



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Εκπαιδευτικό λογισμικό: Σύστημα εκπαίδευσης και εξέτασης υποψηφίων οδηγών
Όνοματεπώνυμο Φοιτητή	Σταματίου Ειρήνη
Πατρώνυμο	Ιωάννης
Αριθμός Μητρώου	ΜΠΠΛ/ 07063
Επιβλέπουσα	Κα. Βίββου Μαρία

ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΑΔΟΣΗΣ ΜΗΝΑΣ ΕΤΟΣ

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

(υπογραφή)

Γεώργιος Τσιχριντζής
Καθηγητής

(υπογραφή)

Ευάγγελος Φούντας
Καθηγητής

Π Ε Ρ Ι Ε Χ Ο Μ Ε Ν Α

Ε Υ Χ Α Ρ Ι Σ Τ Ι Ε Σ	7
Κ Ε Φ Α Λ Α Ι Ο 1	8
1. ΕΙΣΑΓΩΓΗ	8
Κ Ε Φ Α Λ Α Ι Ο 2	10
2. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ	10
2.1. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ	10
2.1.1. ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ	10
2.1.2. ΑΔΥΝΑΜΙΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ	14
2.2. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: http://kok.artinoi.gr	14
2.2.1. ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ	14
2.2.2. ΑΔΥΝΑΜΙΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ	17
Κ Ε Φ Α Λ Α Ι Ο 3	18
3. ΑΝΑΛΥΣΗ ΚΑΙ ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ	18
3.1. ΦΑΣΗ «ΕΝΑΡΞΗ»	20
3.1.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ	20
3.1.1.1. Διαγράμματα περιπτώσεων χρήσης	20
3.1.1.2. Διαγράμματα τάξεων	23
3.1.1.3. Διαγράμματα αντικειμένων	25
3.1.2. Υλοποίηση - Έλεγχος	27

3.2. ΦΑΣΗ «ΕΚΠΟΝΗΣΗ ΜΕΛΕΤΗΣ»	27
3.2.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ	27
3.2.1.1. Διαγράμματα περιπτώσεων χρήσης	27
3.2.1.2. Διαγράμματα τάξεων	29
3.2.1.3. Διαγράμματα αντικειμένων	30
3.2.1.4. Διαγράμματα συνεργασίας	31
3.2.1.5. Διαγράμματα σειράς	32
3.2.1.6. Διαγράμματα καταστάσεων	34
3.2.1.7. Διαγράμματα δραστηριοτήτων	36
3.2.1.8. Διαγράμματα εξαρτημάτων	40
3.2.2. Υλοποίηση - Έλεγχος	41
Κ Ε Φ Α Λ Α Ι Ο 4	42
4. ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ	42
4.1. ΦΑΣΗ «ΚΑΤΑΣΚΕΥΗ»	42
4.1.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ	42
4.1.1.1. Βάση Δεδομένων	42
4.1.1.1.1. Πίνακες.....	42
4.1.1.1.2. Εννοιολογικός σχεδιασμός βάσης.....	44
4.1.1.1.3. Σχεσιακό σχήμα βάσης δεδομένων.....	51
4.1.1.1.4. Δεδομένα.....	52
4.1.1.2. Εφαρμογή	53
4.1.1.2.1. Modules.....	53
4.1.1.3. Περιγραφή εφαρμογής	56

Κ Ε Φ Α Λ Α Ι Ο 5	63
5. ΑΞΙΟΛΟΓΗΣΗ – ΣΥΓΚΡΙΣΗ ΕΦΑΡΜΟΓΩΝ	63
5.1. ΑΞΙΟΛΟΓΗΣΗ ΕΦΑΡΜΟΓΩΝ	63
5.1.1. ΣΥΣΤΗΜΑ ΕΚΠΑΙΔΕΥΣΗΣ ΚΑΙ ΕΞΕΤΑΣΗΣ ΥΠΟΨΗΦΙΩΝ ΟΔΗΓΩΝ	63
5.1.1.1. Ευκολία στην εκμάθηση	63
5.1.1.1.1. Προβλεψιμότητα	63
5.1.1.1.2. Σύνθεση	64
5.1.1.1.3. Οικειότητα	64
5.1.1.1.4. Συνέπεια	65
5.1.1.2. Ευκαμψία του συστήματος διεπαφής	66
5.1.1.2.1. Πολλαπλός έλεγχος	66
5.1.1.2.2. Δυνατότητα αλλαγής εργασιών	66
5.1.1.2.3. Προσαρμοστικότητα	66
5.1.1.3. Ανθεκτικότητα του συστήματος διεπαφής	67
5.1.1.3.1. Δυνατότητα παρατήρησης	67
5.1.1.3.2. Ανανηψιμότητα	68
5.1.1.3.3. Λειτουργικότητα	68
5.1.1.4. Ερωτήσεις γενικού περιεχομένου	69
5.1.2. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ	71
5.1.2.1. Ευκολία στην εκμάθηση	71
5.1.2.2. Ευκαμψία του συστήματος διεπαφής	72
5.1.2.3. Ανθεκτικότητα του συστήματος επαφής	72
5.1.2.4. Ερωτήσεις γενικού περιεχομένου	73
5.1.3. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: //kok.artinoi.gr	74
5.1.3.1. Ευκολία στην εκμάθηση	75
5.1.3.2. Ευκαμψία του συστήματος διεπαφής	75

5.1.3.3. Ανθεκτικότητα του συστήματος επαφής	76
5.1.3.4. Ερωτήσεις γενικού περιεχομένου	76
5.1.4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΑΞΙΟΛΟΓΗΣΗΣ	77
5.1.4.1. Σύστημα εκπαίδευσης και εξέτασης υποψηφίων οδηγών	77
5.1.4.1.1. <i>Ευκολία στην εκμάθηση</i>	77
5.1.4.1.2. <i>Ευκαμψία του συστήματος διεπαφής</i>	78
5.1.4.1.3. <i>Ανθεκτικότητα του συστήματος διεπαφής</i>	79
5.1.4.1.4. <i>Ερωτήσεις γενικού περιεχομένου</i>	80
5.1.4.2. Πρόγραμμα υπουργείου μεταφορών και επικοινωνιών	81
5.1.4.2.1. <i>Ευκολία στην εκμάθηση</i>	81
5.1.4.2.2. <i>Ευκαμψία του συστήματος διεπαφής</i>	82
5.1.4.2.3. <i>Ανθεκτικότητα του συστήματος επαφής</i>	82
5.1.4.2.4. <i>Ερωτήσεις γενικού περιεχομένου</i>	83
5.1.4.3. Πρόγραμμα ιστοσελίδας: kok.artinoi.gr	83
5.1.4.3.1. <i>Ευκολία στην εκμάθηση</i>	84
5.1.4.3.2. <i>Ευκαμψία του συστήματος διεπαφής</i>	84
5.1.4.3.3. <i>Ανθεκτικότητα του συστήματος επαφής</i>	85
5.1.4.3.4. <i>Ερωτήσεις γενικού περιεχομένου</i>	85
5.2. ΣΥΓΚΡΙΣΗ ΠΡΟΓΡΑΜΜΑΤΩΝ	86
5.2.1. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ	86
5.2.2. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: http://kok.artinoi.gr	87
5.2.3. ΠΡΟΓΡΑΜΜΑ «ΣΥΣΤΗΜΑ ΕΚΠΑΙΔΕΥΣΗΣ ΚΑΙ ΕΞΕΤΑΣΗΣ ΥΠΟΨΗΦΙΩΝ ΟΔΗΓΩΝ»	87
5.4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΣΥΓΚΡΙΣΗΣ	88

Κ Ε Φ Α Λ Α Ι Ο 6	91
6. ΣΥΜΠΕΡΑΣΜΑΤΑ	91
Π Α Ρ Α Ρ Τ Η Μ Α Τ Α	93
A. ΚΩΔΙΚΑΣ	93
A.1. ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ	93
A.1.1. MODULES	93
A.1.2. ΦΟΡΜΕΣ	110
B. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ	161
ΓΕΝΙΚΑ	161
ΔΙΑΧΕΙΡΙΣΤΗΣ ΣΥΣΤΗΜΑΤΟΣ	162
ΧΡΗΣΤΕΣ	169

ΕΥΧΑΡΙΣΤΙΕΣ

Με αφορμή την εκπόνηση της μεταπτυχιακής διατριβής, θα ήθελα να ευχαριστήσω μια ομάδα ανθρώπων, οι οποίοι βοήθησαν με διάφορους τρόπους στην ολοκλήρωσή της. Τους καθηγητές του μεταπτυχιακού προγράμματος για τις γνώσεις που μου μετέδωσαν και ιδιαίτερα την κυρία Βίρβου Μαρία για τις σκέψεις και τους προβληματισμούς που ανταλλάξαμε όλο αυτό τον καιρό. Τους φίλους μου που συμμετείχαν στη διαδικασία της αξιολόγησης του λογισμικού και τέλος την οικογένειά μου για την αμέριστη συμπαράστασή τους.

Κ Ε Φ Α Λ Α Ι Ο 1

1. ΕΙΣΑΓΩΓΗ

Σκοπός της εργασίας αυτής είναι η ανάλυση, η σχεδίαση, η υλοποίηση και η αξιολόγηση της εφαρμογής «Σύστημα εκπαίδευσης και εξέτασης υποψηφίων οδηγών». Πρόκειται για μια εφαρμογή κατάλληλη για τη θεωρητική προετοιμασία των υποψηφίων οδηγών αυτοκινήτου, η οποία θα μπορεί να χρησιμοποιηθεί από οποιαδήποτε σχολή οδηγών.

Στο πρώτο κεφάλαιο γίνεται μια περιγραφή του περιεχομένου των κεφαλαίων.

Στο δεύτερο κεφάλαιο παρουσιάζονται δύο παρόμοια προγράμματα με το «Σύστημα εκπαίδευσης και εξέτασης υποψηφίων οδηγών». Το πρώτο είναι το επίσημο πρόγραμμα του υπουργείου μεταφορών και επικοινωνιών και το δεύτερο είναι το πρόγραμμα που βρίσκεται στην ιστοσελίδα: <http://kok.artinoi.gr>. Εφόσον παρουσιαστεί ο τρόπος λειτουργίας του κάθε προγράμματος ακολουθεί αντίστοιχα και μια συνοπτική περιγραφή των αδυναμιών τους.

Στο τρίτο κεφάλαιο γίνεται η ανάλυση και ο σχεδιασμός της εφαρμογής. Στην αρχή αναφέρεται ο στόχος της εφαρμογής. Στη συνέχεια γίνεται μια αναφορά στη διαδικασία Rational Unified Process (RUP) η οποία χρησιμοποιήθηκε προκειμένου να σχεδιαστεί και να κατασκευαστεί η εφαρμογή. Εφόσον γίνει κατανοητή η διαδικασία RUP, ξεκινάει η υλοποίησή της. Συγκεκριμένα στην αρχή έχουμε την πρώτη φάση της διαδικασίας αυτής, την «Έναρξη». Εφόσον γίνεται μια αναφορά για τον τρόπο με τον οποίο θα λειτουργεί το λογισμικό, στη συνέχεια δημιουργώντας τα κατάλληλα διαγράμματα της φάσης αυτής προκειμένου:

- 1) να περιγραφεί με τον καλύτερο τρόπο η αλληλεπίδραση χρηστών και συστήματος (διαγράμματα περιπτώσεων χρήσης),
- 2) να αναπαρασταθεί η στατική δομή του συστήματος μέσω ενός συνόλου αντικειμένων που δημιουργούνται κατά την εκτέλεση του προγράμματος (κλάσεις) αλλά και μέσω των περιεχομένων τους (διαγράμματα τάξεων) και
- 3) να αναπαρασταθούν ορισμένα από τα σημαντικότερα στιγμιότυπα του συστήματος σε κάποια χρονική στιγμή προκειμένου να γίνουν κατανοητά τα διαγράμματα κλάσεων (διαγράμματα αντικειμένων).

Η φάση αυτή ολοκληρώνεται με τον απαραίτητο έλεγχο ανταπόκρισης των απαιτήσεων του πελάτη με όλα όσα υλοποιήθηκαν.

Στη συνέχεια, ακολουθεί η φάση «Εκπόνηση μελέτης», στην οποία εκτός από τις προηγούμενες κατηγορίες διαγραμμάτων προστίθενται άλλες πέντε προκειμένου:

- 1) να βρούμε τις κλάσεις του συστήματος και τη σχέση τους με τις περιπτώσεις χρήσης αλλά και να απεικονίσουμε τις προδιαγραφές υλοποίησης του συστήματος (διαγράμματα συνεργασίας)
- 2) να αναπαραστήσουμε χρονικά τη σειρά εκτέλεσης των γεγονότων που λαμβάνουν χώρα κατά τη διάρκεια αλληλεπίδρασης (διαγράμματα σειράς)
- 3) να παρουσιάσουμε όλες τις πιθανές καταστάσεις που μπορεί να έχει ένα αντικείμενο σε όλη τη διάρκεια ύπαρξής του καθώς και το πως αυτές αλλάζουν στο αντικείμενο (διαγράμματα καταστάσεων)
- 4) να εστιάσουμε σε ροές που προκαλούνται από την εσωτερική επεξεργασία (διαγράμματα δραστηριοτήτων)
- 5) να αναπαραστήσουμε την όψη υλοποίησης του συστήματος (διαγράμματα εξαρτημάτων).

Η φάση αυτή όπως και η προηγούμενη ολοκληρώνεται με τον απαραίτητο έλεγχο που πρέπει να γίνεται σε κάθε φάση.

Στο τέταρτο κεφάλαιο υπάρχει η φάση «Κατασκευή» της RUP. Εφόσον συλλέχθηκαν οι απαραίτητες πληροφορίες αλλά και δημιουργήθηκαν τα κατάλληλα διαγράμματα UML στις προηγούμενες φάσεις, προχωράμε στην κατασκευή της εφαρμογής. Στην αρχή δημιουργούμε τη βάση δεδομένων η οποία υλοποιήθηκε με την Ms Access και στη συνέχεια την εφαρμογή η οποία υλοποιήθηκε στη VB 6.0.

Όσον αφορά τη Βάση Δεδομένων, στην αρχή παρουσιάζονται οι πίνακες από τους οποίους θα αποτελείται η βάση καθώς και τα στοιχεία που θα καταγράφονται σε κάθε πίνακα. Ακολουθεί ο εντοπισμός των οντοτήτων και των γνωρισμάτων τους καθώς και οι σχέσεις τους προκειμένου να δημιουργηθεί το εννοιολογικό σχήμα της Β.Δ., βάση του οποίου δημιουργήθηκε και το εννοιολογικό σχήμα. Τέλος παρουσιάζεται ο κάθε πίνακας με τα δεδομένα του. Το κεφάλαιο ολοκληρώνεται με την παρουσίαση της εφαρμογής και του τρόπου λειτουργίας της, καθώς και την ανάλυση των αλγορίθμων.

Στο πέμπτο κεφάλαιο γίνεται η αξιολόγηση τόσο της δικής μας εφαρμογής όσο και των άλλων δύο, καθώς και η σύγκριση μεταξύ τους. Για την αξιολόγηση της εφαρμογής μας χρησιμοποιήθηκαν ερωτηματολόγια, τα οποία περιλάμβαναν ερωτήσεις που αφορούν:

- 1) το πόσο εύκολο είναι για το χρήστη να μάθει να χρησιμοποιεί τη συγκεκριμένη εφαρμογή,
- 2) τον τρόπο επικοινωνίας του χρήστη με το σύστημα,
- 3) το κατά πόσο η εφαρμογή μπορεί να στηρίξει το χρήστη σε πιθανά λάθη και
- 4) ερωτήσεις γενικού περιεχομένου

Η αξιολόγηση των άλλων δύο προγραμμάτων έγινε με τα ίδια ερωτηματολόγια που χρησιμοποιήθηκαν και στην εφαρμογή μας, με τη διαφορά ότι κάποιες ερωτήσεις παραλείφθηκαν ενώ αντίστοιχα προστέθηκαν κάποιες νέες. Ο λόγος για τον οποίο έγινε αυτό είναι διότι παρόλο που τα προγράμματα εξυπηρετούν τον ίδιο σκοπό, είναι διαφορετικά. Εφόσον ολοκληρωθεί η αξιολόγησή τους ακολουθούν τα συμπεράσματα της αξιολόγησης.

Στη συνέχεια ακολουθεί η σύγκριση των τριών προγραμμάτων, η οποία γίνεται μέσω της καταγραφής των πλεονεκτημάτων και των μειονεκτημάτων του κάθε προγράμματος. Όπως είναι κατανοητό, τα πλεονεκτήματα και τα μειονεκτήματα προέκυψαν από την αξιολόγηση. Το κεφάλαιο ολοκληρώνεται με τα συμπεράσματα που εξάγονται από τη σύγκριση των προγραμμάτων.

Στο έκτο κεφάλαιο παρουσιάζονται τα συμπεράσματα που προκύπτουν από τη χρήση της εφαρμογής μας, δηλαδή το κατά πόσο ανταποκρίνεται στο στόχο της και το αν επιδέχεται επεκτάσεις.

Κ Ε Φ Α Λ Α Ι Ο 2

2. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ

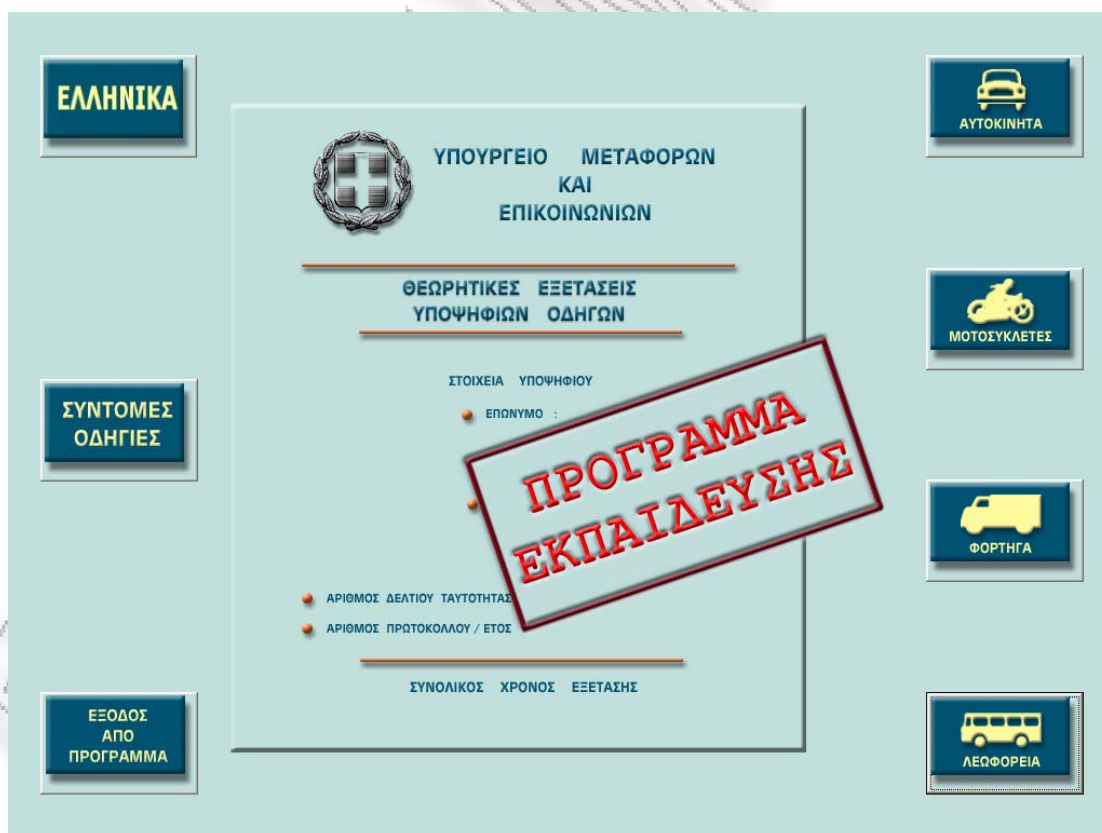
Κάθε υποψήφιος οδηγός έχει πλέον τη δυνατότητα επιλογής μεταξύ διαφόρων εκπαιδευτικών προγραμμάτων προκειμένου να προετοιμαστεί για τις θεωρητικές εξετάσεις απόκτησης του διπλώματος οδήγησης.

Σε αυτό το κεφάλαιο θα παρουσιάσουμε δύο από αυτά εκ των οποίων το ένα είναι του Υπουργείου Μεταφορών και Επικοινωνιών και το άλλο βρίσκεται στην ιστοσελίδα: <http://kok.artinoi.gr>. Εφόσον παρουσιάσουμε το πως λειτουργεί το κάθε πρόγραμμα στη συνέχεια θα αναφέρουμε τις αδυναμίες που παρουσιάζει καθένα από αυτά.

2.1. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

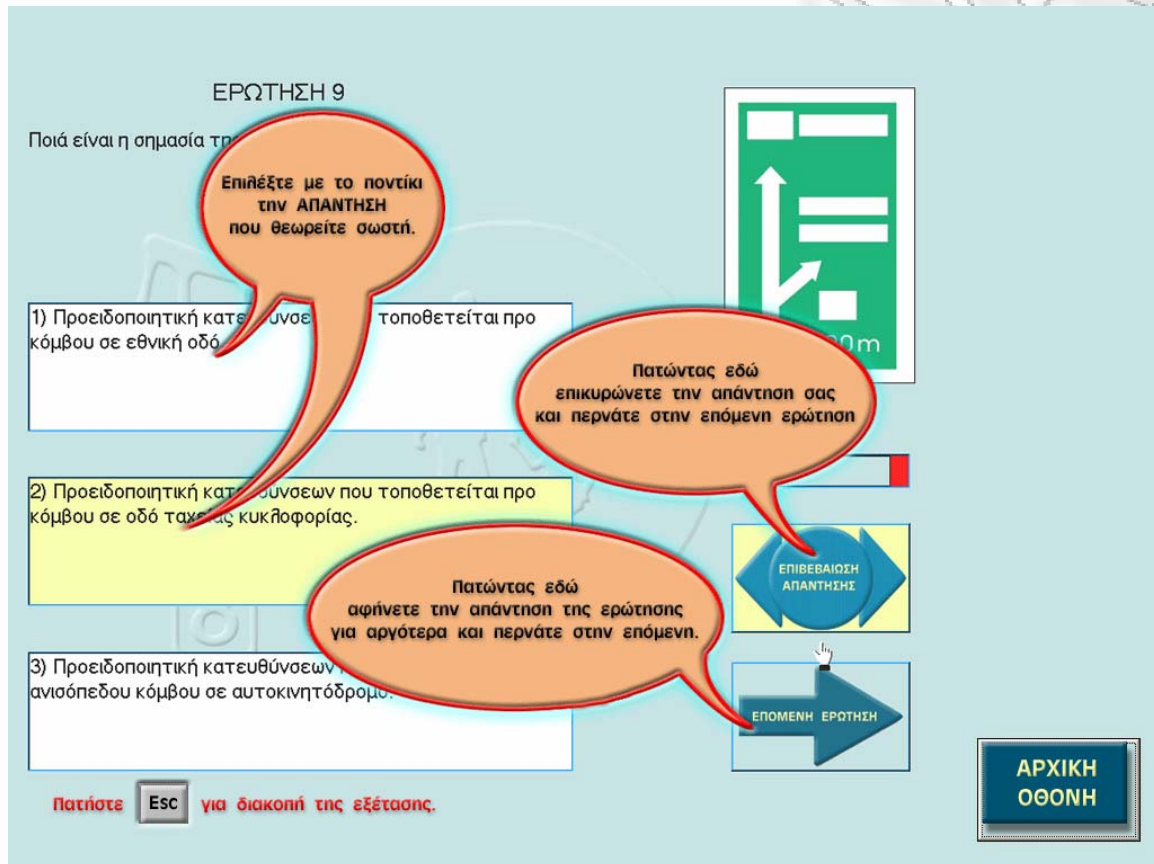
2.1.1. ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ

Το πρόγραμμα ξεκινάει με την παρακάτω οθόνη, από την οποία ο υποψήφιος οδηγός μπορεί να επιλέξει:



A) Γλώσσα εξέτασης. Το πρόγραμμα δίνει τη δυνατότητα να το χρησιμοποιήσουν όχι μόνο όσοι γνωρίζουν ελληνικά αλλά και αυτοί που γνωρίζουν Αγγλικά, Ρωσικά και Αλβανικά.

