζί με την περιγραφή της λειτουργίας τους παρατίθενται στη συνέχεια.

1. `header.php`: Περιέχει την κεφαλίδα (header) όπου αναφέρεται το όνομα του συνδεδεμένου χρήστη καθώς και ένας σύνδεσμος για την αποσύνδεσή του.
2. `index.php`: Η αρχική σελίδα της εφαρμογής. Προσφέρει συνδέσμους προς τις σελίδες της θεωρίας, τις σελίδες των ασκήσεων, τα αποτελέσματα των ασκήσεων και τη σύνδεση (login) του χρήστη. Στην περίπτωση που έχει συνδεθεί χρήστης με δικαιώματα διαχειριστή, εμφανίζεται και σύνδεσμος προς τη διαχείριση χρηστών.
3. `loginmain.php`: Η σελίδα σύνδεσης στην εφαρμογή (Login page).
4. `logintry.php`: ελέγχει τα στοιχεία σύνδεσης στη βάση δεδομένων και αν βρει ταίριασμα δημιουργεί το session cookie με τα στοιχεία του συνδεδεμένου χρήστη.
5. `manageusers.php`: Πρόκειται για σελίδα η οποία υλοποιεί φόρμα για την καταχώρηση στοιχείων νέου χρήστη και διαχείριση των στοιχείων των ήδη καταχωρημένων χρηστών. Επίσης από τη σελίδα αυτή γίνεται και διαγραφή ήδη δημιουργημένου χρήστη.
6. `message.php`: υλοποιεί τα παράθυρα μηνυμάτων που εμφανίζονται από την εφαρμογή σε περίπτωση που ο χρήστης πρέπει να ενημερωθεί για το αποτέλεσμα μιας ενέργειάς του.
7. `results.php`: Πρόκειται για σελίδα η οποία εμφανίζει τα γραφήματα των αποτελεσμάτων των ασκήσεων.
8. `testlist.php`: Σελίδα η οποία δημιουργεί λίστα με τα τεστ που έχει λύσει ο συνδεδεμένος χρήστης. Εάν ο χρήστης έχει δικαιώματα διαχειριστή, βλέπει τα τεστ όλων των χρηστών. Εάν ο συνδεδεμένος χρήστης είναι μαθητής, βλέπει μόνο τα δικά του τεστ.
9. `theory.php`: Δημιουργεί το περιβάλλον ξεφυλλίσματος των σελίδων της θεωρίας. Επίσης προσφέρει σύνδεσμο προς τις αντίστοιχες ασκήσεις του τρέχοντος κεφαλαίου.

Επεξήγηση config.php

Με τη βοήθεια αυτού του αρχείου ορίζονται τα στοιχεία πρόσβασης στη βάση δεδομένων και γίνεται η σύνδεση με τη βάση ενώ αρχικοποιείται το PHP SESSION. Επίσης, ορίζεται το σύνολο χαρακτήρων της βάσης δεδομένων (Unicode, UTF-8), ώστε να είναι δυνατή η διαχείριση των πολυτονικών ελληνικών χαρακτήρων.

Το PHP SESSION χρησιμοποιείται για την αποθήκευση δεδομένων που αφορούν κάποιον χρήστη. Συγκεκριμένα, κάθε μεμονωμένος χρήστης συσχετίζεται μ' ένα μοναδικό αντικείμενο session. Αυτό γίνεται δημιουργώντας ένα μοναδικό ID για τον κάθε χρήστη που αλληλεπιδρά με την εφαρμογή και αποθηκεύοντας το σαν ένα cookie στο μηχάνημα του πελάτη (client). Σε κάθε αίτηση (request) που κάνει ο πελάτης προς την εφαρμογή, χρησιμοποιείται το cookie που περιέχει το ID του πελάτη ώστε να μπορέσει να ξεχωρίσει ανάμεσα από πολλούς άλλους χρήστες. Το ID χρησιμοποιείται για να μπορέσει να καθορίσει το σωστό στιγμιότυπο του session για τον αντίστοιχο χρήστη.

Αυτός ο μηχανισμός επιτρέπει στην εφαρμογή να διατηρεί μια ιδιωτική συνομιλία με τον κάθε χρήστη καθώς αυτός πηγαίνει από σελίδα σε σελίδα. Υπάρχει η δυνατότητα να αποθηκεύουμε και να ανακτούμε τα προσωρινά δεδομένα ενός session χρησιμοποιώντας το αντικείμενο `$_SESSION`. Χαρακτηριστικό παράδειγμα είναι το `$_SESSION['username']` στο οποίο αποθηκεύουμε το username του χρήστη αφού ταυτοποιηθεί.

Κώδικας 7: Κώδικας config.php

```
<?php
```

```

    session_start();
    define("DBHOST","localhost");
    define("DBUSER","root");
    define("DBPWD","11091968");
    define("DBNAME","dglearn");
    $conn=mysqli_connect(DBHOST, DBUSER, DBPWD, DBNAME);
    if(!$conn)
    {die ("ΠΡΟΒΛΗΜΑ ΣΤΗ ΣΥΝΔΕΣΗ ΜΕ ΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ");
    }
    mysqli_query($conn, "SET NAMES UTF8");

?>

```

Επεξήγηση header.php

Το αρχείο αυτό χρησιμοποιείται για το κοινό κομμάτι κάθε ιστοσελίδας (header). Περιέχει πρακτικά την κεφαλίδα (header) της εφαρμογής.

Αρχικά, γίνεται έλεγχος για το αν ο χρήστης έχει συνδεθεί στο σύστημα. Εάν έχει συνδεθεί, τότε το όνομά του εμφανίζεται πάνω αριστερά στη σελίδα. Εάν δεν έχει συνδεθεί κανένας χρήστης, εμφανίζεται στην κεφαλίδα αντίστοιχο μήνυμα ("Δεν έχει συνδεθεί κάποιος χρήστης").

Στη συνέχεια, ακολουθεί ο HTML Header της σελίδας (Κώδικας 8).

Κώδικας 8. Κώδικας header.php

```

<!-- Κεφαλίδα σελίδας -->
<!-- Εμφανίζει το logged χρήστη και την επιλογή αποσύνδεσης -->
<div class ="PageHeader">
    <table style="width: 100%">
        <tr>
            <!-- Αριστερό κελί (αναφέρει το logged χρήστη και
login/logout -->
            <td style="width: 50%; color: white; padding-left:
8px;">
                <?php
                    $LoggedInUser = new User();
                    $LoggedIn_ID = $LoggedInUser->GetLoggedInUser();
                    if (strlen($_SESSION['LoggedInName']) > 0)
                    {
                        // Εμφάνιση username του συνδεδεμένου χρήστη
                        echo "Συνδέθηκε: ".$_SESSION['LoggedInName']."
";
                        echo "<a href=\"message.php?msgid=3\">(κλικ
εδώ για αποσύνδεση)</a>"; // Σύνδεσμος στο μήνυμα αποσύνδεσης
                    }
                    else
                    {
                        // Εάν δε συνδέθηκε χρήστης, ενημέρωση με
κατάλληλο μήνυμα
                        echo "Δεν έχει συνδεθεί κάποιος
χρήστης".<br>";
                        echo "<a href=\"loginmain.php\">Login</a>";
// Σύνδεσμος στη σελίδα σύνδεσης (login)
                    }
                ?>
            </td>
        </tr>

```

```
</table>  
</div> <!-- Div κεφαλίδας σελίδας -->
```

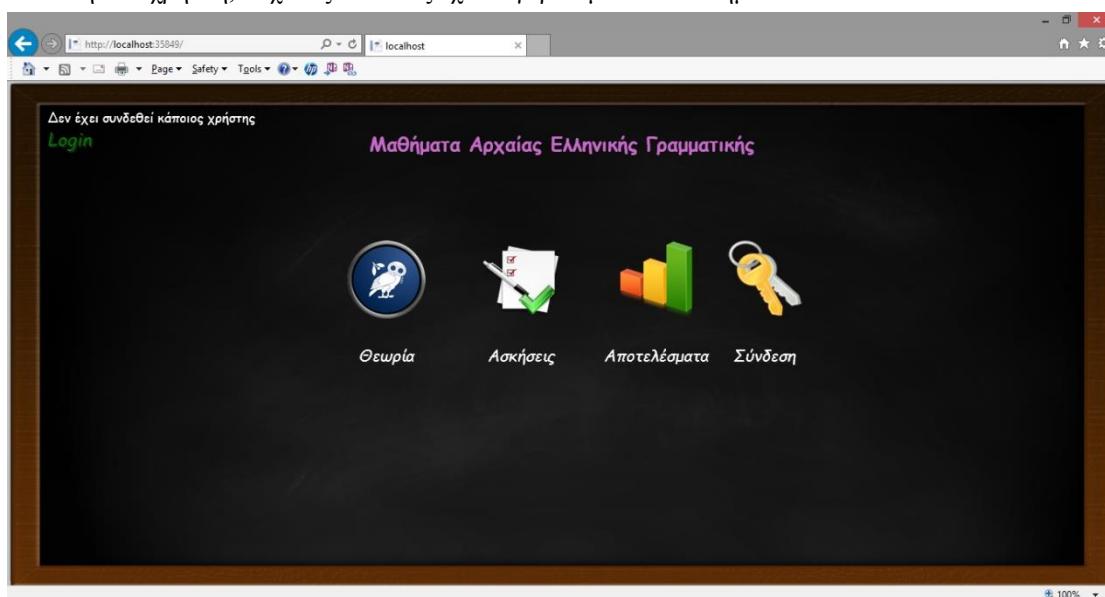
4. Λειτουργικότητα συστήματος

4.1. Εφαρμογή διαχείρισης εκπαιδευτικού μοντέλου

Οι δύο ενότητες της εφαρμογής θα αναφέρονται ως:

- Εφαρμογή εκπαιδευτικού.
- Εφαρμογή μαθητή

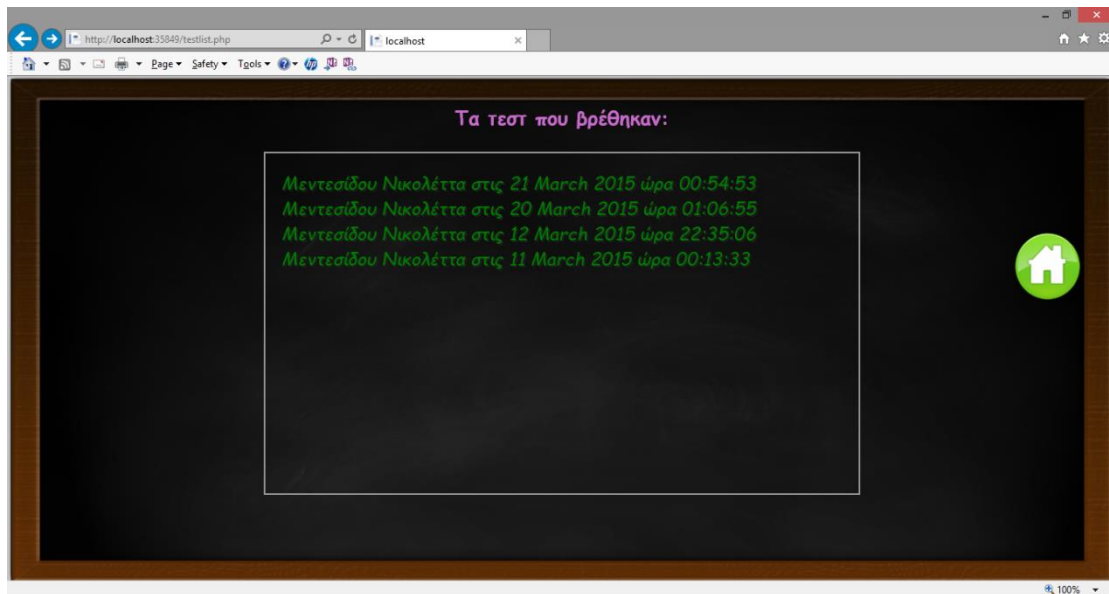
Η εφαρμογή ξεκινά από την αρχική της σελίδα, όπου και προσφέρονται οι βασικές επιλογές οποιουδήποτε χρήστη, ασχέτως αν αυτός έχει λογαριασμό στο σύστημα.



Εικόνα 6. Οθόνη επιλογής τεστ για επισκόπηση

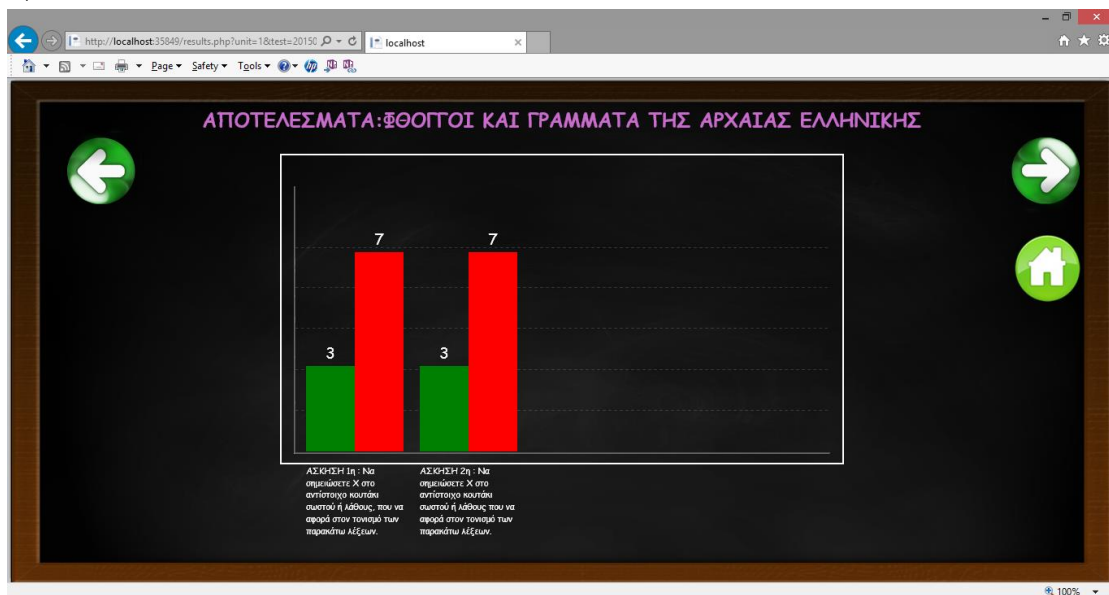
4.1.1. Εφαρμογή εκπαιδευτικού

Η εφαρμογή εκπαιδευτικού δίνει τη δυνατότητα στους εκπαιδευτικούς – διαχειριστές να δουν τα αποτελέσματα των τεστ τα οποία εκτέλεσαν οι μαθητές.



Εικόνα 7. Οθόνη επιλογής τεστ για επισκόπηση

Ο δάσκαλος επιλέγει το μαθητή και το τεστ που θέλει και στη συνέχεια βλέπει τα αποτελέσματα σε κάθε κεφάλαιο του τεστ.



Εικόνα 8. Εμφάνιση αποτελεσμάτων τεστ

Η εφαρμογή εμφανίζει τα αποτελέσματα ανά άσκηση και κεφάλαιο, ώστε ο εκπαιδευτικός να σχηματίζει εικόνα για την επίδοση του μαθητή.

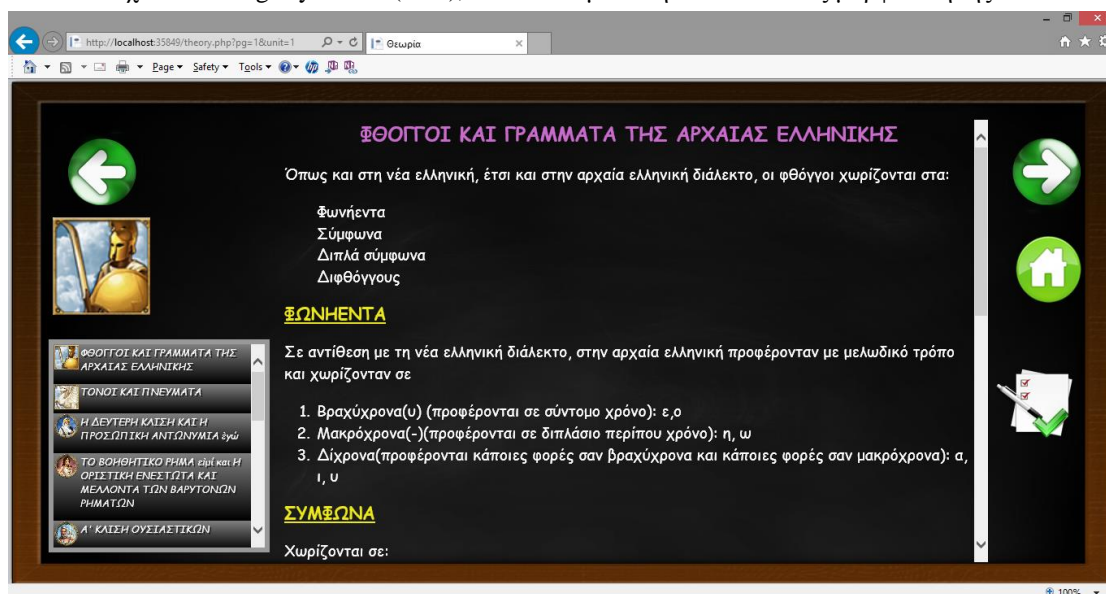
4.1.2. Εφαρμογή μαθητή

Το εκπαιδευτικό υλικό του μαθήματος δομείται σε ενότητες. Η ενότητα προσδιορίζεται από την τιμή του πεδίου `th_class` του πίνακα `theory`. Στη συγκεκριμένη υλοποίηση υλοποιήθηκαν 14 κεφάλαια της 1ης ενότητας των αρχαίων ελληνικών της Α' Γυμνασίου. Τα κεφάλαια που περιλαμβάνονται στο λογισμικό είναι:

- **Κεφάλαιο 1**
ΦΘΟΓΟΙ ΚΑΙ ΓΡΑΜΜΑΤΑ ΤΗΣ ΑΡΧΑΙΑΣ ΕΛΛΗΝΙΚΗΣ

- **Κεφάλαιο 2**
ΤΟΝΟΙ ΚΑΙ ΠΝΕΥΜΑΤΑ
- **Κεφάλαιο 3**
Η ΔΕΥΤΕΡΗ ΚΛΙΣΗ ΚΑΙ Η ΠΡΟΣΩΠΙΚΗ ΑΝΤΩΝΥΜΙΑ εγώ
- **Κεφάλαιο 4**
ΤΟ ΒΟΗΘΗΤΙΚΟ ΡΗΜΑ εἰμί και Η ΟΡΙΣΤΙΚΗ ΕΝΕΣΤΩΤΑ ΚΑΙ ΜΕΛΛΟΝΤΑ ΤΩΝ ΒΑΡΥΤΟΝΩΝ ΡΗΜΑΤΩΝ
- **Κεφάλαιο 5**
Α' ΚΛΙΣΗ ΟΥΣΙΑΣΤΙΚΩΝ
- **Κεφάλαιο 6**
ΟΡΙΣΤΙΚΗ ΠΑΡΑΤΑΤΙΚΟΥ ΤΟΥ Ρ. "εἰμί"
- **Κεφάλαιο 7**
ΔΕΥΤΕΡΟΚΛΙΤΑ ΕΠΙΘΕΤΑ
- **Κεφάλαιο 8**
ΑΝΑΔΙΠΛΑΣΙΑΣΜΟΣ ΣΥΝΤΕΛΙΚΩΝ ΧΡΟΝΩΝ
- **Κεφάλαιο 9**
ΤΡΙΤΟΚΛΙΤΑ ΟΥΣΙΑΣΤΙΚΑ
- **Κεφάλαιο 10**
ΚΑΤΑΛΗΞΕΙΣ ΑΠΑΡΕΜΦΑΤΩΝ ΕΝΕΡΓΗΤΙΚΗΣ ΦΩΝΗΣ
- **Κεφάλαιο 11**
ΟΡΙΣΤΙΚΗ ΜΕΛΛΟΝΤΑ ΜΕΣΗΣ ΦΩΝΗΣ ΤΟΥ ΒΟΗΘΗΤΙΚΟΥ ΡΗΜΑΤΟΣ εἰμι
- **Κεφάλαιο 12**
ΟΡΙΣΤΙΚΗ ΠΑΡΑΤΑΤΙΚΟΥ ΚΑΙ ΑΟΡΙΣΤΟΥ ΜΕΣΗΣ ΦΩΝΗΣ ΒΑΡΥΤΟΝΩΝ ΡΗΜΑΤΩΝ
- **Κεφάλαιο 13**
ΟΡΙΣΤΙΚΗ ΠΑΡΑΚΕΙΜΕΝΟΥ ΚΑΙ ΥΠΕΡΣΥΝΤΕΛΙΚΟΥ ΜΕΣΗΣ ΦΩΝΗΣ
- **Κεφάλαιο 14**
ΑΠΑΡΕΜΦΑΤΟ ΚΑΙ ΜΕΤΟΧΗ ΒΑΡΥΤΟΝΩΝ ΡΗΜΑΤΩΝ ΜΕΣΗΣ ΦΩΝΗΣ

Κάθε κεφάλαιο αντιστοιχεί σε μία html σελίδα, η οποία και αναπτύσσει το αντίστοιχο θεωρητικό κομμάτι. Για την ομοιόμορφη αλλά και κατανοητή παρουσίαση των σελίδων χρησιμοποιήθηκε η γλώσσα HTML και αντίστοιχο Cascading Stylesheet (CSS), όπου ενσωματώθηκαν οι κανόνες μορφοποίησης.



Εικόνα 9 –Περιβάλλον εμφάνισης και ανάγνωσης σελίδων θεωρίας

Ο χρήστης μπορεί να μεταφέρεται στο επόμενο ή προηγούμενο κεφάλαιο χρησιμοποιώντας τα αντίστοιχα βέλη (αριστερό και δεξί) ή να μεταβαίνει στο επιθυμητό κεφάλαιο, επιλέγοντας από τη λίστα των κεφαλαίων στο κάτω αριστερό τμήμα της σελίδας ανάγνωσης της θεωρίας.



Εικόνα 10 –Επιλογές μαθημάτων ανά ενότητα

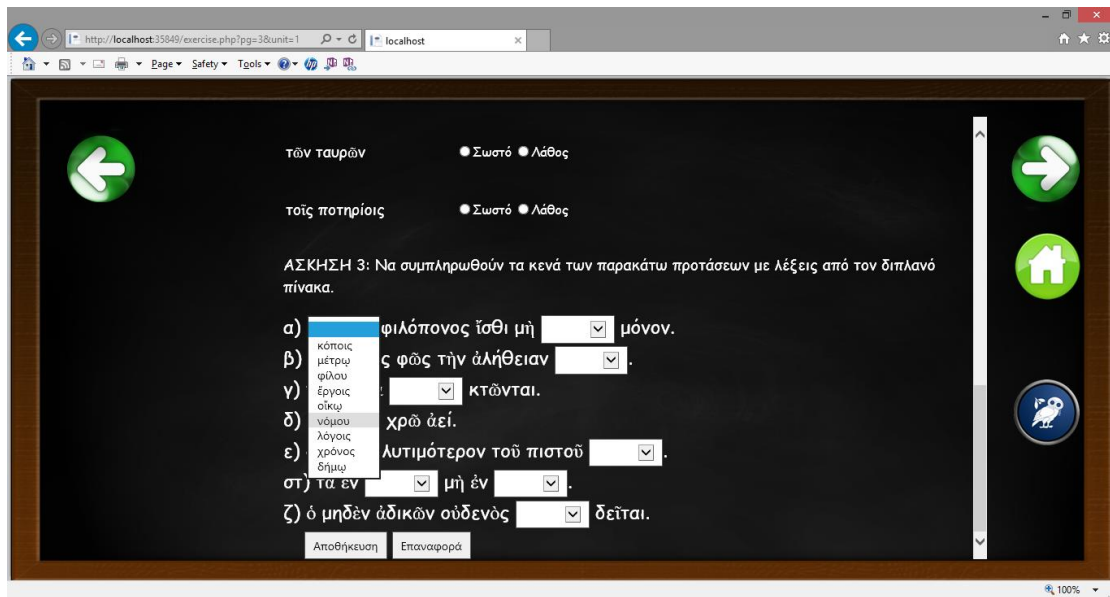
Κάθε κεφάλαιο συνδέεται με έναν αριθμό ασκήσεων. Κάθε άσκηση περιλαμβάνει έναν αριθμό ερωτήσεων και κάθε ερώτηση έχει ένα πλήθος επιλογών.

Κάθε εκπαιδευτική ενότητα, τεστ, άσκηση και ερώτηση έχουν μια απεικόνιση (εγγραφή) στη σχεσιακή βάση δεδομένων και συσχετίσεις όπως αυτές απεικονίζονται στο Διάγραμμα 6.

Οι ασκήσεις ανήκουν σε μία από τις τρεις διαθέσιμες κατηγορίες (κλάσεις):

1. Ερωτήσεις απλής επιλογής (μία σωστή απάντηση από ένα σύνολο επιλογών).
2. Ερωτήσεις πολλαπλής επιλογής (επιλογή μίας ή περισσότερων απαντήσεων από ένα σύνολο προτεινόμενων λύσεων).
3. Ερωτήσεις επιλογής μίας τιμής από μία λίστα επιλογών.

Ανάλογα με την κλάση της ερώτησης ενεργοποιείται ο κατάλληλος μηχανισμός προβολής της ερώτησης και των επιλογών της. Στην οθόνη που ακολουθεί, εμφανίζονται οι υλοποιήσεις των τριών κλάσεων σε αντίστοιχες οθόνες.



Εικόνα 11. Ενδεικτική οθόνη διεξαγωγής μιας άσκησης

5. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Το εκπαιδευτικό λογισμικό έχει σχεδιαστεί με τρόπο ώστε μέσω της προσθήκης νέων εγγραφών στη βάση δεδομένων να επεκτείνει το περιεχόμενό του, τόσο με νέες θεωρητικές ενότητες όσο και με νέες ασκήσεις.

Στο πλαίσιο αυτό, το σύστημα μπορεί να επεκταθεί με νέες θεματικές ενότητες και τα αντίστοιχα κεφάλαια θεωρίας. Επίσης μπορεί να εμπλουτιστεί με πρόσθετες ασκήσεις.

Ένα επίσης ενδιαφέρον θέμα προς υλοποίηση είναι η δημιουργία τεστ μέσα από μια τράπεζα θεμάτων με επιλογή από το σύστημα βάσει αλγορίθμων, οι οποίοι θα μπορούν να λαμβάνουν υπόψη τους παραμέτρους όπως:

- Η δυσκολία των ασκήσεων
- Το επίπεδο του μαθητή

Επιπρόσθετα το σύστημα μπορεί να θέτει χρονικά όρια για τη διεκπεραίωση μιας άσκησης και βαθμολόγηση σε συνάρτηση με το χρόνο που ο μαθητής χρειάστηκε για να δώσει την απάντησή του.

Θα δημιουργηθεί με αυτό τον τρόπο ένα σύστημα με στοιχεία ευφυίας, το οποίο θα εμφανίζεται στο χρήστη με αλλαγμένο το περιεχόμενό του και τις ασκήσεις του κάθε φορά που ο χρήστης θέλει να εκτελέσει ένα τεστ, αλλά και βάσει του επιπέδου του μαθητή, ώστε αυτός να μην αποθαρρύνεται από ασκήσεις υψηλού βαθμού δυσκολίας.

