



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
Πρόγραμμα Μεταπτυχιακών Σπουδών

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ

Τίτλος Διατριβής	ΕΚΠΑΙΔΕΥΤΙΚΗ ΔΙΑΔΙΚΤΥΑΚΗ ΕΦΑΡΜΟΓΗ ΜΕ ΘΕΜΑΤΟ ΜΟΥΣΕΙΟ ΤΗΣ ΑΚΡΟΠΟΛΗΣ A WEB-BASED TUTORING SYSTEM FOR THE ACROPOLIS MUSEUM
Όνοματεπώνυμο Φοιτητή	ΣΟΦΙΑ ΣΙΩΡΟΥ
Πατρώνυμο	ΜΙΛΤΙΑΔΗΣ
Αριθμός Μητρώου	ΜΠΠΛ14074
Επιβλέπων	ΚΑΘΗΓΗΤΡΙΑ ΜΑΡΙΑ ΒΙΡΒΟΥ

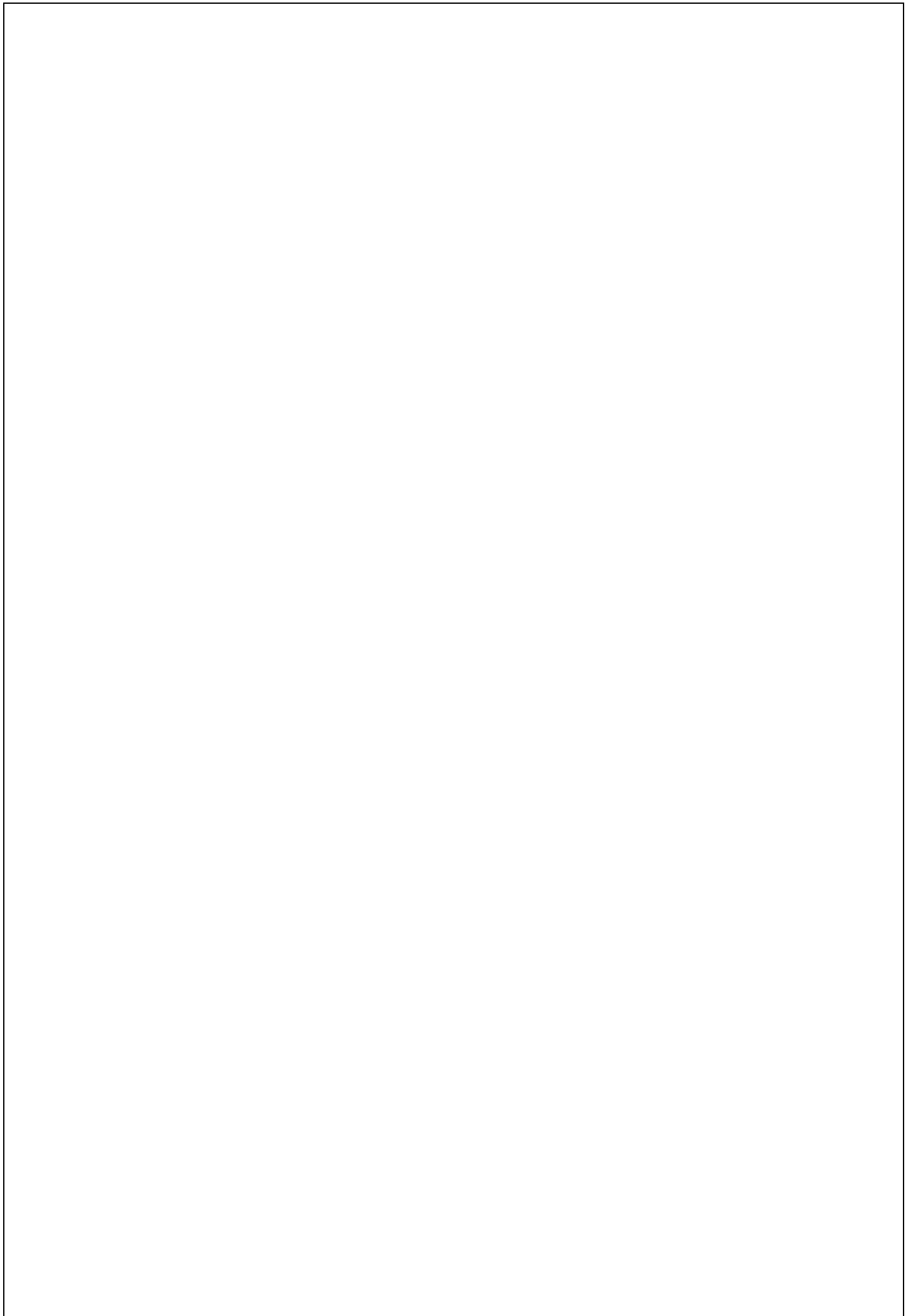
Ημερομηνία Παράδοσης **Οκτώβριος 2017**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)



Πίνακας περιεχομένων

Εισαγωγή	3
1. Ανασκόπηση πεδίου	4
1.1. Η περίπτωση του Smithsonian American Art Museum	4
1.1.1. Εφαρμογή και χρήση στη σχολική αίθουσα	5
1.2. Εγκυκλοπαίδεια Μείζονος Ελληνισμού	6
1.3. Μουσείο Κυκλαδικής Τέχνης	9
1.4. Κέντρο George Pompidou, Παρίσι	11
2. Ανάλυση και Σχεδιασμός Εφαρμογής	12
2.1. Ανάλυση Απαιτήσεων	12
2.1.1. Βασικές Απαιτήσεις	12
2.1.2. Ειδικές Απαιτήσεις	13
2.2. Μοντελοποίηση απαιτήσεων	15
2.2.1. Διάγραμμα περιπτώσεων χρήσης	15
2.2.2. Διαγράμματα δραστηριοτήτων	16
2.3. Σχεδιασμός βάσης δεδομένων	20
Εννοιολογικό μοντέλο	21
2.3.1. Σχεσιακό μοντέλο	24
2.3.2. Φυσικό μοντέλο	25
3. Υλοποίηση εφαρμογής	27
3.1. Αρχιτεκτονική 3-επιπέδων	27
3.1.1. Επίπεδο δεδομένων	28
3.1.2. Επίπεδο επιχειρησιακής λογικής	28
3.1.3. Επίπεδο παρουσίασης	28
3.2. Τεχνολογίες υλοποίησης	28
3.3. Περιγραφή υλοποίησης	31
4. Λειτουργικότητα συστήματος	34
4.1. Εφαρμογή διαχειριστή	34
4.2. Εφαρμογή επισκέπτη	35
4.2.1. Διεξαγωγή ασκήσεων	37

5. Μελλοντικές επεκτάσεις	40
Βιβλιογραφία.....	41
Παράρτημα Α - Κώδικας εφαρμογής	1

Πίνακας εικόνων

Εικόνα 1.	Βήματα δημιουργίας ενός ψηφιακού τεκμηρίου του Smithsonian American Art Museum.....	4
Εικόνα 2.	Περιβάλλον παραγωγής τεκμηριωτικού υλικού από τους μαθητές, μέσα από το περιβάλλον της εφαρμογής. (1) Επιλογή των πρωτογενών εγγράφων (2) τακτοποίησή τους ώστε το αποτέλεσμα να έχει χρονική και λογική συνέχεια (3) προβολή του αποτελέσματος στην οθόνη της αίθουσας.....	5
Εικόνα 3.	Η βασική οργάνωση των επιπέδων νοηματικής δυσκολίας	7
Εικόνα 4.	Τα πλαίσια τα οποία καλύπτει κάθε ενότητα της εγκυκλοπαίδειας.....	7
Εικόνα 5.	Σελίδα πρόσβασης στο υλικό της θεματικής ενότητας και του αντίστοιχου πλαισίου.....	8
Εικόνα 6.	Σελίδα του αντίστοιχου τεκμηριωτικού υλικού.....	9
Εικόνα 7.	Σελίδα παιχνιδιού παζλ.....	10
Εικόνα 8.	Σελίδα εμφάνισης πληροφοριών για το έκθεμα μετά την επιτυχή σύνθεση του παζλ.....	11
Εικόνα 9.	Άσκηση απλής επιλογής.....	14
Εικόνα 10.	Άσκηση πολλαπλής επιλογής.....	14
Εικόνα 11.	Άσκηση συμπλήρωσης κενών.....	14
Εικόνα 12.	Η αρχιτεκτονική τριών επιπέδων του προτεινόμενου συστήματος	27
Εικόνα 13.	Αρχική σελίδα της εφαρμογής.....	34
Εικόνα 14.	Εμφάνιση αποτελεσμάτων τεστ.....	35
Εικόνα 15.	Περιβάλλον εμφάνισης και ανάγνωσης πληροφοριακών σελίδων.....	36
Εικόνα 16.	Λίστα επιλογής των υποενοτήτων μιας ενότητας.....	36
Εικόνα 17.	Εμφάνιση εικονιδίου διαθέσιμου τεστ κάτω δεξιά στη σελίδα επισκόπησης του πληροφοριακού υλικού.....	37
Εικόνα 18.	Ενδεικτική οθόνη διεξαγωγής ασκήσεων διαφορετικών κλάσεων.....	38
Εικόνα 19.	Προτροπή σύνδεσης επισκέπτη με τα διαπιστευτήριά του.....	38
Εικόνα 20.	Οθόνη σύνδεσης επισκέπτη μέσω εισαγωγής των διαπιστευτηρίων του	39
Εικόνα 21.	Ηλεκτρονική φόρμα εγγραφής στο Εκπαιδευτικό λογισμικό.....	39

Πίνακας διαγραμμάτων

Διάγραμμα 1.	Διάγραμμα περίπτωσης χρήσης του συνολικού συστήματος.....	16
Διάγραμμα 2.	Διάγραμμα δραστηριοτήτων διαμόρφωσης του πληροφοριακού υλικού και των αντίστοιχων τεστ	17
Διάγραμμα 3.	Διάγραμμα δραστηριοτήτων μελέτης των πληροφοριακών οντοτήτων (σελίδων)	18
Διάγραμμα 4.	Διάγραμμα δραστηριοτήτων ενεργοποίησης και εκτέλεσης τεστ	19
Διάγραμμα 5.	Διάγραμμα δραστηριοτήτων επισκόπησης αποτελεσμάτων του τεστ	20
Διάγραμμα 6.	Διάγραμμα Οντοτήτων - Συσχετίσεων	20
Διάγραμμα 7.	Σχεσιακό μοντέλο του εκπαιδευτικού λογισμικού	25
Διάγραμμα 8.	Επιδόσεις δοσοληψιών ανάγνωσης του MySQL Server 5.6 (Κάτοχος εικόνας: MySQL)	30
Διάγραμμα 9.	Δυνατότητες διεκπεραίωσης δοσοληψιών ανάγνωσης – εγγραφής για το MySQL Server 5.6 (Κάτοχος εικόνας: MySQL).....	30

Ευρετήριο πινάκων

Πίνακας 1.	Οντότητα “User”	21
Πίνακας 2.	Οντότητα “Exercise”	22
Πίνακας 3.	Οντότητα “question_text”	22
Πίνακας 4.	Οντότητα “question_options”	23
Πίνακας 5.	Η οντότητα “results”	23
Πίνακας 6.	Οντότητα “theory”	24

Ευρετήριο κώδικα προγραμματισμού

Κώδικας 1. Πρόταση SQL για δημιουργία του πίνακα “Users”	25
Κώδικας 2. Πρόταση SQL για δημιουργία του πίνακα “exercises”	25
Κώδικας 3. Πρόταση SQL για δημιουργία του πίνακα “question_text”	26
Κώδικας 4. Πρόταση SQL για δημιουργία του πίνακα “question_options”	26
Κώδικας 5. Πρόταση SQL για δημιουργία του πίνακα “theory”	26
Κώδικας 6. Πρόταση SQL για δημιουργία του πίνακα “results”	26
Κώδικας 7: Κώδικας config.php	33
Κώδικας 8. theory.php.....	1
Κώδικας 9. exercise.php.....	4
Κώδικας 10. manageusers.php	13
Κώδικας 11. Ενδεικτικός κώδικας σελίδας παροχής πληροφοριακών στοιχείων	22

Περίληψη

Τα πολιτιστικά ιδρύματα και ιδιαίτερα του μουσεία έχουν στη διάθεσή τους την απαραίτητη τεχνολογία για την προβολή και τη διάδοση των συλλογών τους. Τα σύγχρονα εργαλεία καθιστούν δυνατή με σχετική ευκολία τη δημιουργία διαδραστικών σελίδων, στις οποίες ο επισκέπτης μπορεί να έρθει σε επαφή με το εκθεσιακό υλικό του μουσείου, αλλά και επιπλέον πολυμεσικό υλικό, μετατρέποντας έτσι την περιήγηση σε μια ενδιαφέρουσα και ελκυστική εμπειρία.

Η ποικιλία των μορφών στις οποίες ένα μουσείο μπορεί να διαθέσει τις πληροφορίες και το υλικό που διαθέτει (κείμενο, αφηγήσεις, βίντεο) δημιουργεί ένα πλαίσιο διάχυσης και αφομοίωσης της γνώσης, το οποίο υπό άλλες συνθήκες θα ήταν δύσκολο να επιτευχθεί. Παράλληλα, η φιλοσοφία των υπερσυνδέσεων του διαδικτύου προσφέρει τη δυνατότητα συνδυασμού γνώσεων από διαφορετικές πηγές, υποβοηθώντας με αυτό τον τρόπο τη μαθησιακή διαδικασία, με τρόπο ο οποίος επεκτείνει τη συμβατική μελέτη του πληροφοριακού υλικού.

Με δεδομένο ότι ένα μουσείο διαθέτει σε οργανωμένη μορφή τις πηγές της γνώσης και τους μηχανισμούς διάχυσής της, δημιουργούνται οι προϋποθέσεις για την ανάπτυξη ενός ολοκληρωμένου περιβάλλοντος, στο οποίο ο επισκέπτης θα μπορεί να δοκιμάζει τις γνώσεις του για τα εκθέματα του μουσείου μέσω διαδραστικών τεστ, τα οποία μπορεί να εκτελέσει μετά τη μελέτη του διαθέσιμου υλικού. Η προσέγγιση αυτή δε χρησιμοποιείται ευρέως, εξ όσων έδειξε η σύντομη έρευνα στο πλαίσιο της εργασίας αυτής.

Με το δεδομένο αυτό, η εργασία αυτή επεκτείνει τους μηχανισμούς διάχυσης της γνώσης ενός μουσείου, ολοκληρώνοντας στη διαδικασία της περιήγησης διαδραστικά τεστ, τα οποία οργανώνονται ανά θεματική ενότητα και υποενότητα των συλλογών του. Για την πρακτική εφαρμογή της μεθόδου επιλέχθηκε το μουσείο της Ακρόπολης. Για δύο ενότητες των εκθεμάτων του αναπτύχθηκε ένα περιβάλλον μελέτης των πληροφοριών που διατίθενται για τα εκθέματα. Το περιβάλλον αυτό συνδέθηκε με ηλεκτρονικές ερωτήσεις, τις οποίες ο επισκέπτης μπορεί να απαντήσει όταν μελετήσει την ενότητα. Επίσης, μπορεί να κάνει και ένα συνολικό τεστ ολοκληρώνοντας την περιήγησή του στο μουσείο. Φιλοδοξία είναι ο μηχανισμός αυτός να δημιουργήσει το ερέθισμα ώστε ο επισκέπτης (είτε μαθητής είτε περιηγητής) να μελετήσει βαθύτερα τα ιστορικά στοιχεία που διαθέτει το μουσείο, με στόχο την επίτευξη καλών επιδόσεων στα τεστ.

Abstract

Museums and cultural organizations in general may take advantage of contemporary technologies, in order to promote and disseminate information about their collections. The ease of use of modern tools make the development of informational pages a rather easy and cost efficient process. These pages may aggregate material from several sources, including text, narrations, still images and video. Such pages turn a visitor's navigation in the "virtual" museum to an attractive and interesting experience.

Museum own huge amount of invaluable information regarding their collections. This information variety is a cornerstone for information dissemination and knowledge transfer to a global audience, which would be impossible in other circumstances. Besides the hyperlink technology makes almost infinite combinations of information sources feasible, allowing the learner to extend conventional information by visiting related sources.

Given that museums are well organized information and knowledge sources, may turn their web sites to integrated learning environments, allowing visitors not only to study informational pages and material but also testing their knowledge by taking tests when they finish studying.

The proposed educational software will extend the informational mechanisms of the Museum of Acropolis, offering not only a study environment but also an integrated knowledge testing facility. For each informational unit a relevant test is maintained and presented to the visitor, in order to solve it and get a grade. At the end of the entire study process, the visitor may take a global test, testing the overall knowledge he acquired during his navigation to the museum's pages. The main idea behind this approach is to give the visitor an incentive to study historical and other relevant aspects in depth, in order to get the highest grade at the competitive tests.

Εισαγωγή

Η εργασία αυτή μελετά και υλοποιεί ένα περιβάλλον εκπαιδευτικού λογισμικού, το οποίο μπορεί να ενταχθεί στην πλατφόρμα ενός μουσείου, ώστε συμπληρωματικά στο πληροφοριακό υλικό το οποίο διαθέτει, ο επισκέπτης να μπορεί να εκτελεί τεστ ερωτήσεων σχετικών με κάθε ενότητα, ώστε να πιστοποιεί κατά πόσο αφομοίωσε το υλικό το οποίο μελέτησε.