B) Σύντομες οδηγίες. Προκειμένου ο χρήστης να μάθει τον τρόπο λειτουργίας του προγράμματος και συγκεκριμένα το πως θα ολοκληρώσει το τεστ, επιλέγει αυτό το κουμπί κι εμφανίζεται η παρακάτω οθόνη.



Σε αυτή την οθόνη απεικονίζεται η μορφή με την οποία θα εμφανίζονται οι ερωτήσεις, ενώ με τις «επεξηγήσεις» που υπάρχουν, γίνεται κατανοητός ο τρόπος με τον οποίο πρέπει να λειτουργήσει ο χρήστης.

Συγκεκριμένα ο χρήστης πρώτα επιλέγει με το ποντίκι την απάντηση που θεωρεί σωστή, στη συνέχεια πατάει το κουμπί «Επιβεβαίωση απάντησης» και μετά προχωράει στην επόμενη ερώτηση πατώντας το αντίστοιχο κουμπί. Σε περίπτωση που ο χρήστης θέλει να διακόψει τη διεξαγωγή του τεστ χωρίς να το ολοκληρώσει, μπορεί να πατήσει esc. Στο δεξιό τμήμα του παραθύρου φαίνεται και ο χρόνος που απομένει για την ολοκλήρωση του τεστ.

Εφόσον ο χρήστης έχει κατανοήσει τον τρόπο με τον οποίο θα πρέπει να λειτουργήσει, επιλέγει το κουμπί «αρχική οθόνη» κι έτσι βρίσκεται και πάλι στην πρώτη οθόνη.


Γ) Κατηγορία διπλώματος. Ανάλογα με το αν ο υποψήφιος οδηγός θέλει να δώσει εξετάσεις για αυτοκίνητο, για μοτοσικλέτα, για φορτηγό ή για λεωφορείο, επιλέγει και την αντίστοιχη κατηγορία. Μόλις κάνει την επιλογή του, το τεστ ξεκινάει.

ΕΡΩΤΗΣΗ 1


Τελειώνετε το προσπεράσμα σας. Για να επανέλθετε στη δεξιά λωρίδα:


1) Περιμένετε να δείτε το όχημα που προσπεράσατε στον καθρέπτη.

2) Αφήνετε να περάσουν 5 δευτερόλεπτα.



Χρόνος






Όπως βλέπουμε το κουμπί «επιβεβαίωση απάντησης» είναι ανενεργό. Το κουμπί αυτό ενεργοποιείται μόνο όταν ο χρήστης επιλέξει μια από τις απαντήσεις.

ΕΡΩΤΗΣΗ 1


Τελειώνετε το προσπεράσμα σας. Για να επανέλθετε στη δεξιά λωρίδα:


1) Περιμένετε να δείτε το όχημα που προσπεράσατε στον καθρέπτη.

2) Αφήνετε να περάσουν 5 δευτερόλεπτα.



Χρόνος





Σε περίπτωση που ο χρήστης δεν έχει απαντήσει σε κάποια ερώτηση ή απάντησε αλλά ξέχασε να πατήσει την επιβεβαίωση, τότε στο τέλος όλων των ερωτήσεων θα εμφανιστεί ξανά αυτή η ερώτηση προκειμένου να απαντηθεί. Όταν το τεστ ολοκληρωθεί ή διακοπεί από το χρήστη με το esc, τότε θα εμφανιστεί πρώτα η παρακάτω οθόνη στην οποία και επιβεβαιώνεται η λήξη της εξέτασης.



Στη συνέχεια, εμφανίζεται η παρακάτω οθόνη, στην οποία φαίνεται το αποτέλεσμα της εξέτασης αλλά και το ποιες ερωτήσεις ήταν σωστές και ποιες λάθος.

ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΞΕΤΑΣΗΣ

A/A	ΕΡΩΤΗΣΗ	ΑΠΑΝΤΗΣΗ	ΧΑΡΑΚΤΗΡΙΣΜΟΣ
1	ΠΡΟΣΠΕΡΑΣΗ 20	1	ΛΑΘΟΣ
2	ΟΔΗΓΟΣ 03	2	ΛΑΘΟΣ
3	ΣΗΜΑΝΣΗ 151	3	ΛΑΘΟΣ
4	ΣΗΜΑΝΣΗ 62	2	ΛΑΘΟΣ
5	ΣΗΜΑΝΣΗ 48	3	ΛΑΘΟΣ
6	ΣΗΜΑΝΣΗ 176	1	ΛΑΘΟΣ
7	ΣΤΡΟΦΕΣ 01	2	ΣΩΣΤΗ
8	ΣΗΜΑΝΣΗ 20	2	ΛΑΘΟΣ
9	ΣΗΜΑΝΣΗ 23	3	ΛΑΘΟΣ
10	ΣΗΜΑΝΣΗ 133	3	ΛΑΘΟΣ
11	ΣΗΜΑΝΣΗ 198	1	ΛΑΘΟΣ
12	ΔΙΑΣΤΑΥΡΩΣΕΙΣ 21	2	ΛΑΘΟΣ
13	ΔΙΑΣΤΑΥΡΩΣΕΙΣ 73	3	ΛΑΘΟΣ
14	ΔΙΑΣΤΑΥΡΩΣΕΙΣ 78	1	ΛΑΘΟΣ
15	ΠΡΟΣΠΕΡΑΣΗ 14	2	ΣΩΣΤΗ
16	ΣΥΝΥΠΑΡΞΗ 12	2	ΛΑΘΟΣ
17	ΟΡΓΑΝΑ 16	2	ΛΑΘΟΣ
18	ΚΟΥΡΑΣΗ 04	2	ΣΩΣΤΗ
19	ΔΙΑΣΤΑΥΡΩΣΕΙΣ 100	1	ΣΩΣΤΗ
20	ΠΡΟΣΠΕΡΑΣΗ 25	2	ΣΩΣΤΗ
21	ΑΠΟΣΤΑΣΕΙΣ 13	2	ΛΑΘΟΣ
22	ΣΥΝΥΠΑΡΞΗ 100	2	ΛΑΘΟΣ

ΑΠΟΡΡΙΦΘΗΚΕ

ΑΡΧΙΚΗ ΟΘΟΝΗ

Αν ο χρήστης επιθυμεί να κάνει κι άλλο τεστ, επιλέγει το κουμπί «Αρχική οθόνη» και ακολουθεί πάλι την ίδια διαδικασία.

Δ) Έξοδος από πρόγραμμα. Πατώντας αυτό το κουμπί κλείνει το πρόγραμμα.

2.1.2. ΑΔΥΝΑΜΙΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Στο σημείο αυτό θα αναλύσουμε τις αδυναμίες που παρουσιάζει το πρόγραμμα του Υπουργείου.

Μια σημαντική αδυναμία του προγράμματος είναι η μη ύπαρξη θεωρίας πάνω στην οποία και βασίζονται οι ερωτήσεις. Το πρόγραμμα στην ουσία δίνει τη δυνατότητα στο χρήστη μόνο για χρήση τεστ και όχι για εκμάθηση. Αν για παράδειγμα ο χρήστης θέλει να δει το κομμάτι της θεωρίας στην οποία αναφέρεται μια ερώτηση, θα πρέπει να ανατρέξει σε δική του πηγή.

Ο χρήστης κατά τη διεξαγωγή ενός τεστ δεν έχει τη δυνατότητα να αλλάξει την απάντηση που έδωσε σε μια ερώτηση. Έτσι, σε περίπτωση που ο χρήστης δώσει λάθος απάντηση σε μια ερώτηση δεν μπορεί να το διορθώσει. Επίσης δεν του δίνεται η δυνατότητα να ξαναδεί τις απαντήσεις που έχει δώσει σε ερωτήσεις.

Η μη δυνατότητα δημιουργίας χρηστών αποτελεί επίσης ένα σημαντικό μειονέκτημα της εφαρμογής, καθώς ο χρήστης δε μπορεί να δει την πρόοδό που έχει κάνει κατά τη διάρκεια χρήσης της εφαρμογής.

Ακόμη, δε δίνεται η δυνατότητα στο χρήστη να επιλέξει βαθμό δυσκολίας του τεστ.

Τέλος, σε περίπτωση αλλαγής του κ.ο.κ. δε μπορούν να προστεθούν ερωτήσεις που θα αφορούν τα καινούρια κομμάτια του κ.ο.κ, όπως και σε περίπτωση κατάργησης κάποιων νόμων ή και αντικατάστασής τους το πρόγραμμα δε δίνει τη δυνατότητα για κατάργηση ή και αλλαγή ερωτήσεων. Με λίγα λόγια η εφαρμογή δεν έχει καμία πρόσβαση στη διαχείριση των ερωτήσεων.

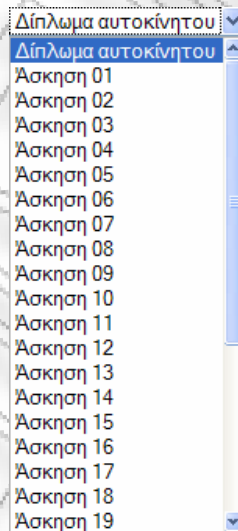
2.2. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: <http://kok.artinoi.gr>

2.2.1. ΠΑΡΟΥΣΙΑΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ

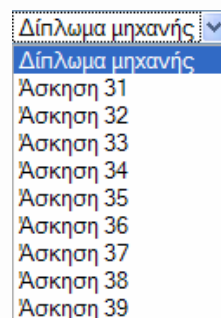
Το πρόγραμμα ξεκινάει με την παρακάτω οθόνη από την οποία μπορούμε να επιλέξουμε είτε την κατηγορία αυτοκίνητο είτε μηχανή.



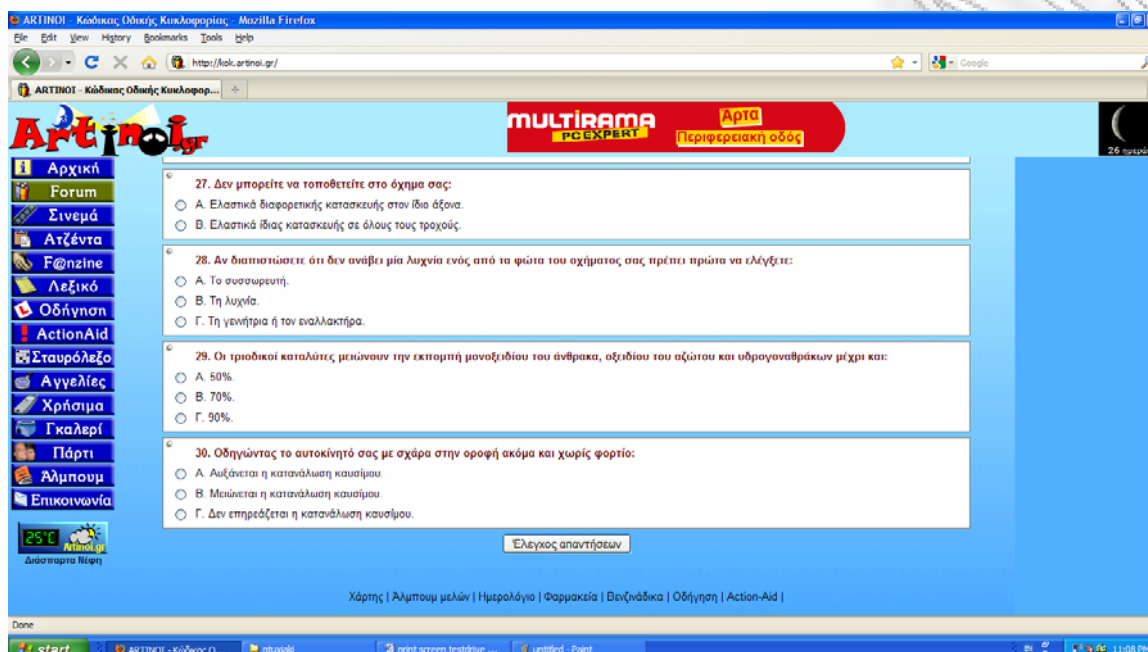
Αν επιλέξουμε «Δίπλωμα αυτοκινήτου», εμφανίζεται μια λίστα με όλα τα τεστ που έχουν καταχωρηθεί.



Το ίδιο συμβαίνει και για το δίπλωμα μηχανής.



Μόλις επιλέξουμε ένα από τα τεστ, εμφανίζεται η παρακάτω οθόνη.



Όπως βλέπουμε, όλες οι ερωτήσεις βρίσκονται σε μια σελίδα. Ο υποψήφιος οδηγός δεν έχει παρά να τσεκάρει κάθε φορά την απάντηση που θεωρεί σωστή. Σε περίπτωση που πιστεύει ότι μια απάντηση που έχει δώσει δεν είναι σωστή του δίνεται η δυνατότητα να τη διορθώσει αρκεί να επιλέξει αυτή που θεωρεί σωστή. Εφόσον έχει απαντήσει σε όλες τις ερωτήσεις ή σε όσες μπόρεσε πατάει το κουμπί «Έλεγχος απαντήσεων», προκειμένου να δει το αποτέλεσμα της εξέτασης.

Εμφανίζεται η παρακάτω οθόνη, στην οποία δεν εμφανίζονται οι σωστές απαντήσεις αλλά μόνο οι λάθος και αυτές που δεν έχουν απαντηθεί. Όπως βλέπουμε αν δεν έχουμε απαντήσει σε μια ερώτηση μας ενημερώνει με το μήνυμα «Δεν δώσατε καμία απάντηση» και σημειώνει με πράσινο ✓, τη σωστή απάντηση.

Αν πάλι η απάντηση που δόθηκε δεν είναι η σωστή, τότε εμφανίζεται το μήνυμα «Απαντήσατε λανθασμένα το...» και το γράμμα της απάντησης ενώ κοκκινίζει τη λάθος απάντηση βάζοντας και από μπροστά ένα κόκκινο X και σημειώνει τη σωστή με ένα πράσινο ✓.

Κάτω ακριβώς από τις ερωτήσεις εμφανίζεται ένα μήνυμα που μας ενημερώνει για το αν επιτύχομε ή όχι. Σε περίπτωση αποτυχίας μας ενημερώνει για τον αριθμό των λανθασμένων απαντήσεων που δώσαμε και μας δίνει τη δυνατότητα να ξανακάνουμε το συγκεκριμένο τεστ. Αν δε θέλουμε το ίδιο τεστ, με τη γραμμή κύλισης πηγαίνουμε στο πάνω μέρος της σελίδας όπου εκ νέου μπορούμε να επιλέξουμε ένα από τα τεστ.



2.2.2. ΑΔΥΝΑΜΙΕΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Η εφαρμογή αυτή παρουσιάζει αρκετές αδυναμίες με αποτέλεσμα να μη τη καθιστούν κατάλληλη για το σκοπό μας.

Τα τεστ που περιλαμβάνει είναι συγκεκριμένα επομένως και οι ερωτήσεις που περιλαμβάνει καθένα από αυτά δε δίδονται με τυχαία σειρά. Όπως είναι κατανοητό ο χρήστης δε μπορεί να χρησιμοποιήσει πολλές φορές το κάθε τεστ.

Η εφαρμογή δε περιλαμβάνει τη θεωρία πάνω στην οποία βασίζονται οι ερωτήσεις με αποτέλεσμα ο χρήστης, παρόλο που μπορεί να δει τα λάθη που έχει κάνει σε ένα τεστ να μη μπορεί να ανατρέξει στη συγκεκριμένη θεωρία και έτσι να γνωρίζει το λόγο που η απάντησή τους είναι λάθος. Κατ' επέκταση ο χρήστης δε μαθαίνει σωστά αλλά το μόνο που μπορεί να κάνει είναι να μάθει μηχανικά κάποιες απαντήσεις.

Σημαντική επίσης αδυναμία της εφαρμογής είναι το γεγονός ότι ο χρήστης δε μπορεί να δει τη πρόοδό του. Για παράδειγμα να κάνει κάποια τεστ και στη συνέχεια να δει συνολικά το ποσοστό επιτυχίας του.

Σε περίπτωση που για κάποιους λόγους ο χρήστης θελήσει να διαχειριστεί τις ερωτήσεις, δεν του δίδεται αυτή η δυνατότητα. Διαγραφή, αλλαγή ή και προσθήκη ερώτησης λοιπόν είναι αδύνατη με αποτέλεσμα το πρόγραμμα να είναι κάπως στατικό.

Μια επίσης σημαντική δυνατότητα που θα μπορούσε να δώσει η εφαρμογή θα ήταν η επιλογή του βαθμού δυσκολίας των τεστ προκειμένου ο χρήστης να μπορεί να επιλέγει, ανάλογα με τις δυνατότητες που πιστεύει ότι έχει.

Τέλος η εφαρμογή εκτελείται μόνο στην ελληνική γλώσσα μη δίδοντας έτσι τη δυνατότητα και σε άλλους χρήστες που δε γνωρίζουν την ελληνική γλώσσα να τη χρησιμοποιήσουν.

Κ Ε Φ Α Λ Α Ι Ο 3

3. ΑΝΑΛΥΣΗ ΚΑΙ ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ

Προτού προχωρήσουμε στην ανάλυση και τη σχεδίαση της εφαρμογής θα πρέπει να αναφερθούμε στο στόχο της, ο οποίος δεν είναι άλλος από τη θεωρητική προετοιμασία των υποψηφίων οδηγών αυτοκινήτου.

Η εφαρμογή λοιπόν δίνει τη δυνατότητα στους υποψήφιους οδηγούς να προετοιμαστούν με τον καλύτερο δυνατό τρόπο για τις θεωρητικές εξετάσεις που απαιτούνται για την απόκτηση διπλώματος αυτοκινήτου.

Ο υποψήφιος οδηγός, μέσω της μελέτης της θεωρία που περιλαμβάνει η εφαρμογή μπορεί να προετοιμαστεί και έτσι να αποκτήσει τις απαραίτητες γνώσεις προκειμένου να προβεί στη διεξαγωγή των τεστ που επίσης περιλαμβάνονται σε αυτή. Ανάλογα με τις γνώσεις που πιστεύει ότι έχει αποκτήσει ο χρήστης επιλέγει και το βαθμό δυσκολίας των τεστ που θα κάνει. Όταν φτάσει στο σημείο που πιστεύει ότι έχει ολοκληρώσει επιτυχώς την εκμάθηση της θεωρίας μπορεί και να επιλέξει και τον πιο δύσκολο τύπο τεστ.

Ο χρήστης επίσης όχι μόνο μπορεί να βλέπει τα λάθη που έχει κάνει σε κάποιο τεστ αλλά να βλέπει και το κομμάτι της θεωρίας στο οποίο αντιστοιχεί η ερώτηση στην οποία έχει κάνει λάθος.

Τέλος, ο χρήστης μπορεί να δει ανά πάσα στιγμή αν είναι έτοιμος να δώσει τις θεωρητικές εξετάσεις καθώς του δίνεται η δυνατότητα να ελέγχει τη πρόοδό του.

Εφόσον έγινε κατανοητός ο στόχος της εφαρμογής μπορούμε να προχωρήσουμε στη δημιουργία της. Τα πρώτα από τα βήματα που θα ακολουθήσουμε είναι η ανάλυση και ο σχεδιασμός της.

Προκειμένου να προχωρήσουμε στην ανάλυση και σχεδίαση του λογισμικού, θα πρέπει να επιλέξουμε ένα από τα μοντέλα του κύκλου ζωής λογισμικού, δηλαδή τον τρόπο με τον οποίο θα περιγράψουμε τις φάσεις από τις οποίες θα περάσει η εφαρμογή μας από τη σύλληψη μέχρι και την απόσυρσή της, καθώς και τις ενέργειες που λαμβάνουν χώρα σε κάθε μια από αυτές. Η επιλογή μας είναι η διαδικασία Rational Unified Process (RUP) στην οποία ο κύκλος ζωής του λογισμικού είναι επαναληπτικός, δηλαδή το λογισμικό αναπτύσσεται μέσα από μια σειρά επαναλήψεων μέχρι το τελικό προϊόν να εξελιχθεί. Η διαδικασία αυτή είναι δομημένη σε δύο διαστάσεις:

- 1) **Χρόνο.** Ο κύκλος ζωής χωρίζεται σε φάσεις και επαναλήψεις.
- 2) **Τμήματα διαδικασίας.** Καλά ορισμένες ενέργειες.

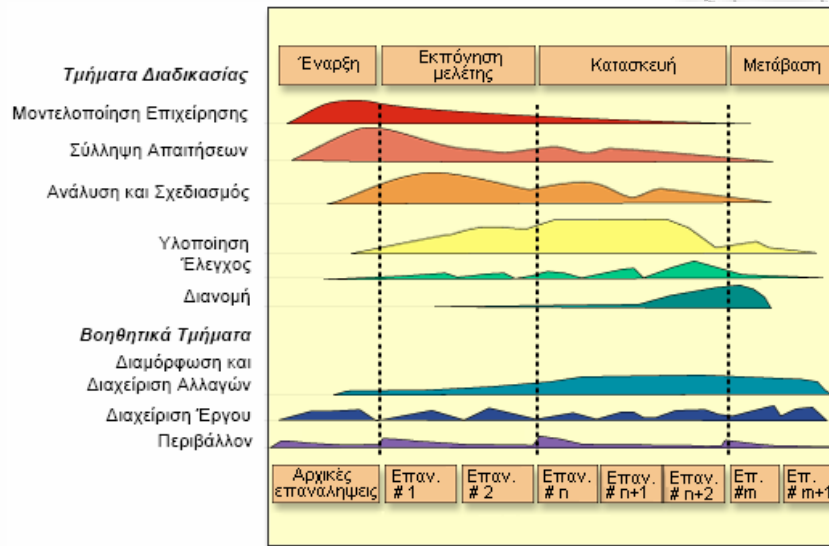
Όσον αφορά το χρόνο, η δόμηση του έργου ακολουθεί τις παρακάτω φάσεις:

- **Έναρξη.** Καθορίζεται η προοπτική του έργου.
- **Εκπόνηση μελέτης.** Σχεδιάζονται οι απαιτούμενες δραστηριότητες και πόροι. Επίσης, καθορίζονται τα χαρακτηριστικά και ο σχεδιασμός της αρχιτεκτονικής.
- **Κατασκευή.** Αναπτύσσεται το προϊόν σε μια σειρά βηματικών επαναλήψεων.
- **Μετάβαση.** Το προϊόν προμηθεύεται στους χρήστες.

Όσον αφορά τα τμήματα διαδικασίας, η δόμηση του έργου περιλαμβάνει τις ακόλουθες δραστηριότητες:

- **Σύλληψη απαιτήσεων.** Αφηγούμαστε τι πρέπει να κάνει το σύστημα.
- **Ανάλυση και σχεδιασμός.** Περιγράφουμε το πως θα υλοποιηθεί το σύστημα.
- **Υλοποίηση.** Παράγουμε τον κώδικα.
- **Έλεγχος.** Επαληθεύουμε το σύστημα.