Παράρτημα Α – Κώδικας εφαρμογής

Κώδικας 9. theory.php

```

<?php
    ob_start();
    require_once 'includes/config.php';
?>

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <link rel="stylesheet" href="styles/styles.css">
        <title>Θεωρία</title>
    </head>
    <body>
        <?php
            // Η τρέχουσα σελίδα περνά ως παράμετρος στο url
            $currentPage = $_GET["pg"];
            // Η τρέχουσα ενότητα περνά ως παράμετρος στο url
            $currentUnit = $_GET["unit"];

            $strSQL = "SELECT * FROM theory WHERE th_class =
$currentUnit ORDER BY th_sorter";
            $result = mysqli_query($conn, $strSQL);
            // Εάν δεν είναι δυνατή η εκτέλεση του ερωτήματος, διακοπή
            της εκτέλεσης
            if (!$result)
                die('Εμφανίσθηκε συνθήκη σφάλματος: ' .
mysqli_error($conn));
            // Πλήθος κεφαλαίων
            $numRecs = mysqli_num_rows($result);
            // Δημιουργία πλαισίου στο κάτω αριστερό τμήμα της σελίδας
            // με ευρειτήριο των κεφαλαίων για άμεση μετάβαση
            echo "<div class=\"TheoryMenu\">";
            echo "<table style=\"border-collapse: collapse;\">";
            $i = 0;
            while($row = mysqli_fetch_array($result))
            {
                $TheoryRecords[]=$row;

                $th_id = $row['th_id'];
                $th_title = $row['th_title'];
                $th_image = $row['th_image'];
                $th_intro = $row['th_intro'];
                $th_class = $row['th_class'];
                $th_page = $row['th_page'];
                $i++;
                // Δημιουργία λίστας με την εικόνα και τον τίτλο του
                κεφαλαίου
                echo "<tr>";
                echo "<td style=\"vertical-align: top; background:
linear-gradient(grey, black);\">";
                echo "<img src=\"\". $th_image. \"\" style=\"width:
30px;\" /></td>";

```

```

        echo "<td style=\"vertical-align: top; background:
linear-gradient(grey, black);\"><a href=\"theory.php?pg=$i&unit=1\"><p
class=\"TheoryMenuText\">$th_title</p></a>";
        echo "</td>";
        echo "<tr>";

    }
    echo "</table>";
    echo "</div>";
    // Υπολογισμός προηγούμενης / επόμενης σελίδας θεωρίας για
εμφάνιση
    // Όταν διαβάζεται η τελευταία σελίδα, το πάτημ "Επόμενο"
προβάλλει το 1ο κεφάλαιο.
    // Όταν διαβάζεται το 1ο κεφάλαιο και πατηθεί "ΠΠροηγούμενο",
προβάλλει το τελευταίο κεφάλαιο
    if ($CurrentPage < $NumRecs)
        $NextPage = $CurrentPage+1;
    else
        $NextPage = 1;
    if ($CurrentPage > 1)
        $PreviousPage = $CurrentPage - 1;
    else
        $PreviousPage = $NumRecs;

    // Δημιουργία των κουμπιών πλοήγησης
    echo "<div class=\"ButtonHome\"><a href=\"index.php\"><img
src=\"images/home.png\" alt=\"Αρχική\"/></a></div>";
    echo "<div class=\"ButtonTest\"><a
href=\"exercise.php?pg=$CurrentPage&unit=1\"><img
src=\"images/tests.png\" alt=\"Αρχική\"/></a></div>";
    echo "<div class=\"ButtonLeft\"><a
href=\"theory.php?pg=$PreviousPage&unit=1\"><img
src=\"images/buttonleft.png\" alt=\"Previous\"/></a></div>";
    echo "<div class=\"ButtonTheoryImage\"><img
src=\"\".$TheoryRecords[$CurrentPage-1]['th_image'].\"\" style=\"width:
120px;\" alt=\"Theory\"/></div>";
    echo "<div class=\"ButtonRight\"><a
href=\"theory.php?pg=$NextPage&unit=1\"><img
src=\"images/buttonright.png\" alt=\"Next\"/></a></div>";
    echo "<div class=\"ContentDiv\">";
    // Εμφάνιση στο κεντρικό πλαίσιο της σελίδας που αντιστοιχεί
στο περιεχόμενο του τρέχοντος κεφαλαίου
    include $TheoryRecords[$CurrentPage-1]['th_page'];
    echo "</div>";

?>
</body>
</html>

```

Κώδικας 10. exercise.php

```

<?php
ob_start();
// Ενσωμάτωση κανόνων σύνδεσης στη βάση δεδομένων
require_once 'includes/config.php';
// Ενσωμάτωση συναρτήσεων διαχείρισης συνδεδεμένου χρήστη
require_once 'includes/functions.php';

```

```

?>

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="styles/styles.css">
    <title></title>
  </head>
  <body>
    <?php
      $hiddendebug = "hidden";

      $currentPage = $_GET["pg"]; // Κεφάλαιο
      $currentUnit = $_GET["unit"]; // Ενότητα θεωρίας (εδώ
      χρησιμοποιείται μόνο η τιμή 1). Για μελλοντική χρήση
      // Εύρεση συνδεδεμένου χρήστη από το session cookie
      $loggedUser = new User();
      $logged_ID = $loggedUser->GetLoggedUser();
      $adminAccess = $loggedUser->IsAdmin($logged_ID);

      if ($logged_ID > 0)
      {
        $sessionID = $_SESSION['SessionID'];
        echo "<div class=\"ContentDiv\">";

        // Βρίσκει τα κεφάλαια της θεωρίας που ανήκουν σε
        συγκεκριμένη ενότητα.
        // Στην εφαρμογή έχει υλοποιηθεί μόνο μία ενότητα.
        // Η παράμετρος $currentUnit χρησιμοποιείται για
        μελλοντική επέκταση του περιεχομένου της εφαρμογής
        // με επιπλέον θεματικές ενότητες της αρχαίας
        ελληνικής
        $strSQL = "SELECT * FROM theory WHERE th_class =
        $currentUnit ORDER BY th_sorter";
        $result = mysqli_query($conn, $strSQL);
        // Εάν δεν είναι δυνατή η εκτέλεση του ερωτήματος,
        διακοπή της εκτέλεσης
        // και εμφάνιση του μηνύματος λάθους στην οθόνη.
        if (!$result)
          die('Εμφανίσθηκε συνθήκη σφάλματος: ' .
        mysqli_error($conn));

        $numRecs = mysqli_num_rows($result); // Πλήθος
        κεφαλαίων θεωρίας.
        $rowNum = 0;

        // Βρίσκει όλες τις ασκήσεις του τρέχοντος κεφαλαίου
        $strSQL = "SELECT exercises.exe_id,
        exercises.exe_text, exercises.exe_unit, exercises.exe_class,
        exercises.exe_sorter, exercises.exe_unit
        FROM exercises
        WHERE exercises.exe_unit=".$currentPage."
        ORDER BY exercises.exe_sorter";
        $result = mysqli_query($conn, $strSQL);

```

```

        // Εάν δεν είναι δυνατή η εκτέλεση του ερωτήματος,
διακοπή της εκτέλεσης
        if (!$result )
            die('Εμφανίσθηκε συνθήκη σφάλματος: ' .
mysqli_error($conn));

        // Δημιουργία φόρμας η οποία περιέχει τις αντίστοιχες
ασκήσεις
        echo "<form name=\"questionnaire\"
id=\"questionnaire\"
action=\"".$_SERVER['PHP_SELF']."?pg=".$_CurrentPage."&unit=1\".\"\"
method=\"POST\">";
        while ($row = mysqli_fetch_array($result))
        {
            // Ανάκληση της εκφώνησης και των δεδομένων των
ασκήσεων από τη βάση δεδομένων
            // και ανάθεσή τους σε αντίστοιχες μεταβλητές
            $exe_id = $row['exe_id'];
            $exe_text = $row['exe_text'];
            $exe_unit = $row['exe_unit'];
            $exe_class = $row['exe_class'];
            $qt_id = $row['qt_id'];
            $qt_text = $row['qt_text'];
            $qo_id = $row['qo_id'];
            $qo_text = $row['qo_text'];
            echo "<p>".$exe_text."</p>";

            // Loop που βρίσκει τα λεκτικά των ερωτημάτων της
ερώτησης
            $strSQLQuestionText = "SELECT exercises.exe_id,
exercises.exe_unit, question_text.qt_id, question_text.qt_text,
                                question_text.qt_sorter
FROM exercises INNER JOIN
question_text ON exercises.exe_id = question_text.qt_exercise
WHERE
exercises.exe_unit=".$_exe_unit." AND exercises.exe_id=".$_exe_id.
                                " ORDER BY
question_text.qt_sorter, question_text.qt_id";
            $resultQuestionText = mysqli_query($conn,
$strSQLQuestionText);
            $qt_text = "";
            $options_number = 0;
            while ($rowQuestionText =
mysqli_fetch_array($resultQuestionText))
            {
                $placeholder = $rowQuestionText['qt_text'];
                $qt_text = $placeholder;
                $qt_id = $rowQuestionText['qt_id'];
                if ($exe_class != 3)
                {
                    echo "<table style=\"border-style: solid;
border-width: 0px; border-color: red;\">";
                    echo "<tr style=\"border-style: solid;
border-width: 0px; border-color: red;\">";
                    echo "<td style=\"width: 60%;\">";
                    echo "<p>".$qt_text;

```

```

        echo "</td>";
    }
    else
        echo "<span style=\"font-size: 18pt;
color: #fff;\">".$qt_text;
        // Loop που βρίσκει τις επιλογές κάθε
ερωτήματος
        $strSQLQuestionOptions = "SELECT
question_options.qo_id, question_options.qo_qt_id,
question_options.qo_text,

question_options.qo_correct_answer, question_options.qo_optionsfile,
question_options.qo_shorter

FROM
question_options

WHERE
question_options.qo_qt_id=".$qt_id." ORDER BY
question_options.qo_shorter";
        $resultQuestionOptions = mysqli_query($conn,
$strSQLQuestionOptions);
        $qt_id_aggregate = "";
        while ($rowQuestionOptions =
mysqli_fetch_array($resultQuestionOptions))
        {
            $qo_id = $rowQuestionOptions['qo_id'];
            // Τα id των επιλογών συνενώνονται σε ένα
string
            $qt_id_aggregate .= $qo_id."#";
            $qo_correct_answer =
$rowQuestionOptions['qo_correct_answer']; // Τιμή σωστής απάντησης
            $qo_qt_id =
$rowQuestionOptions['qo_qt_id']; // id της επιλογής
            $qo_text = $rowQuestionOptions['qo_text'];
            // Λεκτικό επιλογής (π.χ. "Σωστό"
            $qo_optionsfile =
$rowQuestionOptions['qo_optionsfile']; // Αρχείο επιλογών (σε
περίπτωση λίστας επιλογών)

            // Εύρεση της απάντησης που έχει δοθεί και
ενημέρωση του περιβάλλοντος των
            // ασκήσεων με την ήδη καταχωρημένη τιμή
            $checked = '';

            $strSQLGivenAnswer = "SELECT res_answer
FROM results WHERE res_session_id='$SessionID' AND
res_question_option=$qo_id";
            $resultGivenAnswer = mysqli_query($conn,
$strSQLGivenAnswer);
            while ($rowGivenAnswer =
mysqli_fetch_array($resultGivenAnswer))
            {
                $GivenAnswer =
$rowGivenAnswer['res_answer'];
            }