Για το σκοπό αυτό, στο πλαίσιο της εργασίας μελετήθηκαν αντίστοιχες υλοποιήσεις και εξήχθησαν συμπεράσματα σχετικά με την υφιστάμενη εκπαιδευτική διαδικασία μουσείων και γενικότερα φορέων διάδοσης πολιτισμού. Βάσει των συμπερασμάτων αυτών, δημιουργήθηκε το λογισμικό το οποίο δίνει τη δυνατότητα στον επισκέπτη ενός μουσείου να απαντά ερωτήματα ποικίλων μορφών σε ηλεκτρονικό διαδικτυακό περιβάλλον. Σαν φορέας δοκιμής του λογισμικού επιλέχθηκε το Μουσείο Ακρόπολης, δύο θεματικές ενότητες του οποίου υλοποιήθηκαν ως ηλεκτρονικό μάθημα και αναπτύχθηκαν αντίστοιχα τεστ. Επίσης δημιουργήθηκε και ένα γενικευμένο (συλλογικό) τεστ, το οποίο ο επισκέπτης μπορεί να διεξάγει όταν ολοκληρώσει τη μελέτη των αντίστοιχων εννοιών.

Στο Κεφάλαιο 1 της εργασίας γίνεται ανασκόπηση του πεδίου. Έγινε διερεύνηση στους διαδικτυακούς τόπους διάφορων μουσείων παγκοσμίως αλλά και γενικότερα πολιτισμικών φορέων, ώστε να διαπιστωθεί εάν η προτεινόμενη προσέγγιση έχει υλοποιηθεί σε κάποιο απ' αυτά. Έγινε παρουσίαση της εκπαιδευτικής προσέγγισης των φορέων αυτών και εξάχθηκε το συμπέρασμα ότι σε κανένα από αυτά ο επισκέπτης δεν έχει την ευκαιρία να εξεταστεί πάνω στις γνώσεις που αποκόμισε.

Στο Κεφάλαιο 2 καταγράφεται η φάση ανάλυσης των απαιτήσεων από το λογισμικό και έγινε ο σχεδιασμός της, βάσει του μοντέλου καταρράκτη, το οποίο ακολουθήθηκε. Η ανάλυση συστήματος κατέγραψε τις προδιαγραφές που θα πρέπει να πληροί ένα τέτοιο σύστημα ώστε να είναι εύχρηστο και λειτουργικό. Στη φάση σχεδίασης έγινε η μοντελοποίηση των απαιτήσεων και καταγράφηκαν οι οντότητες του προβλήματος καθώς και οι συσχετίσεις τους.

Στο Κεφάλαιο 3 περιγράφονται οι τεχνολογίες που επιλέχθηκαν για την υλοποίηση της εφαρμογής και ο τρόπος που αυτές συνεργάζονται για τη δημιουργία του εκπαιδευτικού λογισμικού.

Στο Κεφάλαιο 4 περιγράφεται η λειτουργικότητα του τελικού συστήματος, ενώ δίνονται και οδηγίες για τη χρήση του, με περιγραφές των σελίδων που προβάλλονται τόσο στον επισκέπτη όσο και στο διαχειριστή.

Το Κεφάλαιο 5 αναφέρεται σε χαρακτηριστικά τα οποία δεν περιλήφθηκαν στη συγκεκριμένη υλοποίηση και τα οποία μπορούν να προσδώσουν πρόσθετη λειτουργικότητα και μεγαλύτερη προσαρμοστικότητα στο κοινό που απευθύνεται στο εκπαιδευτικό λογισμικό.

1. Ανασκόπηση πεδίου

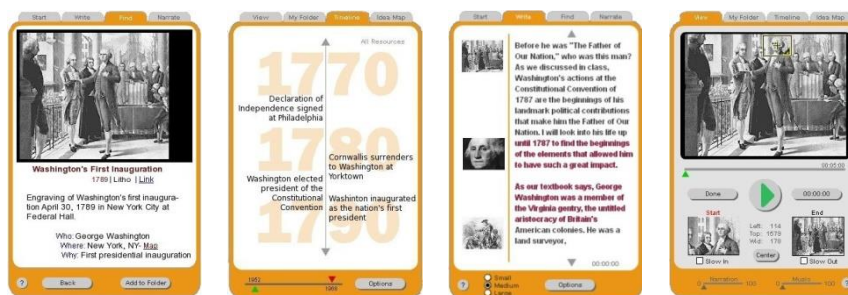
Η ευκολία χρήσης και η προσβασιμότητα των νέων τεχνολογιών και του διαδικτύου, δημιουργούν ένα ευνοϊκό πλαίσιο για την ανάπτυξη εφαρμογών παρουσίασης και προβολής φορέων όπως τα μουσεία και γενικότερα οντοτήτων που χειρίζονται θέματα γνώσης και πολιτισμού.

Τα μεγάλα μουσεία του κόσμου διαθέτουν πληρέστατους δικτυακούς τόπους για την προβολή των συλλογών τους, ενώ προωθημένες υλοποιήσεις δημιουργούν περιβάλλοντα εικονικής περιήγησης στους χώρους τους, δίνοντας στον επισκέπτη τους την αίσθηση ότι βρίσκονται στους χώρους του.

Στο πλαίσιο της έρευνας που διεξήχθη, διαπιστώθηκε ότι πολλά μουσεία ενσωματώνουν στις σελίδες τους ενότιες οι οποίες προσφέρουν εκπαιδευτική προσέγγιση στα εκθέματά τους και την ιστορία τους. Στις παραγράφους που ακολουθούν γίνεται αναφορά και συνοπτική περιγραφή ενδεικτικών υλοποιήσεων, με έμφαση στην εκπαιδευτική διάσταση των διαδικτυακών τόπων.

1.1. Η περίπτωση του Smithsonian American Art Museum

Το Smithsonian American Art Museum δίνει έμφαση στην προβολή της τεκμηρίωσης των εκθεμάτων του, δημιουργώντας ολοκληρωμένες συλλογές τεκμηρίων, ενώ δίνει ιδιαίτερη φροντίδα στην παρουσίαση του επεξεργασμένου υλικού σε καλαίσθητες και λειτουργικές σελίδες (Smithsonian American Art Museum, 2017). Το μουσείο δημιούργησε ένα δίκτυο συνεργαζόμενων φορέων, οι οποίοι συνεισφέρουν στην επεξεργασία του υλικού. Η συνέργεια αυτή ήταν αποτέλεσμα της εκτίμησης του όγκου του υλικού και του χρόνου, παράγοντες οι οποίοι θα καθιστούσαν το έργο αυτό μη βιώσιμο και μη ολοκληρώσιμο σε εύλογο χρονικό ορίζοντα. Τέτοιοι συνεργαζόμενοι φορείς είναι το Virginia Center for Digital History και το University of North Carolina's Documenting the American South archive.



Επιλογή
ενημερωτικών
πηγών

Ορισμός χρονικής
αλληλουχίας

Επιμέλεια του
κειμένου της
αφήγησης

Ρυθμίσεις
κινούμενων εικόνων
και βίντεο

Εικόνα 1. Βήματα δημιουργίας ενός ψηφιακού τεκμηρίου του Smithsonian American Art Museum

Πηγή: National Archives and Records Administration

Το λογισμικό αυτό απευθύνεται κυρίως σε εκπαιδευτικά ιδρύματα. Οι μαθητές έχουν στη διάθεσή τους ένα εργαλείο αναδίφησης στο υλικό καθώς και αναζήτησης όρων σε

αυτό μέσω του διαδικτύου. Η περιήγησή τους γίνεται μέσα σε τεράστιο όγκο πολυμεσικής πληροφορίας (ιστορικών εγγράφων, εικόνων, βίντεο και ηχογραφήσεων).

Με σκοπό την επαυξημένη διαδραστικότητα της εφαρμογής, δίνεται η δυνατότητα στους εκπαιδευόμενους να δημιουργήσουν δικές τους παρουσιάσεις του μουσειακού υλικού, χρησιμοποιώντας τις πρωτογενείς αυτές πολυμεσικές πληροφορίες και προσθέτοντας δικές τους αφηγήσεις, μοντάζ εικόνων και βίντεο και κείμενα τα οποία επιμελούνται οι ίδιοι.

1.1.1. Εφαρμογή και χρήση στη σχολική αίθουσα

Η εκπαιδευτική εφαρμογή του μουσείου επιτρέπει την ενσωμάτωση των διαφορετικών πληροφοριακών οντοτήτων σε σενάρια σε πραγματικό χρόνο. Οι μαθητές καταλήγουν στις πηγές τις οποίες θέλουν να ενσωματώσουν στην εργασία τους και μέσα από ένα γραφικό περιβάλλον τις σύρουν στις επιθυμητές θέσεις, ενώ προσθέτουν σε άλλες νέα κείμενα.



Εικόνα 2. Περιβάλλον παραγωγής τεκμηριωτικού υλικού από τους μαθητές, μέσα από το περιβάλλον της εφαρμογής. (1) Επιλογή των πρωτογενών εγγράφων (2) τακτοποίησή τους ώστε το αποτέλεσμα να έχει χρονική και λογική συνέχεια (3) προβολή του αποτελέσματος στην οθόνη της αίθουσας

Οι δυνατότητες αυτές μπορούν να χρησιμοποιηθούν και από τον καθηγητή για τη δημιουργία εκπαιδευτικού υλικού για κάποιο συγκεκριμένο θέμα το οποίο πραγματεύεται στο πλαίσιο ενός μαθήματος. Η εφαρμογή του μουσείου δίνει επιπλέον τη δυνατότητα της τροποποίησης του τεκμηριωτικού υλικού, μέσω της προσθήκης νέων πρωτογενών πληροφοριών από τις ψηφιακές βιβλιοθήκες του μουσείου ακόμη και κατά τη διάρκεια του μαθήματος.

Τα ψηφιακά τεκμήρια τα οποία παράγονται μέσω της εφαρμογής μπορούν να διατίθενται μέσω του διαδικτύου. Μέσα από το μηχανισμό αυτό οι μαθητές ενθαρρύνονται να προτείνουν τη δική τους άποψη για τα ιστορικά γεγονότα, ξεκινώντας την επεξεργασία μέσα στη σχολική τάξη και ολοκληρώνοντάς της στο σπίτι, μέσω τις διαδικτυακής τους σύνδεσης.

Η φιλοσοφία αυτή δημιουργεί ένα «δίκτυο γνώσεων», στο οποίο συμμετέχουν τόσο οι εκπαιδευτικοί μέσω της συνεισφοράς τους με εκπαιδευτικό υλικό, όσο και οι μαθητές, οι οποίοι προτείνουν τις δικές τους απόψεις. Στο δίκτυο αυτό, τα παραγόμενα τεκμήρια μπορούν να χρησιμοποιηθούν από άλλους εκπαιδευτικούς ή μαθητές και να ενσωματωθούν στη δική τους εργασία, ώστε να εκπληρώνονται οι στόχοι τους πιο γρήγορα.

Αυτό που παρατηρείται στο λογισμικό αυτό είναι η απουσία διαδραστικών τεστ τα οποία να μπορεί να διεκπεραιώσει ο μαθητής ή ευρύτερα ο επισκέπτης.

1.2. Εγκυκλοπαίδεια Μείζονος Ελληνισμού

Το Ίδρυμα Μείζονος Ελληνισμού ως φορέας διάδοσης του ελληνικού πολιτισμού Ο ΠΛΑΤΩΝ είναι μια ψηφιακή εγκυκλοπαίδεια προσβάσιμη από το διαδίκτυο. Με αφορμή την πλατωνική σκέψη διαθέτει ένα ευρύ όγκο πληροφοριών για τον αρχαίο πολιτισμό. Σημαντική συνεισφορά του έργου είναι η συσχέτιση των φιλοσοφικών προσεγγίσεων με την πολιτεία, την κοινωνική δράση, την οικονομία, την ιστορία και την καθημερινή ζωή της εποχής (Η Εγκυκλοπαίδεια του Πλάτωνα, 2017).

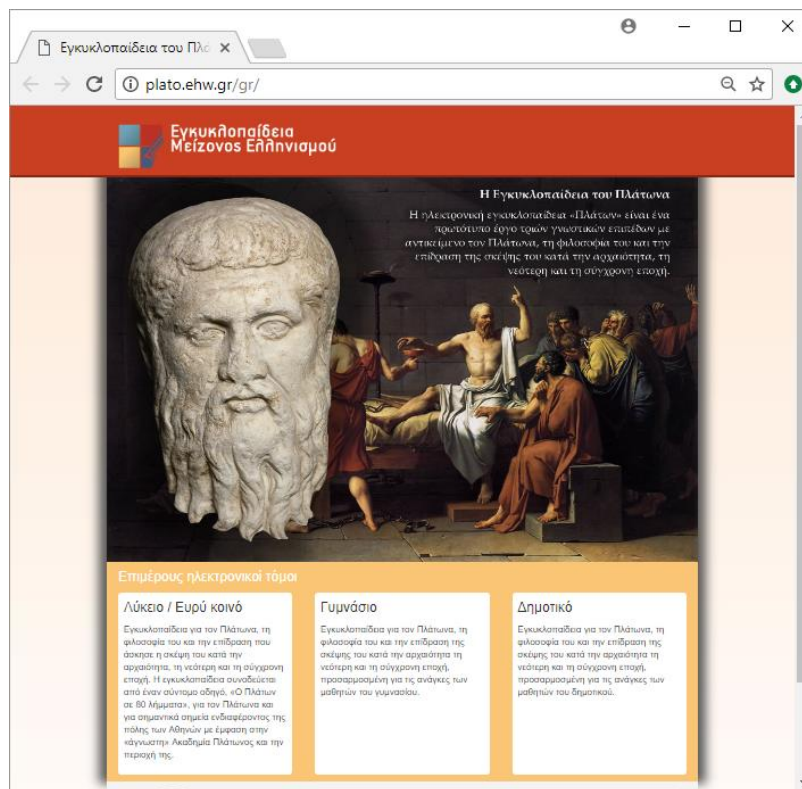
Μέσα από τις σελίδες της εγκυκλοπαίδειας γίνεται προβολή τόσο της περιοχής της Ακαδημίας Πλάτωνος (γεωγραφικά), όσο και της κληρονομιάς της πλατωνικής σκέψης προς το παγκόσμιο κοινό με πολιτισμικά ενδιαφέροντα.

Το περιεχόμενο της ψηφιακής εγκυκλοπαίδειας έχει χαρακτήρα κειμενικό και πολυμεσικό. Οργανώνεται σε γνωστικές ενότητες ή θέματα που περιλαμβάνουν κείμενο συνοπτικό ή/και αναλυτικό, σταθερή ή κινούμενη εικόνα και ήχο. Οι γνωστικές ενότητες του έργου περιλαμβάνουν κατάλογους, εργογραφίες, χρονολόγια, χάρτες, μαθησιακά παιχνίδια και γραφήματα.

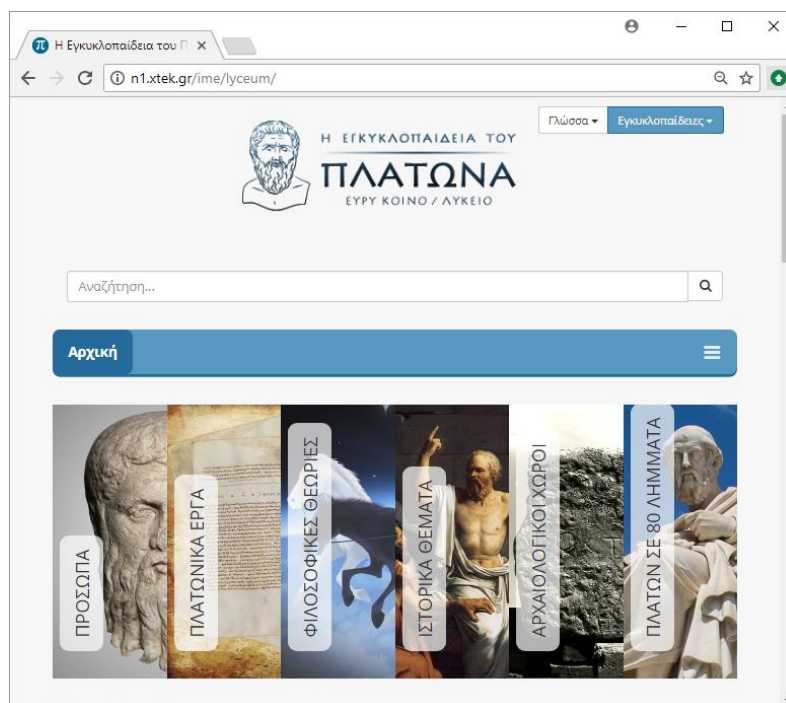
Η κειμενική πληροφορία και το πολυμεσικό υλικό της τεκμηρίωσης κατανέμονται σε τρία επίπεδα ως προς τη νοηματική τους δυσκολία ώστε να απευθύνονται αντίστοιχα σε μαθητές δημοτικού σχολείου, γυμνασίου και λυκείου. Επίσης δημιουργείται ένα τέταρτο επίπεδο, το οποίο απευθύνεται στους επισκέπτες του διαδικτυακού τόπου και οι οποίοι δε έχουν την ιδιότητα του μαθητή. Στο επίπεδο αυτό οι πληροφοριακές οντότητες συνοδεύονται κατά περίπτωση από επιπλέον πληροφορία (τοπικού, τουριστικού ή επιστημονικού).

Το περιβάλλον περιήγησης εμφανίζει στον χρήστη τις γνωστικές ενότητες οργανωμένες θεματικά και αλφαβητικά και επιτρέπει απλή και σύνθετη αναζήτηση στους τίτλους τους και στο κειμενικό περιεχόμενό τους.