Στο παρακάτω σχήμα φαίνεται πως αυτές οι δύο διαστάσεις επηρεάζονται η μία από την άλλη καθώς και πως συνυπάρχουν.



Χρησιμοποιώντας τη UML θα μοντελοποιήσουμε τη διαδικασία RUP. Η UML είναι μια οπτική αντικειμενοστραφής γλώσσα μοντελοποίησης που χρησιμοποιείται για απεικόνιση, προδιαγραφή, τεκμηρίωση και κατασκευή των δομικών συστατικών ενός συστήματος. Τα κυριότερα χαρακτηριστικά της είναι τα παρακάτω:

- Εκφραστική.
- Εκτενής σημασιολογικά καθώς υποστηρίζει τη σημασιολογία τύπων και μοντέλων για όλα τα μοντέλα ενός συστήματος.
- Απλή και επεκτάσιμη καθώς το μετα-μοντέλο μπορεί να εμπλουτιστεί με κλάσεις, ιδιότητες και σημασιολογία.
- Επακριβώς ορισμένη με βάση τα δομικά συστατικά ενός αντικειμενοστραφούς συστήματος.

Η UML ορίζει 9 είδη διαγραμμάτων:

- 1) **Διαγράμματα τάξεων.** Παρουσιάζουν κάποια στατικά στοιχεία μοντελοποίησης και τις σχέσεις μεταξύ τους.
- 2) **Διαγράμματα αντικειμένων.** Αναπαριστούν αντικείμενα και τις σχέσεις του και αντιστοιχούν σε απλοποιημένα διαγράμματα συνεργασίας που δεν αναπαριστούν μετάδοση μηνυμάτων.
- 3) **Διαγράμματα συνεργασίας.** Παρουσιάζουν μια αλληλεπίδραση αντικειμένων με έμφαση στη δομική τους οργάνωση.
- 4) **Διαγράμματα σειρές.** Παρουσιάζουν μια αλληλεπίδραση αντικειμένων με έμφαση στη χρονική σειρά ανταλλαγής μηνυμάτων. Μαζί με τα διαγράμματα συνεργασίας μπορούν να εντοπισθούν σε διαγράμματα αλληλεπίδρασης.
- 5) **Διαγράμματα καταστάσεων.** Παρουσιάζουν τις καταστάσεις του κύκλου ζωής των αντικειμένων.
- 6) **Διαγράμματα δραστηριοτήτων.** Παρουσιάζουν την ακολουθιακή ροή των δραστηριοτήτων και περιέχουν προσδιορισμούς των μηνυμάτων που στέλνονται.

- 7) **Διαγράμματα εξαρτημάτων.** Δείχνουν τα συστατικά μέρη του κώδικα και τη φυσική τους δομή. Παρουσιάζουν την οργάνωση και τις εξαρτήσεις των μελών ενός συνόλου συνιστωσών.
- 8) **Διαγράμματα διανομής.** Παρουσιάζουν την τοπολογία των υπολογιστικών κόμβων ενός συστήματος και τύπο των συνδέσεων ενώ περιέχουν εκτελέσιμα αντικείμενα που φανερώνουν ποιες μονάδες λογισμικού εκτελούνται σε κάθε κόμβο.
- 9) **Διαγράμματα περιπτώσεων χρήσης.** Περιγράφουν τη λειτουργικότητα του συστήματος όπως αυτή γίνεται αντιληπτή από εξωτερικές οντότητες.

3.1. ΦΑΣΗ «ΕΝΑΡΞΗ»

3.1.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ

Η εφαρμογή που θα δημιουργήσουμε θα πρέπει να παρέχει την κατάλληλη θεωρητική προετοιμασία για τους υποψήφιους οδηγούς αυτοκινήτου και θα μπορεί να χρησιμοποιηθεί από σχολές οδηγών.

Συγκεκριμένα, ο υποψήφιος οδηγός προκειμένου να κάνει χρήση του προγράμματος θα πρέπει πρώτα να εγγραφεί σε αυτό. Την εγγραφή του την αναλαμβάνει ένας από τους διαχειριστές του συστήματος. Οι διαχειριστές συστήματος μπορεί να είναι ο ιδιοκτήτης της σχολής αλλά και κάποιοι υπάλληλοί του.

Εφόσον γίνει η εγγραφή του υποψηφίου οδηγού, ο χρήστης μπορεί να χρησιμοποιήσει την εφαρμογή. Εισάγει το μοναδικό κωδικό του και στη συνέχεια προχωράει στη διεξαγωγή των τεστ. Συγκεκριμένα, επιλέγει το βαθμό δυσκολίας του τεστ και εφόσον το τεστ φορτώσει, το κάνει. Μόλις ολοκληρώσει το τεστ θα εμφανιστεί ένα μήνυμα στο οποίο θα φαίνεται αν το ολοκληρώσει επιτυχώς ή όχι.

Στην περίπτωση που ο υποψήφιος οδηγός έχει απαντήσει λανθασμένα σε κάποιες ερωτήσεις, έχει τη δυνατότητα να τσεκάρει το τεστ που έκανε. Έτσι, μπορεί να δει ακριβώς τις ερωτήσεις με τις λάθος απαντήσεις αλλά και να διαβάσει το θεωρητικό κομμάτι στο οποίο αναφέρονται, προκειμένου όχι μόνο να δει το λάθος του αλλά να το κατανοήσει και κατ'επέκταση να μάθει και τη σωστή απάντηση. Όσα τεστ κι αν κάνει ο χρήστης μπορεί ανά πάσα στιγμή να τα ελέγξει. Όπως κάθε εφαρμογή, έτσι και η συγκεκριμένη έχει τη «βοήθεια» που αφορά τον τρόπο λειτουργίας της. Αν λοιπόν, ο υποψήφιος οδηγός δεν γνωρίζει την εφαρμογή, μέσω της «βοήθειας» μπορεί να κατατοπιστεί πλήρως.

Όσον αφορά τώρα τους διαχειριστές του συστήματος, αυτοί, εκτός από τη διαχείριση των λογαριασμών των χρηστών, έχουν τη δυνατότητα να ελέγξουν όλα τα τεστ που έχουν κάνει όλοι οι υποψήφιοι οδηγοί. Μια πολύ σημαντική λειτουργία της εφαρμογής είναι η δυνατότητα που δίνει στον διαχειριστή να διαχειρίζεται τις ερωτήσεις αλλά και τη θεωρία. Σε περίπτωση λοιπόν που είτε αλλάξει η ύλη των εξετάσεων είτε προστεθεί είτε αφαιρεθεί κάτι, ο διαχειριστής μπορεί μέσω της εφαρμογής να προσθέσει ή και να αφαιρέσει ερωτήσεις αλλά και θεωρητικά κομμάτια. Τέλος, μπορεί να δημιουργήσει καινούριο τύπο τεστ με βάση το βαθμό δυσκολίας που επιθυμεί να έχει.

3.1.1.1. Διαγράμματα περιπτώσεων χρήσης

Προκειμένου να ανταποκριθούμε στις απαιτήσεις αυτής της φάσης θα χρησιμοποιήσουμε το διάγραμμα περιπτώσεων χρήσης. Μέσω του διαγράμματος περιπτώσεων χρήσης μοντελοποιούμε το πλαίσιο λειτουργίας του συστήματος καθώς και τις προδιαγραφές

του και έτσι περιγράφεται η αλληλεπίδραση μεταξύ χρηστών και συστήματος. Συγκεκριμένα αποτυπώνεται η συμπεριφορά ενός συστήματος υποσυστήματος ή κλάσης όπως αυτή εμφανίζεται σε ένα εξωτερικό χρήστη.

Επίσης χωρίζει τη λειτουργικότητα σε λειτουργίες του συστήματος οι οποίες εκκινούνται από τους ενεργοποιητές και ονομάζονται περιπτώσεις χρήσης. Ως ενεργοποιητής θεωρείται οποιοσδήποτε αλληλεπιδρά με το σύστημα και μπορεί να είναι είτε οι χρήστες του συστήματος είτε άλλα συστήματα είτε ο χρόνος. Ως περίπτωση χρήσης εννοούμε ένα σύνολο σεναρίων που συνδέονται με ένα κοινό σκοπό (για το χρήστη).

Συνήθως περιλαμβάνει:

- Ένα πρωταρχικό σενάριο (σαν μια ακολουθία από βήματα).
- Τις εναλλακτικές περιπτώσεις (σαν παραλλαγές της κύριας ακολουθίας βημάτων).

Πρόκειται λοιπόν, για μια τεχνική προτυποποίησης που περιγράφει το τι θα πρέπει να κάνει ένα καινούριο σύστημα. Οικοδομείται δια μέσω μιας επαναληπτικής διαλογικής διαδικασίας μεταξύ ειδικών και πελατών που οδηγεί σε προδιαγραφές απαιτήσεων που όλοι συμφωνούν και αποδέχονται.

Σημειολογικά οι ενεργοποιητές:

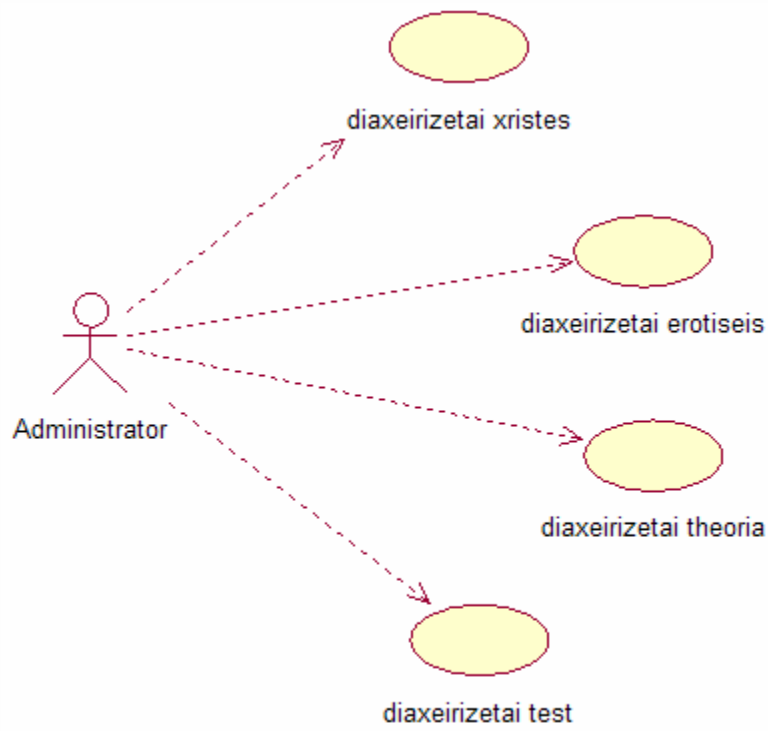


Σημειολογικά οι περιπτώσεις χρήσης:

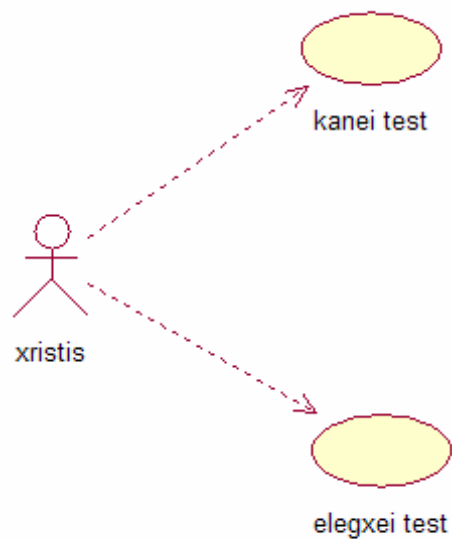


Βάση αυτών που αναφέραμε, θα δημιουργήσουμε το διάγραμμα περιπτώσεων χρήσης του διαχειριστή συστήματος και του απλού χρήστη.

Διάγραμμα περιπτώσεων χρήσης administrator



Διάγραμμα περιπτώσεων χρήσης απλού χρήστη



3.1.1.2. Διαγράμματα τάξεων

Σε ένα διάγραμμα τάξεων αναπαρίστανται η στατική δομή του συστήματος σε επίπεδο κλάσεων και τα περιεχόμενά τους. Επίσης, αναπαρίστανται οι σχέσεις μεταξύ των κλάσεων, χρησιμοποιώντας σχέσεις όπως η συσχέτιση, η εξειδίκευση, η γενίκευση και άλλες.






Με τον όρο κλάση εννοούμε ένα σύνολο αντικειμένων που δημιουργούνται κατά τη διάρκεια της εκτέλεσης του προγράμματος και τα οποία έχουν όμοιες ιδιότητες, κοινή συμπεριφορά και κοινές συσχετίσεις με άλλα αντικείμενα. Τα αντικείμενα με τη σειρά τους αποτελούν στιγμιότυπα κλάσεων.

Αρχικά το διάγραμμα τάξεων περιλαμβάνει τις βασικές κλάσεις του συστήματος, το οποίο στη συνέχεια εμπλουτίζεται με νέες κλάσεις οι οποίες προκύπτουν κατά την ανάλυση του συστήματος και τελικά προστίθενται κλάσεις και διεπαφές που σχετίζονται με τον τρόπο υλοποίησης του συστήματος.

Σημειολογικά η κλάση:

Όνομα κλάσης
Ιδιότητες
Λειτουργίες

Σημειολογικά οι σχέσεις:

Συσχέτιση	- Μια σχέση ανάμεσα σε δύο ή περισσότερες κλάσεις ή ταξινομητές.	
Συσσώρευση & Σύνθεση	- Μια ειδική περίπτωση εταιρικότητας που περιγράφει τη σχέση «σύνθετου όλου» με τα «συστατικά μέρη».	 
Γενίκευση	- Μια σχέση ταξινόμησης ανάμεσα σε μια γενική και πιο ειδική οντότητα.	
Εξάρτηση	- Η σχέση ανάμεσα σε δυο οντότητες που περιγράφει σχέση εξάρτησης σύμφωνα με την οποία αλλαγές στη μία οντότητα (εξαρτούσα) επηρεάζουν την άλλη οντότητα (εξαρτώμενη).	

Υλοποίηση - Η σχέση ανάμεσα σε μία προδιαγραφή.



Πλοηγισμότητα - Η σχέση ανάμεσα σε δύο κλάσεις που περιγράφει τη δυνατότητα από τη μία κλάση ανάκτησης αντικειμένων της άλλης

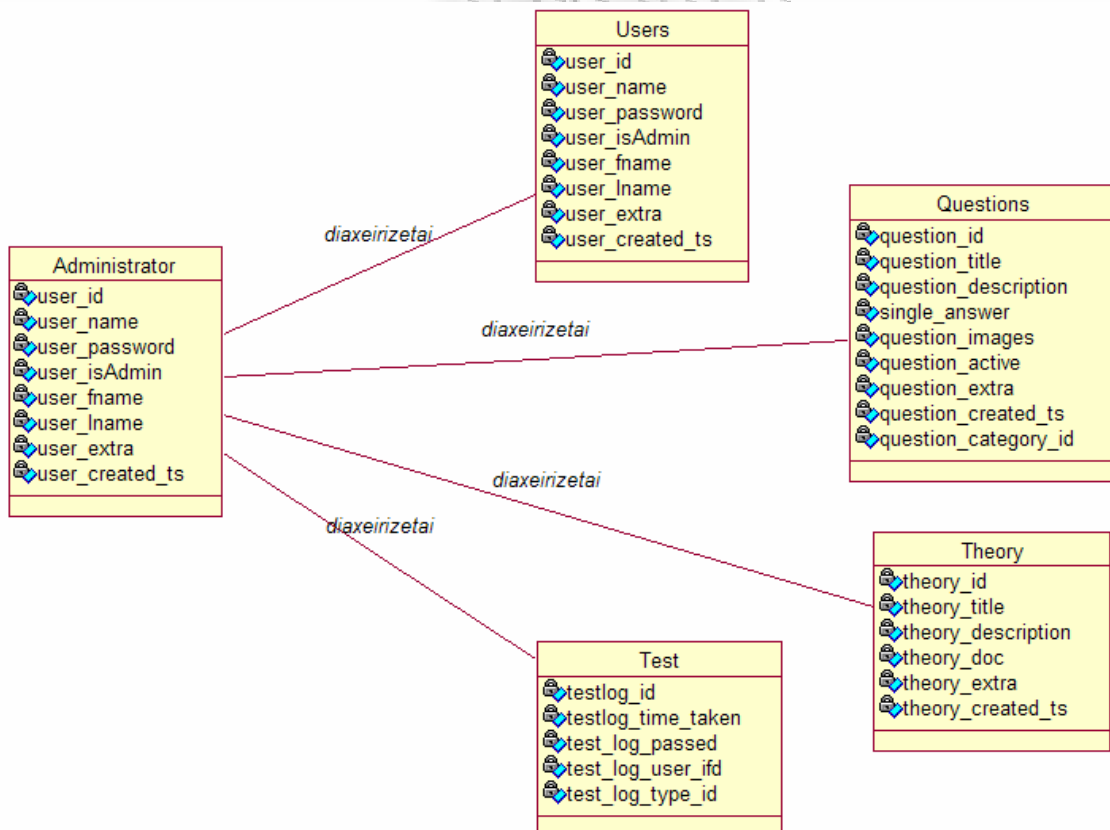


Συνήθως, οι ιδιότητες των κλάσεων είναι ιδιωτικές, ενώ οι λειτουργίες είναι δημόσιες. Με τον όρο ιδιωτικές εννοούμε ότι τα χαρακτηριστικά μιας κλάσης δεν είναι προσπελάσιμα από άλλες, σε αντίθεση με τις δημόσιες.

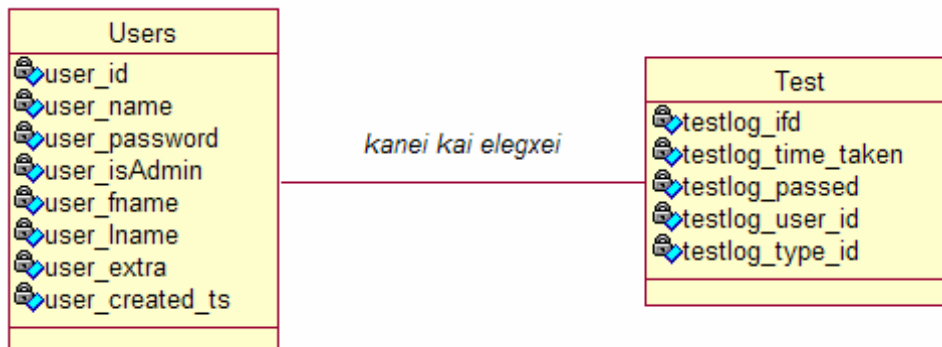
Η ιδιωτική πρόσβαση συμβολίζεται με «-» ενώ η δημόσια με «+».

Στη συνέχεια προχωράμε στη δημιουργία των διαγραμμάτων τάξεων. Ακολουθούν δύο διαγράμματα.

Διάγραμμα τάξεων administrator



Διάγραμμα τάξεων απλού χρήστη



3.1.1.3. Διαγράμματα αντικειμένων

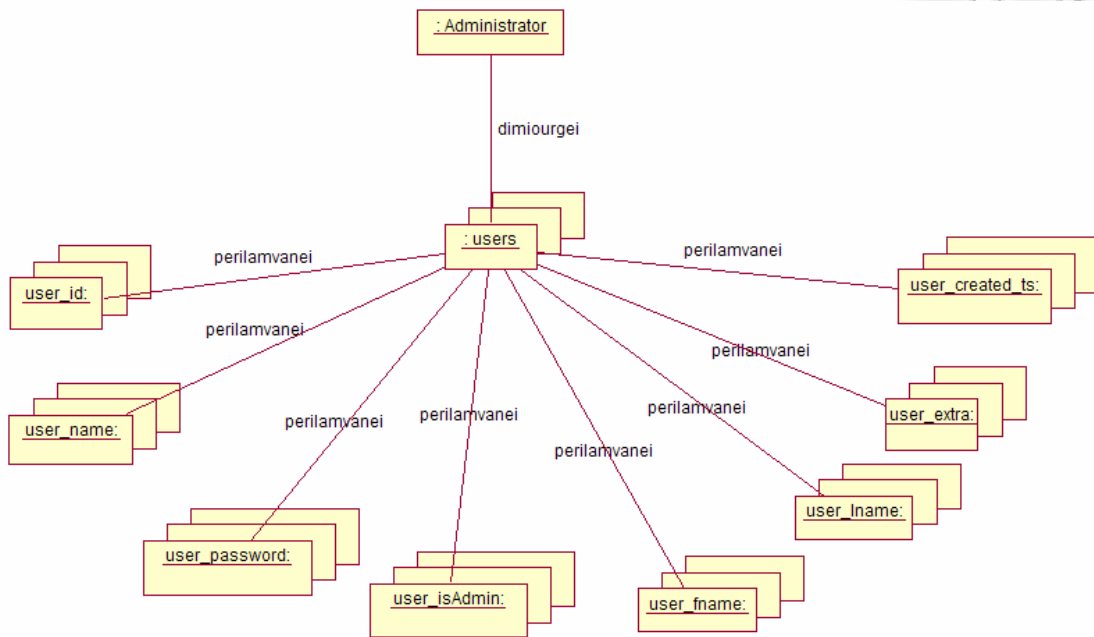
Το διάγραμμα αντικειμένων είναι ένα στιγμιότυπο του συστήματος σε κάποια χρονική στιγμή στο οποίο απεικονίζονται αντικείμενα που προέρχονται από κλάσεις σε ένα διάγραμμα κλάσεων καθώς και οι σύνδεσμοι μεταξύ των αντικειμένων που προέρχονται από τις συσχετίσεις μεταξύ των κλάσεων. Καταλαβαίνουμε λοιπόν ότι μέσω αυτού του διαγράμματος μπορούμε να κατανοήσουμε ένα διάγραμμα κλάσεων.

Όσον αφορά τα αντικείμενα, αυτά συμβολίζονται όπως και οι κλάσεις με τη διαφορά ότι μπροστά από το όνομα της κλάσης βάζουμε το όνομα του αντικειμένου και μια άνω – κάτω τελεία.

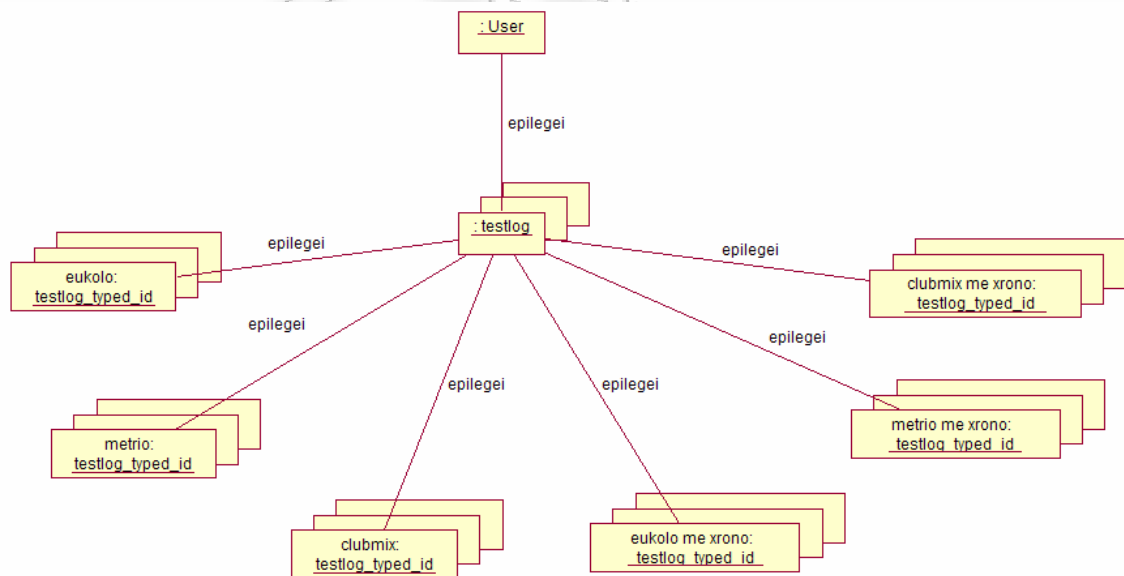
Στην ουσία το διάγραμμα αντικειμένων είναι ένα παράδειγμα για να κατανοήσουμε ή να εξηγήσουμε το διάγραμμα κλάσεων.