```

```

// Ανάλογα με την κλάση της ερώτησης
δημιουργούνται:
// 1. Radio buttons
// 2. Check boxes
// 3. List boxes
switch ($exe_class)
{
    // Εάν η κλάση της άσκησης είναι 1,
δημιουργία radio buttons για τις επιλογές
    case 1:
        echo "<td>";
        if ((strcmp ($GivenAnswer, '1') ==
0) && (strcmp($qo_correct_answer, '1') == 0 ))
            $checked = 'checked';
        if ((strcmp ($GivenAnswer, '0') ==
0) && (strcmp($qo_correct_answer, '0') == 0 ))
            $checked = 'checked';
        echo "<input type=\"radio\"
name=\"Data_Array[$rowNum][qo_qt_id]\"
value=\"\".$qo_correct_answer.\"\" id=\"\".$qo_id.\"\" $checked>";
        echo $qo_text." ";
        echo "</td>";
        break;
    // Εάν η κλάση της άσκησης είναι 2,
δημιουργία check box για τις επιλογές
    case 2:
        echo "<td>";
        if (strcmp ($GivenAnswer, '1') ==
0)
            $checked = 'checked';
        echo "<input type=\"checkbox\"
name=\"Data_Array[$rowNum][qo_qt_id]\" value=\"1\" $checked>";
        echo $qo_text." ";
        echo "</td>";
        break;
    // Εάν η κλάση της άσκησης είναι 3,
δημιουργία list box για τις επιλογές
    case 3:
        echo "<select
name=\"Data_Array[$rowNum][qo_qt_id]\">";
        // Το list box γεμίζει με τις
τιμές το text αρχείο $qo_optionsfile
        if (strlen($qo_optionsfile) > 0)
// Έλεγχος εάν έχει δοθεί τιμή στη μεταβλητή $qo_optionsfile
        {
            $file =
fopen($qo_optionsfile,"r"); // Εάν έχει δηλωθεί αρχείο τιμών, άνοιγμα
και ανάγνωση των περιεχομένων του
            while(! feof($file))
            {
                $line =
trim(fgets($file));
                if (strcmp ($GivenAnswer,
$line) == 0)
                    $checked = 'selected';

```

```

                                echo "<option
value=\"\"$.line.\" \" $checked>$.line.</option>"; // Προσθήκη των
τιμών στο list box

                                $checked = '';
                                }
                                fclose($file);
                                }
                                echo "</select>";
                                break;
                                }
                                }

                                $options_number++; // Αύξηση του πλήθους
επιλογών που δίνει η ερώτηση
                                // Οι ερωτήσεις με επιλογές radio button και
check box μορφοποιούνται σε κελιά πίνακα
                                // Όταν διαβάζονται όλες οι επιλογές, κλείσιμο
της γραμμής του πίνακα και του πίνακα
                                if ($exe_class != 3)
                                {
                                    echo "</tr>";
                                    echo "</table>";
                                }
                                // Κρυφά πεδία για πέρασμα του array των
ιδιοτήτων κάθε ερώτησης
                                echo "<input type=\"text\" $hiddendebug
name=\"Data_Array[$rowNum][qo_id]\" value=\"\$qt_id_aggregate\">";
                                echo "<input type=\"text\" $hiddendebug
name=\"Data_Array[$rowNum][exe_class]\" value=\"\$exe_class\">";
                                echo "<input type=\"text\" $hiddendebug
name=\"Data_Array[$rowNum][exe_id]\" value=\"\$exe_id\">";
                                echo "<input type=\"text\" $hiddendebug
name=\"Data_Array[$rowNum][exe_unit]\" value=\"\$exe_unit\">";

                                $rowNum++;
                                }
                                }
                                if ($options_number > 0)
                                    echo "<input style=\"margin-left: 26px;\"
type=\"submit\" value=\"Save\"
onclick=\"ResetFieldChange('UnsavedChanges');\" /><input
style=\"margin-left: 6px;\" type=\"reset\" value=\"Reset\"/>";
                                else
                                    echo "<p>Αυτό το κεφάλαιο δεν έχει ασκήσεις.</p>";
                                echo "</form>";
                                echo "</div>";

                                // Υπολογισμός επόμενου / προηγούμενου κεφαλαίου για
την προβολή των αντίστοιχων ασκήσεων
                                // όταν πατιέται το αντίστοιχο κουμπί στην οθόνη
(επόμενο / προηγούμενο)
                                if ($currentPage < $NumRecs)
                                    $NextPage = $currentPage+1;
                                else
                                    $NextPage = 1;
                                if ($currentPage > 1)

```



```

        $PreviousPage = $CurrentPage - 1;
    else
        $PreviousPage = $NumRecs;

    // Δημιουργία των κουμπιών οθόνης για πλοήγηση στην
εφαρμογή
    // Κουμπί επιστροφής στην αρχική σελίδα
    echo "<div class=\"ButtonHome\"><a
href=\"index.php\"><img src=\"images/home.png\"
alt=\"Αρχική\"/></a></div>";
    // Κουμπί μετάβασης στη θεωρία του αντίστοιχου
κεφαλαίου
    echo "<div class=\"ButtonTest\"><a
href=\"theory.php?pg=$CurrentPage&unit=1\"><img
src=\"images/theoryhome.png\" alt=\"Αρχική\"/></a></div>";
    // Κουμπί μετάβασης στις ασκήσεις του προηγούμενου
κεφαλαίου
    echo "<div class=\"ButtonLeft\"><a
href=\"exercise.php?pg=$PreviousPage&unit=1\"><img
src=\"images/buttonleft.png\" alt=\"Previous\"/></a></div>";
    // Κουμπί μετάβασης στις ασκήσεις του επόμενου
κεφαλαίου
    echo "<div class=\"ButtonRight\"><a
href=\"exercise.php?pg=$NextPage&unit=1\"><img
src=\"images/buttonright.png\" alt=\"Next\"/></a></div>";

    }
    else
    {
        // Μήνυμα εάν ο χρήστης προσπαθήσει να μεταβεί σε
σελίδα ασκήσεων χωρίς πρώτα
        // να έχει συνδεθεί στο σύστημα (login)
        header("Location: message.php?msgid=2");
    }

    if ($_SERVER["REQUEST_METHOD"] == "POST")
    {
        // Εάν πατηθεί το κουμπί "Αποθήκευση" των απαντήσεων
        // Εύρεση των στοιχείων σύνδεσης από το session cookie
        $SessionID = $_SESSION['SessionID'];
        $UserID = $_SESSION['LoggedUserID'];
        $Timestamp = $_SESSION['SessionStart'];
        // Τα δεδομένα κάθε άσκησης ανατίθενται σε αντίστοιχες
μεταβλητές
        foreach ($_POST["Data_Array"] as $dt)
        {
            $qo_id = $dt['qo_id'];
            $exe_class = $dt['exe_class'];
            $exe_id = $dt['exe_id'];
            $exe_unit = $dt['exe_unit'];
            if (strlen($dt['qo_qt_id']) > 0)
                $qo_qt_id = $dt['qo_qt_id'];
            else
                $qo_qt_id = NULL;
        }
    }

```

```

        // Ανάλυση του string των id των επιλογών κάθε
άσκησης
        // Κάθε id διαχωρίζεται από το επόμενο με το
σύμβολο #
        $sids = explode("#", $qo_id);
        for($i = 0; $i < count($sids)-1; $i++)
        {
            // Εισαγωγή των τιμών των απαντήσεων και του
session στο αρχείο αποτελεσμάτων των test
            $strSQLInsert = "INSERT INTO results
(res_session_id, res_user_id, res_timestamp, res_question_option,
res_answer, res_exe_class, res_exe_id, res_exe_unit) VALUES ".
                "('$SessionID', $UserID,
'$Timestamp', $sids[$i], '$qo_qt_id', $exe_class, $exe_id, $exe_unit)";
            $result = mysqli_query($conn, $strSQLInsert);
            if (!$result)
            {
                // Εάν η sql INSERT αποτύχει, αυτό
σημαίνει ότι ήδη υπάρχει καταχωρημένη η συγκεκριμένη απάντηση
                // στο τρέχον session. Σε αυτή την
περίπτωση, ενημέρωσε την υπάρχουσα εγγραφή
                // Πρόταση sql για την ενημέρωση
υφιστάμενης εγγραφής
                $strSQLUpdate = "UPDATE results
                SET res_answer =
                '$qo_qt_id'
                WHERE res_session_id =
                '$SessionID' AND res_question_option = $sids[$i]";
                $resultUpdate = mysqli_query($conn,
                $strSQLUpdate);
            }
        }
        header("Location:
exercise.php?pg=$currentPage&unit=1");
    }

    ?>
</body>
</html>

```

Κώδικας 11. manageusers.php

```

<?php
//
*****
// Δημιουργεί το περιβάλλον διαχείρισης χρηστών
// 1. Προσθήκη νέου χρήστη
// 2. Επεξεργασία στοιχείων ήδη καταχωρημένου χρήστη
// 3. Διαγραφή χρήστη
//
//
*****

ob_start();
require 'includes/config.php';