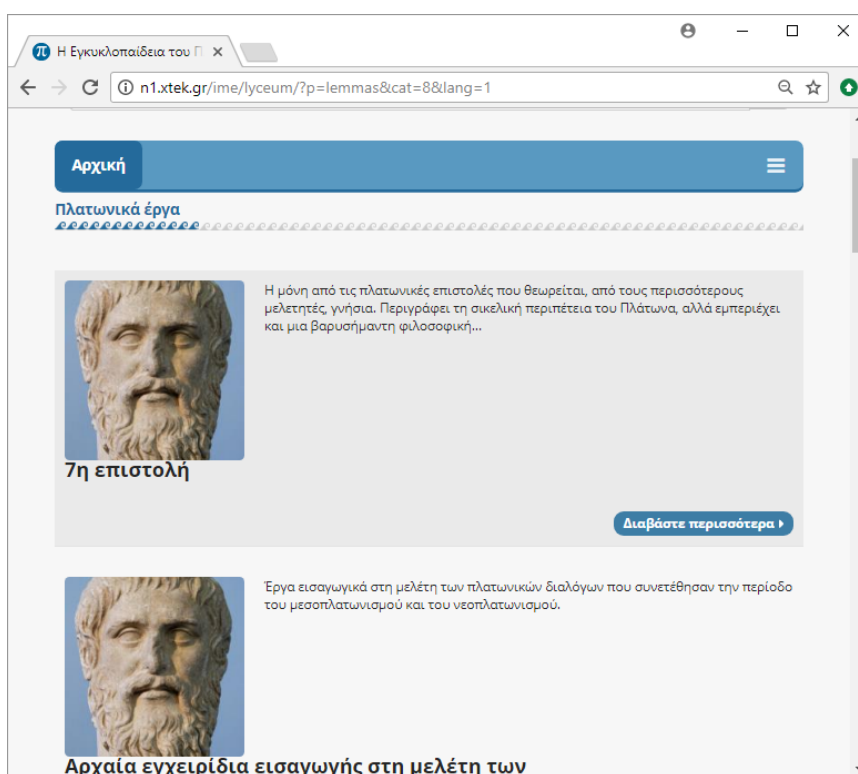
Η θεματολογία του υλικού καλύπτει τα εξής εξής πλαίσια: τοπικό (φυσικό και δομημένο περιβάλλον), πρόσωπα, ιστορία, φιλοσοφία (ιδέες και έργα), πρόσληψη και επίδραση του πλατωνικού έργου.



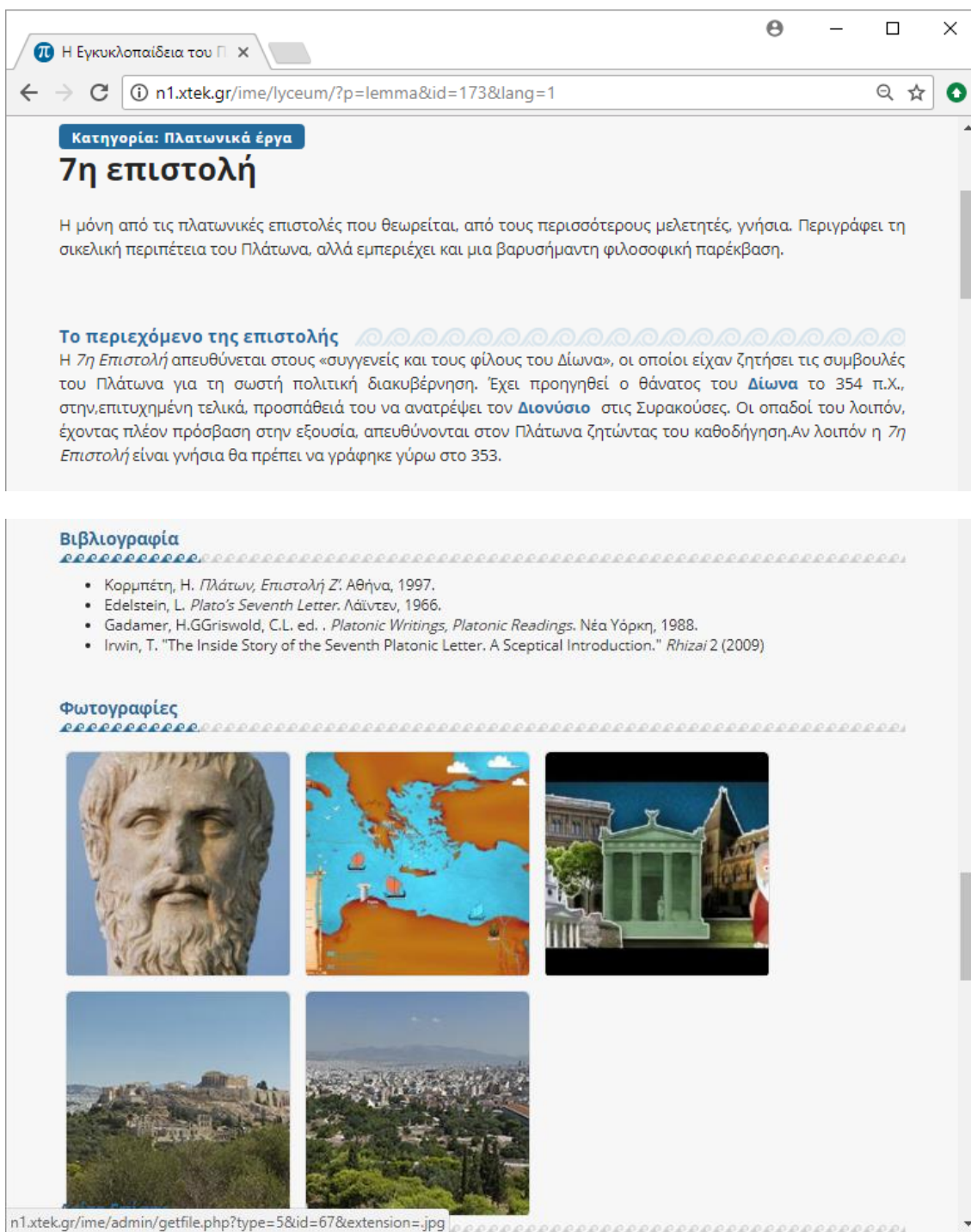
Εικόνα 3. Η βασική οργάνωση των επιπέδων νοηματικής δυσκολίας



Εικόνα 4. Τα πλαίσια τα οποία καλύπτει κάθε ενότητα της εγκυκλοπαίδειας



Εικόνα 5. Σελίδα πρόσβασης στο υλικό της θεματικής ενότητας και του αντίστοιχου πλαισίου.



Εικόνα 6. Σελίδα του αντίστοιχου τεκμηριωτικού υλικού

1.3. Μουσείο Κυκλαδικής Τέχνης

Το Μουσείο Κυκλαδικής Τέχνης είναι αφιερωμένο στη μελέτη και προβολή των αρχαίων πολιτισμών του Αιγαίου και της Κύπρου, με έμφαση στην Κυκλαδική τέχνη της 3ης π.Χ. χιλιετίας. Στις αίθουσες του μουσείου, παρουσιάζονται τρεις πολιτισμικές ενότητες (Μουσείο Κυκλαδικής Τέχνης - Συλλογές, 2017):

1. Ο Κυκλαδικός Πολιτισμός της Πρώιμης Εποχής του Χαλκού (3200 - 2000 π.Χ.) .
2. Η Αρχαία Ελληνική Τέχνη, από τους προϊστορικούς χρόνους μέχρι τη Ρωμαϊκή εποχή (2η χιλιετία π.Χ. - 4ος αι. μ.Χ.).
3. Ο Κυπριακός Πολιτισμός από τη Χαλκολιθική περίοδο μέχρι τα πρώιμα βυζαντινά χρόνια (4η χιλιετία π.Χ. - 6ος αι. μ.Χ.)

Το Μουσείο Κυκλαδικής Τέχνης είναι αφιερωμένο στη μελέτη και προβολή της Αρχαίας Ελληνικής τέχνης. Ο δικτυακός τόπος του μουσείου παρέχει πληροφορίες για την τέχνη των περιόδων που καλύπτει, δε δίνει ωστόσο στους επισκέπτες του την δυνατότητα να περιηγηθούν εικονικά στους χώρους του μέσω ενός είδους εικονικής ξενάγησης.

Ο προσανατολισμός του διαδικτυακού τόπου είναι κυρίως να προβάλλει τις εκδηλώσεις και τις συλλογές του μουσείου χωρίς να προσφέρει συγκροτημένη εκπαιδευτική παρουσίαση του ιστορικού πλαισίου μέσα στο οποίο εντάσσονται τα εκθέματα.

Ο διαδικτυακός τόπος του μουσείου φιλοξενεί τρεις ψηφιακές εκπαιδευτικές εφαρμογές -παιχνίδια (κλασικό, παζλ, παιχνίδι μνήμης).



Εικόνα 7. Σελίδα παιχνιδιού παζλ

Οι εφαρμογές αυτές απευθύνονται σε παιδιά, όντας σχεδιασμένες με έντονα χρώματα ώστε να τραβούν την προσοχή τους. Το παιχνίδι ενημερώνει το παιδί για το χρόνο που έχει δαπανήσει, το ποσοστό ολοκλήρωσης καθώς και τον αριθμό των κινήσεων που έχει κάνει.

Με την ολοκλήρωση του παζλ ο χρήστης επιβραβεύεται μέσω της πρόσβασής του σε μια σελίδα, στην οποία μπορεί να μάθει επιπλέον πληροφορίες για την ιστορία του έργου που μόλις σχημάτισε στο παζλ.



Εικόνα 8. Σελίδα εμφάνισης πληροφοριών για το έκθεμα μετά την επιτυχή σύνθεση του παζλ

Στο παιχνίδι μνήμης ο παίκτης πρέπει να συνδυάσει τις δυάδες οι οποίες απεικονίζουν το ίδιο έργο. Μόλις βρεθεί κάποια δυάδα, στην οθόνη τους προβάλλεται μια σύντομη περιγραφή του αντικειμένου.

1.4. Κέντρο George Pompidou, Παρίσι

Το κέντρο Ζορζ Πομπιντού (Centre Georges Pompidou), ή κέντρο Μπομπούρ (Beaubourg), φιλοξενεί την Δημόσια Βιβλιοθήκη Πληροφοριών και το Εθνικό Μουσείο Μοντέρνας Τέχνης της Γαλλίας (Centre Pompidou, 2017).

Ο διαδικτυακός κόμβος του μουσείου διατηρεί μια σειρά εκπαιδευτικών εφαρμογών η οποία προωθεί την ενεργή συμμετοχή των επισκεπτών του στην οικοδόμηση των γνώσεων τους σε θέματα τέχνης και ιστορίας μέσα από διάφορες δημιουργικές δραστηριότητες. Το κέντρο διαθέτει ένα ευρύ φάσμα επιμέρους ηλεκτρονικών βιβλιοθηκών, αφιερωμένων στους πιο σημαντικούς καλλιτέχνες που φιλοξενεί. Πρόκειται περισσότερο για ένα κέντρο τεκμηρίωσης του καλλιτεχνικού έργου. Οι εκπαιδευτικές δραστηριότητες περιορίζονται στη δια ζώσης παρουσία του επισκέπτη και δεν προσφέρονται εκπαιδευτικά παιχνίδια ή κουίζ στο διαδικτυακό του τόπο.

2. Ανάλυση και Σχεδιασμός Εφαρμογής

Σύμφωνα με το μοντέλο καταρράκτη για την υλοποίηση ενός έργου λογισμικού, της ανάπτυξης προηγούνται οι φάσεις της Ανάλυσης των απαιτήσεων και του Σχεδιασμού της εφαρμογής.

2.1. Ανάλυση Απαιτήσεων

Η συγκεκριμένη εφαρμογή θα καλύψει την πρόθεση ενός μουσείου (στην συγκεκριμένη περίπτωση του Μουσείου της Ακρόπολης) να προσφέρει ένα περιβάλλον εκμάθησης των ιστορικών στοιχείων γύρω από τα εκθέματά του αλλά και την πιστοποίηση των γνώσεων που αποκόμισε ο επισκέπτης μέσω απλών δοκιμασιών (τεστ). Η απαίτηση αυτή θα οδηγήσει στη δημιουργία ενός περιβάλλοντος το οποίο είναι διαθέσιμο σε επισκέπτες του διαδικτυακού τύπου και σε διαχειριστές και προσφέρει τόσο θεωρητική κατάρτιση σχετική με τα εκθέματά του (ανά θεματική ενότητα) αλλά και αντίστοιχες ασκήσεις.

2.1.1. Βασικές Απαιτήσεις

Οι απαιτήσεις που θα καλύψει η εφαρμογή όσον αφορά την διάθεση του εκπαιδευτικού υλικού αλλά και την πρακτική εξάσκηση μέσω διαδραστικών ασκήσεων, παρουσιάζονται παρακάτω (Pallud & Straub, 2014):

- **Φιλικό περιβάλλον.**
Η διεπαφή θα πρέπει να είναι φιλική και να έλκει το χρήστη του συστήματος. Επίσης η αλληλεπίδραση για την αναζήτηση και επισκόπηση των θεματικών εννοιών να είναι η ελάχιστη και να μην απαιτεί πολύπλοκα περιβάλλοντα αναζήτησης πληροφορίας.
- **Πρόσβαση από οποιοδήποτε γεωγραφικό σημείο χωρίς την ανάγκη εγκατάστασης τρίτου λογισμικού.**
Η μη απαίτηση τρίτου λογισμικού για τη λειτουργία της εφαρμογής ελαχιστοποιεί την ανάγκη ο επισκέπτης να χρειάζεται εξειδικευμένες γνώσεις για την εγκατάσταση και χρήση του συστήματος.
- **Γρήγορη πρόσβαση στην εφαρμογή.**
Καθώς η εφαρμογή είναι δικτυακή επηρεάζεται από την απόδοση του δικτύου. Συνεπώς θα πρέπει τα δεδομένα τα οποία ανταλλάσσονται μέσω του δικτύου να είναι τα ελάχιστα δυνατά, ώστε η επικοινωνία να είναι όσο το δυνατόν γρηγορότερη. Όταν η μετάβαση στα διαφορετικά στοιχεία της διεπαφής είναι αργή, αυτό επηρεάζει αρνητικά τους χρήστες με αποτέλεσμα να μειώνεται σημαντικά η χρηστικότητα της.
- **Ταυτοποίηση.**
Η φύση της εφαρμογής είναι τέτοια η οποία πρέπει να επιτρέπει στον επισκέπτη την ελεύθερη πρόσβαση στις εκπαιδευτικές ενότητες, οι οποίες προάγουν τη γνώση σχετικά με την Ακρόπολη των Αθηνών, στο πλαίσιο της παρουσίας των μνημείων στο μουσείο. Το εκπαιδευτικό λογισμικό που θα αναπτυχθεί θα δημιουργήσει μια ελεύθερη πλατφόρμα διάχυσης εκπαιδευτικού υλικού σχετικού με τα εκθέματα του Μουσείου της Ακρόπολης. Συνεπώς η πρόσβαση στις θεωρητικές ενότητες δεν προϋποθέτει καμία ταυτοποίηση του επισκέπτη του διαδικτυακού τύπου.

Οι ενότητες των ασκήσεων αντιμετωπίζονται υπό το πρίσμα της παρακολούθησης της προόδου του επισκέπτη. Για το λόγο αυτό αυτός θα πρέπει να διαθέτει ένα λογαριασμό ώστε να ταυτοποιείται μέσω ενός ζεύγους διαπιστευτηρίων (ονόματος χρήστη και συνθηματικού). Η επιλογή αυτή εξασφαλίζει την καταχώρηση των επιδόσεων κάθε εγγεγραμμένου χρήστη και τη μετέπειτα επισκόπησή τους. Για να έχει ο επισκέπτης πρόσβαση στο περιβάλλον εκτέλεσης ασκήσεων πρέπει προηγουμένως να εγγραφεί στην εφαρμογή, ώστε να ορίσει ένα όνομα (username) και έναν κωδικό (password). Το ζεύγος Όνομα χρήστη – Συνθηματικό είναι μοναδικό για κάθε χρήστη και χρησιμοποιούνται για την εισαγωγή του στο σύστημα διεξαγωγής των ασκήσεων.

2.1.2. Ειδικές Απαιτήσεις

Στόχος της εργασίας είναι η δημιουργία ενός ενδεικτικού περιβάλλοντος εκμάθησης των ιστορικών στοιχείων των εκθεμάτων του Μουσείου της Ακρόπολης. Συγκεκριμένα καλύπτονται οι εξής συλλογές:

- Η Αίθουσα των Κλιτύων της Ακρόπολης
- Η Αίθουσα του Παρθενώνα

Επιπρόσθετα, το περιβάλλον διαθέτει μεθόδους εξάσκησης των γνώσεων του επισκέπτη μέσω ενός συνόλου ασκήσεων που αυτός μπορεί να εκτελέσει, ώστε να δοκιμάσει τις γνώσεις που απέκτησε βάσει της μελέτης του θεωρητικού μέρους. Στη συνέχεια αναπτύσσεται η λειτουργικότητα και οι ειδικές απαιτήσεις του προτεινόμενου συστήματος.

Γενικές αρχές σχεδιασμού

Προκειμένου η εφαρμογή να είναι προσβάσιμη από οποιοδήποτε γεωγραφικό σημείο, επιλέχθηκε σχεδιασμός και υλοποίηση του συστήματος σε μορφή εφαρμογής διαδικτύου (web application). Ο φυλλομετρητής επιτρέπει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω υπερσυνδέσμων (hyperlinks).

Για την ενσωμάτωση περισσότερων διαδραστικών χαρακτηριστικών και δυνατοτήτων στις σελίδες του συστήματος θα ενσωματωθεί στις σελίδες κώδικας λογισμικού σε συγκεκριμένες γλώσσες προγραμματισμού:

- PHP για την ανάπτυξη διεπαφών με τη βάση δεδομένων και εμφάνιση των αποτελεσμάτων που προκύπτουν από τις επερωτήσεις προς αυτή.
- JavaScript, για την ενσωμάτωση διαδραστικών και οπτικών στοιχείων που κάνουν τις σελίδες πιο ελκυστικές και διαδραστικές (Pallud & Straub, 2014).