Με βάση λοιπόν το διάγραμμα τάξεων δημιουργούμε τα παρακάτω διαγράμματα αντικειμένων.

Διάγραμμα αντικειμένων administrator



Διάγραμμα αντικειμένων επιλογής τεστ



3.1.2. ΥΛΟΠΟΙΗΣΗ - ΈΛΕΓΧΟΣ

Στην αρχή αυτής της φάσης καταγράψαμε τις απαιτήσεις ενός τέτοιου προγράμματος και στη συνέχεια προχωρήσαμε στη μοντελοποίηση των σημαντικότερων περιπτώσεων χρήσης καθώς και στη δημιουργία διαγραμμάτων τάξεων και αντικειμένων.

Στο σημείο αυτό θα πρέπει να ελέγξουμε αν όλα όσα υλοποιήσαμε ανταποκρίνονται στις απαιτήσεις του πελάτη. Προκειμένου να προχωρήσουμε στον έλεγχο θα αξιολογήσουμε κάποια βασικά κριτήρια ελέγχου της συγκεκριμένης φάσης. Τα κριτήρια αυτά είναι τα παρακάτω:

- Συμφωνία μεριδιούχων έργου όσον αφορά το σκοπό του έργου καθώς και για τις εκτιμήσεις που έχουν γίνει και αφορούν το κόστος αλλά και το χρονοδιάγραμμα υλοποίησης.
- Κατανόηση των απαιτήσεων του πελάτη καθώς και ακριβής απόδοση των πρωτεύοντων περιπτώσεων χρήσης.
- Εύρος και βαθμός αρχιτεκτονικού προτύπου που έχει αναπτυχθεί.
- Λόγος πραγματικών προς υποθετικών δαπανών
- Ακύρωση ή επανεξέταση έργου σε περίπτωση αποτυχία ικανοποίησης ορόσημου.

Εφόσον αξιολογήσαμε τα κριτήρια αυτά καταλήξαμε στο ότι κατανοήσαμε πλήρως τις απαιτήσεις του πελάτη με συνέπεια τη δυνατότητα μετάβασης στην επόμενη φάση.

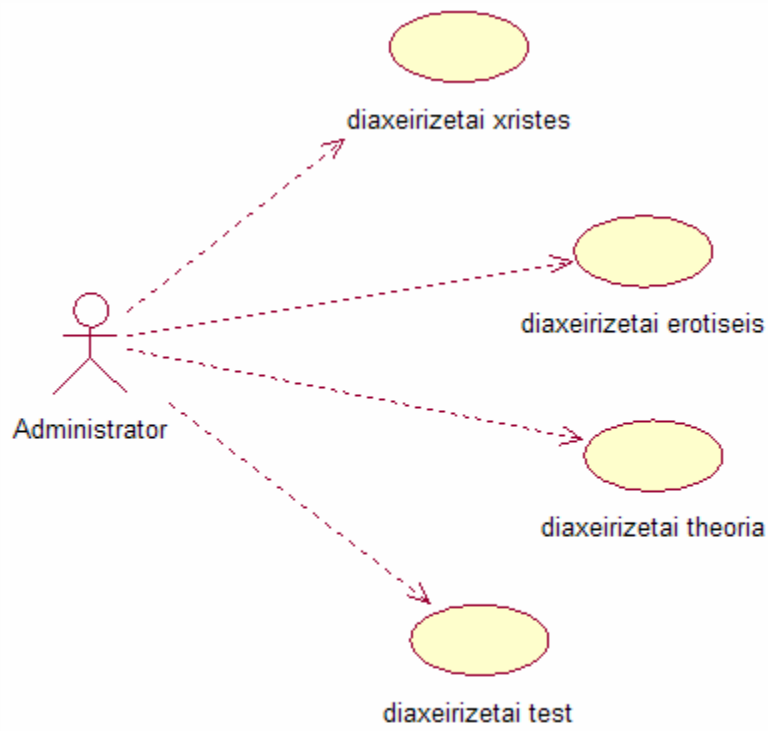
3.2. ΦΑΣΗ «ΕΚΠΟΝΗΣΗ ΜΕΛΕΤΗΣ»

3.2.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ

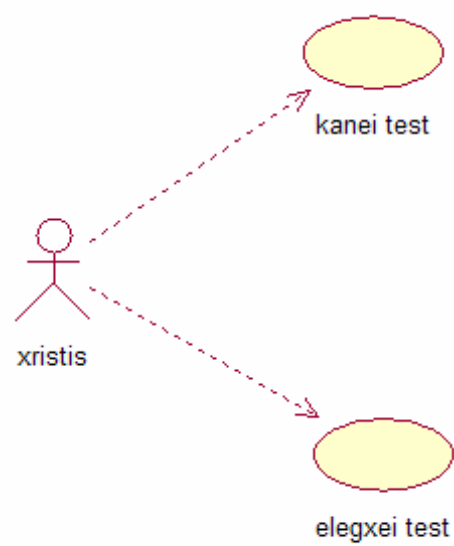
3.2.1.1. Διαγράμματα περιπτώσεων χρήσης

Ακολουθούν δύο διαγράμματα περιπτώσεων χρήσης

Διάγραμμα περιπτώσεων χρήσης administrator



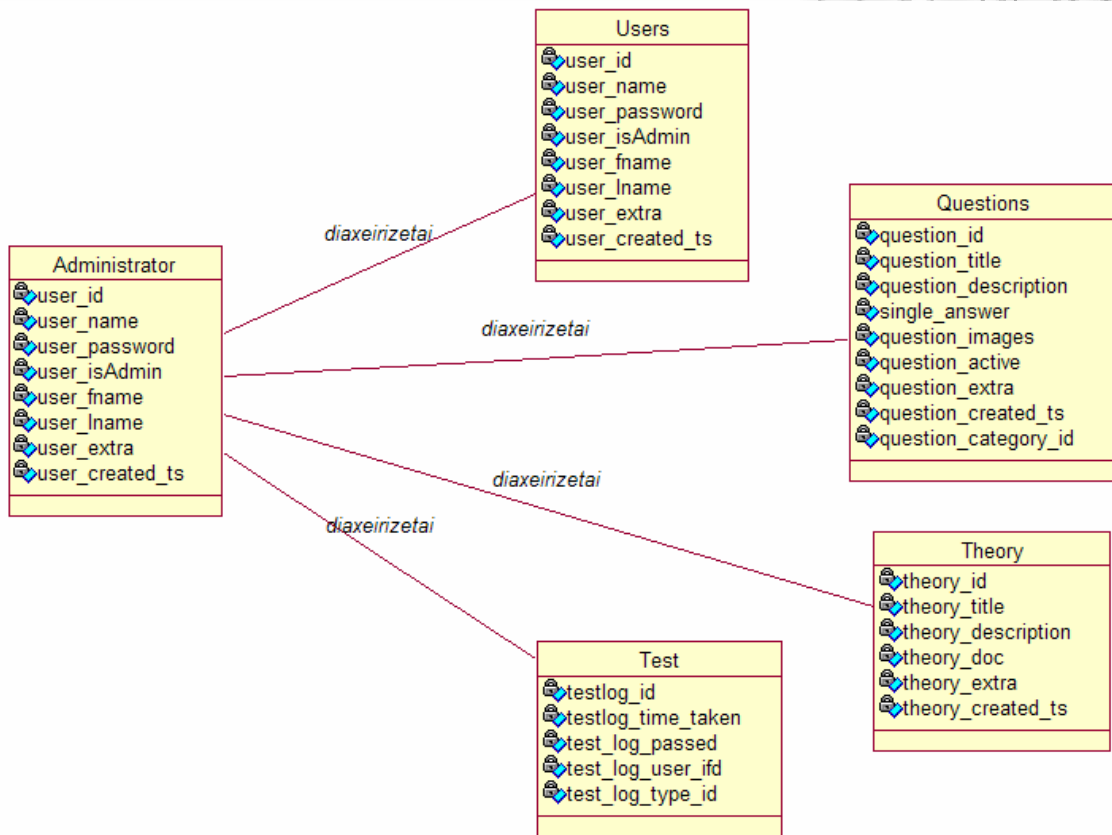
Διάγραμμα περιπτώσεων χρήσης απλού χρήστη



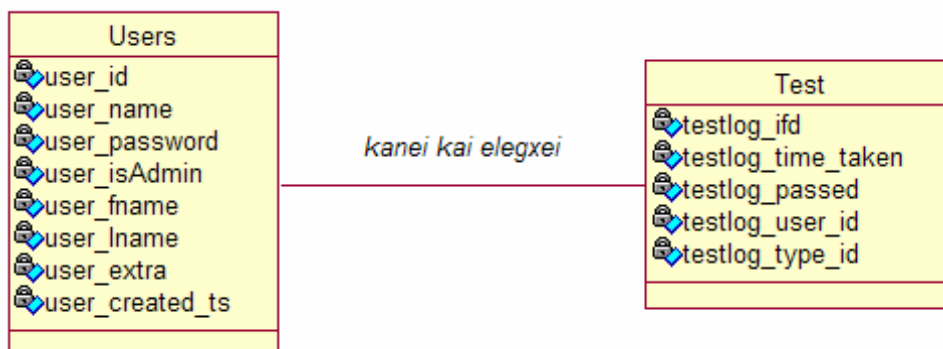
3.2.1.2. Διαγράμματα τάξεων

Ακολουθούν δύο διαγράμματα περιπτώσεων χρήσης

Διάγραμμα τάξεων administrator



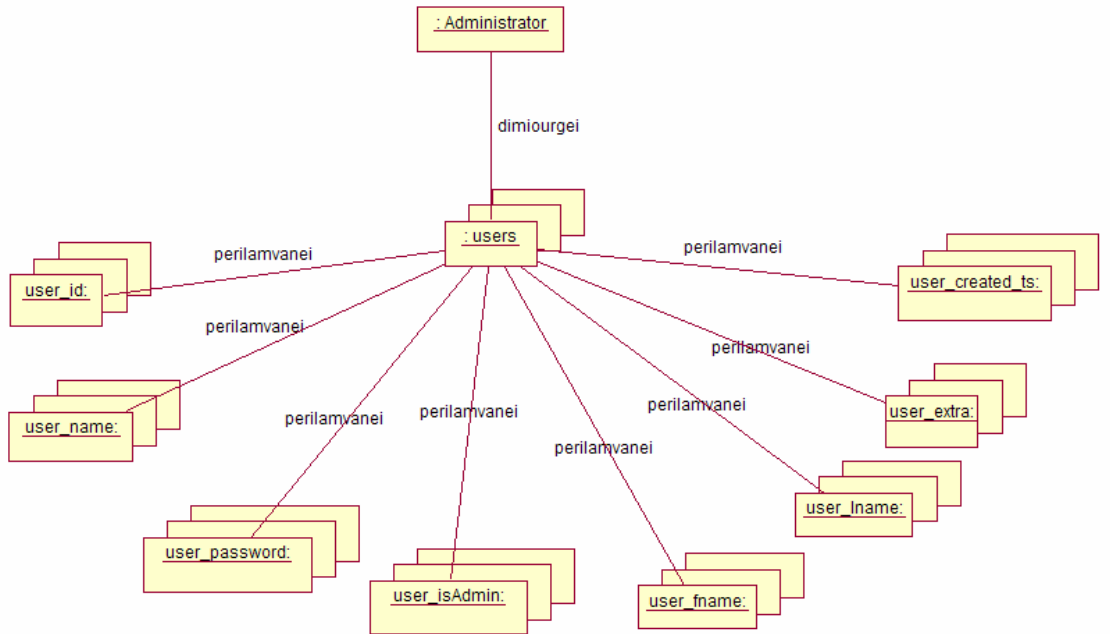
Διάγραμμα τάξεων απλού χρήστη



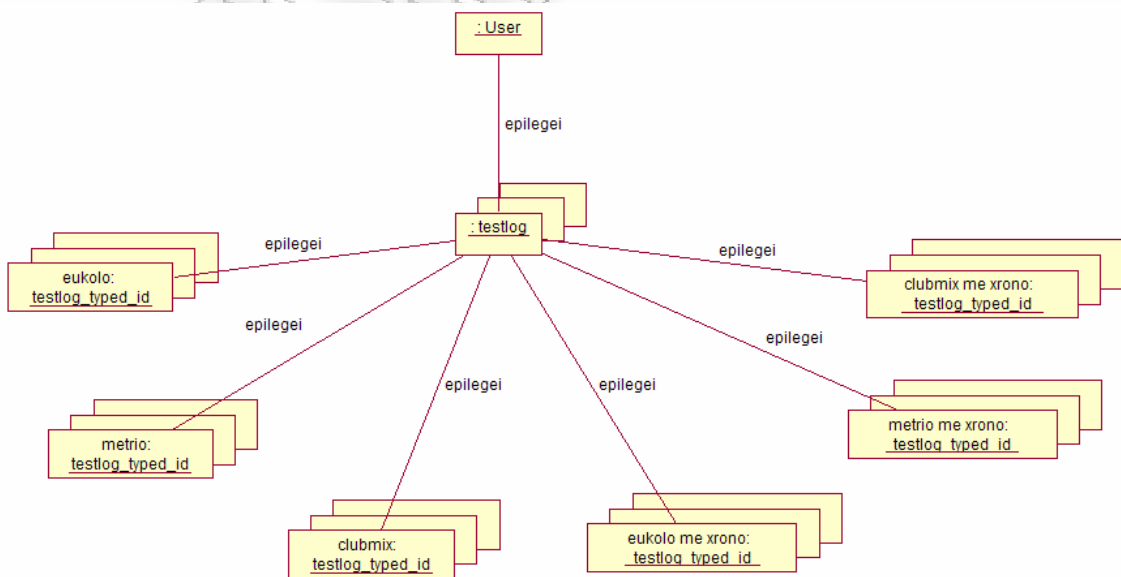
3.2.1.3. Διαγράμματα αντικειμένων

Ακολουθούν δύο διαγράμματα αντικειμένων

Διάγραμμα αντικειμένων administrator



Διάγραμμα αντικειμένων επιλογής τεστ



3.2.1.4. Διαγράμματα συνεργασίας

Τα διαγράμματα συνεργασίας μαζί με τα διαγράμματα σειράς μπορούν να ομαδοποιηθούν κάτω από τον τίτλο «διαγράμματα αλληλεπίδρασης». Το διάγραμμα συνεργασίας ή επικοινωνίας όπως αλλιώς ονομάζεται, δείχνει τις μεθόδους που θα υλοποιεί κάθε κλάση και ποια άλλη κλάση θα τις καλεί. Στην ουσία το διάγραμμα συνεργασίας έχει σαν σκοπό να μοντελοποιήσει:

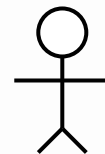
- Τη συνεργασία ανάμεσα στα αντικείμενα, η οποία και ορίζει τους ρόλους που παίζουν τα αντικείμενα όταν συμμετέχουν σε μια περίπτωση χρήσης.
- Την αλληλεπίδραση ανάμεσα σε αντικείμενα η οποία και ορίζει το τρόπο και το μοτίβο της επικοινωνίας (ανταλλαγής μηνυμάτων) ανάμεσα σε αντικείμενα που παίζουν κάποιο ρόλο σε μια συνεργασία.

Τα διαγράμματα συνεργασίας τα χρησιμοποιούμε:

- Σαν εργαλείο για να βρούμε τις κλάσεις του συστήματος.
- Για να βρούμε τη σχέση ανάμεσα σε περιπτώσεις χρήσης και τις απαραίτητες κλάσεις.
- Για να απεικονίσουμε τις προδιαγραφές ενός συστήματος στην υλοποίησή του.

Σημειολογικά τα διαγράμματα συνεργασίας:

Ενεργοποιός



Αντικείμενο → μια οντότητα με μοναδικά ορισμένη υπόσταση η οποία μπορεί να λαμβάνει μηνύματα και να βρίσκεται σε διαφορετικές καταστάσεις.

Όνομα

Σύνδεσμος → σύνδεση ανάμεσα σε υλοποιήσεις

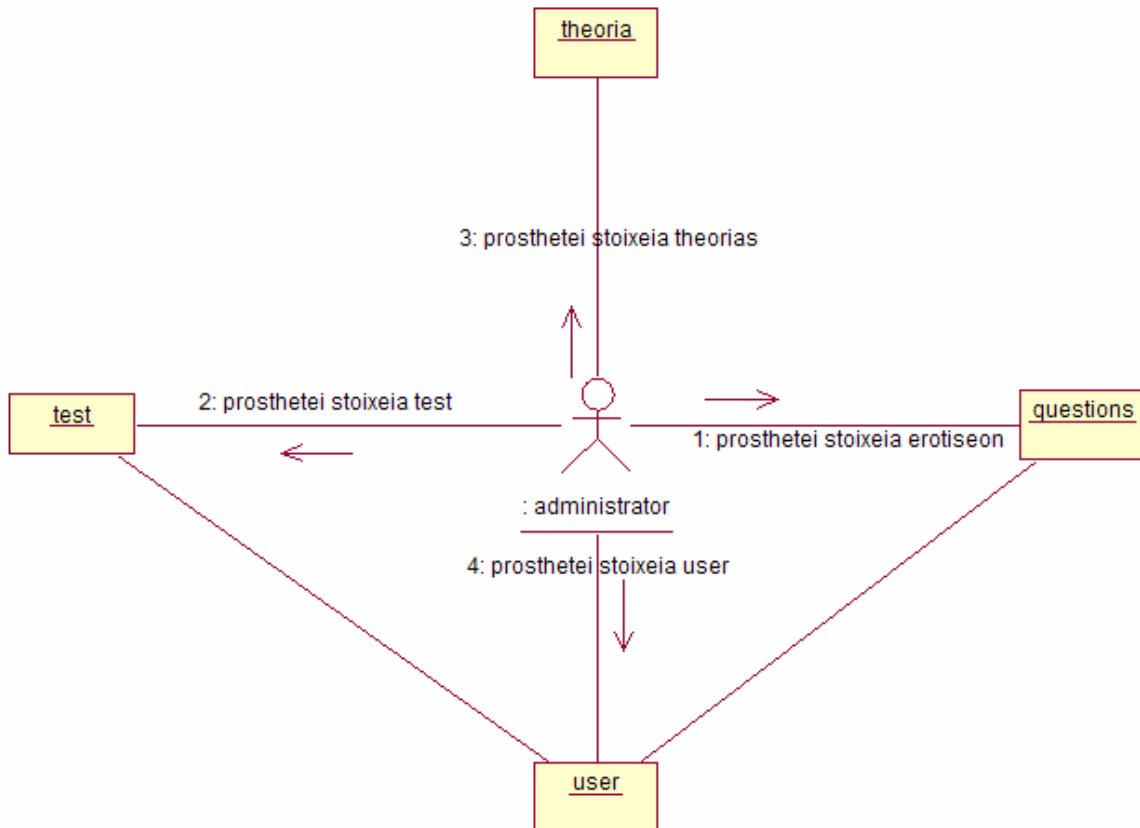


Μήνυμα → ορίζει την επικοινωνία ανάμεσα σε δυο υλοποιήσεις

Μήνυμα



Διάγραμμα συνεργασίας administrator



3.2.1.5. Διαγράμματα σειράς

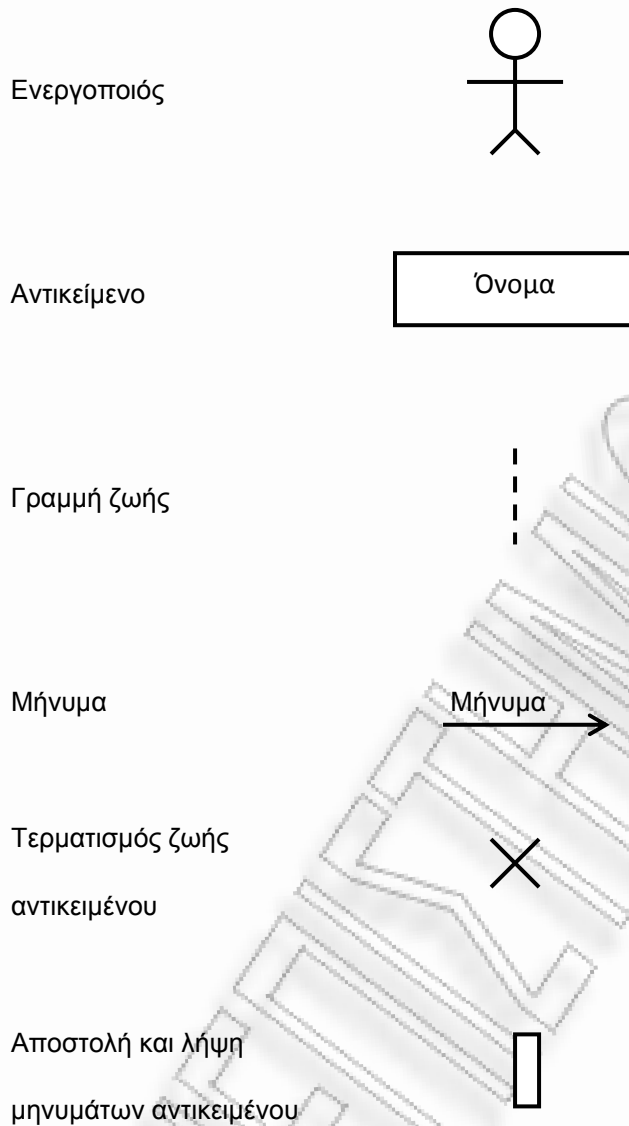
Όπως είπαμε και στα διαγράμματα συνεργασίας, τα διαγράμματα σειράς ανήκουν στα διαγράμματα αλληλεπίδρασης.

Η διαφορά των διαγραμμάτων σειράς από τα διαγράμματα συνεργασίας είναι ότι τα πρώτα εστιάζουν στη χρονική ακολουθία των μηνυμάτων, ενώ τα δεύτερα εστιάζουν στις σχέσεις μεταξύ των αντικειμένων που ανταλλάσσουν μηνύματα.

Το διάγραμμα σειράς, ή ακολουθίας όπως αλλιώς ονομάζεται, αναπαριστά τη χρονική σειρά εκτέλεσης των γεγονότων που λαμβάνουν χώρα κατά τη διάρκεια της αλληλεπίδρασης και έχει δυο διαστάσεις:

- Την κάθετη διάσταση που αναπαριστά το χρόνο.
- Την οριζόντια διάσταση που αναπαριστά τα διάφορα αντικείμενα.