```

```
require 'includes/functions.php';

// Βρίσκει τον logged χρήστη και το ρόλο του
// Αν είναι administrator, το $AdminAccess παίρνει τιμή True
$LoggedInUser = new User();
$LoggedIn_ID = $LoggedInUser->GetLoggedInUser();
$AdminAccess = $LoggedInUser->IsAdmin($LoggedIn_ID);
?>

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <script type="text/javascript"
src="scripts/manageusers.js"></script>
    <link href="styles/usrmanagement.css" rel="stylesheet"
type="text/css" />
    <title>Διαχείριση χρηστών</title>
  </head>
  <body>

    <!-- Το div με id=NewRecord δημιουργεί παράθυρο για συμπλήρωση
    νέας εγγραφής χρήστη -->
    <div id="NewRecord" class="FormDialog" style="height: 500px;
width: 540px; z-index: 2000;">
      <?php
        echo "<form name=\"newuserform\"
action=\"\".$_SERVER['PHP_SELF'].\"\" method=\"POST\">";
        ?>
        <table style="margin-left: auto; margin-right: auto;
margin-top: 50px;">
          <!-- Γραμμή εισαγωγής Username -->
          <tr>
            <td class="Label" style="background-color:
transparent;">
              <ul><li style="color: white;">Username:</li></ul>
            </td>
            <td class="InputField">
              <input type="text" required name="usr_username"
value="" size="10" ></input>
            </td>
          </tr>

          <!-- Γραμμή εισαγωγής Password -->
          <tr>
            <td class="Label" style="background-color:
transparent;">
              <ul><li style="color: white;">Password:</li></ul>
            </td>
            <td class="InputField">
              <input type="text" required name="usr_password"
value="" size="10" ></input>
            </td>
          </tr>
        </table>
      </div>
    </body>
</html>
```

```

        <!-- Γραμμή εισαγωγής Ονόματος -->
        <tr>
        <td class="Label" style="background-color:
transparent;">
            <ul><li style="color: white;">Όνομα:</li></ul>
        </td>
        <td class="InputField">
            <input type="text" name="usr_firstname" value=""
size="10" ></input>
        </td>
        </tr>

        <!-- Γραμμή εισαγωγής Επωνύμου -->
        <tr>
        <td class="Label" style="background-color:
transparent;">
            <ul><li style="color: white;">Επώνυμο:</li></ul>
        </td>
        <td class="InputField">
            <input type="text" name="usr_lastname" value=""
size="10" ></input>
        </td>
        </tr>

        <!-- Γραμμή εισαγωγής email -->
        <tr>
        <td class="Label" style="background-color:
transparent;">
            <ul><li style="color: white;">E-mail:</li></ul>
        </td>
        <td class="InputField">
            <input type="text" name="usr_email" value=""
size="20" ></input>
        </td>
        </tr>

        <!-- Γραμμή εισαγωγής τηλεφώνου -->
        <tr>
        <td class="Label" style="background-color:
transparent;">
            <ul><li style="color: white;">Τηλέφωνο:</li></ul>
        </td>
        <td>
            <input type="text" name="usr_phone" value=""
size="10" ></input>
        </td>
        </tr>

        <!-- Γραμμή εισαγωγής ρόλου -->
        <tr>
        <td class="Label" style="color: white; background-
color : transparent;">
            <input type="radio" name="usr_admin"
value="True">Διαχειριστής</input>
        </td>

```

```

        <td class="Label" style="color: white; background-
color : transparent;">
        <input type="radio" name="usr_admin" value="False"
checked>Μαθητής</input>
    </td>
</tr>
<tr>
    <td><p></p></td>
    <td>
    </td>
</tr>
<tr>
<td>
<td>
    <input class="dbButton" type="submit"
name="AddRecord" value="Αποθήκευση"
onclick="ResetFieldChange('UnsavedChanges');"></input>
    <input class="dbButton" type="reset"
value="Ακύρωση" onclick="collapseElement('NewRecord') "></input>
    </td>
</tr>
</table>
</form>
</div>

<?php
// Αποθήκευση εγγραφής
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    if (isset ($_POST["AddRecord"])) // Εάν πατήθηκε το κουμπί
αποθήκευση
    {
        $usr_id = $_POST['usr_id'];
        $usr_username = $_POST['usr_username'];
        $usr_password = $_POST['usr_password'];
        $usr_lastname = $_POST['usr_lastname'];
        $usr_firstname = $_POST['usr_firstname'];
        $usr_email = $_POST['usr_email'];
        $usr_phone = $_POST['usr_phone'];
        $usr_admin = $_POST['usr_admin'];
        $usr_contributor = "False";
        if (strcmp($usr_admin, "False") == 0)
            $usr_contributor = "True";

        // Εύρεση χρήστη που έχει συνδεθεί
        $LoggedInUser = new User();
        $rec_Creator = $LoggedInUser->GetLoggedInUser();
        $rec_Timestamp = date('Y-m-d H:i:s');

        // Επερώτηση εισαγωγής στοιχείων χρήστη στη βάση δεδομένων
        $strSQL = "INSERT INTO users "
            "(Usr_Username, Usr_Password, Usr_Lastname,
Usr_Firstname, Usr_Email, "
            "Usr_Phone, Usr_Role_Admin,
Usr_Role_Contributor) VALUES ("

```



```

        // Επικεφαλίδα λίστας χρηστών
        echo "<tr>";
        echo "<td></td>";
        echo "<td><p class=\"Header\">\".\"ID\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Username\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Password\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Επώνυμο\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Όνομα\".</p></td>";
        echo "<td><p class=\"Header\">\".\"E-
mail\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Τηλέφωνο\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Διαχειριστής\".</p></td>";
        echo "<td><p
class=\"Header\">\".\"Μαθητής\".</p></td>";
        echo "</tr>";

        // Δημιουργία διαδοχικών γραμμών με τα δεδομένα
των χρηστών
        $i = 1;
        while ($row = mysqli_fetch_array($result))
        {
            $usr_id = $row['Usr_ID'];
            $usr_username = $row['Usr_Username'];
            $usr_password = $row['Usr_Password'];
            $usr_lastname = $row['Usr_Lastname'];
            $usr_firstname = $row['Usr_Firstname'];
            $usr_email = $row['Usr_Email'];
            $usr_phone = $row['Usr_Phone'];
            $usr_admin = $row['Usr_Role_Admin'];
            $usr_contributor =
            $row['Usr_Role_Contributor'];

            echo "<tr>";
            // 1η στήλη (checkbox επιλογής γραμμής για
μετέπειτα διαγραφή)
            echo "<td>";
            echo "<input type=\"checkbox\"
name=\"Data_Array[$i][recSelected]\"></input>";
            echo "</td>";
            // 2η στήλη (id της εγγραφής όπως έχει
καταχωρηθεί στη βάση δεδομένων)
            echo "<td>";
            echo "<input type=\"text\" readonly
name=\"Data_Array[$i][usr_id]\" value=\"\".$usr_id.\"\"
size=\"1\"></input>";
            echo "</td>";
            // 3η στήλη (username)
            echo "<td>";
            echo "<input type=\"text\"
name=\"Data_Array[$i][usr_username]\" value=\"\".$usr_username.\"\"

```

```

size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 4η στήλη (password)
    echo "<td>";
    echo "<input type='text'
name='Data_Array[$i][usr_password]' value='".$usr_password."'
size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 5η στήλη (Επώνυμο χρήστη)
    echo "<td>";
    echo "<input type='text'
name='Data_Array[$i][usr_lastname]' value='".$usr_lastname."'
size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 6η στήλη (Όνομα χρήστη)
    echo "<td>";
    echo "<input type='text'
name='Data_Array[$i][usr_firstname]' value='".$usr_firstname."'
size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 7η στήλη (e-mail χρήστη)
    echo "<td>";
    echo "<input type='text'
name='Data_Array[$i][usr_email]' value='".$usr_email."'
size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 8η στήλη (τηλέφωνο χρήστη)
    echo "<td>";
    echo "<input type='text'
name='Data_Array[$i][usr_phone]' value='".$usr_phone."'
size="10"
onchange="HandleFieldChange('UnsavedChanges')"></input>;
    echo "</td>";
    // 9η στήλη (ένδειξη radio button ανάλογα με
το αν ο χρήστης είναι Διαχειριστής ή Μαθητής)
    // Ανάλογα επιλέγεται το κατάλληλο radio
button
    if (strcmp ($usr_admin, "True") == 0)
    {
    echo "<td>";
        echo "<input type='radio'
name='Data_Array[$i][usr_admin]' value='True' checked
style='margin-left: 40px;'></input>";
        echo "</td>";
        echo "<td>";
            echo "<input type='radio'
name='Data_Array[$i][usr_admin]' value='False' style='margin-
left: 40px;'></input>";
        echo "</td>";
    }
    else

```



```

        {
            echo "<td>";
            echo "<input type=\"radio\"
name=\"Data_Array[$i][usr_admin]\" value=\"True\" style=\"margin-left:
40px; color: #fff;\"></input>";
            echo "</td>";
            echo "<td>";
            echo "<input type=\"radio\"
name=\"Data_Array[$i][usr_admin]\" value=\"False\" checked
style=\"margin-left: 40px;\"></input>";
            echo "<td>";
        }
        echo "</tr>";
        $i++;
    }
    echo "</table>";
}
else
{
    // Εάν ο χρήστης δεν είναι διαχειριστής, εμφάνιση
προειδοποιητικού μηνύματος
    header("Location: message.php?msgid=7");
}
?>
</div>      <!-- formarea -->
<?php
    // Περιοχή κουμπιών εντολών (Διαγραφή, Νέος,
Αποθήκευση, Επαναφορά)
    echo "<div class=\"formarea\" style=\"height: 120px;
width: 850px; overflow-y: hidden; border-width: 0px;\">";
    echo "<br><br>";
    echo "<input class=\"dbButton\" type=\"submit\"
name=\"Delete\" value=\"Διαγραφή\" onclick=\"return
action_confirm('Delete selected records?');\"/>";
    echo "<input class=\"dbButton\" type=\"reset\"
name=\"Reset\" value=\"Επαναφορά\" style=\"margin-left: 20px;\"/>";
    echo "<input class=\"dbButton\" type=\"button\"
onclick=\"expandElement('NewRecord');\" value=\"Νέος\" style=\"margin-
left: 313px;\">";
    echo "<input class=\"dbButton\" type=\"submit\"
name=\"Save\" value=\"Αποθήκευση\"
onclick=\"ResetFieldChange('UnsavedChanges');\" style=\"margin-left:
20px;\"/>";

    echo "</div>";
    echo "<div class=\"ButtonHome\"><a
href=\"index.php\"><img src=\"images/home.png\"
alt=\"Αρχική\"/></a></div>";

?>
</form>
</div>      <!-- div class: section -->
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    // Εάν έχει πατηθεί το πλήκτρο "Αποθήκευση"
    if (isset ($_POST["Save"]))