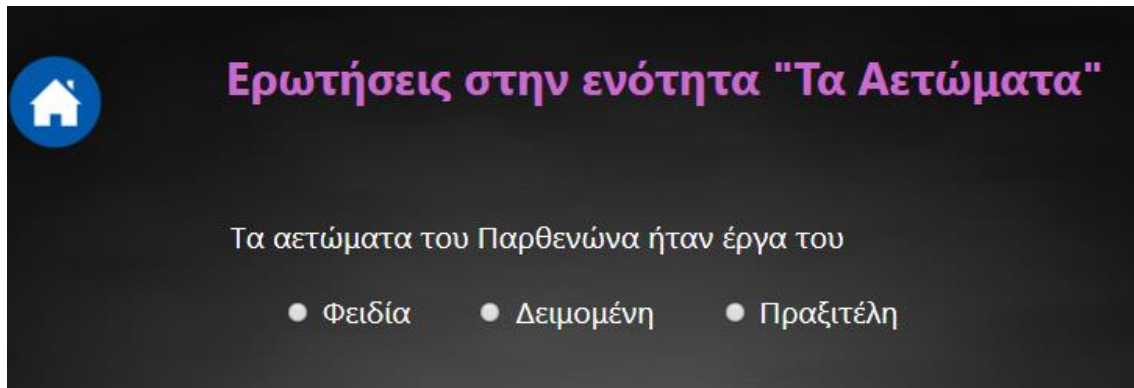
Περιβάλλον παρουσίασης πληροφοριακού υλικού

- Το λογισμικό θα περιλαμβάνει περιβάλλον περιήγησης στις θεματικές ενότητες σε μορφή διαδικτυακών σελίδων, οι οποίες ενσωματώνουν τα αντίστοιχα πληροφοριακά στοιχεία των αιθουσών και των εκθεμάτων τους.
- Κάθε ενότητα μπορεί να απαρτίζεται από υποενότητες, δημιουργώντας έτσι μια ιεραρχία δύο επιπέδων.
- Θα παρέχει ακολουθιακή πρόσβασης στις υποενότητες (ξεφύλλισμα).
- Θα παρέχει την άμεση μετάβαση στην επιθυμητή ενότητα μέσω κατάλληλου ευρετηρίου.

Ασκήσεις & είδη ασκήσεων

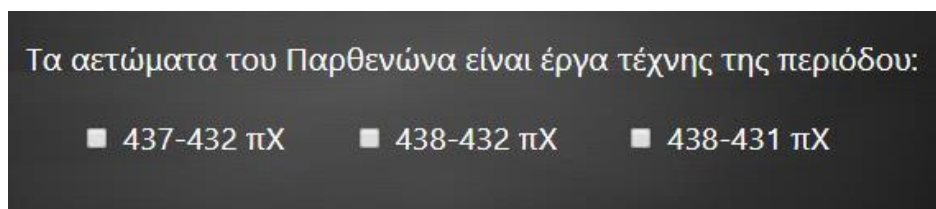
- Κάθε θεματική ενότητα και υποενότητα μπορεί να διαθέτει μη προκαθορισμένο πλήθος ερωτήσεων στο πλαίσιο ενός τεστ.
- Η μετάβαση στις ασκήσεις θα γίνεται μέσα από το περιβάλλον επισκόπησης του πληροφοριακού υλικού της ενότητας ή της υποενότητας.

- Υποστήριξη απεριόριστου αριθμού ασκήσεων ανά θεματική ενότητα ή υποενότητα.
- Κάθε ερώτηση θα προσφέρει μη προκαθορισμένο πλήθος δυνατών απαντήσεων.
- Οι τύποι των υποστηριζόμενων διαδραστικών ασκήσεων θα είναι οι εξής:
 - Ασκήσεις απλής επιλογής. Η άσκηση θα προσφέρει ένα μη προκαθορισμένο σύνολο δυνατών απαντήσεων και ο χρήστης μπορεί να επιλέξει μόνο μία απάντηση. Ενδείκνυται για απαντήσεις τύπου Σωστό / Λάθος.



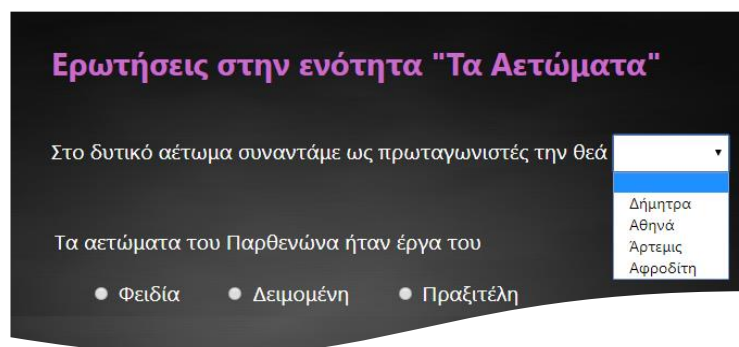
Εικόνα 9. Άσκηση απλής επιλογής

- Ασκήσεις πολλαπλής επιλογής. Η ερώτηση προτείνει ένα σύνολο επιλογών και ο επισκέπτης μπορεί να επιλέξει όσες απαντήσεις θεωρεί ότι είναι σωστές.



Εικόνα 10. Άσκηση πολλαπλής επιλογής

- Συμπλήρωση κενού. Το ερώτημα περιλαμβάνει κενές θέσεις σε μια πρόταση. Ο επισκέπτης επιλέγει τη σωστή από μία λίστα τιμών.



Εικόνα 11. Άσκηση συμπλήρωσης κενών

Γραφική απεικόνιση αποτελεσμάτων των τεστ

- Οι απαντήσεις των ασκήσεων καταχωρούνται σε βάση δεδομένων για κάθε επισκέπτη.
- Η επισκόπηση των αποτελεσμάτων μπορεί να γίνεται από τον ίδιο τον επισκέπτη.
- Ο διαχειριστής μπορεί να δει τα αποτελέσματα όλων των επισκεπτών.
- Τα αποτελέσματα θα αναπαρίστανται με αριθμητικές τιμές (πλήθος λανθασμένων και σωστών απαντήσεων επί συνόλου).
- Τα αποτελέσματα θα πρέπει να παρουσιάζονται και με μορφή διαγράμματος χρήση ραβδογραμμάτων, ώστε η επισκόπησή τους να είναι εποπτική και να δίνει πιο εύληπτη εικόνα της επίδοσης.

2.2. Μοντελοποίηση απαιτήσεων

Η σχεδίαση του συστήματος καταγράφει τις ιδιότητες που πρέπει να έχει αυτό για να ικανοποιεί τις απαιτήσεις και επεξεργάζεται τον τρόπο που το σύστημα θα υλοποιήσει τη λειτουργικότητα που .

Για τη αποτύπωση των σχεδιαστικών παραμέτρων του συστήματος χρησιμοποιείται η γλώσσα μοντελοποίησης UML (Rumbaugh, Jacobson, & Booch, 1999). Από τα 9 βασικά διαγράμματα που υποστηρίζει η UML, στην υλοποίηση του έργου αυτού χρησιμοποιούνται τα διαγράμματα use case και activity. Η σχεδίαση των διαγραμμάτων UML προϋποθέτει την περιγραφή των ρόλων των βασικών δρώντων (actors) του συστήματος.

Στο Διάγραμμα 1 απεικονίζεται το συνολικό εύρος του συστήματος, οι λειτουργικές ενότητες του καθώς και οι διεπαφές των διαφορετικών κατηγοριών χρηστών με την καθεμιά ενότητα.

2.2.1. Διάγραμμα περιπτώσεων χρήσης

Δρώντες (Actors)



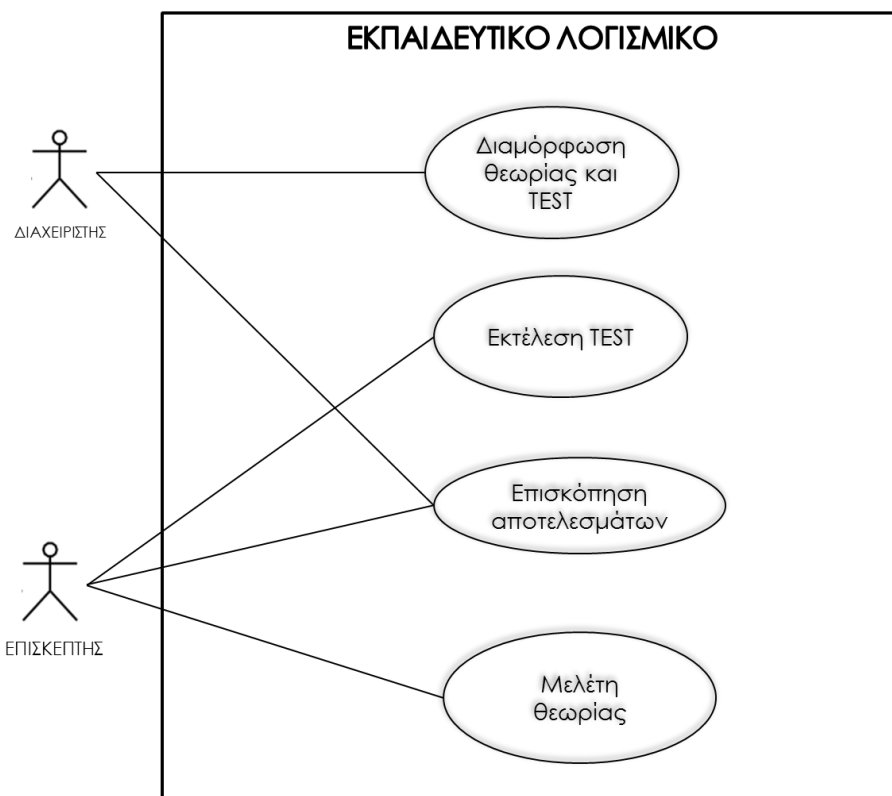
ΔΙΑΧΕΙΡΙΣΤΗΣ

Ο διαχειριστής του συστήματος διαμορφώνει κάθε φορά το θεωρητικό υλικό που θα προβάλλεται στον επισκέπτη καθώς και τις αντίστοιχες ασκήσεις. Επίσης, έχει τη δυνατότητα επισκόπησης των αποτελεσμάτων των ασκήσεων για οποιονδήποτε επισκέπτη επιθυμεί..



ΕΠΙΣΚΕΠΤΗΣ

Ο επισκέπτης είναι ο εκάστοτε χρήστης του συστήματος ο οποίος επισκέπτεται τις σελίδες για να μελετά το θεωρητικό υλικό . Εάν συνδεθεί στο σύστημα (μέσω διαδικασίας login) ενεργοποιεί τη δυνατότητα να εκτελεί τα τεστ γνώσεων κάθε ενότητας. Σε οποιαδήποτε στιγμή επιθυμεί, μπορεί να προβάλλει τα αποτελέσματα ώστε να διαπιστώνει τις επιδόσεις του.



Διάγραμμα 1. Διάγραμμα περίπτωσης χρήσης του συνολικού συστήματος

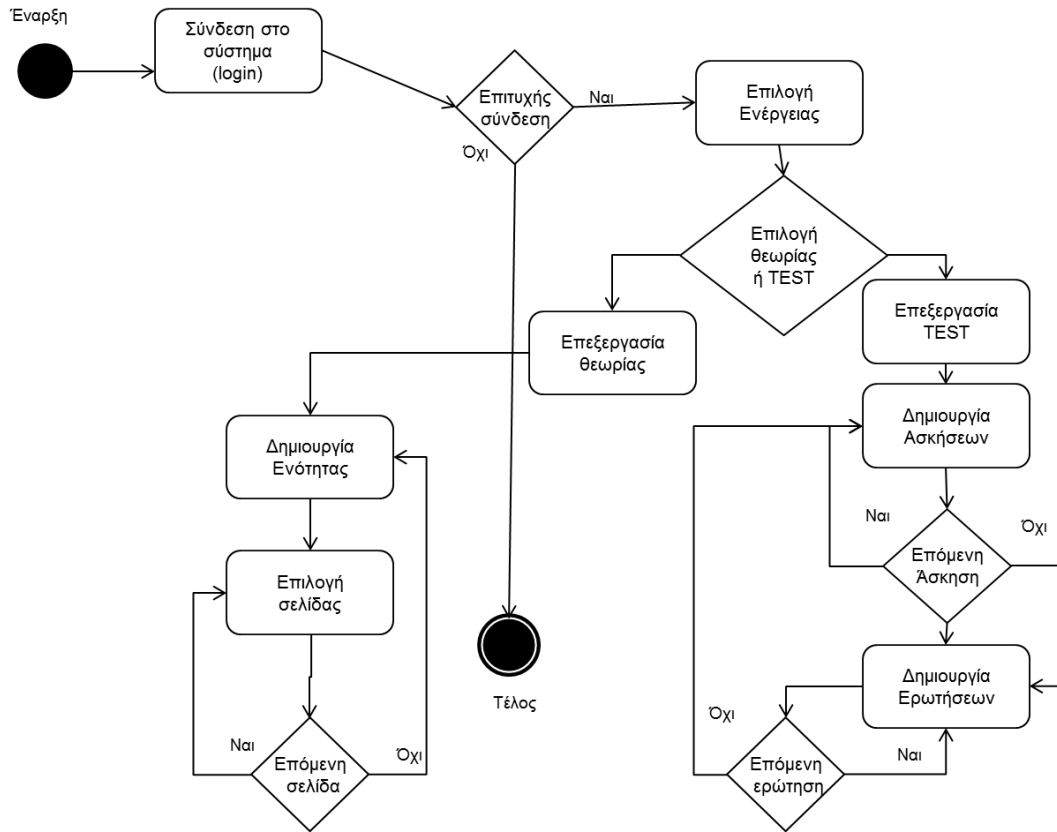
2.2.2. Διαγράμματα δραστηριοτήτων

Σα διαγράμματα δραστηριοτήτων αποτυπώνεται η συμπεριφορά μιας διαδικασίας ως ένα σύνολο από διαδοχικές ενέργειες. Οι βασικές δραστηριότητες του συστήματος αναπαρίστανται ως κόμβοι με μορφή ορθογώνιου, ενώ οι κόμβοι αποφάσεων με ρόμβο.

Οι κύριες δραστηριότητες που μοντελοποιούνται είναι:

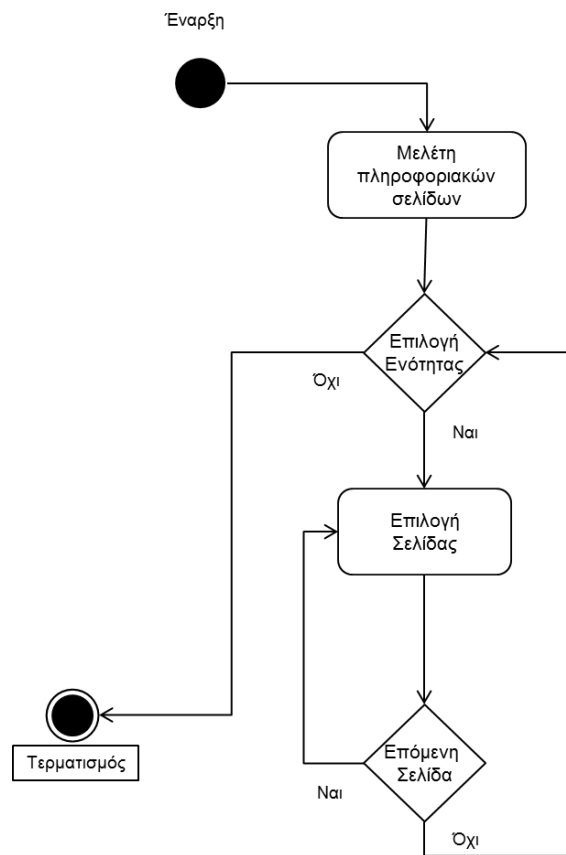
1. Διαμόρφωση πληροφοριακού / εκπαιδευτικού υλικού και τεστ
2. Μελέτη των πληροφοριακών ενότητων του λογισμικού.
3. Ενεργοποίηση και επίλυση των τεστ
4. Επισκόπηση αποτελεσμάτων

1. Διαμόρφωση πληροφοριακού / εκπαιδευτικού υλικού και τεστ



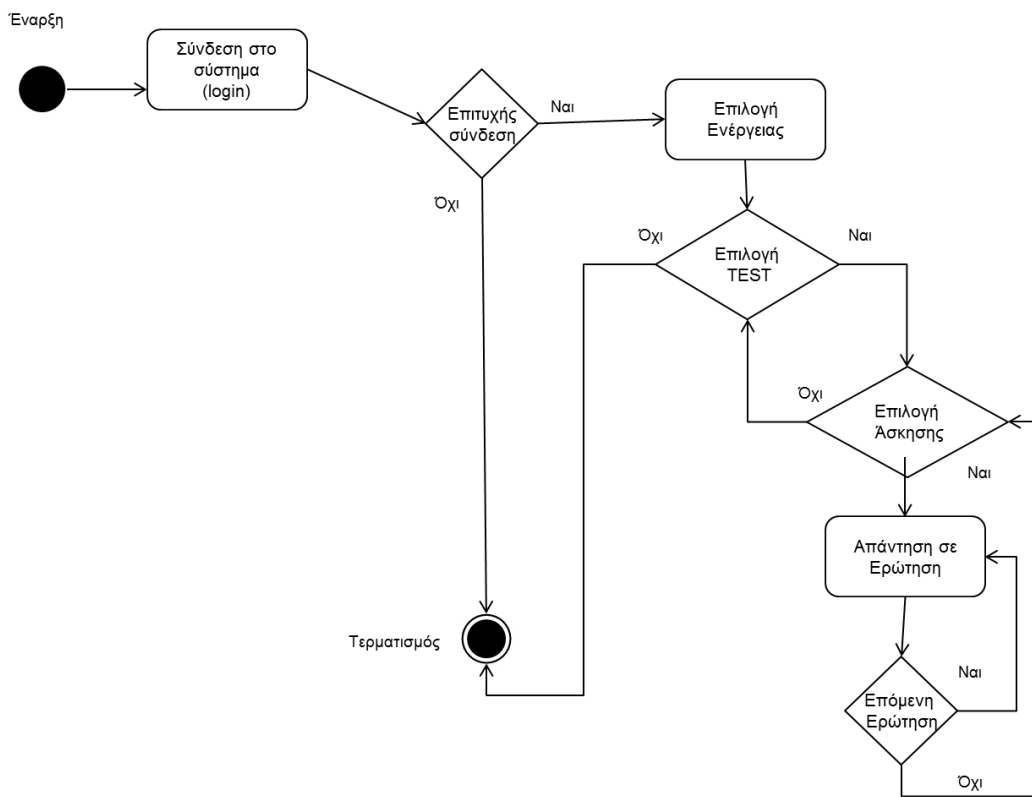
Διάγραμμα 2. Διάγραμμα δραστηριοτήτων διαμόρφωσης του πληροφοριακού υλικού και των αντίστοιχων τεστ