Σημειολογικά τα διαγράμματα σειράς:

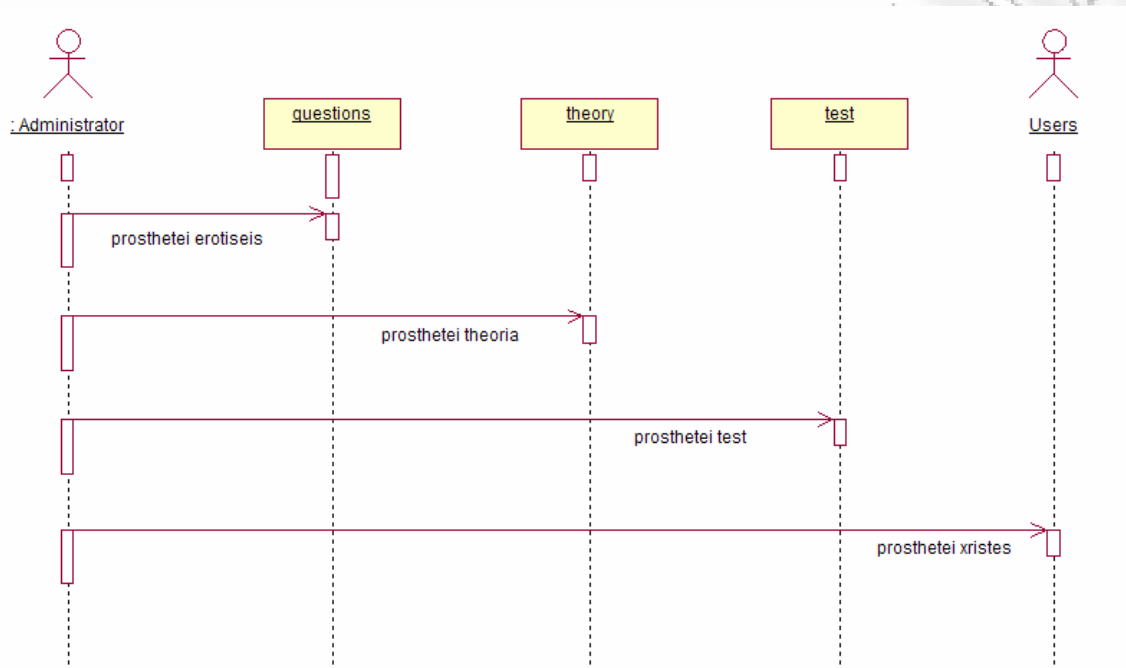


Οι ενεργοποιοί και τα αντικείμενα τοποθετούνται πάνω – πάνω στο διάγραμμα. Κάτω από τον κάθε ενεργοποιοί τοποθετούμε τη γραμμή ζωής. Υπάρχουν περιπτώσεις που ένα αντικείμενο δημιουργεί ένα προσωρινό αντικείμενο και τότε τοποθετούμε το X στο τέλος της «γραμμή ζωής» προκειμένου να δείξουμε ότι το αντικείμενο έχει καταστραφεί. Τέλος, το λεπτό ορθογώνιο κουτί το τοποθετούμε σε κάποια σημεία στη γραμμή ζωής για να δείξουμε πότε οι κλάσεις στέλνουν και δέχονται μηνύματα.

Βάση αυτών που αναφέραμε για τα διαγράμματα σειράς, δημιουργήσαμε το παρακάτω διάγραμμα.

Ακολουθεί το διάγραμμα σειράς του administrator.

Διάγραμμα σειράς administrator



3.2.1.6. Διαγράμματα καταστάσεων

Τα διαγράμματα αυτά χρησιμοποιούνται για μια κλάση, προκειμένου να μοντελοποιήσουν τη δυναμική συμπεριφορά των αντικειμένων μιας κλάσης και τον τρόπο που μεταβάλλεται η κατάστασή τους ως ανάδραση σε συμβάντα.

Το διάγραμμα καταστάσεων απεικονίζει το σύστημα σαν μια μηχανή πεπερασμένων καταστάσεων και περιλαμβάνει:

- Διάφορες καταστάσεις στις οποίες μπορεί να βρεθεί το αντικείμενο κατά τη λειτουργία.
- Τις μεταβάσεις από τη μια κατάσταση στην άλλη ανταποκρινόμενο σε εσωτερικά ή εξωτερικά συμβάντα/γεγονότα.
- Δραστηριότητες που ενδεχομένως αντιδρούν σε διάφορα συμβάντα.

Με τον όρο κατάσταση εννοούμε τη συνθήκη στην οποία βρίσκεται ένα αντικείμενο κατά τη διάρκεια της ζωής του. Ένα αντικείμενο βρίσκεται σε μια κατάσταση για κάποιο χρονικό διάστημα. Στο διάστημα αυτό αποκρίνεται σε γεγονότα που βρίσκονται στις μεταβάσεις που ξεκινούν από την κατάσταση αυτή.

Στην ουσία λοιπόν ένα διάγραμμα καταστάσεων παρουσιάζει όλες τις πιθανές καταστάσεις που μπορεί να έχει ένα συγκεκριμένο αντικείμενο (σε όλη τη διάρκεια ύπαρξής του) και πως αυτές αλλάζουν ανάλογα με τα γεγονότα που φθάνουν στο αντικείμενο.

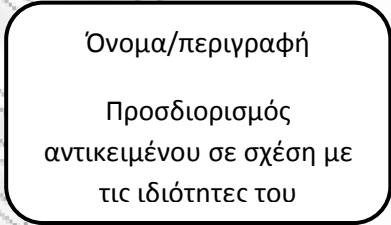
Από αυτά καταλαβαίνουμε ότι δε περιοριζόμαστε σε μια περίπτωση χρήσης. Στην ουσία το διάγραμμα καταστάσεων αφορά μια κλάση και συγκεκριμένα όχι όλες τις κλάσεις αλλά μόνο αυτές που παρουσιάζουν μια έντονα δυναμική συμπεριφορά.

Για να δημιουργήσουμε ένα διάγραμμα καταστάσεων, ακολουθούμε τα παρακάτω βήματα:

- 1) Επιλέγουμε το αντικείμενο
- 2) Εντοπίζουμε τις διακριτές καταστάσεις που μπορεί να μπει αυτό το αντικείμενο.
- 3) Εντοπίζουμε συμβάντα που προκαλούν μετάβαση ανάμεσα στις καταστάσεις.

Σημειολογικά τα διαγράμματα καταστάσεων:

Κατάσταση αντικειμένου:



Μετάβαση από τη μια κατάσταση στην άλλη

συμβάν/γεγονός που πυροδοτεί τη μετάβαση



Αρχική κατάσταση

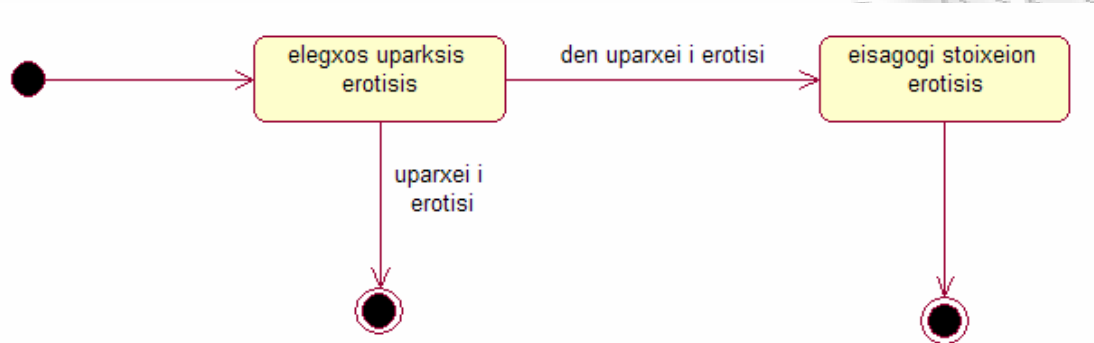


Τελική κατάσταση

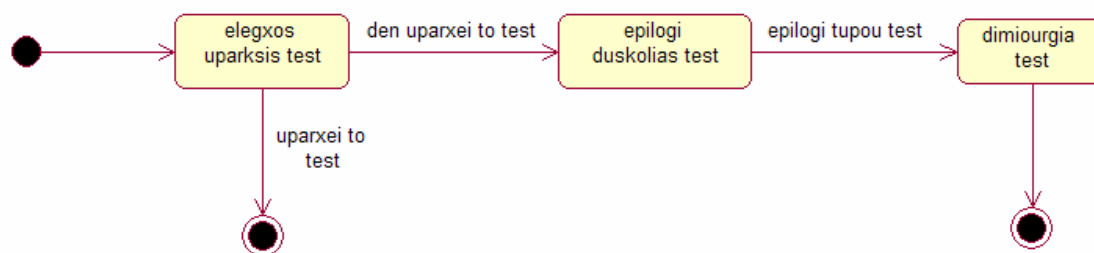


Ακολουθούν δύο διαγράμματα καταστάσεων:

Διάγραμμα καταστάσεων εισαγωγής ερωτήσεων



Διάγραμμα καταστάσεων εισαγωγής τεστ



3.2.1.7. Διαγράμματα δραστηριοτήτων

Τα διαγράμματα δραστηριοτήτων είναι μια παραλλαγή του διαγράμματος καταστάσεων στο οποίο οι περισσότερες από τις μεταβάσεις προκαλούνται από την ολοκλήρωση των δράσεων ή των υποδραστηριοτήτων των πηγαίων καταστάσεων. Το διάγραμμα δραστηριοτήτων προσκολλάται σε μια κλάση, όπως μια περίπτωση χρήσης, ή σε ένα πακέτο ή στην υλοποίηση μιας πράξης.

Το διάγραμμα αυτό έχει σαν σκοπό να εστιάσει σε ροές που προκαλούνται από την εσωτερική επεξεργασία (σε αντιδιαστολή με τα εξωτερικά συμβάντα). Τα χρησιμοποιούμε σε περιπτώσεις όπου όλα ή τα περισσότερα από τα συμβάντα αναπαριστούν την ολοκλήρωση εσωτερικά παραγόμενων δράσεων.

Σημειολογικά τα διαγράμματα δραστηριοτήτων:

Αρχική δραστηριότητα



Δραστηριότητα

«δομή»
Δραστηριότητα 1

Αποφάσεις → σημείο απόφασης που οδηγεί σε δυο
δρόμους ανάλογα με την απόφαση



Ενώσεις → σημείο ένωσης χρησιμοποιείται συνήθως
μετά από ένα σημείο απόφασης



Διακλάδωση/ένωση

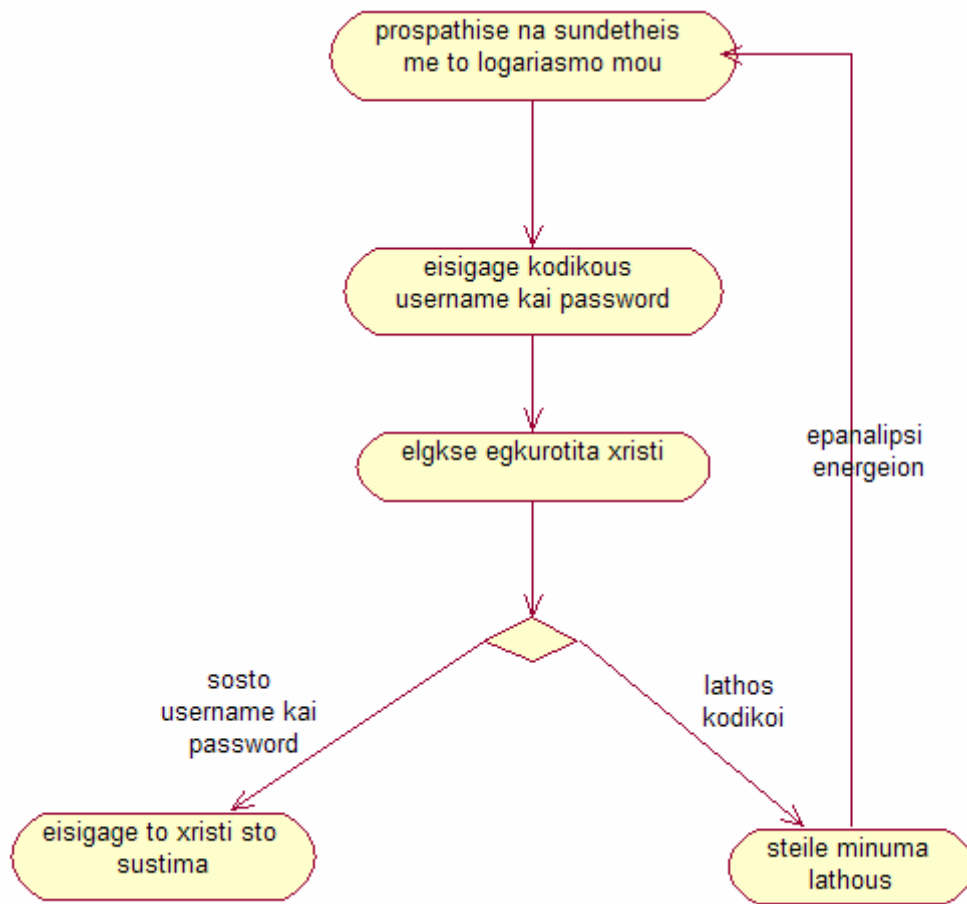


Τελική δραστηριότητα

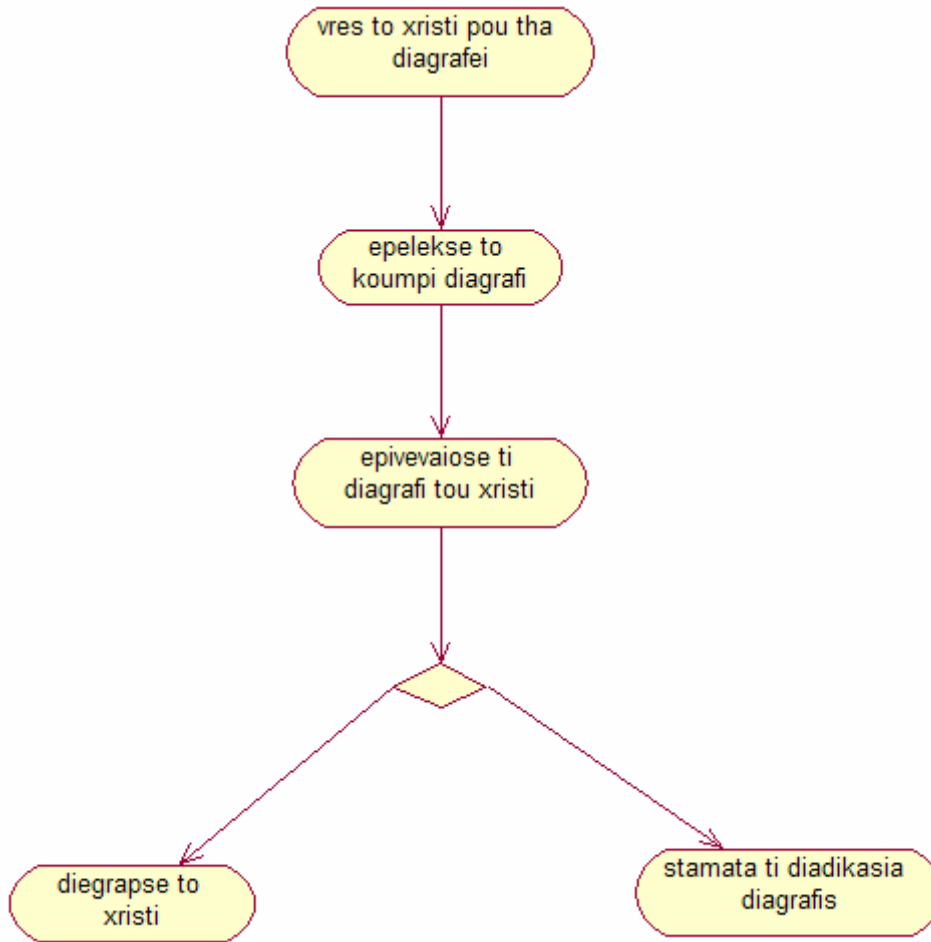


Ακολουθούν τρία διαγράμματα δραστηριοτήτων.

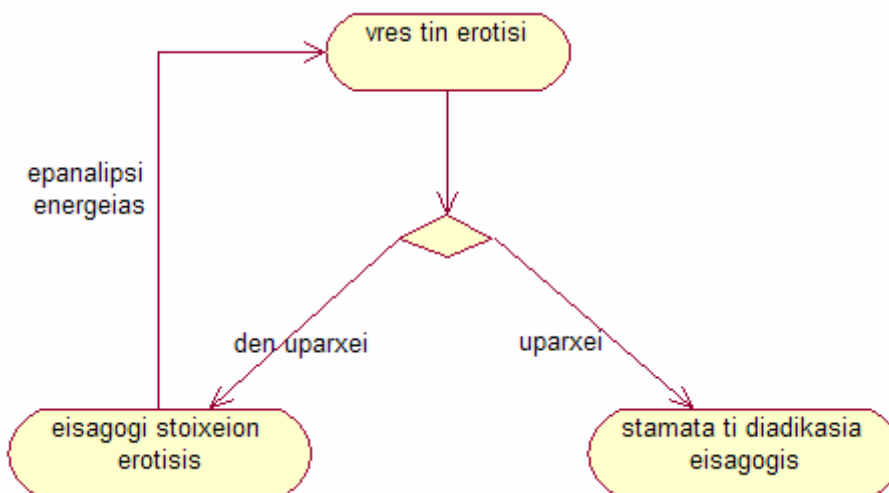
Διάγραμμα δραστηριοτήτων για τη σύνδεση χρήστη με το λογαριασμό του



Διάγραμμα δραστηριοτήτων για τη διαγραφή χρήστη



Διάγραμμα δραστηριοτήτων για την εισαγωγή ερώτησης



3.2.1.8. Διαγράμματα εξαρτημάτων

Τα διαγράμματα εξαρτημάτων ή συστατικών όπως αλλιώς ονομάζονται, περιέχουν συστατικά και εξαρτήσεις. Τα συστατικά αναπαριστούν το φυσικό πακετάρισμα μιας ομάδας κώδικα και οι εξαρτήσεις είναι οι σχέσεις μεταξύ των συστατικών και δείχνουν πως οι αλλαγές που πραγματοποιούνται σε ένα συστατικό μπορούν να επηρεάσουν τα άλλα συστατικά.

Τα διαγράμματα εξαρτημάτων αναπαριστούν επίσης τα συστήματα διεπαφής που χρησιμοποιούνται από τα συστατικά για να επικοινωνήσουν μεταξύ τους. Ένα συστατικό είναι μια φυσική μονάδα υλοποίησης κώδικα.

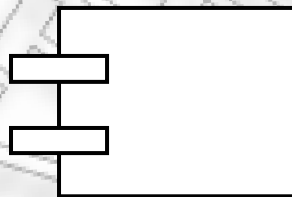
Η κυριότερη διαφορά τους με τα υπόλοιπα διαγράμματα της UML είναι ότι τα διαγράμματα αυτά αναπαριστούν την όψη υλοποίησης του συστήματος.

Οι σημαντικότερες ιδιότητες των συστατικών είναι οι παρακάτω:

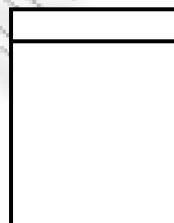
- Είναι επαναχρησιμοποιήσιμα και μπορούν να αντικατασταθούν
- Πρέπει να παρέχουν μια διεπαφή για να επιτρέπουν σε άλλα συστατικά να αλληλεπιδρούν με αυτά και να χρησιμοποιούν τις υπηρεσίες τους.
- Αποτελούνται ή υλοποιούνται από Artifacts τα οποία μπορεί να είναι εκτελέσιμα.

Σημειολογικά τα διαγράμματα εξαρτημάτων:

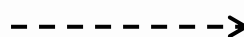
Εξάρτημα



Υποπρόγραμμα



Σχέση εξάρτησης

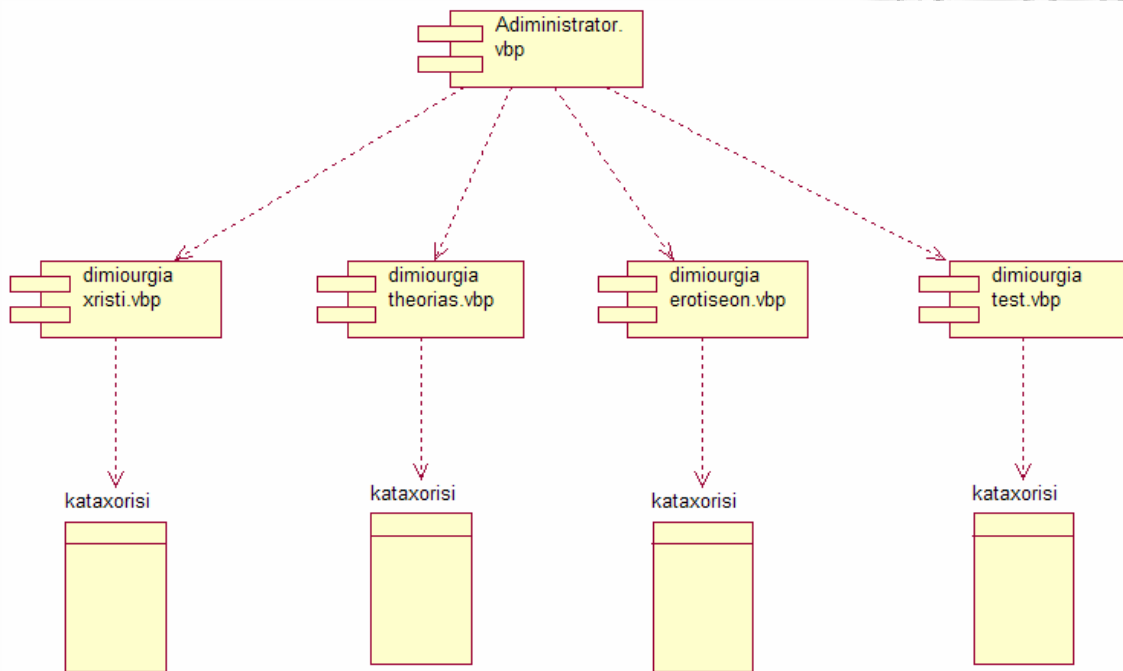


Interface



Ακολουθεί το παρακάτω διάγραμμα εξαρτημάτων:

Διάγραμμα εξαρτημάτων Administrator



3.2.2. ΥΛΟΠΟΙΗΣΗ - ΈΛΕΓΧΟΣ

Στη φάση αυτή δημιουργήσαμε όλα τα διαγράμματα της UML, επομένως μπορούμε να προχωρήσουμε και στον έλεγχο της. Προκειμένου να το πράξουμε αυτό θα αξιολογήσουμε κάποια βασικά κριτήρια ελέγχου της συγκεκριμένης φάσης. Τα κριτήρια αυτά είναι τα παρακάτω:

- Έλεγχος του βαθμού σταθερότητας και προοπτικής του προϊόντος.
- Έλεγχος της σταθερότητας της αρχιτεκτονικής.
- Έλεγχος της αναγνώρισης και αντιμετώπισης των κύριων στοιχείων επικινδυνότητας μέσω μιας επίδειξης του εκτελέσιμου.
- Έλεγχος τεκμηρίωσης και ακρίβειας του σχεδίου οργάνωσης.
- Έλεγχος συμφωνίας όλων των μεριδιούχων για το ότι το όραμα μπορεί να επιτευχθεί αν το παρόν σχέδιο οργάνωσης του έργου εκτελεστεί ώστε να υλοποιηθεί το πλήρες σύστημα μέσα στα πλαίσια της παρούσας αρχιτεκτονικής.
- Έλεγχος συμφωνίας πραγματικών εξόδων με τα αρχικός εκτιμώμενα.
- Ακύρωση ή επανεξέταση έργου σε περίπτωση αποτυχίας ικανοποίησης ορόσημου

Εφόσον αξιολογήσαμε τα κριτήρια αυτά καταλήξαμε στο ότι ικανοποιήθηκαν πλήρως και έτσι μπορούμε να προχωρήσουμε στην επόμενη φάση.