```

```

    {
        $TotalErrors = 0;
        // Εξαγωγή των δεδομένων από το array $_POST και ανάθεσή
        τους σε αντίστοιχες μεταβλητές
        foreach ($_POST["Data_Array"] as $dt)
        {
            $usr_id = $dt['usr_id'];
            $usr_username = stripslashes($dt['usr_username']);
            $usr_username = mysqli_real_escape_string($conn,
$usr_username);
            $usr_password = stripslashes($dt['usr_password']);
            $usr_password = mysqli_real_escape_string($conn,
$usr_password);
            $usr_lastname = stripslashes($dt['usr_lastname']);
            $usr_lastname = mysqli_real_escape_string($conn,
$usr_lastname);
            $usr_firstname = stripslashes($dt['usr_firstname']);
            $usr_firstname = mysqli_real_escape_string($conn,
$usr_firstname);
            $usr_email = stripslashes($dt['usr_email']);
            $usr_email = mysqli_real_escape_string($conn,
$usr_email);
            $usr_phone = stripslashes($dt['usr_phone']);
            $usr_phone = mysqli_real_escape_string($conn,
$usr_phone);
            $usr_admin = stripslashes($dt['usr_admin']);
            $usr_admin = mysqli_real_escape_string($conn,
$usr_admin);
            if (strcmp ($usr_admin, "True") == 0)
                $usr_contributor = "False";
            else
                $usr_contributor = "True";

            // Επερώτηση ενημέρωσης των στοιχείων της εγγραφής του
            χρήστη
            $strSQL = "UPDATE users SET ".
                "Usr_Username = '$usr_username', ".
                "Usr_Password = '$usr_password', ".
                "Usr_Lastname = '$usr_lastname', ".
                "Usr_Firstname = '$usr_firstname', ".
                "Usr_Email = '$usr_email', ".
                "Usr_Phone = '$usr_phone', ".
                "Usr_Mobile = '', ".
                "Usr_Role_Admin = '$usr_admin', ".
                "Usr_Role_Contributor =
'$usr_contributor', ".
                "Usr_Status = '' ".
                "WHERE Usr_ID = $usr_id";

            $result = mysqli_query($conn, $strSQL);
            if (!$result)
                $TotalErrors++;
        }
        if ($TotalErrors > 0)
            header("Location: message.php?msgid=30"); // Μήνυμα
            αποτυχίας αποθήκευσης

```

```

        else
            header("Location: message.php?msgid=26"); // Μήνυμα
επιτυχούς αποθήκευσης
        }

        // Εάν πατήθηκε το πλήκτρο "Διαγραφή"
        if (isset ($_POST["Delete"]))
        {
            $TotalErrors = 0;
            // Για όλες τις εγγραφές χρηστών
            foreach ($_POST["Data_Array"] as $dt)
            {
                // Εντοπίζονται οι εγγραφές για τις οποίες έχει
επιλεγεί το
                // check box της 1ης στήλης
                if (isset($dt['recSelected']))
                {
                    // Επερώτηση διαγραφής εγγραφής
                    $strSQL = "DELETE FROM users WHERE Usr_ID =
". $dt['usr_id'];
                    $result = mysqli_query($conn, $strSQL);
                    if (!$result)
                        $TotalErrors++; // Καταγραφή κατάστασης λάθους
με αύξηση της τιμής της μεταβλητής
                }
                if ($TotalErrors > 0)
                    header("Location: message.php?msgid=27"); // Μήνυμα
αποτυχίας ενημέρωσης στοιχείων χρήστη
                else
                    header("Location: message.php?msgid=28"); // Μήνυμα
επιτυχούς ενημέρωσης
                }
            } // Ολοκλήρωσης διαδικασίας διαγραφής
        }
    ?>
</body>
</html>

```

Κώδικας 12. Ενδεικτικός κώδικας σελίδας θεωρίας

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <link rel="stylesheet"
href="styles/styles.css"><title>Τίτλος</title>
        <title></title>
    </head>

    <body>
        <h1>ΤΟ ΒΟΗΘΗΤΙΚΟ ΡΗΜΑ "εἶμι"</h1>
        <p>Παρατηρήσεις:</p>
        <ol>
            <li>Όταν το ρήμα εἶμι είναι σύνθετο, ανεβάζει τον τόνο στην
Οριστική Ενεστώτα και στο β' ενικό - πληθυντικό Προστακτικής (π.χ εἶμι ->
σύνειμι).</li>

```

Οι τόνοι της οριστικής του εἶμι, εκτός από το β' ενικό, δεν είναι σταθεροί.

<h2>ΚΛΙΣΗ ΤΟΥ ΒΟΗΘΗΤΙΚΟΥ ΡΗΜΑΤΟΣ "εἶμι":</h2>
<table>
<tr>
<td colspan="2"><p class="tabletext">ΟΡΙΣΤΙΚΗ ΕΝΕΣΤΩΤΑ</p></td>
</tr>
<tr>
<td><p class="tabletext">εἶμι</p></td><td><p class="tabletext">(=εἶμαι)</p></td>
</tr>
<tr>
<td><p class="tabletext">εἶ</p></td><td><p class="tabletext">(=εἶσαι)</p></td>
</tr>
<tr>
<td><p class="tabletext">ἐστί (v)</p></td><td><p class="tabletext">(=εἶναι)</p></td>
</tr>
<tr>
<td><p class="tabletext">ἐσμέν</p></td><td><p class="tabletext">(=εἶμαστε)</p></td>
</tr>
<tr>
<td><p class="tabletext">ἐστέ</p></td><td><p class="tabletext">(=εἶσθε)</p></td>
</tr>
<tr>
<td><p class="tabletext">εἶσι (v)</p></td><td><p class="tabletext">(=εἶναι)</p></td>
</tr>
</table>
<h2>ΔΙΑΙΡΕΣΗ ΤΩΝ ΡΗΜΑΤΩΝ</h2>
<p>Τα ρήματα της αρχαίας ελληνικής διαλέκτου χωρίζονται στις παρακάτω κατηγορίες :</p>
<h3>Α' ΣΥΖΥΓΙΑ (ρήματα σε -ω)</h3>
<p>1) ΦΩΝΗΕΝΤΟΛΗΚΤΑ :</p>

α) Ασυναίρετα (βαρύτονα) :

λύ-ω, παιδεύ-ω
β) Συνηρημένα (περισπώμενα) :

σε -άω (τιμά-ω = ᾶ)

σε -έω (ποιέ-ω = ῶ)

σε -όω (δηλό-ω = ῶ)

<p>2) ΣΥΜΦΩΝΟΛΗΚΤΑ :</p>

α) Αφωνόληκτα (βαρύτονα) :

- ουρανικόληκτα π.χ ἄγ-ω, τάσσ-ω (θ. ταγ-)

- χειλικόληκτα π.χ γράφ-ω, βλάπτ-ω (θ. βλαβ-)

```

) </li>
      <br>- οδοντικόληκτα π.χ σπεύδ-ω, ἐλπίζ-ω (θ.ἐλπιδ-
</li>
      <li>β) Ημίφωνα (βαρύτονα) :
      <br>- ενρινόληκτα π.χ νέμ-ω, μέν-ω
      <br>- υγρόληκτα π.χ ἀγέλ-λ-ω, φέρ-ω</li>
</ul>
<h2>Β' ΣΥΖΥΓΙΑ (ρήματα σε -μι)</h2>
<p><strong>2) ΣΥΜΦΩΝΟΛΗΚΤΑ :</strong></p>
<ul>
  <li>1) ΦΩΝΗΕΝΤΟΛΗΚΤΑ:
    <br>ἴ-στη-μι
    <br>ἴ-η-μι
    <br>τί-θη-μι
    <br>δί-δω-μι</li>
  <li>2) ΣΥΜΦΩΝΟΛΗΚΤΑ:
    <br>δείκ-νυ-μι
    <br>μείγ-νυ-μι
    <br>ᾔμ-νυ-μι
    <br>πιάρ-νυ-μαι</li>
</ul>
<h2>ΚΛΙΣΗ ΒΑΡΥΤΟΝΩΝ ΡΗΜΑΤΩΝ</h2>
<table>
  <tr>
    <td colspan="4"><p class="tabletext">ΕΝΕΡΓΗΤΙΚΗ
ΦΩΝΗ</p></td><td><p class="tabletext"></p></td><td><p
class="tabletext"></p></td><td><p class="tabletext"></p></td>
  </tr>
  <tr>
    <td colspan="4"><p class="tabletext">Οριστική
ΕΝΕΣΤΩΤΑ</p></td><td><p class="tabletext"></p></td><td><p
class="tabletext"></p></td><td><p class="tabletext"></p></td>
  </tr>
  <tr>
    <td><p class="tabletext">λύ-ω</p></td><td><p
class="tabletext">πείθ-ω</p></td><td><p class="tabletext">τρέπ-
ω</p></td><td><p class="tabletext">λήγ-ω</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">λύ-εις</p></td><td><p
class="tabletext">πείθ-εις</p></td><td><p class="tabletext">τρέπ-
εις</p></td><td><p class="tabletext">λήγ-εις</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">λύ-ει</p></td><td><p
class="tabletext">πείθ-ει</p></td><td><p class="tabletext">τρέπ-
ει</p></td><td><p class="tabletext">λήγ-ει</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">λύ-ομεν</p></td><td><p
class="tabletext">πείθ-ομεν</p></td><td><p class="tabletext">τρέπ-
ομεν</p></td><td><p class="tabletext">λήγ-ομεν</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">λύ-ετε</p></td><td><p
class="tabletext">πείθ-ετε</p></td><td><p class="tabletext">τρέπ-
ετε</p></td><td><p class="tabletext">λήγ-ετε</p></td>
  </tr>

```