2. Μελέτη πληροφοριακών ενότητων



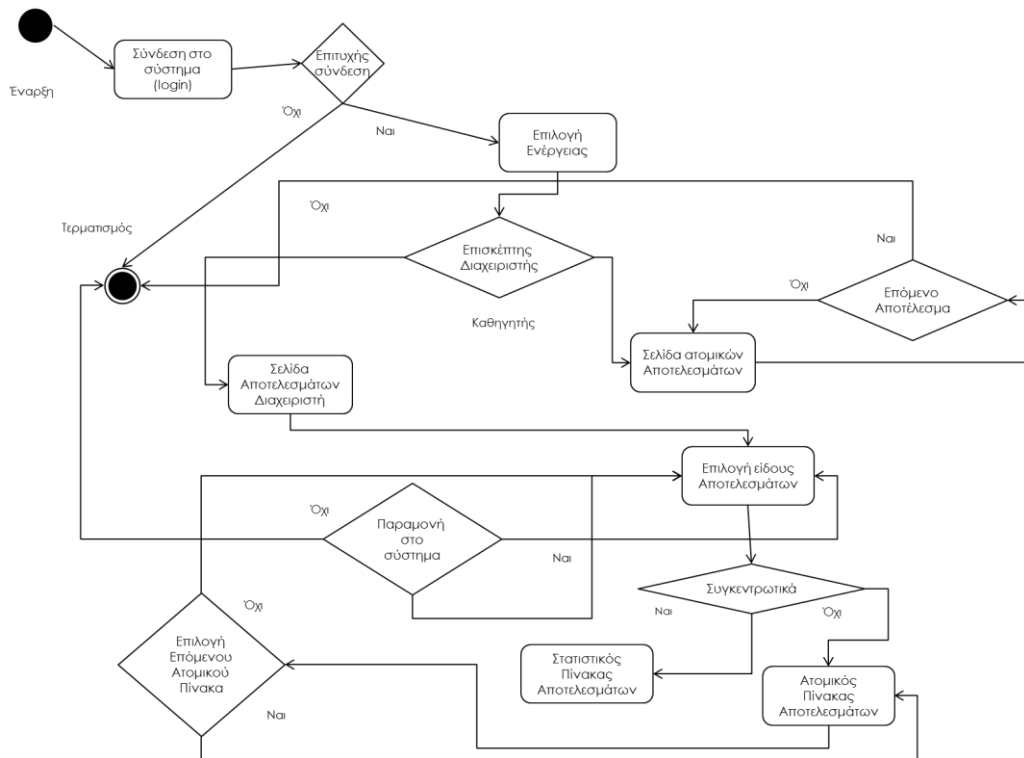
Διάγραμμα 3. Διάγραμμα δραστηριοτήτων μελέτης των πληροφοριακών οντοτήτων (σελίδων)

3. Ενεργοποίηση και επίλυση τεστ



Διάγραμμα 4. Διάγραμμα δραστηριοτήτων ενεργοποίησης και εκτέλεσης τεστ

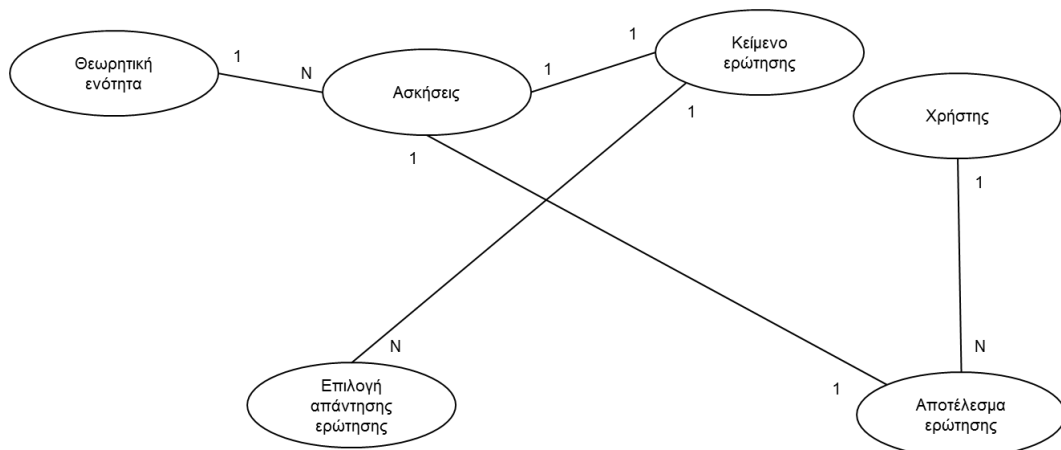
4. Επισκόπηση αποτελεσμάτων



Διάγραμμα 5. Διάγραμμα δραστηριοτήτων επισκόπησης αποτελεσμάτων του ΤΕΣΤ

2.3. Σχεδιασμός βάσης δεδομένων

Η σχεδίαση της βάσης δεδομένων βασίστηκε στο Μοντέλο Οντοτήτων - Συσχετίσεων (Entity-Relationship - ER) για τη μοντελοποίηση των δεδομένων του συστήματος και την περιγραφή της δομής των δεδομένων του . Βάσει του μοντέλου αυτού γίνεται η μετατροπή στο σχεσιακό μοντέλο , το οποίο και θα υλοποιηθεί στο σύστημα διαχείρισης βάσεων δεδομένων.



Διάγραμμα 6. Διάγραμμα Οντοτήτων - Συσχετίσεων

Εννοιολογικό μοντέλο

Βάσει των παραπάνω απαιτήσεων, σχεδιάστηκε το διάγραμμα οντοτήτων – συσχετίσεων, το οποίο απεικονίζει τις οντότητες του προβλήματος και τις σχέσεις που αναπτύσσονται μεταξύ τους, όπως και η πληθικότητα των σχέσεων αυτών.

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. Οντότητα “User”

Όνομα οντότητας	User
Περιγραφή	Περιέχει τα στοιχεία των εγγεγραμμένων χρηστών
Σχεσιακός πίνακας	Users
Ιδιότητες	
Usr_ID	Πρωτεύον κλειδί. Μοναδικός εσωτερικός κωδικός. Παράγεται αυτόματα από τη βάση δεδομένων
Usr_Username	Μοναδικό αναγνωριστικό όνομα χρήστη. Δεν είναι δυνατό να υπάρχουν δύο χρήστες με το ίδιο αναγνωριστικό όνομα. Χαρακτηρίζεται ως UNIQUE στη βάση δεδομένων και δεν επιτρέπει κενές τιμές (Not Null)
Usr_Password	Συνθηματικό. Το κρυφό συνθηματικό που χρησιμοποιεί ο επισκέπτης για τη σύνδεσή του. Δεν επιτρέπει κενές τιμές (Not Null)
Usr_Lastname	Επώνυμο χρήστη. Το πραγματικό επώνυμο του χρήστη.
Usr_Firstname	Όνομα χρήστη Το όνομα του χρήστη.
Usr_Role_Admin	Παίρνει τιμή True εάν ο χρήστης στο πλαίσιο του συστήματος έχει το ρόλο του διαχειριστή.
Usr_Role_Contributor	Παίρνει τιμή True εάν ο χρήστης είναι επισκέπτης και θα χρησιμοποιεί το λογισμικό για διεξαγωγή των τεστ.
Usr_Status	Καθορίζει την κατάσταση του χρήστη στο πλαίσιο του συστήματος (ενεργός / ανενεργός).

Πίνακας 2. Οντότητα “Exercise”

Όνομα οντότητας	Exercise
Περιγραφή	Διαχειρίζεται τις εκφωνήσεις των ασκήσεων
Σχισιακός πίνακας	Exercises
Ιδιότητες	
exe_id	Πρωτεύον κλειδί. Μοναδικός εσωτερικός κωδικός. Παράγεται αυτόματα από τη βάση δεδομένων (auto increment).
exe_text	Κείμενο εκφώνησης
exe_image	Προβεβλημένη εικόνα της άσκησης. Προβάλλεται στη σελίδα δίπλα στην αντίστοιχη άσκηση
exe_unit	Κωδικός της ενότητας στην οποία ανήκει η άσκηση
exe_page	Αριθμητική τιμή, η οποία αντιστοιχεί στον αριθμό ενημερωτικής σελίδας της ενότητας στην οποία θα αντιστοιχεί η συγκεκριμένη άσκηση.
exe_class	Αντιστοιχεί στην κλάση της άσκησης. Παίρνει τις τιμές: <ul style="list-style-type: none"> • 1: Απλής επιλογής • 2: Πολλαπλής επιλογής • 3: Συμπλήρωσης κενών
exe_sorter	Καθορίζει την αλληλουχία εμφάνισης των ασκήσεων στο τεστ. Το σύστημα προβάλλει τις ασκήσεις της ενότητας κατ' αύξουσα σειρά της τιμής exe_sorter.

Πίνακας 3. Οντότητα “question_text”

Όνομα οντότητας	question_text
Περιγραφή	Περιλαμβάνει τα ερωτήματα μιας άσκησης. Μία άσκηση μπορεί να περιέχει πολλά ερωτήματα
Σχισιακός πίνακας	question_text
Ιδιότητες	
qt_id	Πρωτεύον κλειδί. Μοναδικός εσωτερικός κωδικός. Παράγεται αυτόματα από το σύστημα βάσης.
qt_exercise	Ο εσωτερικός κωδικός της άσκησης στην οποία ανήκει το ερώτημα. Περιγράφει τη σχέση exercise -> question_text
qt_text	Κείμενο της ερώτησης
qt_sorter	Σειρά εμφάνισης της ερώτησης. Οι ερωτήσεις μιας άσκησης εμφανίζονται σύμφωνα με την τιμή qt_sorter κατ' αύξουσα σειρά.

Πίνακας 4. Οντότητα “question_options”

Όνομα οντότητας	question_options
Περιγραφή	Περιλαμβάνει τις επιλογές που είναι διαθέσιμες για απάντηση κάθε ερωτήματος της άσκησης
Σχεσιακός πίνακας	question_options
Ιδιότητες	
qo_id	Πρωτεύον κλειδί. Μοναδικός εσωτερικός κωδικός. Παράγεται αυτόματα από το σύστημα βάσης δεδομένων (auto increment)
qo_qt_id	Εσωτερικός κωδικός του ερωτήματος στο οποίο αντιστοιχεί η επιλογή
qo_text	Κείμενο της επιλογής (Ενδεικτικές τιμές: Σωστό, Λάθος)
qo_correct_answer	Σωστή απάντησης για την επιλογή. Ακολουθούνται οι εξής κανόνες: 1: Σωστή επιλογή σε ερωτήσεις απλής / πολλαπλής επιλογής 0: Λάθος επιλογή σε ερωτήσεις απλής / πολλαπλής επιλογής <Κείμενο απάντησης>: Το ακριβές κείμενο της απάντησης σε ερωτήσεις συμπλήρωσης κενών.
qo_optionsfile	Όνομα αρχείου κειμένου, το οποίο σε διαδοχικές γραμμές περιέχει τις επιλογές για συμπλήρωση κενού σε ασκήσεις κλάσης συμπλήρωσης κενών.
qo_shorter	Σειρά εμφάνισης της επιλογής (ταξινόμηση κατ' αύξουσα τιμή της ιδιότητας αυτής).

Πίνακας 5. Η οντότητα “results”

Όνομα οντότητας	Results
Περιγραφή	Καταγραφή απαντήσεων του χρήστη σε κάθε ερώτηση της άσκησης
Σχεσιακός πίνακας	Results
Ιδιότητες	
res_id	Μοναδικός εσωτερικός Παράγεται αυτόματα από το σύστημα βάσης δεδομένων (auto increment)
res_session_id Σύνθετο πρωτεύον κλειδί, Απαγορεύει την καταχώρηση δύο απαντήσεων για το ίδιο τεστ και την ίδια ερώτηση.	Μοναδικός εσωτερικός κωδικός του τεστ Σχηματίζεται την ημερομηνία σύνδεσης του επισκέπτη, την ώρα σύνδεσης και τον μοναδικό εσωτερικό του κωδικό
res_question_option	Εσωτερικός κωδικός της ερώτησης της άσκησης στην οποία απάντησε ο επισκέπτης

res_user_id	Εσωτερικός κωδικός του επισκέπτη που εκτελεί το τεστ
res_timestamp	Ημερομηνία και ώρα σύνδεσης του επισκέπτη στο σύστημα
res_answer	Τιμή της απάντησης στην ερώτηση res_question_option
res_exe_class	Κλάση της ερώτησης 1: Ερωτήσεις απλής επιλογής 2: Ερωτήσεις πολλαπλής επιλογής 3: Ερωτήσεις συμπλήρωσης κενού
res_exe_id	Μοναδικός εσωτερικός κωδικός στην οποία ανήκει η ερώτηση
res_exe_unit	Ενότητα στην οποία ανήκει η άσκηση

Πίνακας 6. Οντότητα “theory”

Όνομα οντότητας	Theory
Περιγραφή	Πληροφορίες για τις πληροφοριακές / ενημερωτικές σελίδες Οργανώνει τις ενημερωτικές σελίδες του συστήματος. Κάθε σελίδα ανήκει σε μία ενότητα (th_class).
Σχεσιακός πίνακας	Theory
Ιδιότητες	
th_id	Πρωτεύον κλειδί. Μοναδικός εσωτερικός κωδικός. Παράγεται αυτόματα από το σύστημα βάσης δεδομένων
th_title	Κείμενο τίτλου της ενημερωτικής σελίδας
th_image	Αρχείο εικόνας που συνδέεται με τη σελίδα.
th_class	Κωδικός της πληροφοριακής ενότητας
th_sorter	Σειρά εμφάνισης των σελίδων στο περιβάλλον ανάγνωσης. Οι σελίδες εμφανίζονται κατ’ αύξουσα τιμή του πεδίου
th_page	Όνομα αρχείου html το οποίο περιέχει το ενημερωτικό

2.3.1. Σχεσιακό μοντέλο

Στο Διάγραμμα 7, απεικονίζονται οι πίνακες της βάσης δεδομένων και οι οποίοι αντιστοιχούν στο διάγραμμα Ο/Σ.

Βάσει του διαγράμματος Οντοτήτων – Συσχετίσεων προκύπτει το σχεσιακό σχήμα, το οποίο περιγράφεται στο Διάγραμμα 7.

Σε κάθε πίνακα, με το εικονίδιο κλειδιού υποδηλώνεται το πρωτεύον κλειδί. Το πρωτεύον κλειδί είναι ένα πεδίο (ή σύνολο πεδίων) το οποίο έχει μοναδική τιμή για κάθε γραμμή (tuple) του πίνακα. Οι γραμμές αντιστοιχούν σε συσχετίσεις μεταξύ των δύο πινάκων που συνδέονται με αυτές.



Διάγραμμα 7. Σχισιακό μοντέλο του εκπαιδευτικού λογισμικού

2.3.2. Φυσικό μοντέλο

Η δήλωση του σχήματος ο προσδιορισμός των δεδομένων της βάσης, έγινε μέσω της γλώσσας 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”