Κ Ε Φ Α Λ Α Ι Ο 4

4. ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ

Στο κεφάλαιο αυτό δημιουργήσαμε τη βάση δεδομένων καθώς και την εφαρμογή μας. Για την υλοποίηση της εφαρμογής, ως γλώσσα προγραμματισμού χρησιμοποιήθηκε η Visual Basic 6.0. ενώ για τη βάση δεδομένων η MsAccess.

Η επιλογή της Visual Basic έγινε με βάση το γεγονός ότι η Visual Basic προσφέρει ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών για Windows, καθώς ως γλώσσα υψηλού επιπέδου εκμεταλλεύεται όλες τις δυνατότητες που προσφέρουν τα Windows. Η επιλογή της MsAccess βασίστηκε στο ότι πρόκειται για ένα εύχρηστο, απλό και λειτουργικό πρόγραμμα το οποίο μπορεί να δημιουργήσει μια ολοκληρωμένη εφαρμογή χωρίς τις περισσότερες φορές να χρειαστεί ο χρήστης να γράψει ούτε μια γραμμή κώδικα προγράμματος, ενώ παράλληλα οι απαιτήσεις της MsAccess σε CPU και μνήμη είναι ελάχιστες.

Συγκεκριμένα, δημιουργήσαμε τη βάση δεδομένων η οποία αποτελείται από επτά πίνακες και στους οποίους σε γενικές γραμμές κρατώνται τα στοιχεία για τη δημιουργία των τεστ, τα στοιχεία των χρηστών καθώς και τα στοιχεία που μας δίνουν τη πρόοδό τους. Έπειτα προχωρήσαμε στον εννοιολογικό σχεδιασμό της βάσης και στη συνέχεια δημιουργήσαμε το σχεσιακό σχήμα της προκειμένου να φανούν οι σχέσεις μεταξύ των πινάκων.

Εφόσον ολοκληρώσαμε τις εργασίες στη βάση δεδομένων προχωρήσαμε στη δημιουργία της εφαρμογής. Σαν κανόνα για τη δημιουργία της είχαμε το ότι έπρεπε να δημιουργήσουμε μια εφαρμογή με το μικρότερο δυνατό κόστος, όσον αφορά τον προγραμματισμό αλλά και το καλύτερο δυνατό αποτέλεσμα όσον αφορά τη λειτουργία της.

Η χρήση των modules όπως και σε κάθε σχεδόν πρόγραμμα έτσι κι εδώ κρίθηκε απαραίτητη. Δημιουργήσαμε λοιπόν modules που αφορούν τα μηνύματα που εμφανίζονται στην εφαρμογή, το αρχείο help καθώς και τις πληροφορίες των χρηστών, τη σύνδεση της εφαρμογής με τη βάση δεδομένων αλλά και modules για τη σωστή λειτουργία της εφαρμογής.

4.1. ΦΑΣΗ «ΚΑΤΑΣΚΕΥΗ»

4.1.1. ΑΝΑΛΥΣΗ - ΣΧΕΔΙΑΣΜΟΣ

Εφόσον συλλέξαμε τις απαραίτητες πληροφορίες και δημιουργήσαμε τα κατάλληλα διαγράμματα UML στο προηγούμενο κεφάλαιο, προχωράμε στη δημιουργία της Βάσης Δεδομένων η οποία θα αποτελείται από τους πίνακες: users, questions, answers, theory, types, testlogs και testlog_entries.

4.1.1.1. Βάση Δεδομένων

4.1.1.1.1. Πίνακες

1^{ος} πίνακας: Users

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε χρήστη (`user_id`), το όνομά του (`user_name`), ο κωδικός πρόσβασής του στο λογαριασμό του (`user_password`), αν είναι ή όχι ο διαχειριστής του συστήματος (`user_isAdmin`), το μικρό του όνομα (`user_fname`), το επίθετό του (`user_lname`), κάποιες επιπλέον πληροφορίες για το χρήστη (`user_extra`) και τέλος η ημερομηνία δημιουργίας του χρήστη (`user_created_ts`).

Το Primary Key του πίνακα είναι το `user_id`.

2^{ος} πίνακας: Questions

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε ερώτηση (`question_id`), ο αριθμός της ερώτησης (`question_title`), η περιγραφή της (`question_description`), σε ποιο κομμάτι θεωρίας ανήκει αυτή η ερώτηση (`question_category_id`), το αν έχει σωστή απάντηση η ερώτηση (`single_answer`), η εικόνα στην οποία αναφέρεται η ερώτηση (`question_images`), το αν είναι ή όχι ενεργή η ερώτηση δηλαδή το αν μπορεί να χρησιμοποιηθεί σε ένα τεστ (`question_active`), κάποιες επιπλέον πληροφορίες για την ερώτηση (`question_extra`) και η ημερομηνία δημιουργίας της ερώτησης (`question_created_ts`). Εκτός όμως από αυτά τα στοιχεία θα πρέπει να υπάρχει και αυτό το οποίο είναι απαραίτητο για την ένωση αυτού του πίνακα με τον πίνακα "theory". Το στοιχείο αυτό είναι ο μοναδικός κωδικός της θεωρίας (`question_category_id`).

Το Primary Key αυτού του πίνακα είναι το `question_id`.

3^{ος} πίνακας: Answers

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε απάντηση (`answer_id`), η περιγραφή της απάντησης (`answer_description`) και το αν η απάντηση είναι η σωστή ή όχι (`answer_correct`). Εκτός όμως από αυτά τα στοιχεία θα πρέπει να υπάρχει και αυτό το οποίο είναι απαραίτητο για την ένωση αυτού του πίνακα με τον πίνακα "questions". Το στοιχείο αυτό είναι ο μοναδικός κωδικός της ερώτησης (`answer_question_id`).

Το Primary Key αυτού του πίνακα είναι το `answer_id`.

4^{ος} πίνακας: Theory

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε κομμάτι θεωρίας (`theory_id`), ο τίτλος της θεωρίας (`theory_title`), το κείμενο της θεωρίας (`theory_description`), το αρχείο από το οποίο παίρνουμε τη συγκεκριμένη θεωρία (`theory_doc`), κάποιες επιπλέον πληροφορίες για τη θεωρία (`theory_extra`) και τέλος η ημερομηνία δημιουργίας της θεωρίας (`theory_created_ts`).

Το Primary Key αυτού του πίνακα είναι το `theory_id`.

5^{ος} πίνακας: Types

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε τύπο τεστ (types_id), ένα χαρακτηρισμό δυσκολίας του τεστ (types_name), μια περιγραφή του τεστ (types_description), ο αριθμός των ερωτήσεων που θα περιλαμβάνει το τεστ (types_questions_num), ο αριθμός των ερωτήσεων που θα πρέπει να απαντηθούν σωστά προκειμένου ο χρήστης να περάσει το τεστ (types_questions_num_pass), ο αριθμός των εύκολων, των μέτριων αλλά και των δύσκολων ερωτήσεων που θα περιλαμβάνει το τεστ (types_categories_mix), το αν αυτό το τεστ θα είναι ενεργό ή όχι (type_active), το χρονικό όριο ολοκλήρωσης του τεστ (type_time_limit), επιπλέον πληροφορίες για τον τύπο του τεστ (type_extra), και τέλος η ημερομηνία δημιουργίας του τύπου (type_created_ts).

Το Primary Key αυτού του πίνακα είναι το types_id.

6^{ος} πίνακας: Testlogs

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός για κάθε του τεστ (testlog_id), ο χρόνος που διήρκεσε το τεστ (testlog_time_taken), το αν ο χρήστης πέρασε το τεστ (testlog_passed) και το πότε δημιουργήθηκε (testlog_created_ts). Εκτός όμως από αυτά τα στοιχεία θα πρέπει να υπάρχουν και αυτά τα οποία είναι απαραίτητα για την ένωση αυτού του πίνακα με τους πίνακες "users" και "types". Τα στοιχεία αυτά είναι το "testlog_user_id" και "testlog_type_id" αντίστοιχα.

Το Primary Key αυτού του πίνακα είναι το testlog_id.

7^{ος} πίνακας: Testlog entries

Τα στοιχεία που θα καταγράφονται σε αυτό το πίνακα είναι ένας μοναδικός κωδικός που αφορά τη διόρθωση των ερωτήσεων των τεστ (testlog_entry_id) ενώ τα στοιχεία που περιλαμβάνει αφορούν την ένωση αυτού του πίνακα με τον πίνακα των τεστ (testlogs) μέσω του πεδίου "testlog_entry_testlog_id", την ένωσή του με τον πίνακα των ερωτήσεων (question) μέσω του πεδίου "testlog_entry_questions_id" και κατ' επέκταση τα πεδία "testlog_entry_answer_id" και "testlog_entry_answer_correct_id" για την ένωσή του με τον πίνακα "answers". Προκειμένου να γίνει πιο κατανοητό, σε αυτόν τον πίνακα φαίνεται αν έχει απαντηθεί μια ερώτηση σε ένα τεστ και αν έχει απαντηθεί αν η απάντηση που δόθηκε είναι η σωστή.

Το Primary Key αυτού του πίνακα είναι το testlog_entry_id.

4.1.1.1.2. Εννοιολογικός σχεδιασμός βάσης

Για να δημιουργήσουμε το εννοιολογικό σχήμα της Β.Δ., θα πρέπει να ακολουθήσουμε κάποια βήματα:

α) Εντοπισμός οντοτήτων

Οι οντότητες που εντοπίζουμε στη Β.Δ. είναι:

- 1) Users
- 2) Questions
- 3) Answers
- 4) Theory
- 5) Types
- 6) Testlogs

β) Εντοπισμός γνωρισμάτων οντοτήτων

- 1) Users → user_id, user_name, user_password, user_isAdmin, user_fname, user_lname, user_extra, user_created_ts.
- 2) Questions → question_id, question_title, question_description, question_theory_id, question_category_id, single_answer, question_images, question_active, question_extra, question_created_ts.
- 3) Answers → answer_id, answer_description, answer_correct, answer_question_id.
- 4) Theory → theory_id, theory_title, theory_description, theory_doc, theory_extra, theory_created_ts.
- 5) Types → types_id, types_name, types_description, types_questions_num, types_questions_num_pass, types_categories_mix, type_active, type_time_limit, type_extra, type_created_ts.
- 6) Testlogs → testlog_id, testlog_user_id, testlog_type_id, testlog_time_taken, testlog_passed, testlog_created_ts.
- 7) Testlog_entries → testlog_entry_id, testlog_entry_testlog_id, testlog_entry_question_id, testlog_entry_answer_id, testlog_entry_answer_correct_id.

γ) Επιλογή και περιγραφή σχέσεων

Οι σχέσεις που εντοπίζουμε είναι:

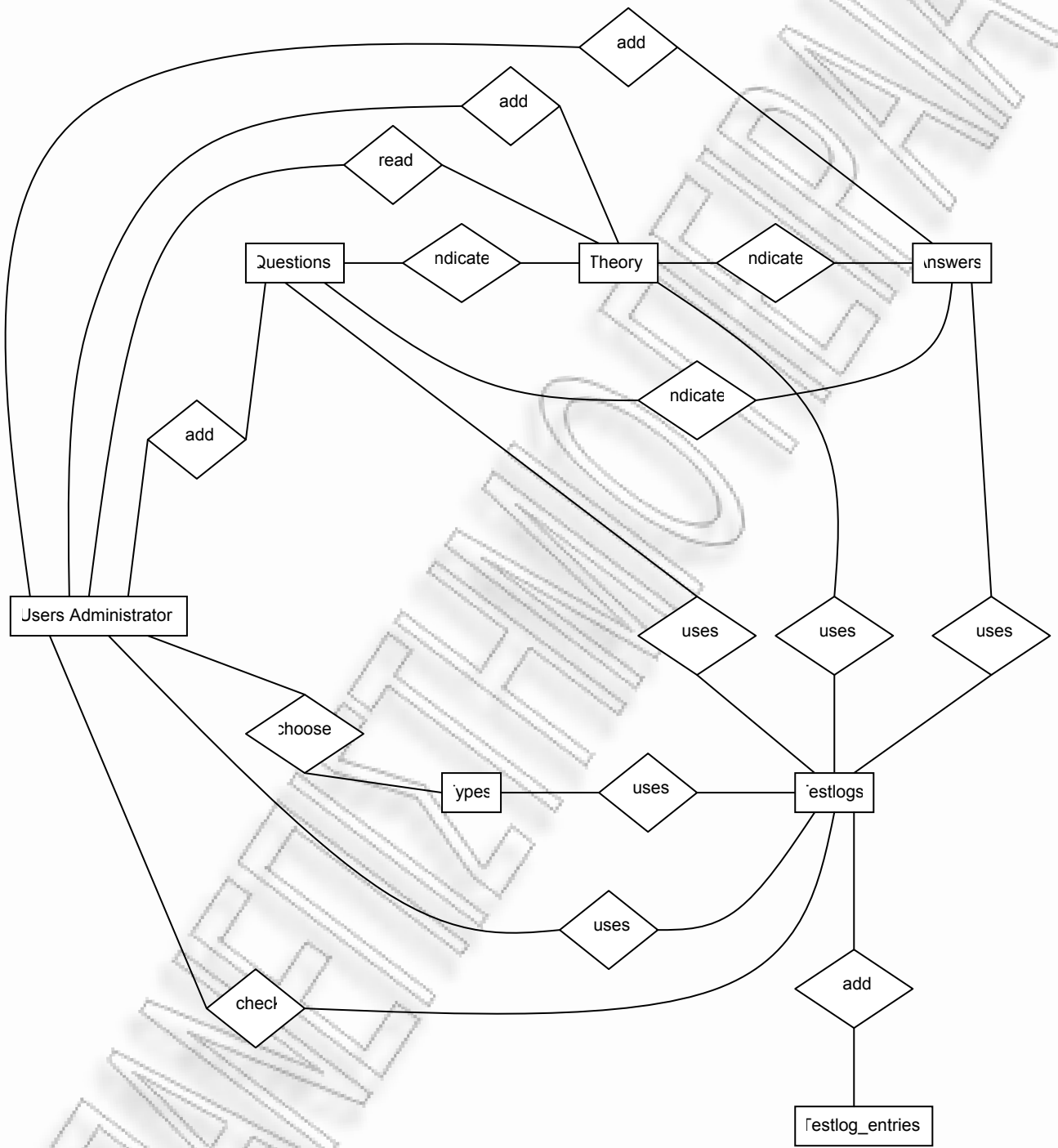
- 1) Αποθηκεύονται οι χρήστες ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Users και Administrator.
- 2) Οι χρήστες χρησιμοποιούν τα τεστ ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Users και Testlog.
- 3) Οι χρήστες επιλέγουν τύπο τεστ ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Users και Types.
- 4) Οι χρήστες διαβάζουν θεωρία ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Users και Theory.
- 5) Οι χρήστες ελέγχουν τα τεστ που έκαναν ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Users και Testlog.
- 6) Αποθηκεύονται οι ερωτήσεις ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Administrator και Questions.
- 7) Οι ερωτήσεις χρησιμοποιούνται στα τεστ ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Questions και Testlog.
- 8) Οι ερωτήσεις αντιστοιχούν σε κομμάτια θεωρίας ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Questions και Theory.
- 9) Αποθηκεύονται οι απαντήσεις ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Administrator και Answers.
- 10) Οι απαντήσεις χρησιμοποιούνται στα τεστ ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Answers και Testlog.
- 11) Οι απαντήσεις αντιστοιχούν σε ερωτήσεις ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Answers και Questions.
- 12) Οι απαντήσεις αντιστοιχούν σε κομμάτια θεωρίας ⇒ Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Answers και Theory.

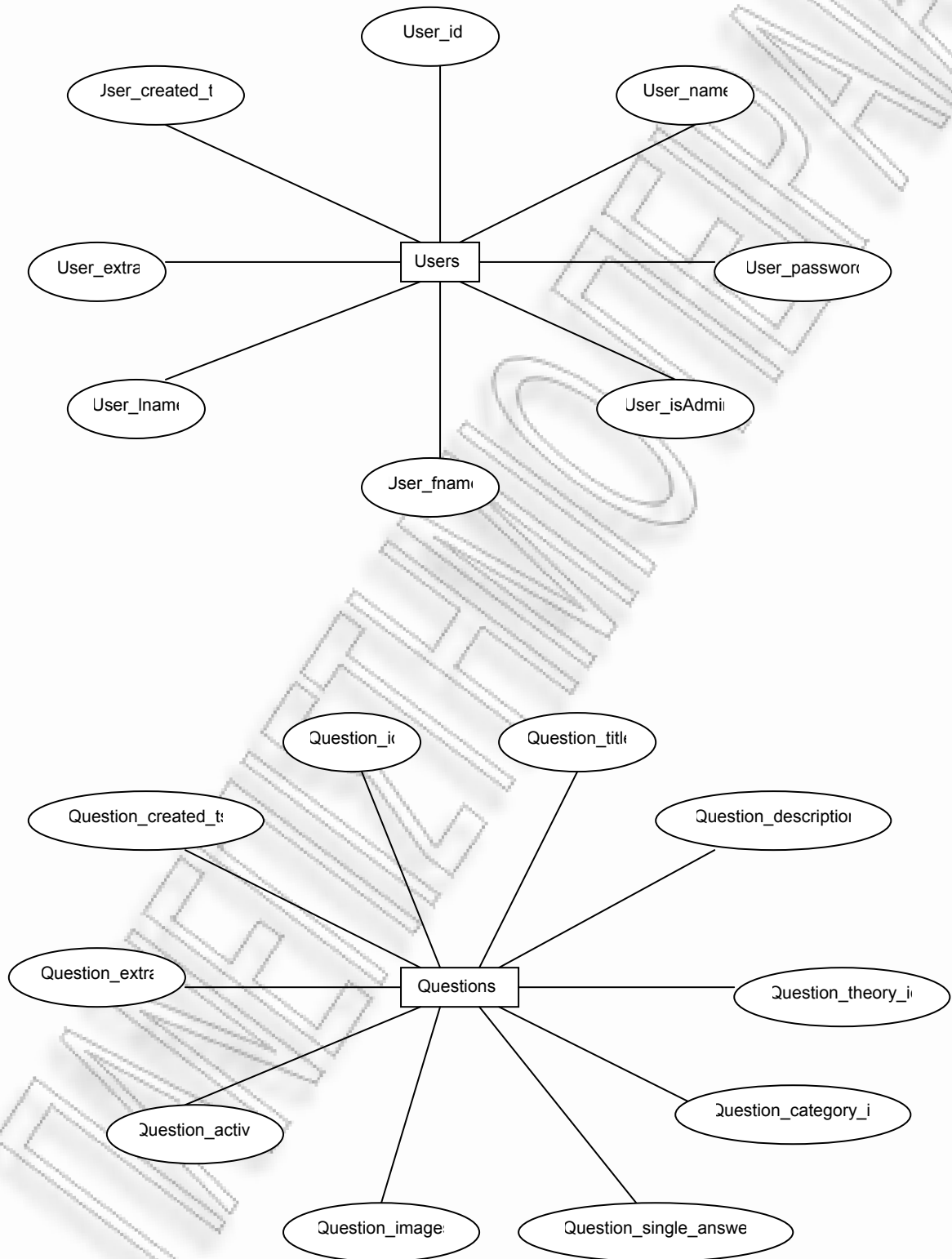
- 13) Αποθηκεύεται η θεωρία \Leftrightarrow Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Administrator και Theory.
- 14) Η θεωρία χρησιμοποιείται στα τεστ \Leftrightarrow Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Theory και Testlog.
- 15) Οι τύποι των τεστ χρησιμοποιούν τις ερωτήσεις \Leftrightarrow Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Types και Questions.
- 16) Οι Administrator χρήστες μπορούν να ελέγχουν όλα τα τεστ που έχουν κάνει οι χρήστες \Leftrightarrow Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Administrator και Testlog.
- 17) Αποθηκεύονται τα τεστ \Leftrightarrow Οι οντότητες που εμπλέκονται σε αυτή τη σχέση είναι: Testlog και Testlog_entries.

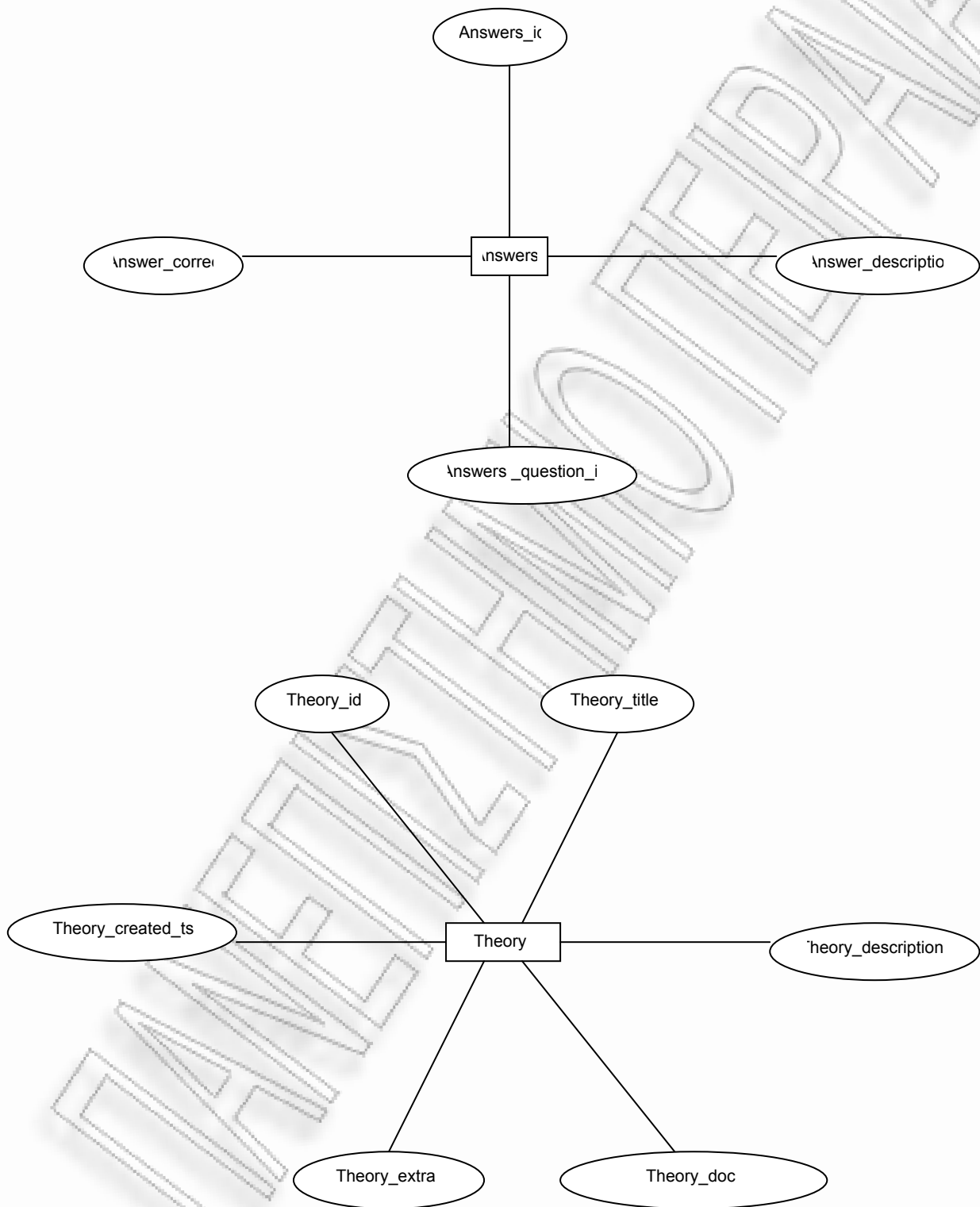
Αντίστοιχα, η περιγραφή των σχέσεων είναι:

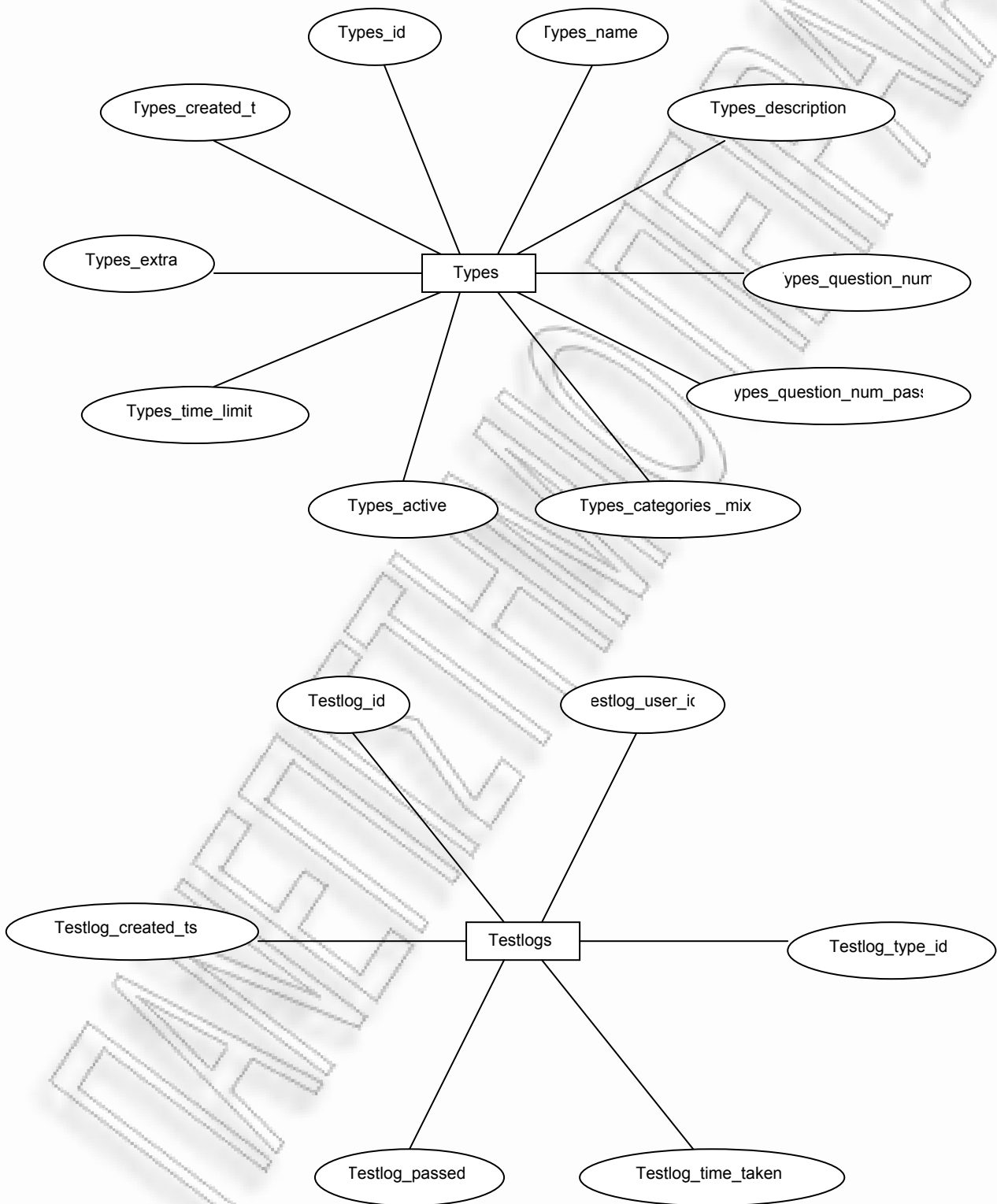
- 1) M προς N. Πολλοί Administrator μπορούν να καταχωρούν πολλούς χρήστες.
- 2) M προς N. Πολλοί χρήστες κάνουν πολλά τεστ.
- 3) M προς N. Πολλοί χρήστες επιλέγουν πολλούς τύπους τεστ.
- 4) M προς N. Πολλοί χρήστες διαβάζουν θεωρία.
- 5) 1 προς N. Ο χρήστης ελέγχει πολλά δικά του τεστ.
- 6) M προς N. Πολλοί Administrator μπορούν να καταχωρήσουν πολλές ερωτήσεις.
- 7) M προς N. Πολλές ερωτήσεις χρησιμοποιούνται σε πολλά τεστ.
- 8) 1 προς N. Ένα κομμάτι θεωρίας μπορεί να αντιστοιχεί σε πολλές ερωτήσεις.
- 9) M προς N. Πολλοί Administrator καταχωρούν πολλές απαντήσεις.
- 10) M προς N. Πολλές απαντήσεις χρησιμοποιούνται σε πολλά τεστ.
- 11) 1 προς N. Μια ερώτηση αντιστοιχεί σε πολλές απαντήσεις.
- 12) M προς N. Πολλές απαντήσεις αντιστοιχούν σε πολλά κομμάτια θεωρίας.
- 13) M προς N. Πολλοί Administrator αποθηκεύουν πολλά κομμάτια θεωρίας.
- 14) M προς N. Πολλά κομμάτια θεωρίας χρησιμοποιούνται σε πολλά τεστ.
- 15) M προς N. Πολλοί τύποι τεστ χρησιμοποιούν πολλές ερωτήσεις.
- 16) M προς N. Πολλοί Administrator χρήστες ελέγχουν πολλά τεστ.
- 17) 1 προς N. Ένα τεστ αντιστοιχεί σε μία αποθήκευσή του.

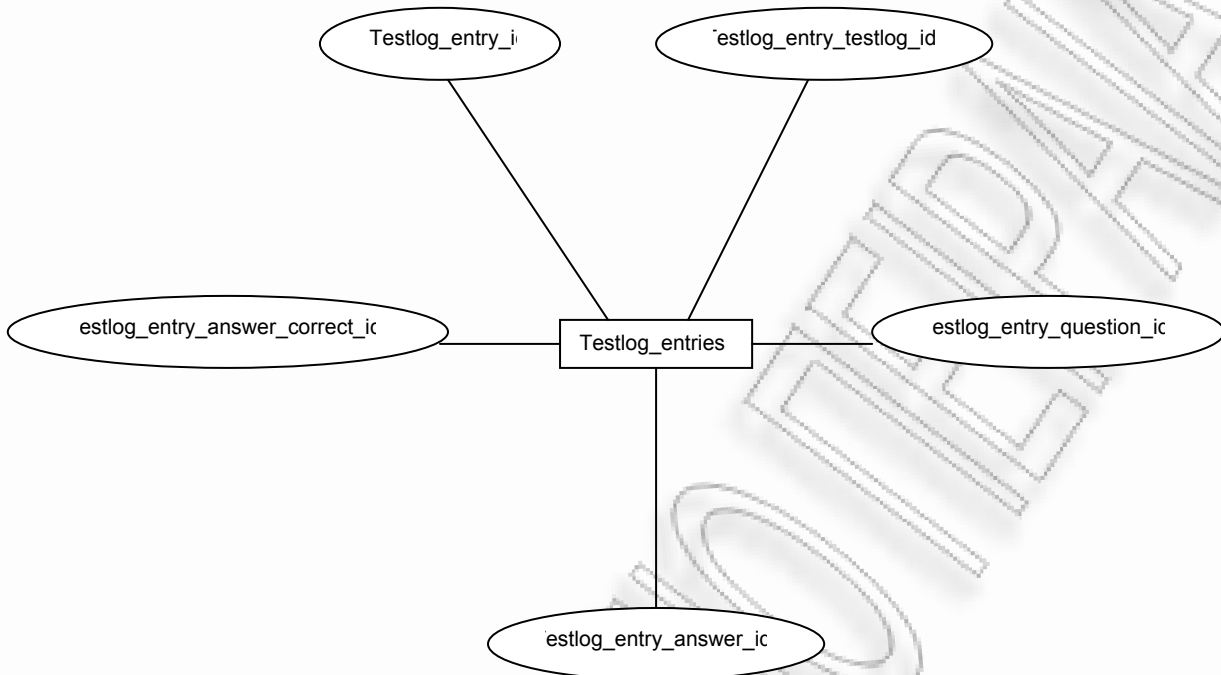
Εφόσον αναγνωρίσαμε τις οντότητες, τα γνωρίσματα και τις σχέσεις μπορούμε να δημιουργήσουμε το εννοιολογικό σχήμα της Βάσης Δεδομένων. Λόγω του ότι τα χαρακτηριστικά των οντοτήτων είναι πολλά, το εννοιολογικό σχήμα θα παρουσιαστεί χωρίς τα χαρακτηριστικά των οντοτήτων του και στη συνέχεια θα ακολουθήσουν χωριστά σχήματα των οντοτήτων με τα χαρακτηριστικά τους.





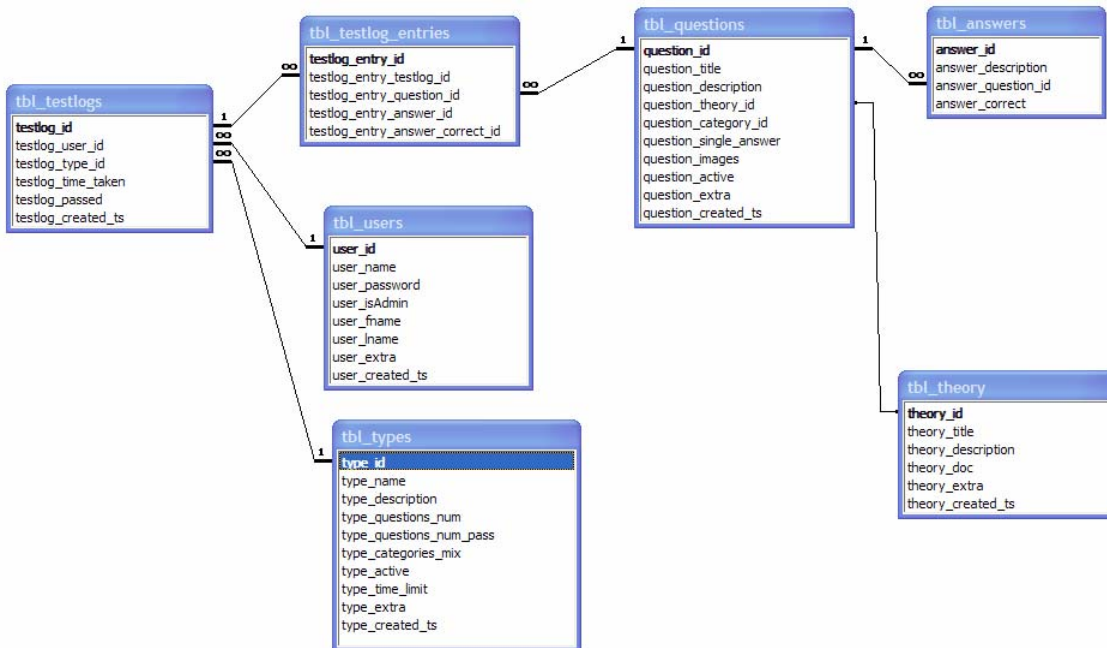






4.1.1.1.3. Σχεσιακό σχήμα βάσης δεδομένων

Με βάση τον εννοιολογικό σχεδιασμό δημιουργούμε και το σχεσιακό σχήμα της Βάσης Δεδομένων.



4.1.1.1.4. Δεδομένα

Πίνακας Users

user_id	user_name	user_password	user_isAdmin	user_fname	user_lname	user_extra	user_created_ts
1	admin	admin	<input checked="" type="checkbox"/>	Διαχειριστής	Συστήματος		2010/02/02
2	user	user	<input type="checkbox"/>	Απλός	Χρήστης		2010/02/02
3	user2	user2	<input type="checkbox"/>	Απλός	Χρήστης 2		2010/03/14

Πίνακας Questions

question_id	question_title	question_description	question_theory_id	question_category_id	question_single_answer	question_images	question_active	question_extra	question_created_ts
1	Ερώτηση 1	Το χειρόφρενο πρέπει ν	4	1	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		2006/05/01
6	Ερώτηση 2	Το χειρόφρενο μπορεί ν	4	1	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		2010/02/12
7	Ερώτηση 3	Η νεκρή γωνία είναι μια	4	1	<input checked="" type="checkbox"/>	02.jpg,	<input checked="" type="checkbox"/>		2010/02/12
8	Ερώτηση 4	Ποια είναι η σημασία τι	5	1	<input checked="" type="checkbox"/>	π61.bmp,	<input checked="" type="checkbox"/>		2010/02/13

Πίνακας Answers

answer_id	answer_description	answer_question_id	answer_correct
5	Ναι	5	<input checked="" type="checkbox"/>
7	Όχι	5	<input type="checkbox"/>
8	Ναι	6	<input checked="" type="checkbox"/>
9	Όχι	6	<input type="checkbox"/>
10	Μη ορατή από τη θέση του οδηγού μέσω των καθρεπτών.	7	<input checked="" type="checkbox"/>

Πίνακας Theory

theory_id	theory_title	theory_description	theory_doc	theory_extra	theory_created_ts
4	Κεφάλαιο 1	Γνωρίστε το αυτοκίνητο	chapter1.chm		2010/04/13
5	Κεφάλαιο 2	Η σήμανση και η σηματοδότηση	chapter2.chm		2010/04/13
6	Κεφάλαιο 3	Οι διασταυρώσεις	chapter3.chm		2010/04/13
7	Κεφάλαιο 4	Η ταχύτητα	chapter4.chm		2010/04/13
8	Κεφάλαιο 5	Η θέση στο οδόστρωμα	chapter5.chm		2010/04/13
9	Κεφάλαιο 6	Οι αποστάσεις ασφαλείας πεδήσεως και ακινητοποιήσεως	chapter6.chm		2010/04/13

Πίνακας Types

type_id	type_name	type_description	type_questions_num	type_questions_num_pass	type_categories_mix	type_active	type_time_limit	type_extra	type_created_ts
1	Εύκολο	Ερωτήσεις για αρχάριους	32	30	32/0/0	<input checked="" type="checkbox"/>	0		2010/02/02
2	Μέτριο	Ερωτήσεις μέτριας δυσκ	32	30	0/32/0	<input checked="" type="checkbox"/>	2		2010/02/02
3	Κλαμπ μιλξ	Όλες οι κατηγορίες από	33	30	11/11/11	<input checked="" type="checkbox"/>	0		2010/02/28
4	Εύκολο με χρόνο	Εύκολες ερωτήσεις με χ	30	28	30/0/0	<input checked="" type="checkbox"/>	15		2010/04/13
5	Μέτριο με χρόνο	Μέτριας δυσκολίας ερω	30	28	0/30/0	<input checked="" type="checkbox"/>	15		2010/04/13

Πίνακας Testlogs

testlog_id	testlog_user_id	testlog_type_id	testlog_time_taken	testlog_passed	testlog_created_ts
1	2	1	-1	<input type="checkbox"/>	2010/03/06
2	2	2	120	<input type="checkbox"/>	2010/03/06
3	2	3	-1	<input checked="" type="checkbox"/>	2010/03/24
4	2	2	20	<input type="checkbox"/>	2010/04/07
5	2	1	-1	<input type="checkbox"/>	2010/04/13

Πίνακας Testlog_entries

testlog_entry_id	testlog_entry_testlog_id	testlog_entry_question_id	testlog_entry_answer_id	testlog_entry_answer_correct_id
1	1	53	138	139
2	1	48	126	125
3	1	35	91	90
4	1	59	154	154

4.1.1.2. Εφαρμογή

Μετά το σχεδιασμό και την ανάπτυξη της Β.Δ. σειρά έχει η υλοποίηση του πηγαίου κώδικα με τη χρήση της V.B. 6.0. Στο σημείο αυτό θα αναφέρουμε αναλυτικά το λόγο δημιουργίας του κάθε module καθώς και το πώς λειτουργεί η κάθε φόρμα.

4.1.1.2.1. *Modules***MsgProperties**

Σκοπός του module αυτού είναι η δημιουργία κάποιων βασικών αντικειμένων των μηνυμάτων του προγράμματος, έτσι ώστε να μη χρειάζεται κάθε φορά να φτιάχνουμε ένα μήνυμα από την αρχή. Στο module αυτό κρατώνται οι επικεφαλίδες των μηνυμάτων, το κείμενο των μηνυμάτων και τα 3 κουμπιά που βρίσκονται στο τέλος της επικεφαλίδας του μηνύματος. Όλες οι μεταβλητές που χρησιμοποιήθηκαν εδώ μπορούν να χρησιμοποιηθούν και από άλλα modules και φόρμες. Τα μηνύματα που υπάρχουν εδώ είναι:

- 1) Μηνύματα λάθους που σαν κουμπί εμπεριέχουν μόνο το «ok».
- 2) Μηνύματα προειδοποίησης που εμπεριέχουν μια ερώτηση, μια προειδοποίηση για το αποτέλεσμα της ενέργειας που πρόκειται να κάνουμε, έπειτα μια ερώτηση για το αν θέλουμε να εκτελέσουμε την ενέργεια αυτή και τα κουμπιά «ok» και «cancel».
- 3) Μηνύματα πληροφόρησης που εμπεριέχουν μια πληροφορία καθώς και το κουμπί «ok».

MsgSubMain

Μέσω αυτού του module τσεκάρουμε αν η εφαρμογή είναι ήδη ανοιχτή. Σε περίπτωση που η εφαρμογή είναι ανοιχτή εμφανίζεται μήνυμα λάθους και τερματίζεται η εφαρμογή, διαφορετικά εμφανίζεται η splash form.

RegistryModule

Στο module αυτό ελέγχεται και ενημερώνεται το registry του λειτουργικού συστήματος του υπολογιστή και κατά συνέπεια σε κάθε έλεγχο αυξάνεται το progress bar που υπάρχει στη splash form.

ApplicationProperties

Στο module αυτό δημιουργήθηκαν οι απαραίτητες ιδιότητες για τη διαμόρφωση της εφαρμογής. Συγκεκριμένα ενημερώνεται το αρχείο «settings.ini» για τη θέση στην οποία είναι αποθηκευμένη η βάση δεδομένων μας, καθώς και ορίζεται ο μέγιστος αριθμός εικόνων που μπορεί να εμπεριέχει μια ερώτηση όπως και ο μέγιστος αριθμός απαντήσεων. Οι μεταβλητές που χρησιμοποιούνται σε αυτό το module μπορούν επίσης να χρησιμοποιηθούν και από όλες τις άλλες φόρμες αλλά και τα modules.

MsgAccessSQL

Στο module αυτό αναπαρίστανται όλα τα SQL Queries μέσω μεταβλητών. Οι μεταβλητές αυτές μπορούν να χρησιμοποιηθούν και από όλα τα άλλα modules αλλά και όλες τις φόρμες.

PublicVariables

Το module αυτό δημιουργήθηκε για να:

- 1) κρατάει όλες τις τιμές για τη διαμόρφωση της εφαρμογής οι οποίες και έχουν οριστεί στο αρχείο «settings.ini»
- 2) κρατάει όλες τις πληροφορίες των χρηστών που αφορούν το όνομα, το επίθετο, τον τύπο και τον κωδικό τους.
- 3) καταχωρήσουμε τον χρήστη ασφαλείας και τον κωδικό πρόσβασής του
- 4) κρατάει το path των εικόνων και των κειμένων που χρησιμοποιούνται και τέλος
- 5) κρατάει το αρχείο help.

MsgBoxModule

Στο module αυτό κρατάμε:

- 1) τα μηνύματα λάθους στα οποία υπάρχει μόνο το κουμπί «ok» το οποίο όταν ο χρήστης το πατήσει, το πρόγραμμα δεν κάνει κάτι. Στην ουσία το πρόγραμμα ενημερώνει το χρήστη για κάποιο λάθος και με το πάτημα του «ok» το πρόγραμμα δεν επιστρέφει κάτι.
- 2) τα μηνύματα πληροφοριών, στα οποία υπάρχει η πληροφορία που δίδεται στο χρήστη και το κουμπί «ok». Όταν ο χρήστης πατήσει «ok» το πρόγραμμα δεν επιστρέφει κάτι.
- 3) τα μηνύματα ειδοποίησης, στα οποία υπάρχει η ερώτηση και τα κουμπιά «ok» και «cancel». Όπως καταλαβαίνουμε σε αυτή την περίπτωση, ανάλογα με το τι θα επιλέξει ο χρήστης το πρόγραμμα θα επιστρέψει ή όχι κάτι.

FileSystemModule

Το module αυτό δημιουργήθηκε προκειμένου να διαχειριζόμαστε τα αρχεία του συστήματος του υπολογιστή. Συγκεκριμένα χρησιμοποιήθηκαν μέθοδοι για να:

- 1) ελέγχουν τη δομή της εφαρμογής, η οποία ορίζεται στο αρχείο «settings.ini»
- 2) αποθηκεύουν τη δομή στο αρχείο του υπολογιστή.
- 3) φορτώνουν τη δομή από το αρχείο.
- 4) επικυρώνουν τη δομή που έχουμε φορτώσει
- 5) διαμορφώνουν νέα περιεχόμενα στη δομή της εφαρμογής.

DBModule

Το module αυτό χρησιμοποιείται για τη σύνδεση της εφαρμογής με τη βάση δεδομένων. Η σύνδεση της βάσης με την εφαρμογή δεν είναι πάντοτε ανοιχτή αλλά ανοίγει κάθε φορά που εμείς είτε θέλουμε να αποθηκεύσουμε κάτι στη βάση είτε να διαγράψουμε είτε να αλλάξουμε είτε απλά να δούμε κάποια δεδομένα.

Συγκεκριμένα τα βήματα που ακολουθούνται κάθε φορά, όποια ενέργεια από τις παραπάνω κι αν θέλουμε να εκτελέσουμε, είναι τα παρακάτω:

- 1) Ανοίγει η σύνδεση με τη βάση
- 2) Εκτελείται το SQL query
- 3) Διαχειρίζεται το recordset το οποίο δημιουργήθηκε από το query
- 4) Κλείνει το recordset
- 5) Κλείνει η σύνδεση

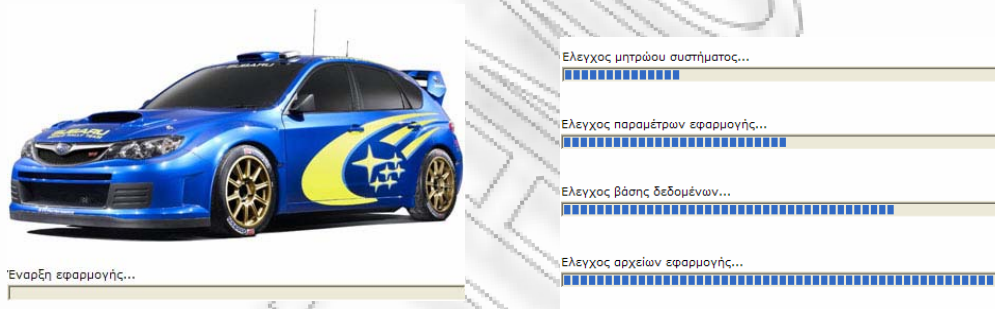
4.1.1.3. Περιγραφή εφαρμογής

FRMSplash

Με το που ανοίγει αυτή η φόρμα, ρυθμίζονται εκ νέου οι τιμές της progress bar. Συγκεκριμένα η αρχική της τιμή είναι το 0, το minimum είναι το 0 και το maximum το 4.