```

        </tr>
        <tr>
            <td><p class="tabletext">λύ-ουσι (ν) </p></td><td><p
class="tabletext">πείθ-ουσι (ν) </p></td><td><p class="tabletext">τρέπ-
ουσι (ν) </p></td><td><p class="tabletext">λήγ-ουσι (ν) </p></td>
        </tr>
    </table>
<p></p>
<table>
    <tr>
        <td colspan="4"><p class="tabletext">ΕΝΕΡΓΗΤΙΚΗ
ΦΩΝΗ</p></td><td><p class="tabletext"></p></td><td><p
class="tabletext"></p></td><td><p class="tabletext"></p></td>
    </tr>
    <tr>
        <td colspan="4"><p class="tabletext">Οριστική
ΜΕΛΛΟΝΤΑ</p></td><td><p class="tabletext"></p></td><td><p
class="tabletext"></p></td><td><p class="tabletext"></p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σω</p></td><td><p
class="tabletext">πεί-σω</p></td><td><p class="tabletext">τρέ-
ψω</p></td><td><p class="tabletext">λή-ξω</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σεις</p></td><td><p
class="tabletext">πεί-σεις</p></td><td><p class="tabletext">τρέ-
ψεις</p></td><td><p class="tabletext">λή-ξεις</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σει</p></td><td><p
class="tabletext">πεί-σει</p></td><td><p class="tabletext">τρέ-
ψει</p></td><td><p class="tabletext">λή-ξει</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σομεν</p></td><td><p
class="tabletext">πεί-σομεν</p></td><td><p class="tabletext">τρέ-
ψομεν</p></td><td><p class="tabletext">λή-ξομεν</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σειτε</p></td><td><p
class="tabletext">πεί-σειτε</p></td><td><p class="tabletext">τρέ-
ψειτε</p></td><td><p class="tabletext">λή-ξειτε</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">λύ-σουσι (ν) </p></td><td><p
class="tabletext">πεί-σουσι (ν) </p></td><td><p class="tabletext">τρέ-
ψουσι (ν) </p></td><td><p class="tabletext">λή-ξουσι (ν) </p></td>
    </tr>
</table>
<h2>ΒΑΣΙΚΟΙ ΚΑΝΟΝΕΣ ΣΧΗΜΑΤΙΣΜΟΥ ΤΟΥ ΜΕΛΛΟΝΤΑ των βαρύτωνων
ρημάτων:</h2>
<p>Ο μέλλοντας των βαρύτωνων ρημάτων σχηματίζεται ως εξής:</p>
<ol>

```

για τα Φωνηεντόληκτα ρήματα, ο μέλλοντας σχηματίζεται με την προσθήκη του χρονικού χαρακτήρα "σ" ανάμεσα στο θέμα και τις καταλήξεις του ενεστώτα (πχ. λύω -> λύσω).

για τα Ουρανικόληκτα, σχηματίζεται ως εξής:

κ+σ = ξ (πχ. διώκω -> διώξω)

γ+σ = ξ (πχ. ἄγω -> ἄξω)

χ+σ = ξ (πχ. βρέχω -> βρέξω)

τι+σ = ξ (πχ. ταράττω -> ταράξω)

για τα Χειλικόληκτα, σχηματίζεται ως εξής:

π+σ = ψ (πχ. βλέπω -> βλέψω)

β+σ = ψ (πχ. ἀμείβω -> ἀμείψω)

φ+σ = ψ (πχ. γράφω -> γράψω)

πι+σ = ψ (πχ. βλάπτω -> βλάψω)

για τα Οδοντικόληκτα, σχηματίζεται ως εξής:

θ+σ = σ (πχ. πείθω -> πείσω)

δ+σ = σ (πχ. φράζω -> φράσω)

<h3>ΚΛΙΣΗ ΟΡΙΣΤΙΚΗ ΕΝΕΣΤΩΤΑ ΚΑΙ ΜΕΛΛΟΝΤΑ ΤΩΝ ΕΞΗΣ ΡΗΜΑΤΩΝ:</h3>

<tr>	<td><p class="tabletext">Ενεστώτας</p></td><td><p class="tabletext">Μέλλοντας</p></td>	
</tr>	<tr>	<td><p class="tabletext">πλάτιω</p></td><td><p class="tabletext">πλάσω</p></td>
</tr>	<tr>	<td><p class="tabletext">πλάττεις</p></td><td><p class="tabletext">πλάσεις</p></td>
</tr>	<tr>	<td><p class="tabletext">πλάττει</p></td><td><p class="tabletext">πλάσει</p></td>
</tr>	<tr>	<td><p class="tabletext">πλάτιομεν</p></td><td><p class="tabletext">πλάσομεν</p></td>
</tr>	<tr>	<td><p class="tabletext">πλάττετε</p></td><td><p class="tabletext">πλάσετε</p></td>
</tr>	<tr>	<td><p class="tabletext">πλάττουσιν</p></td><td><p class="tabletext">πλάσουσιν</p></td>
</tr>	</table>	<p></p>
<table>	<tr>	<td><p class="tabletext">Ενεστώτας</p></td><td><p class="tabletext">Μέλλοντας</p></td>
</tr>		

<p>κρύπτω</p>	<p>κρύψω</p>
<p>κρύπτεις</p>	<p>κρύψεις</p>
<p>κρύπτει</p>	<p>κρύψει</p>
<p>κρύπτομεν</p>	<p>κρύψομεν</p>
<p>κρύπτετε</p>	<p>κρύψετε</p>
<p>κρύπτουσι (ν)</p>	<p>κρύψουσι (ν)</p>

<p>Ενεσιώτας</p>	<p>Μέλλονιας</p>
<p>φυλάτιω</p>	<p>φυλάξω</p>
<p>φυλάτιεις</p>	<p>φυλάξεις</p>
<p>φυλάτιει</p>	<p>φυλάξει</p>
<p>φυλάτιομεν</p>	<p>φυλάξομεν</p>
<p>φυλάτιετε</p>	<p>φυλάξετε</p>
<p>φυλάτιουσι (ν)</p>	<p>φυλάξουσι (ν)</p>


```

<p></p>
<table>
  <tr>
    <td><p class="tabletext">Ενεσιώτας</p></td><td><p
class="tabletext">Μέλλοντιας</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττω</p></td><td><p
class="tabletext">ἀρμόσω</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττεις</p></td><td><p
class="tabletext">ἀρμόσεις</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττει</p></td><td><p
class="tabletext">ἀρμόσει</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττομεν</p></td><td><p
class="tabletext">ἀρμόσομεν</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττετε</p></td><td><p
class="tabletext">ἀρμόσετε</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀρμόττουσι (v)</p></td><td><p
class="tabletext">ἀρμόσουσι (v)</p></td>
  </tr>
</table>
<p></p>
<table>
  <tr>
    <td><p class="tabletext">Ενεσιώτας</p></td><td><p
class="tabletext">Μέλλοντιας</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀλλάττω</p></td><td><p
class="tabletext">ἀλλάξω</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀλλάττεις</p></td><td><p
class="tabletext">ἀλλάξεις</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀλλάττει</p></td><td><p
class="tabletext">ἀλλάξει</p></td>
  </tr>
  <tr>
    <td><p class="tabletext">ἀλλάττομεν</p></td><td><p
class="tabletext">ἀλλάξομεν</p></td>
  </tr>
  <tr>

```

```
        <td><p class="tabletext">ἀλλάττετε</p></td><td><p
class="tabletext">ἀλλάξετε</p></td>
    </tr>
    <tr>
        <td><p class="tabletext">ἀλλάττουσι (v) </p></td><td><p
class="tabletext">ἀλλάξουσι (v) </p></td>
    </tr>
</table>
<p></p>
</body>
</html>
```

Βιβλιογραφία

1. Liu, X., Heo, J., & Sha, L. (2009). *Modeling 3-Tiered Web Applications*. University of Illinois.
2. Pallud, J., & Straub, D. (2014). Effective website design for experience-influenced environments. *Information & Management*, 359–373.
3. Rumbaugh, J., Jacobson, I., & Booch, G. (1999). *The Unified Modeling Language Reference Manual*. ADDISON-WESLEY.
4. Ε. Λιακοπούλου, Β. Ν. (2010). Το ΕΛ/ΛΑΚ στην Εκπαίδευση. *Workshop on Informatics in Education* (σ. 20). Tripoli: GREEK COMPUTER SOCIETY(GCS).
5. Νικολουδάκης, Ε. Η. (2010). Δημιουργία Σκαλωσιάς με τη βοήθεια των ΤΠΕ σε ένα Δομημένης Μορφής Φύλλο Εργασίας. *Workshop on Informatics in Education* (σ. 118). Tripoli: GREEK COMPUTER SOCIETY(GCS).