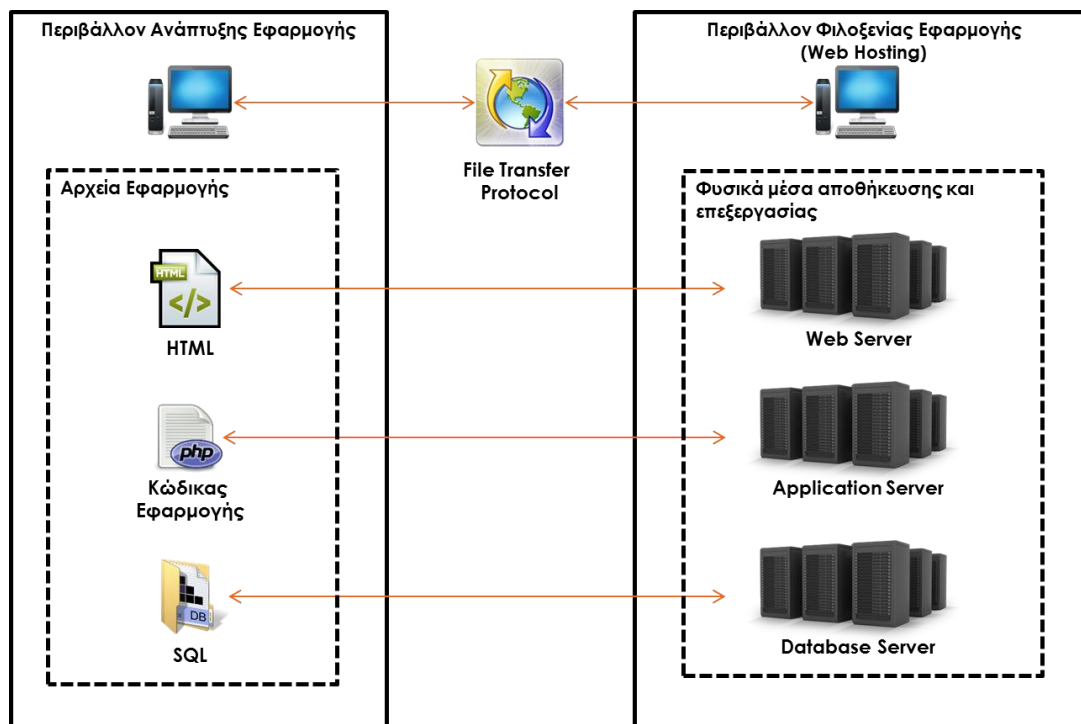
```
CREATE TABLE `results` (  
  `res_id` int(11) NOT NULL AUTO_INCREMENT,  
  `res_session_id` varchar(25) NOT NULL,  
  `res_question_option` int(11) NOT NULL,  
  `res_user_id` int(11) DEFAULT NULL,  
  `res_timestamp` datetime DEFAULT NULL,  
  `res_answer` varchar(45) DEFAULT NULL,  
  `res_exe_class` 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 12. 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. Εξασφαλίζεται ανεξαρτησία από το γραφικό περιβάλλον διεπαφής. Η μετάβαση σε επιπλέον πλατφόρμες παρουσίασης των εφαρμογών (κινητά τηλέφωνα, tablets, σύγχρονες τηλεοπτικές συσκευές, κ.λπ.) είναι σχετικά απλή, δεδομένου ότι επηρεάζεται μόνο το επίπεδο παρουσίασης του μοντέλου 3-tier. 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. Παρέχεται ευελιξία στην τροποποίηση του επιπέδου παρουσίασης των πληροφοριών, ενώ τα επίπεδα επιχειρησιακής λογικής και διαχείρισης δεδομένων παραμένουν αμετάβλητα.
4. Κλιμάκωση. Η δόμηση του συστήματος σε επίπεδα δίνει την ευχέρεια ανεξάρτητης αναβάθμισης κάθε επιπέδου, ανάλογα με το πού εντοπίζονται αυξημένες ανάγκες επιδόσεων.

5. Απλή εγκατάσταση εφαρμογών στις εγκαταστάσεις του παρόχου διαδικτυακής φιλοξενίας.
6. Προσαρμογή της ασφάλειας της εφαρμογής στα επανξιμένα χαρακτηριστικά ασφάλειας που διαθέτουν οι πάροχοι φιλοξενίας στις εγκαταστάσεις τους.

Στη συνέχεια περιγράφονται τα αρχιτεκτονικά επίπεδα που συγκροτούν το λογισμικό.

3.1.1. Επίπεδο δεδομένων

Το επίπεδο δεδομένων υλοποιείται από το διακομιστή βάσης δεδομένων όπου αποθηκεύονται οι πληροφοριακές οντότητες του λογισμικού και τα δεδομένα των χρηστών. Τα δεδομένα αποθηκεύονται και διατηρούνται στο διακομιστή του επιπέδου δεδομένων ανεξάρτητα από το διακομιστή εφαρμογών. Το επίπεδο δεδομένων διαχειρίζεται τα δεδομένα της εφαρμογής με ασφάλεια και διεκπεραιώνει τις δοσοληψίες καταχώρησης, αναζήτησης και ανάκλησης πληροφοριών προς και από τα ανώτερα επίπεδα του μοντέλου.

3.1.2. Επίπεδο επιχειρησιακής λογικής

Το επίπεδο λογικής της εφαρμογής υλοποιήθηκε μέσω της γλώσσας προγραμματισμού PHP. Ο κώδικας PHP εκτελείται στον web server του συστήματος παράγοντας δυναμικά HTML περιεχόμενο το οποίο διατίθεται στον φυλλομετρητή του χρήστη ο οποίος ενεργοποίησε τον αντίστοιχο κώδικα. Ο υπολογιστής του χρήστη δεν απαιτεί πρόσθετο λογισμικό για την εκτέλεση του κώδικα PHP (plugin ή runtime) για να εκτελείται η εφαρμογή.

3.1.3. Επίπεδο παρουσίασης

Η διάθεση των δυναμικών HTML σελίδων που δημιουργούνται μέσω του κώδικα PHP χρησιμοποιείται ο HTTP server Apache. Πρόκειται για έναν ευρύτατα διαδεδομένο διακομιστή HTTP ανοικτού κώδικα, ο οποίος χρησιμοποιείται τους παρόχους διαδικτυακής φιλοξενίας.

Η γλώσσα περιγραφής του περιεχομένου είναι η HTML5. Οι κανόνες μορφοποίησης των σελίδων που παράγονται δυναμικά χρησιμοποιούν τεχνολογία Cascading Stylesheets (CSS). Η CSS είναι γλώσσα προσδιορισμού των χαρακτηριστικών εμφάνισης μιας ιστοσελίδας, ορίζοντας χαρακτηριστικά όπως γραμματοσειρές, χρώματα, στοίχιση, θέση των στοιχείων στη σελίδα, μορφοποίηση κεφαλίδων και σώματος κειμένου, πινάκων, κ.ο.κ..

3.2. Τεχνολογίες υλοποίησης

Στόχος ήταν η δημιουργία μιας απλής και εύχρηστης εφαρμογής, η οποία να προσφέρει ένα εποπτικό και εργονομικό περιβάλλον χρήσης, το οποίο απαιτεί ελάχιστη έως καθόλου εκπαίδευση στη χρήση του.

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 δόμηση σελίδων του World Wide Web (ή απλά ιστού, Web) (Pallud & Straub, 2014).

Η HTML έχει σχεδιαστεί με σκοπό:

- Τη μορφοποίηση 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.

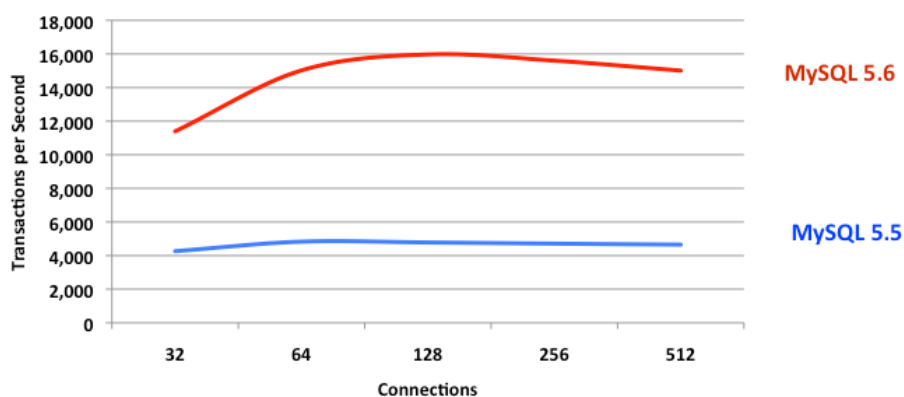
Η php δεν απαιτεί αυξημένους υπολογιστικούς πόρους του διακομιστή. Αυτό την καθιστά μια γρήγορη γλώσσα στην εκτέλεση του προγράμματος χωρίς να επιβραδύνει τις υπόλοιπες διεργασίες του διακομιστή. Επίσης η php είναι ευρύτατα αποδεκτή ως σταθερή γλώσσα η οποία χρησιμοποιείται κατά κόρον για ανάπτυξη εφαρμογών βάσεων δεδομένων. Ένα σημαντικό χαρακτηριστικό της php είναι ότι υλοποιεί πολλαπλά επίπεδα ασφάλειας για την προστασία των εφαρμογών από κακόβουλες επιθέσεις. Επίσης η php 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 php για διασύνδεση με συστήματα βάσεων δεδομένων. Διατίθενται διεπαφές για συστήματα βάσεων δεδομένων MySQL, Microsoft SQL Server, Oracle, Informix, Postgres, κ.α. Επίσης η php είναι συμβατή και εκτελείται στην πλειονότητα των application servers (Microsoft Internet Information Server, Apache, THTTPD, κ.α.).

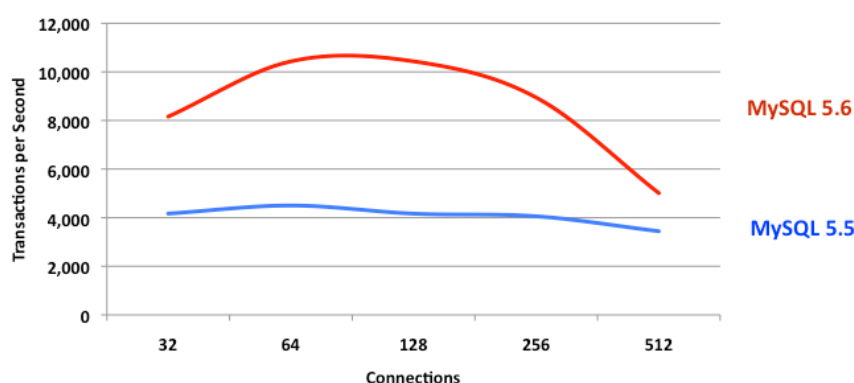
MySQL

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

1. Είναι διαθέσιμο χωρίς κόστος. Πρόκειται για ελεύθερο λογισμικό, το οποίο επιτρέπει τη δημιουργία περιβάλλοντος 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 ικανότητα διαχείρισης επερωτήσεων 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 ανάγνωσης 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

Το CSS (Cascading Style Sheets) είναι γλώσσα ορισμού χαρακτηριστικών, η οποία βοηθά στην περιγραφή του τρόπου με τον οποίο θα εμφανίζονται τα διάφορα στοιχεία HTML σε μια ιστοσελίδα.

Τα πλεονεκτήματα που έχει η χρήση CSS έναντι της μορφοποίησης μέσω χαρακτηριστικών εμφυτευμένων στον κώδικα HTML με μορφή attributes, είναι τα εξής:

- Εύκολη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός διαδικτυακού τόπου μπορεί να ελέγχεται από ένα και 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.
- Βελτιωμένη SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν επεξεργάζονται κώδικα μορφοποίησης, αλλά αναλύουν και δεικτοδοτούντο πληροφοριακό περιεχόμενο της σελίδας και μόνο.
- Ταχύτερη φόρτωση σελίδων. Ένα εξωτερικό αρχείο CSS επεξεργάζεται από το φυλλομετρητή την πρώτη φορά που το χρησιμοποιεί κάποια σελίδα του ιστότοπου και μόνο και αποθηκεύεται στην προσωρινή μνήμη. Επόμενες κλήσεις σε αυτό εξυπηρετούνται από το τοπικά αποθηκευμένο και επεξεργασμένο αντίγραφο, χωρίς να απαιτείται η εκ νέου φόρτωσή του.

3.3. Περιγραφή υλοποίησης

Οι αρχές υλοποίησης του λογισμικού αναπτύσσονται στην παράγραφο αυτή..

Ο σχεδιασμός έγινε με γνώμονα το διαχωρισμό των χαρακτηριστικών εμφάνισης των ιστοσελίδων από τη λειτουργικότητά του. Για το σκοπό αυτό, έχει χρησιμοποιηθεί αρχείο cascading stylesheet, το οποίο ορίζει τους κανόνες εμφάνισης των στοιχείων HTML πάνω στη σελίδα.

Έπειτα, τα επαναχρησιμοποιούμενα στοιχεία των σελίδων (όπως η κεφαλίδα των σελίδων) έχουν δημιουργηθεί σε ανεξάρτητα αρχεία, ώστε να μην επαναλαμβάνεται συνεχώς ο ίδιος κώδικας και να είναι ευκολότερο να γίνονται αλλαγές.

Φάκελοι της εφαρμογής

Images	Στον κατάλογο αυτό περιέχονται οι εικόνες της εφαρμογής.
Styles	Είναι ο κατάλογος που περιέχει τα αρχεία css.
includes:	Είναι ο κατάλογος που περιέχει τα αρχεία ρυθμίσεων και τη βιβλιοθήκη των συναρτήσεων που χρησιμοποιούνται από άλλες σελίδες της εφαρμογής:
Scripts	Είναι ο κατάλογος που περιέχει τα αρχεία γλώσσας javascript, συναρτήσεις της οποίας που χρησιμοποιούνται από τις σελίδες της εφαρμογής.

Αρχεία εφαρμογής

header.php	Βρίσκει το όνομα του συνδεδεμένου χρήστη και το εμφανίζει πάνω αριστερά στην αρχική σελίδα. Επίσης, δημιουργεί ένα σύνδεσμο τον οποίο ο επισκέπτης μπορεί να χρησιμοποιήσει για να αποσυνδεθεί.
index.php	Αρχική σελίδα της εφαρμογής. Δημιουργεί συνδέσμους προς τις σελίδες των πληροφοριακών ενότητων, τις σελίδες των ασκήσεων, τα αποτελέσματα των ασκήσεων και τη σύνδεση (login) του χρήστη. Εάν ο συνδεδεμένος χρήστης έχει δικαιώματα διαχειριστή, εμφανίζεται και

σύνδεσμος προς τη διαχείριση χρηστών.

loginmain.php	Η σελίδα εισαγωγής των διαπιστευτηρίων για σύνδεση στην εφαρμογή (Login page).
logintry.php	Ελέγχει τα στοιχεία σύνδεσης στη βάση δεδομένων, τα οποία πληκτρολογεί ο επισκέπτης. Εάν το ζεύγος των διαπιστευτηρίων βρίσκεται καταχωρημένο στη βάση δεδομένων, δημιουργεί το session cookie με τα στοιχεία του συνδεδεμένου χρήστη.
manageusers.php	Σελίδα η οποία δημιουργεί ηλεκτρονική φόρμα για την καταχώρηση στοιχείων νέου χρήστη. Επίσης χρησιμοποιείται για τη διαχείριση ή και διαγραφή των δεδομένων των ήδη καταχωρημένων χρηστών.
message.php	Κώδικας των παραθύρων μηνυμάτων που εμφανίζονται από την εφαρμογή, όταν το σύστημα πρέπει να προσφέρει ανάδραση σε μορφή κειμένου στην οθόνη
results.php	Σελίδα εμφάνισης των αποτελεσμάτων των ασκήσεων σε μορφή γραφήματος.
testlist.php	Δημιουργεί κατάλογο με τα τεστ που έχουν διεξαχθεί από συνδεδεμένους επισκέπτες. Εάν ο χρήστης έχει δικαιώματα διαχειριστή, έχει πρόσβαση στα τεστ όλων των χρηστών. Εάν ο συνδεδεμένος χρήστης είναι επισκέπτης (με ρόλο “contributor”), έχει πρόσβαση μόνο στα τεστ που έχει επιλύσει ο ίδιος.
theory.php	Δημιουργεί το περιβάλλον περιήγησης στις πληροφοριακές σελίδες της εφαρμογής. Επίσης δημιουργεί υπερσύνδεσμο προς τις ασκήσεις του τρέχοντος κεφαλαίου.
config.php	Περιέχει τις ρυθμίσεις της βάσης δεδομένων MySQL και τους κανόνες σύνδεσης της εφαρμογής σε αυτήν.
functions.php	Περιέχει τους ορισμούς και την υλοποίηση όλων των συναρτήσεων που χρησιμοποιούνται στον κώδικα PHP της εφαρμογής.

Αρχείο ρυθμίσεων config.php

Στο αρχείο αυτό ορίζονται τα δεδομένα σύνδεσης της εφαρμογής στη βάση δεδομένων, ενώ ενεργοποιείται και η σύνδεση αυτή. Στο ίδιο αρχείο γίνεται η ρύθμιση του συνόλου χαρακτήρων της βάσης δεδομένων (Unicode, UTF-8), το οποίο θα χρησιμοποιεί ξ εφαρμογή για την ανταλλαγή δεδομένων με τη βάση και να είναι δυνατή η χρήση των ελληνικών χαρακτήρων.

Η συγκεκριμένη σελίδα δημιουργεί και την `php session` (σύνοδο), η οποία χρησιμοποιείται για την προσωρινή αποθήκευση δεδομένων που αφορούν τη σύνδεση του χρήστη στην εφαρμογή και τις δοσοληψίες του με αυτή. Κάθε επισκέπτης συσχετίζεται με μία σύνοδο (session). Αυτό γίνεται μέσω δημιουργίας μιας μοναδικής ταυτότητας για τον εκάστοτε χρήστη που αλληλεπιδρά με την εφαρμογή και αποθήκευσή του σε μορφή cookie στον υπολογιστή. Σε κάθε αίτηση (request) που κάνει ο επισκέπτης προς την εφαρμογή, χρησιμοποιείται το cookie που περιέχει το τη μοναδική αυτή ταυτότητα του επισκέπτη, ώστε όπου αυτό είναι απαραίτητο να διακρίβώνεται η ταυτότητά του (όπως για παράδειγμα στις σελίδες των τεστ, οι οποίες απαιτούν τη σύνδεση του επισκέπτη με τα διαπιστευτήριά του..

Αυτός ο μηχανισμός επιτρέπει στην εφαρμογή να διατηρεί μια ιδιωτική συνομιλία με τον εκάστοτε χρήστη καθώς αυτός περιηγείται μεταξύ των σελίδων. Η PHP δίνει τη δυνατότητα αποθήκευσης και ανάκτησης προσωρινών δεδομένων μιας συνόδου, χρησιμοποιώντας το αντικείμενο `$_SESSION`. Ενδεικτικά, η κλήση `$_SESSION['username']` επιστρέφει στην εφαρμογή το username του χρήστη ο οποίος έχει συνδεθεί μέσω της διαδικασίας login στην εφαρμογή

Κώδικας 7: Κώδικας config.php

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

Επεξήγηση header.php

Η σελίδα αυτή ενσωματώνεται σε όσες σελίδες η εφαρμογή θα εμφανίζει το όνομα του συνδεδεμένου επισκέπτη και τον αντίστοιχο σύνδεσμο για την αποσύνδεσή του.

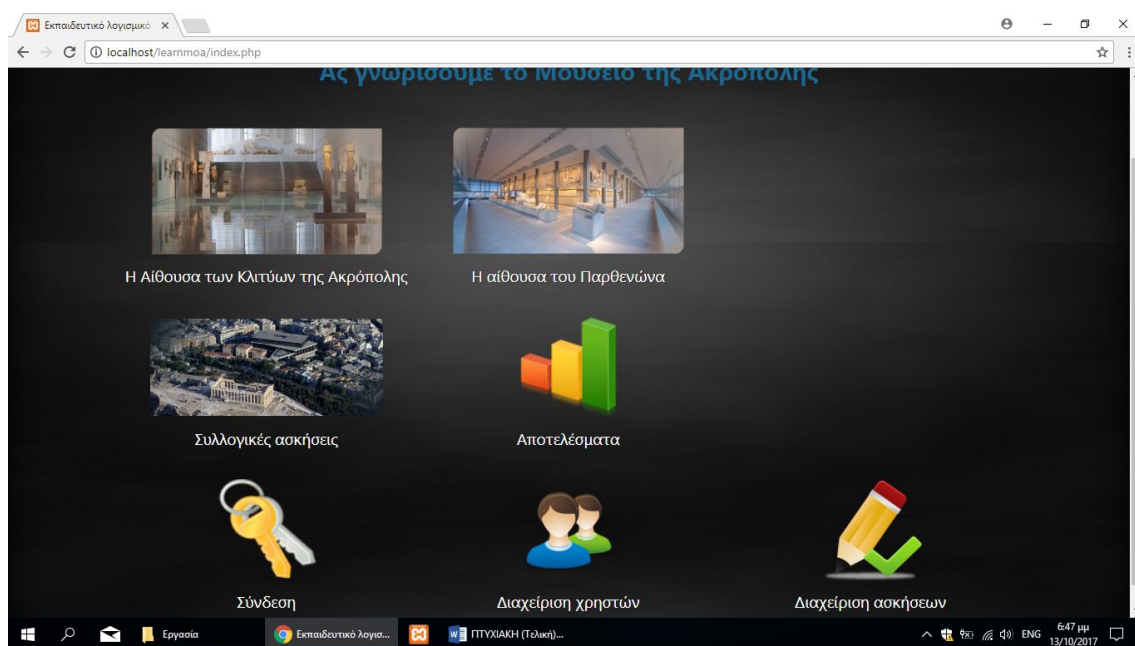
Ο κώδικας της σελίδας ελέγχει εάν υπάρχει χρήστης ο οποίος έχει συνδεθεί στο σύστημα. Εάν έχει συνδεθεί, τότε στη σελίδα η οποία καλεί το αρχείο header.php εμφανίζει τα στοιχεία σύνδεσης. Εάν ο επισκέπτης δεν έχει συνδεθεί, εμφανίζεται στην κεφαλίδα αντίστοιχο μήνυμα ("Δεν έχει συνδεθεί κάποιος χρήστης") και η προτροπή για τη σύνδεσή του (login).

4. Λειτουργικότητα συστήματος

Η εφαρμογή έχει δύο επίπεδα πρόσβασης:

- Εφαρμογή διαχειριστή.
- Εφαρμογή επισκέπτη.

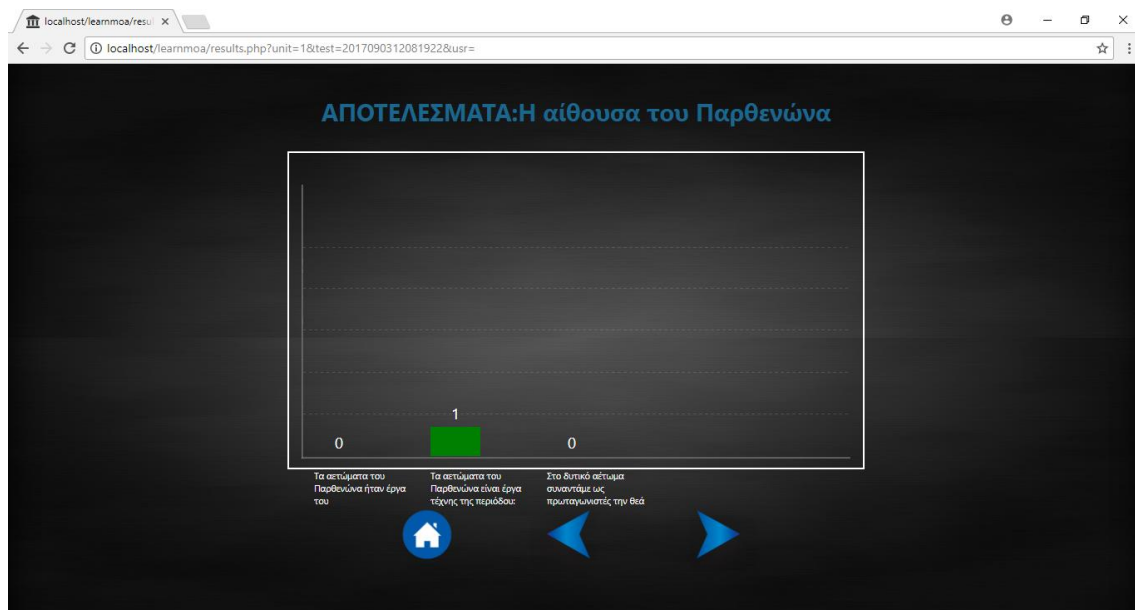
Η αρχική σελίδα της εφαρμογής εμφανίζει τις βασικές επιλογές οποιουδήποτε επισκέπτη, ασχέτως εάν αυτός έχει εγγραφεί στο σύστημα.



Εικόνα 13. Αρχική σελίδα της εφαρμογής

4.1. Εφαρμογή διαχειριστή

Η εφαρμογή διαχειριστή δίνει τη δυνατότητα επισκόπησης των τεστ, τα οποία συμπλήρωσαν όλοι οι επισκέπτες της εφαρμογής. Ο διαχειριστής επιλέγει την αντίστοιχη γραμμή με το όνομα του επισκέπτη και το τεστ που επιθυμεί.



Εικόνα 14. Εμφάνιση αποτελεσμάτων τεστ

Η εφαρμογή εμφανίζει τα αποτελέσματα ανά υποενότητα..

4.2. Εφαρμογή επισκέπτη

Το ενημερωτικό / εκπαιδευτικό υλικό του μαθήματος δομείται σε ενότητες. Η ενότητα προσδιορίζεται από την τιμή του πεδίου `th_class` του πίνακα `theory`. Στη συγκεκριμένη υλοποίηση υλοποιήθηκαν οι εξής ενότητες και υποενότητες::

- **Ενότητα 1: Η Αίθουσα των Κλιτύων της Ακρόπολης**
Υποενότητα 1: Ο Οικισμός

Υποενότητα 2: Τα Ιερά
- **Ενότητα 2: Η Αίθουσα του Παρθενώνα**
Υποενότητα 1: Το Μνημείο

Υποενότητα 2: Οι Μετόπες

Υποενότητα 3: Τα Αετώματα

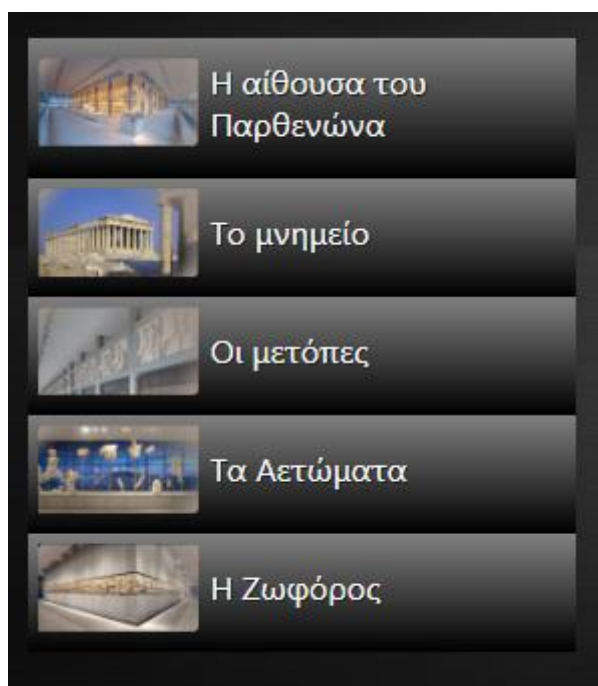
Υποενότητα 4: Η Ζωφόρος

Για κάθε υποενότητα έχει αναπτυχθεί μία html σελίδα, η οποία και παρέχει τις πληροφορίες του αντίστοιχου εκθέματος. Για την ομοιόμορφη αλλά και κατανοητή παρουσίαση των σελίδων χρησιμοποιήθηκε η γλώσσα HTML και ένα κοινό Cascading Stylesheet (CSS), όπου ενσωματώθηκαν οι κανόνες μορφοποίησης.



Εικόνα 15. Περιβάλλον εμφάνισης και ανάγνωσης πληροφοριακών σελίδων

Ο επισκέπτης μπορεί να μεταφέρεται στο επόμενο ή προηγούμενο κεφάλαιο πιέζοντας τα αντίστοιχα πλήκτρα στο κάτω μέρος της σελίδας. Επίσης μπορεί να εμφανίσει άμεσα το επιθυμητό κεφάλαιο από τη λίστα των κεφαλαίων στο κάτω αριστερό τμήμα της σελίδας.



Εικόνα 16. Λίστα επιλογής των υποενοτήτων μιας ενότητας

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

Κάθε εκπαιδευτική ενότητα, τεστ, άσκηση και ερώτηση διαθέτει μια απεικόνιση στη βάση δεδομένων και συσχετίζεται με τις άλλες οντότητες όπως απεικονίζονται στο Διάγραμμα 7.

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 απαντήσεων).
3. Ερωτήσεις συμπλήρωσης κενών σε μία πρόταση από μία λίστα προτάσεων.

Ανάλογα με την κλάση της άσκησης, ενεργοποιείται ο αντίστοιχος μηχανισμός προβολής της ερώτησης και των δυνατών απαντήσεών της. Η Εικόνα 18 απεικονίζει υλοποιημένες τις κλάσεις ασκήσεων.

4.2.1. Διεξαγωγή ασκήσεων

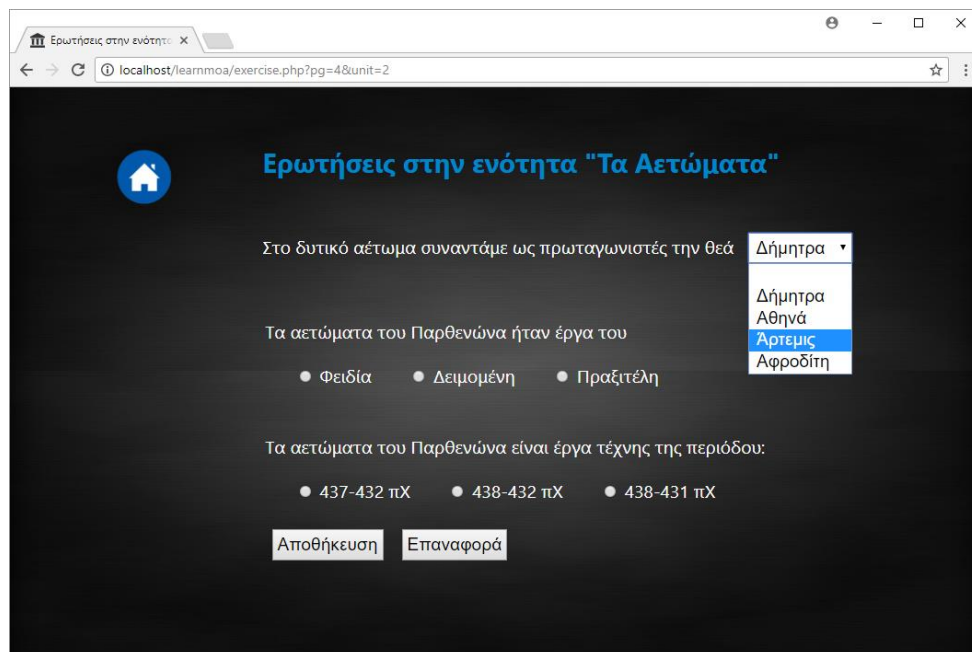
Όταν σε μία υποενότητα υπάρχει διαθέσιμο τεστ με ερωτήσεις που αφορούν την αντίστοιχη θεματολογία, στη σελίδα και κάτω από το πληροφοριακό περιεχόμενο εμφανίζεται κατάλληλο εικονίδιο, το οποίο όταν πατήσει ο επισκέπτης μεταβαίνει στη σελίδα του τεστ. Εάν δεν υπάρχει τεστ για τη συγκεκριμένη υποενότητα, τότε το εικονίδιο αυτό δεν εμφανίζεται.



Εικόνα 17. Εμφάνιση εικονιδίου διαθέσιμου τεστ κάτω δεξιά στη σελίδα επισκόπησης του πληροφοριακού υλικού

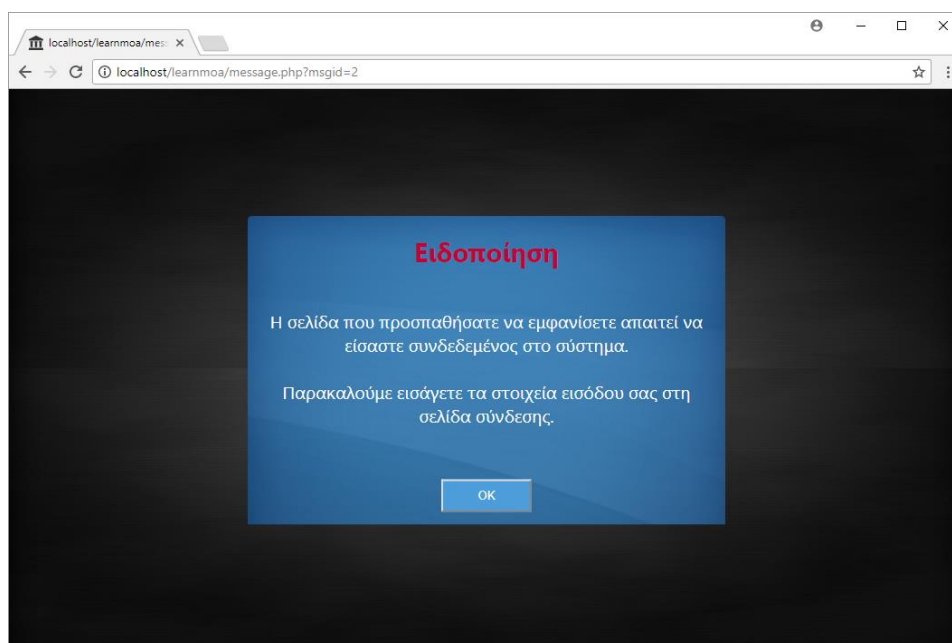
Πατώντας το κουμπί του τεστ, ο επισκέπτης (εφόσον έχει συνδεθεί στο σύστημα) θα μεταβεί στη σελίδα του τεστ (Εικόνα 18), όπου και θα απαντήσει τις ερωτήσεις.

Ολοκληρώνοντας τη δοκιμασία πρέπει να πατήσει το κουμπί Αποθήκευση, ώστε τα αποτελέσματα του τεστ να αποθηκευτούν στη βάση δεδομένων και να είναι δυνατή η επισκόπησή τους.



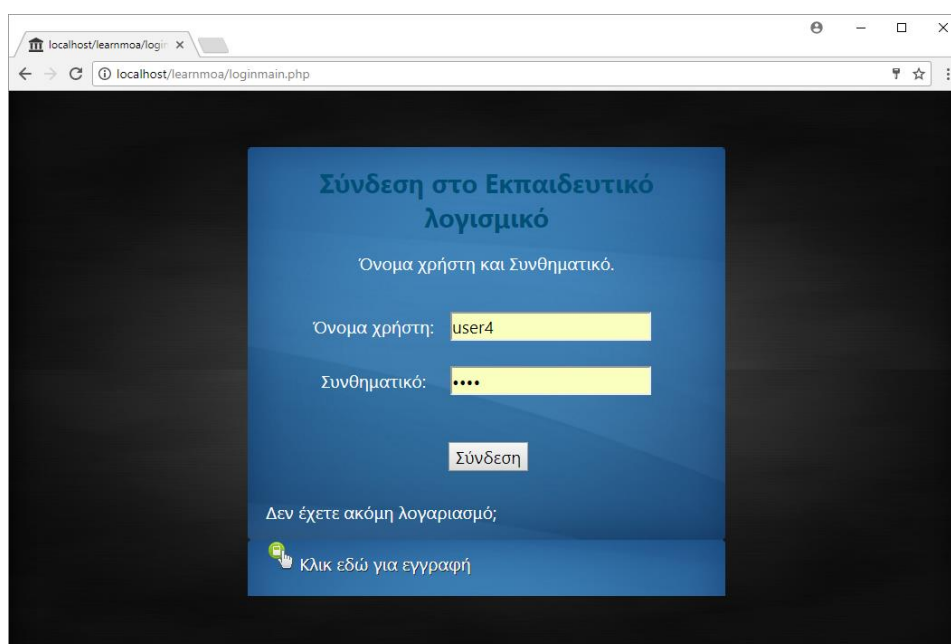
Εικόνα 18. Ενδεικτική οθόνη διεξαγωγής ασκήσεων διαφορετικών κλάσεων

Εάν ο επισκέπτης δεν έχει συνδεθεί με τα διαπιστευτήριά του, θα εμφανιστεί σε αυτόν κατάλληλο μήνυμα, το οποίο τον προτρέπει να συνδεθεί (Εικόνα 19)



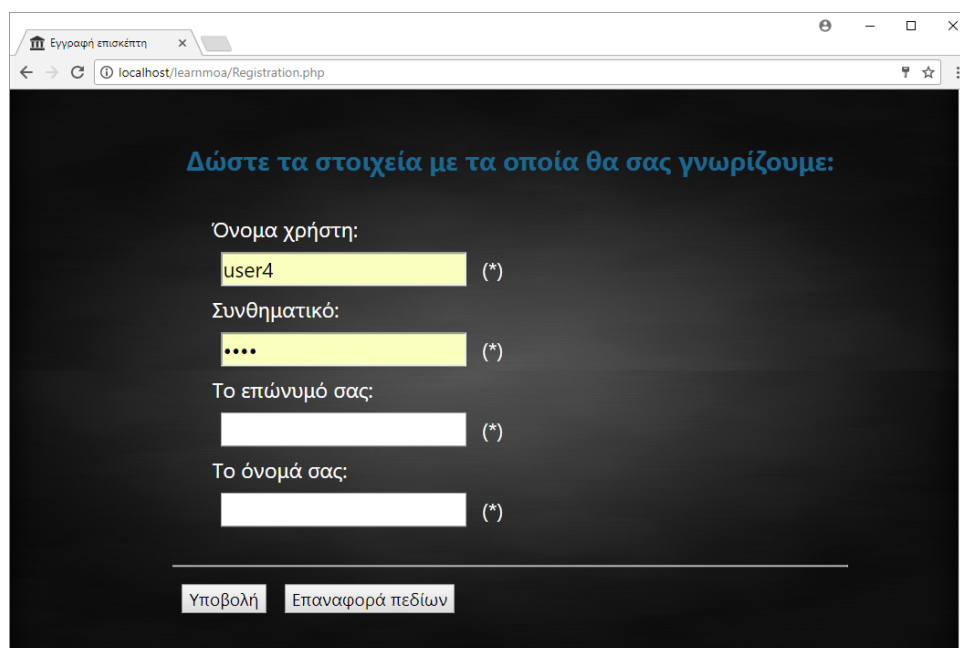
Εικόνα 19. Προτροπή σύνδεσης επισκέπτη με τα διαπιστευτήριά του.

Πατώντας OK, θα εμφανιστεί η σελίδα σύνδεσης, όπου ο χρήστης εισάγει τα διαπιστευτήριά του (Εικόνα 20)



Εικόνα 20. Οθόνη σύνδεσης επισκέπτη μέσω εισαγωγής των διαπιστευτηρίων του

Εάν ο επισκέπτης δε διαθέτει λογαριασμό, μπορεί κάνοντας κλικ στην υπερσύνδεση «Πατήστε εδώ για εγγραφή» να εγγραφεί στο εκπαιδευτικό λογισμικό. Συμπληρώνοντας τα στοιχεία του στην ηλεκτρονική φόρμα της Εικόνας 21.



Εικόνα 21. Ηλεκτρονική φόρμα εγγραφής στο Εκπαιδευτικό λογισμικό

5. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Το εκπαιδευτικό λογισμικό έχει σχεδιαστεί με τρόπο ώστε μέσω της προσθήκης νέων εγγραφών στη βάση δεδομένων να επεκτείνεται το περιεχόμενό του, τόσο με νέες πληροφοριακές ενότητες όσο και με νέες 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 επισκέπτη

Το εκπαιδευτικό λογισμικό συνεπώς θα αποκτήσει μια διάσταση ευφυΐας, επιτρέποντάς του να προσαρμόζει τη διεπαφή του με τον επισκέπτη με διαφορετικό τρόπο, κάθε φορά που αυτός συνδέεται στο σύστημα.

Ενδιαφέρον επίσης αποτελεί η υλοποίηση του λογισμικού με τεχνολογίες οι οποίες του επιτρέπουν να είναι προσβάσιμο από φορητές συσκευές (έξυπνα τηλέφωνα ή tablets), ώστε οι επισκέπτες να μπορούν να το χρησιμοποιούν ακόμη και κατά την περιήγησή τους στους χώρους του μουσείου, απαντώντας επί τόπου στις ασκήσεις και έχοντας μια άμεση, ενδιαφέρουσα και διαδραστική εμπειρία με τα εκθέματα.

Βιβλιογραφία

1. *Centre Pompidou*. (2017). Ανάκτηση Αύγουστος 2017, από Centre Pompidou: <https://www.centrepompidou.fr/en>
2. Liu, X., Heo, J., & Sha, L. (2009). *Modeling 3-Tiered Web Applications*. University of Illinois.
3. Pallud, J., & Straub, D. (2014). Effective website design for experience-influenced environments. *Information & Management*, 359–373.
4. Rumbaugh, J., Jacobson, I., & Booch, G. (1999). *The Unified Modeling Language Reference Manual*. ADDISON-WESLEY.
5. Smithsonian American Art Museum. (2017). *Smithsonian American Art Museum*. Ανάκτηση Ιούνιος 2017, από Smithsonian American Art Museum: <http://americanart.si.edu/>
6. Ε. Λιακοπούλου, Β. Ν. (2010). Το ΕΛ/ΛΑΚ στην Εκπαίδευση. *Workshop on Informatics in Education* (σ. 20). Tripoli: GREEK COMPUTER SOCIETY(GCS).
7. *Η Εγκυκλοπαίδεια του Πλάτωνα*. (2017). Ανάκτηση Ιούλιος 2017, από Εγκυκλοπαίδεια Μείζονος Ελληνισμού: <http://plato.ehw.gr/gr/>
8. *Μουσείο Κυκλαδικής Τέχνης - Συλλογές*. (2017). Ανάκτηση Ιούνιος 2017, από Μουσείο Κυκλαδικής Τέχνης: <https://www.cycladic.gr/>
9. 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).

Παράρτημα Α – Κώδικας εφαρμογής

Κώδικας 8. 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"];

            // Loop που βρίσκει εάν υπάρχουν ασκήσεις στη συγκεκριμένη σελίδα
            $strSQLQuestionText = "SELECT exercises.exe_id, exercises.exe_unit, question_text.qt_id,
question_text.qt_text,
                                question_text.qt_sorter
question_text.qt_exercise
                                FROM exercises INNER JOIN question_text ON exercises.exe_id =
exercises.exe_page=". $currentPage.
                                WHERE exercises.exe_unit=". $currentUnit. " AND
                                " ORDER BY question_text.qt_sorter, question_text.qt_id";
            $resultQuestionText = mysqli_query($conn, $strSQLQuestionText);
            $options_number = 0;
            while ($rowQuestionText = mysqli_fetch_array($resultQuestionText))
                $options_number++;

            $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 πλαϊσίου στο κάτω αριστερό τμήμα της σελίδας
// με ευρετήριο των κεφαλαίων για άμεση μετάβαση
echo "<div class=\"TheoryMenu\">";
echo "<table style=\"border-collapse: collapse; width: 100%\">";
$i = 0;
while($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'];
    $i++;
    // Δημιουργία λίστας με την εικόνα και τον τίτλο του κεφαλαίου
    echo "<tr>";
    echo "<td style=\"vertical-align: middle; background: linear-gradient(grey, black);\">";
    echo "<img class=\"TheoryMenuImage\" src=\"\".$th_image.\" /></td>";
    echo "<td style=\"vertical-align: middle; background: linear-gradient(grey, black);\"><a
href=\"theory.php?pg=$i&unit=\".$CurrentUnit.\"><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 πάτημα "Επόμενο" προβάλλει το 1ο κεφάλαιο.
// Όταν διαβάζεται το 1ο κεφάλαιο και πατηθεί "Προηγούμενο", 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 class=\"ButtonHome\" src=\"images/home.png\"
alt=\"Αρχική\"/></a></div>";
    if ($options_number > 0)
        echo "<div class=\"ButtonTest\"><a href=\"exercise.php?pg=$CurrentPage&unit=".$CurrentUnit."\"><img
src=\"images/tests.png\" alt=\"Τεστ\"/></a></div>";
        echo "<div class=\"ButtonLeft\"><a href=\"theory.php?pg=$PreviousPage&unit=".$CurrentUnit."\"><img
class=\"ButtonLeft\" src=\"images/buttonleft.png\" alt=\"Previous\"/></a></div>";
        echo "<div class=\"TheoryFeaturedImage\"><img class=\"TheoryFeaturedImage\"
src=\"".$TheoryRecords[$CurrentPage-1]['th_image']."\" alt=\"Theory\"/></div>";
        echo "<div class=\"ButtonRight\"><a href=\"theory.php?pg=$NextPage&unit=".$CurrentUnit."\"><img
class=\"ButtonRight\" 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 9. 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';

    $hiddendebug = "hidden";

    $currentPage = $_GET["pg"]; // Κεφάλαιο
    $currentUnit = $_GET["unit"]; // Ενότητα θεωρίας (εδώ χρησιμοποιείται μόνο η τιμή 1). Για μελλοντική χρήση

    // Βρίσκει τον τίτλο της ενότητας για να προβληθεί στην κεφαλίδα
    $strSQL = "SELECT * FROM theory WHERE th_class = $currentUnit AND th_sorter = $currentPage";
    $result = mysqli_query($conn, $strSQL);
    while ($row = mysqli_fetch_array($result))
    {
        $th_title = $row['th_title'];
    }
?>

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <link rel="stylesheet" href="styles/styles.css">
        <title>Ερωτήσεις στην ενότητα <?php>$th_title<?></title>
    </head>
    <body>
        <?php
            // 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=\"TestDiv\">";
    echo "<h1 style=\"text-align: left;\">Ερωτήσεις στην ενότητα \"".$th_title."</h1>";

// 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
$strSQL = "SELECT exercises.exe_id, exercises.exe_text, exercises.exe_unit, exercises.exe_class,
exercises.exe_sorter, exercises.exe_unit
FROM exercises
WHERE exercises.exe_unit=\".$CurrentUnit.\" AND exercises.exe_page=\".$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 = $row['qo_text'];
        //echo "<p class=\"Question_exe_text\">".$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_page=\".$currentPage.\" AND exercises.exe_id=\".$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 = $rowQuestionText['qt_id'];
                                // Εάν δεν πρόκειται για άσκηση συμπλήρωσης κενού δημιουργεί κατάλληλο πίνακα
                                // με γραμμές και στήλες όπου τοποθετούνται οι επιλογές
            if ($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 class=\"Question_qt_text\">".$qt_text."</p>";
                echo "</td>";
            }
            else
                echo "<p class=\"Question_qt_text\"></p><span
class=\"Question_qt_text\">".$qt_text."</span>";
            // Loop που βρίσκει τις επιλογές κάθε ερωτήματος
            echo "</tr>";
            echo "</table>";

```



```

        $strSQLQuestionOptions = "SELECT question_options.qo_id, question_options.qo_qt_id,
question_options.qo_text,
                                question_options.qo_correct_answer,
question_options.qo_optionsfile, question_options.qo_shorter
                                FROM question_options
                                WHERE question_options.qo_qt_id=".$qt_id." ORDER BY
question_options.qo_shorter";
        $resultQuestionOptions = mysqli_query($conn, $strSQLQuestionOptions);
        $qt_id_aggregate = "";
        echo "</table>";
        echo "<tr>";
        while ($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"
περίπτωση λίστας επιλογών)
            $qo_optionsfile = $rowQuestionOptions['qo_optionsfile']; // Αρχείο επιλογών (σε

            // 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 ))
            $checked = 'checked';
        echo "<input type=\"radio\" class=\"radio_button\"
name=\"Data_Array[$rowNum][qo_qt_id]\" value=\"\".$qo_correct_answer.\"\" id=\"\".$qo_id.\"\" $checked>";
        echo "<span class=\"radio_button_label\" style=\"color: #fff;\">\".$qo_text.\"
</span>";

        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)
            $checked = 'checked';
        echo "<input type=\"checkbox\" class=\"radio_button\"
name=\"Data_Array[$rowNum][qo_qt_id]\" value=\"1\" $checked>";
        echo "<span class=\"radio_button_label\" style=\"color: #fff;\">\".$qo_text.\"
</span>";

        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]\" style=\"font-size: 1em; color:
#000; padding: 4px; margin: 10px;\">";
        // To 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;
}
}
echo "</tr>";
echo "</table>";

$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

// 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\">";

    $rowNum++;
}
}
echo "<p></p>";
if ($options_number > 0)
    echo "<input style=\"font-size: 1em; color: #000; padding: 4px; margin: 10px;\" type=\"submit\"
value=\"Αποθήκευση\" onclick=\"ResetFieldChange('UnsavedChanges');\" /><input style=\"font-size: 1em; color: #000;
margin: 10px; padding: 4px; \" type=\"reset\" value=\"Επαναφορά\"/>";
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=\"ButtonHomeEx\"><a href=\"index.php\"><img class=\"ButtonHomeEx\"
src=\"images/home.png\" alt=\"Αρχική\"/></a></div>";
    // Κουμπί μετάβασης στη θεωρία του αντίστοιχου κεφαλαίου
    // echo "<div class=\"ButtonHome\"><a href=\"theory.php?pg=$CurrentPage&unit=1\"><img
class=\"ButtonHome\" src=\"images/theoryhome.png\" alt=\"Αρχική\"/></a></div>";
    // Κουμπί μετάβασης στις ασκήσεις του προηγούμενου κεφαλαίου
    //echo "<div class=\"ButtonLeft\"><a href=\"exercise.php?pg=$PreviousPage&unit=1\"><img
class=\"ButtonLeft\" 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
class=\"ButtonRight\" 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 #
    $ids = explode("#", $qo_id);
    for($i = 0; $i < count($ids)-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', $ids[$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 = $ids[$i]";
            $resultUpdate = mysqli_query($conn, $strSQLUpdate);
        }
    }
}

```

```
        }
    }
    header("Location: exercise.php?pg=$currentPage&unit=1");
}

?>
</body>
</html>
```

Κώδικας 10. 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/usermanagement.css" rel="stylesheet" type="text/css" />
    <title>Διαχείριση χρηστών</title>
  </head>
  <body>

    <!-- To 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>

<!-- 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 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>Επισκέπτης</input>
        </td>
        </tr>
        <tr>
        <td><p></p></td>
        <td>
        </td>
        </tr>
        <tr>
        <td>
            <input class="dbButton" type="submit" name="AddRecord" value="Αποθήκευση"
onclick="ResetFieldChange('UnsavedChanges');"></input>
        </td>
        <td>
            <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_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
$strSQL = "INSERT INTO users ".
    "(Usr_Username, Usr_Password, Usr_Lastname, Usr_Firstname, Usr_Email, ".
    "Usr_Phone, Usr_Role_Admin, Usr_Role_Contributor) VALUES (.
    '$usr_username', '$usr_password', '$usr_lastname', '$usr_firstname', ".
    '$usr_admin', '$usr_contributor)";
$result = mysqli_query($conn, $strSQL);
if (!$result)
{
    // Μήνυμα για διπλή εγγραφή
    header("Location: message.php?msgid=26");
}
}
}
?>
<!-- Τέλος του Popup εισαγωγής νέου χρήστη -->

<?php
    include 'header.php';          // Κεφαλίδα σελίδας (εμφάνιση στοιχείων συνδεδεμένου χρήστη)
?>
<div class="section">
    <?php
        echo "<h1 style=\"color: #4084b9; margin-left: 12px;\">Διαχείριση χρηστών</h1>";
        echo "<form name=\"datasetform\" action=\"".$_SERVER['PHP_SELF']."\" method=\"POST\">";
        ?>
        <div class="formarea" style="width: 1060px;">
            <!-- Η γραμμή εισαγωγής κειμένου UnsavedChanges τροποποιείται από 0 (default) -->
            <!-- σε 1 κάθε φορά που τροποποιείται ένα πεδίο εισαγωγής της φόρμας. -->
            <!-- Ανάλογα με την τιμή του πεδίου UnsavedChanges εμφανίζεται μήνυμα -->
```

```
<!-- προειδοποίησης όταν ο χρήστης εγκαταλείπει τη σελίδα χωρίς να αποθηκεύσει -->
<!-- τις αλλαγές -->
<input type="text" hidden name="UnsavedChanges" id="UnsavedChanges" value="0"></input>
<?php
// Πρόσβαση στη σελίδα διαχείρισης μόνο εάν ο συνδεδεμένος χρήστης είναι διαχειριστής
if (strcmp ($AdminAccess,"True")==0)
{
    // Εύρεση των καταχωρημένων χρηστών και εμφάνισή τους σε λίστα
    $strSQL = "SELECT Usr_ID, Usr_Username, Usr_Password, Usr_Lastname, ".
        "Usr_Firstname, Usr_Role_Admin, ".
        "Usr_Role_Contributor, Usr_Status ".
        "FROM users ".
        "ORDER BY Usr_Role_Admin, Usr_LastName, Usr_FirstName";
    $result = mysqli_query($conn, $strSQL);
    echo "<table>";
    // 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\">\".\"Διαχειριστής\".</p></td>";
    echo "<td><p class=\"Header\">\".\"Επισκέπτης\".</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_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=\"\".$usr_firstname.\"\"
size=\"10\" onchange=\"HandleFieldChange('UnsavedChanges')\"></input>";
        echo "</td>";
        // 7η στήλη (ένδειξη radio button ανάλογα με το αν ο χρήστης είναι Διαχειριστής ή Μαθητής)
        // Ανάλογα επιλέγεται το κατάλληλο radio button
        if (strcmp ($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);
            $usr_admin = stripslashes($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_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>
```

Κώδικας11. Ενδεικτικός κώδικας σελίδας παροχής πληροφοριακών στοιχείων

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

  <body>
    <h1>Η Αίθουσα των Κλιτύων της Ακρόπολης</h1>
    <p>
      Ο επισκέπτης του Μουσείου της Ακρόπολης, αφού διασχίσει τον προθάλαμο του ισογείου, βρίσκεται στον
      πρώτο εκθεσιακό χώρο. Στην ευρύχωρη αίθουσα με το κεκλιμένο γυάλινο δάπεδο παρουσιάζονται ευρήματα από τις κλιτύς
      (πλαγιές) του βράχου της Ακρόπολης. Το γυάλινο δάπεδο επιτρέπει την θέαση στην αρχαιολογική ανασκαφή, ενώ η ανηφορική
      κλίση της παραπέμπει στην ανάβαση προς την Ακρόπολη.
    </p>
    <p>
      Η Αίθουσα των κλιτύων της Ακρόπολης φιλοξενεί ευρήματα από τα ιερά που ήταν ιδρυμένα στις πλαγιές
      της Ακρόπολης, καθώς και από τον οικισμό που αναπτύχθηκε στις υπώρειες του λόφου, σε όλες τις ιστορικές περιόδους. Στα
      αριστερά του επισκέπτη εκτίθενται ευρήματα από τα μεγάλα ιερά που ιδρύθηκαν στις πλαγιές του βράχου, ενώ στα δεξιά του
      ευρήματα από τα μικρότερα ιερά και από τον οικισμό που αναπτύχθηκε στις πλαγιές της Ακρόπολης. Στην αρχαιότητα, οι
      πλαγιές αυτές ήταν η μεταβατική ζώνη ανάμεσα στην πόλη και το πιο διάσημο ιερό της. Ήταν η περιοχή όπου επίσημες και
      λαϊκές λατρείες, μικρά και μεγαλύτερα ιερά συνυπήρχαν με σπίτια ιδιωτών.
    </p>
  </body>
</html>
```