Χρησιμοποιείται επίσης ένας timer, στη διάρκεια του οποίου θα πρέπει να εκτελεστούν έλεγχοι για: 1) Registry 2) Configuration 3) Database 4) File System, όσον αφορά τη σωστή λειτουργία τους.

Προκειμένου να γίνουν αυτοί οι έλεγχοι θα έπρεπε να χρησιμοποιήσουμε αρκετές ένθετες εντολές «if...else», κάτι το οποίο θα μπορούσε να περιπλέξει το πρόγραμμά μας. Η χρήση της εντολής «select case», την οποία ονομάσαμε «checkster», αποδείχτηκε η καταλληλότερη για τον έλεγχο των πολλαπλών συνθηκών που έχουμε. Οι έλεγχοι γίνονται ένας προς έναν. Συγκεκριμένα πρώτα ελέγχεται η σωστή λειτουργία της registry. Αν όλα είναι εντάξει προχωράει στον επόμενο έλεγχο, διαφορετικά εμφανίζεται μήνυμα λάθους και τερματίζει η εφαρμογή. Κάθε φορά που ολοκληρώνεται ένας έλεγχος η progress bar γεμίζει κατά ένα «τμήμα», επομένως όταν τελειώσουν όλοι οι έλεγχοι θα έχουν γεμίσει όλα τα τμήματα της progress bar.

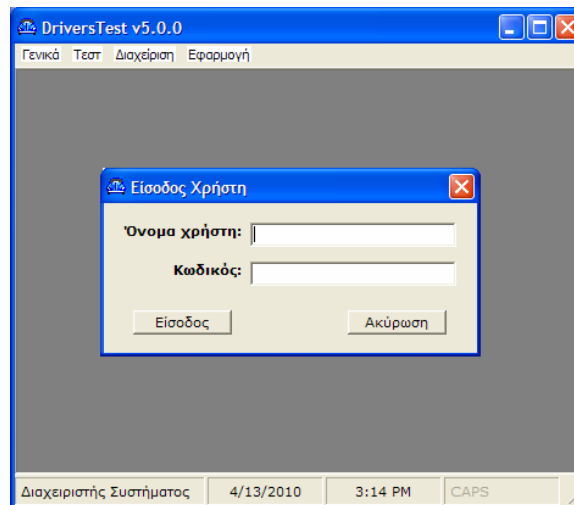


FRMLogin

Με το που ανοίγει η φόρμα εμφανίζεται ο τίτλος του παραθύρου και με το που ενεργοποιείται, τα πεδία «username» και «password» είναι κενά και ο δρομέας βρίσκεται στο «username».

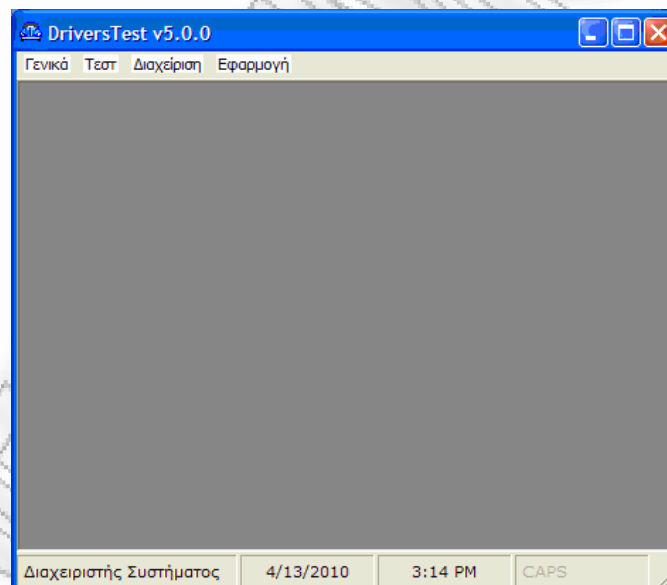
Μόλις ο χρήστης πληκτρολογήσει το «username» και το «password» ελέγχεται αν ο χρήστης είναι ο χρήστης ασφαλείας ο οποίος έχει κωδικό username αλλά και password το «letmein». Αν είναι ο χρήστης ασφαλείας τότε του επιτρέπεται η είσοδος. Αν δεν είναι, το πρόγραμμα ελέγχει αν το username και το password υπάρχουν. Για να γίνει αυτό ανοίγει η βάση δεδομένων και έπειτα ελέγχεται η ύπαρξη του συγκεκριμένου recordset. Εφόσον βρεθεί, ελέγχεται και το αν ο χρήστης είναι ο administrator ή είναι ένας απλός χρήστης και στη συνέχεια κλείνει το recordset και η σύνδεση με τη βάση.

Σε περίπτωση που ο χρήστης δεν υπάρχει εμφανίζεται αντίστοιχο μήνυμα ενώ σε περίπτωση που είτε το recordset είτε η σύνδεση με τη βάση δεν κλείσει εμφανίζονται αντίστοιχα μηνύματα λάθους.



MDImainform

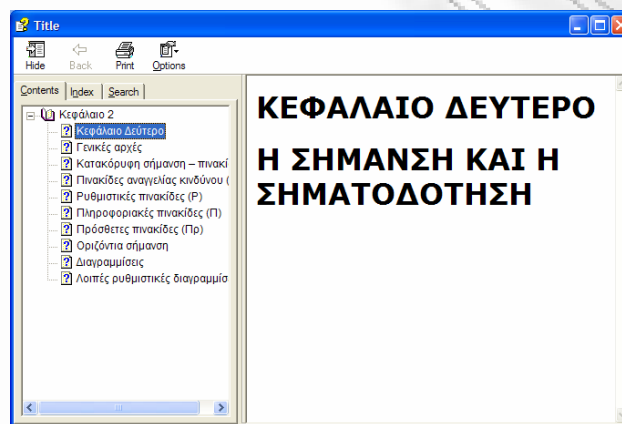
Όταν ανοίγει η φόρμα, στη γραμμή τίτλου εμφανίζεται ο τίτλος αλλά και η έκδοση του προγράμματος. Πριν αυτά δηλωθούν στον κώδικα της φόρμας τα δηλώνουμε στο «Project Properties» και συγκεκριμένα στη καρτέλα «Make».



Εφόσον η φόρμα είναι ενεργή και δεν υπάρχει «username» δηλαδή κάποιος ενεργός χρήστης τότε εμφανίζεται η φόρμα «Login» προκειμένου να εισαχθεί ο χρήστης. Εφόσον γίνει αυτό, τότε καλείται το «reset menus» στο οποίο έχουμε δηλώσει ότι όλα τα μενού της φόρμα είναι ενεργά και ορατά. Όταν εκτελεστεί αυτό, καλείται το «Set menus» στο οποίο δηλώνεται ποια μενού θα είναι ενεργά και μη, ανάλογα με τους τύπους χρήστη. Προκειμένου να το κάνουμε αυτό χρησιμοποιούμε με την εντολή «Select Case». Όπως είναι κατανοητό ο αριθμός των cases είναι 3 εφόσον και οι τύποι χρηστών είναι 3.

Όταν υπάρχει ενεργός χρήστης, επομένως είναι ενεργές και οι εντολές των μενού που τον αφορούν, ανάλογα με το τι επιλέγει εμφανίζονται και τα αντίστοιχα παράθυρα. Στο σημείο αυτό θα πρέπει να σταθούμε σε 2 από τις εντολές των μενού:

- 1) Έξοδος χρήστη. Σε αυτή την περίπτωση καλούνται 3 εντολές. Η «clear User Session», η «reset menus» και η «MDIform.Activate». Με την εκτέλεση της πρώτης, το όνομα, το επίθετο και ο τύπος χρήστη παρουσιάζονται κενά και ο κωδικός τους μηδενίζεται. Με την εκτέλεση της δεύτερης ενεργοποιούνται όλα τα μενού και οι εντολές τους και με την εκτέλεση της τρίτης εμφανίζεται η φόρμα «Login».
- 2) Βοήθεια. Για να δημιουργήσουμε τη βοήθεια χρησιμοποίησαμε το πρόγραμμα «kel chm creator». Τα κείμενα της βοήθειας αρχικά είχαν τη μορφή αρχείων html και στη συνέχεια με τη βοήθεια του προγράμματος μετατράπηκαν σε ένα αρχείο «chm» το οποίο βρίσκεται μέσα στο φάκελο της εφαρμογής. Η θέση στην οποία βρίσκεται το αρχείο της βοήθειας είναι συγκεκριμένη προκειμένου να μπορεί να βρεθεί από το πρόγραμμα. Αν λοιπόν το αρχείο υπάρχει στο path που ορίζεται, τότε ανοίγει όταν ο χρήστης επιλέξει «βοήθεια» διαφορετικά εμφανίζεται μήνυμα λάθους.



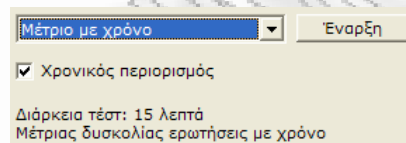
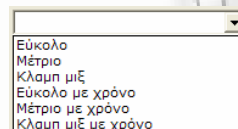
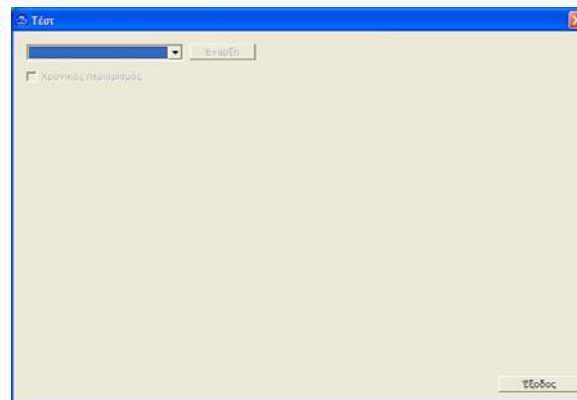
FrmTestRun

Όταν ανοίγει αυτή η φόρμα, καλούνται τρεις υπορουτίνες:

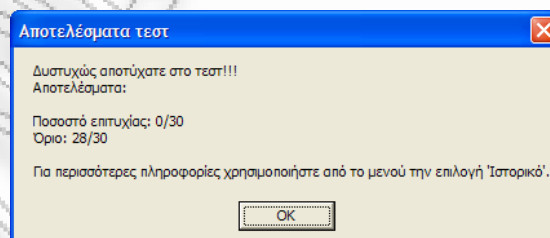
- 1) HideQuestionArea. Μέσω αυτής της υπορουτίνας δε φαίνεται η περιοχή των ερωτήσεων αλλά ούτε και το χρονόμετρο ενώ ορίζεται και το χρώμα των αριθμών του χρονόμετρου ως μαύρο.
- 2) ClearCategoryDetails. Μέσω αυτής της υπορουτίνας ενεργοποιείται ο χρονικός περιορισμός αλλά δεν είναι τσεκαρισμένος, επίσης ο χώρος στον οποίο εμφανίζονται οι πληροφορίες του τεστ που επιλέγει ο χρήστης είναι κενός.
- 3) PrepareCategoryCombo. Η υπορουτίνα αυτή χρησιμοποιείται προκειμένου να δημιουργήσουμε το sql query με το οποίο επιλέγονται όλα από τον πίνακα «Type» με βάση το «id». Ανοίγει η βάση δεδομένων, καθαρίζει το πεδίο «category» και «categoryid» και ελέγχεται η ύπαρξη ή μη του recordset. Εφόσον υπάρχουν εγγραφές ο δρομέας πηγαίνει εγγραφή – εγγραφή και για να φτάσει μέχρι την τελευταία, του δηλώνουμε να προσθέτει και από ένα τύπο κι ένα id κάθε φορά και να προχωράει στο επόμενο. Έπειτα κλείνει το recordset και η σύνδεση με τη βάση. Ενεργοποιείται επίσης και το κουμπί «Έναρξη», καθώς και ο τύπος αλλά απενεργοποιείται το id.

Εφόσον ο χρήστης επιλέξει τον τύπο του τεστ που θέλει να ακολουθήσει πατάει το κουμπί «Έναρξη». Με το που πατηθεί η έναρξη ετοιμάζεται η λίστα των ερωτήσεων. Μπορούν να επιλέγουν ερωτήσεις και από τις τρεις κατηγορίες ερωτήσεων εφόσον έτσι δηλώνεται από τη

κατηγορία του τεστ που έχει επιλέξει ο χρήστης. Στη συνέχεια διαμορφώνει τη λίστα με τις ερωτήσεις οι οποίες είναι τυχαίες για κάθε τεστ.



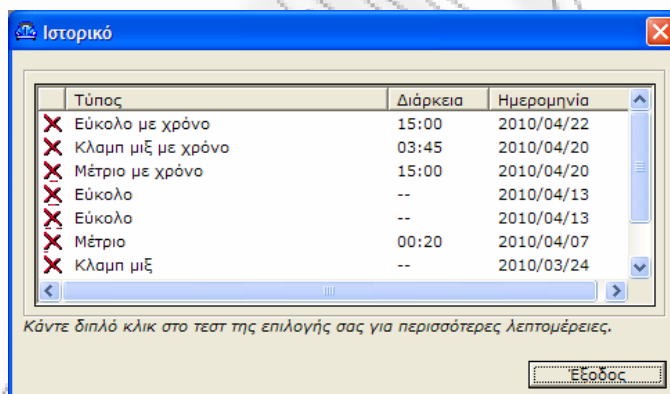
Αν ο χρήστης επιλέξει την «ολοκλήρωση», τότε εκτελείται η υπορουτίνα «finishTest». Αυτό που θα γίνει με την εκτέλεσή της είναι να πάρουμε πρώτα όλες τις απαντήσεις να γίνει έλεγχος για το αν το τεστ είναι ακόμη σε εξέλιξη και εν συνέχεια θα γίνει έλεγχος για το αν υπάρχουν αναπάντητες ερωτήσεις. Το τεστ δε θα ολοκληρωθεί και θα εμφανιστεί η φόρμα «frmTestRunFinish». Όπως είναι λογικό θα σταματήσει και ο χρόνος σε περίπτωση που το τεστ έχει χρονομέτρηση. Στη συνέχεια αν ο χρήστης σταματήσει τη διακοπή του τεστ ο χρόνος θα συνεχίσει να μετράει. Αν πάλι ο χρήστης αποφασίσει να διακόψει το τεστ θα καλέσουμε την υπορουτίνα «CalculateResults» με την οποία θα υπολογιστεί το αποτέλεσμα. Όταν ξεκινάει η διαδικασία εμφανίζεται το μήνυμα που μας ενημερώνει για το ότι πρέπει να περιμένουμε μέχρι να βγουν τα αποτελέσματα. Το μήνυμα χάνεται όταν εμφανιστούν τα αποτελέσματα.



FrmTestReports

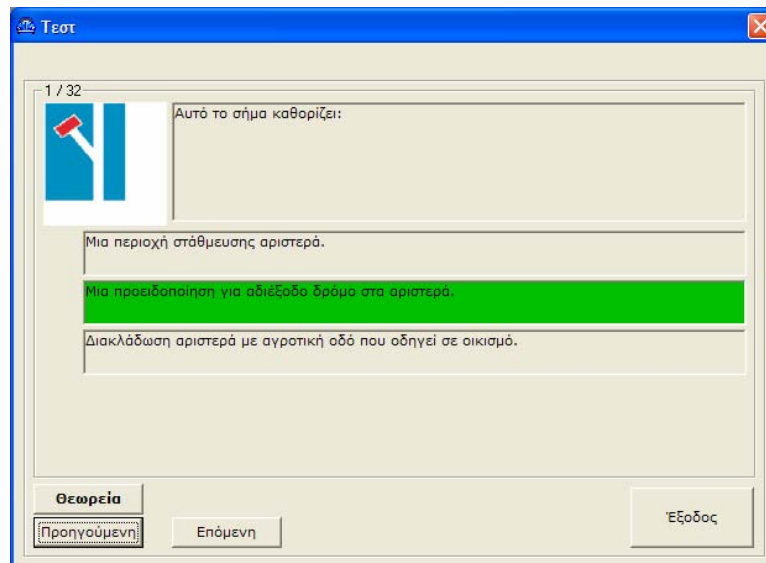
Όταν ανοίγει το «ιστορικό» καλούνται 2 υπορουτίνες:

- 1) SetformCation. Σε αυτή την υπορουτίνα δηλώνουμε τι θα εμφανίζεται στη γραμμή τίτλου του παραθύρου ανάλογα με το ποιος είναι ο χρήστης δηλαδή το αν είναι ο administrator ή απλός χρήστης. Το τι θα εμφανίζεται έχει οριστεί στο module «MsgProperties».
- 2) ShowTestList. Με αυτή την υπορουτίνα θέλουμε να εμφανίσουμε τη λίστα με τα τεστ που έχουν ολοκληρωθεί. Μέσα από αυτή την υπορουτίνα καλείται και η υπορουτίνα «GetTestList». Σε αυτή την υπορουτίνα δηλώνουμε το query μέσω του οποίου θα επιλέξουμε τα στοιχεία των τεστ που θα φαίνονται. Στη συνέχεια ανοίγει η σύνδεση με τη βάση, ελέγχεται η ύπαρξη του recordset και καλείται η υπορουτίνα «PrepareTestList» στην οποία καθαρίζονται οι επικεφαλίδες των στηλών και τα περιεχόμενά τους και ορίζουμε τις επικεφαλίδες των στηλών. Ανάλογα με τον τύπο του χρήστη αλλάζουν και οι επικεφαλίδες των στηλών. Εφόσον ολοκληρωθεί η εκτέλεση αυτής της υπορουτίνας εμφανίζονται τα αποτελέσματα. Στη συνέχεια καθαρίζουμε τη λίστα καλώντας την υπορουτίνα «ClearTestList» μέσω της οποίας καθαρίζονται οι επικεφαλίδες των στηλών και τα περιεχόμενά τους, δημιουργούνται οι νέες επικεφαλίδες και στη συνέχεια τις κάνουμε μη ορατές. Τέλος, κλείνει το recordset και η σύνδεση με τη βάση.



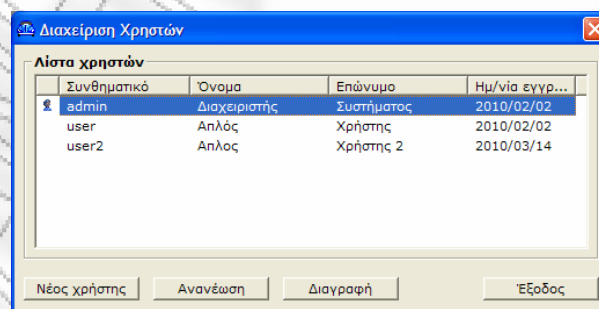
FrmTestReportsBD

Με το που ανοίγει αυτή η φόρμα ενημερώνεται ο τίτλος της. Το τι θα εμφανίζεται στη γραμμή τίτλου ορίζεται στο module «MsgProperties». Στη συνέχεια προετοιμάζεται η λίστα των ερωτήσεων και των απαντήσεων και εμφανίζεται η πρώτη εγγραφή, ενώ σε περίπτωση λάθους, εμφανίζεται αντίστοιχο μήνυμα. Για να εμφανιστούν οι εγγραφές, χρησιμοποιούμε την υπορουτίνα «getReportArray» κι έτσι δημιουργούμε το SQLquery, ανοίγει σύνδεση με τη βάση, ελέγχεται αν υπάρχει το συγκεκριμένο recordset και εμφανίζεται η πρώτη εγγραφή. Στη συνέχεια κλείνει η σύνδεση με το recordset και με τη βάση.

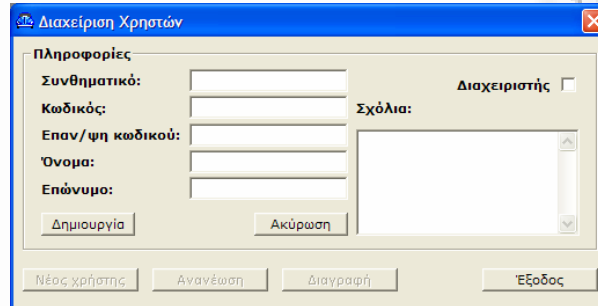


FrmUserMgt

FrmUserList. Με το που ανοίγει αυτή η φόρμα δηλώνεται ο τίτλος της φόρμας ο οποίος ορίζεται στο module «MsgProperties» και καλείται και η υπορουτίνα «ShowUserList». Μέσω αυτής της υπορουτίνας δηλώνουμε τη μη εμφάνιση του frame με τις πληροφορίες των χρηστών και ενεργοποιούνται τα κουμπιά «Νέος χρήστης», «Ανανέωση» και «Διαγραφή». Στη συνέχεια ενεργοποιείται η λίστα των χρηστών και καλείται η υπορουτίνα «GetUserList». Με αυτή την υπορουτίνα δηλώνουμε το sql query με το οποίο επιλέγουμε τα πάντα από τον πίνακα «users» με βάση το id τους. Έπειτα ανοίγει σύνδεση με τη βάση, ελέγχεται αν υπάρχει το συγκεκριμένο recordset και καλείται η υπορουτίνα «PrepareUserList». Σε αυτή την υπορουτίνα καθαρίζουμε τις επικεφαλίδες των στηλών και τα περιεχόμενά τους, τις ορίζουμε από την αρχή και μετά εμφανίζουμε τις εγγραφές. Στη συνέχεια καλείται η υπορουτίνα «ClearUserList» την οποία έχουμε δημιουργήσει προκειμένου να καθαρίζεται η λίστα σε περίπτωση που δεν υπάρχουν εγγραφές. Στην ουσία αυτό που γίνεται εδώ είναι να καθαρίζονται οι επικεφαλίδες των στηλών και τα περιεχόμενά τους, να φτιάχνει τις νέες επικεφαλίδες και μετά να τις κάνει μη ορατές. Τέλος κλείνει η σύνδεση με το recordset και τη βάση.



FrUserDetails. Όσον αφορά τώρα το άλλο frame (FrUserDetails) αυτό που θα πρέπει να δούμε είναι πως δημιουργούμε νέο χρήστη και πως ανανεώνουμε ένα ήδη υπάρχον χρήστη. Όσον αφορά τη δημιουργία χρήστη πρώτα γίνεται έλεγχος του username που εισάγαμε σε περίπτωση που υπάρχει ήδη οπότε και εμφανίζεται μήνυμα που μας ενημερώνει, διαφορετικά δημιουργούμε το sql εισαγωγής στοιχείων χρήστη. Στην περίπτωση της ανανέωσης των πληροφοριών του χρήστη χρησιμοποιούμε το sql για την ανανέωση των στοιχείων του χρήστη.



FrmAbout

Όταν ανοίγει αυτή η φόρμα δηλώνουμε τον τίτλο του παραθύρου αλλά και το περιεχόμενο του παραθύρου τα οποία ορίζονται στο module «MsgProperties».



Κ Ε Φ Α Λ Α Ι Ο 5

5. ΑΞΙΟΛΟΓΗΣΗ – ΣΥΓΚΡΙΣΗ ΕΦΑΡΜΟΓΩΝ

5.1. ΑΞΙΟΛΟΓΗΣΗ ΕΦΑΡΜΟΓΩΝ

5.1.1. ΣΥΣΤΗΜΑ ΕΚΠΑΙΔΕΥΣΗΣ ΚΑΙ ΕΞΕΤΑΣΗΣ ΥΠΟΨΗΦΙΩΝ ΟΔΗΓΩΝ

Η αξιολόγηση του λογισμικού είναι πολύ σημαντική καθώς και η συνεισφορά του εκπαιδευτικού λογισμικού στην εκπαιδευτική διαδικασία είναι επίσης ιδιαίτερα σημαντική. Κατά την αξιολόγηση θα εξετάσουμε τον τρόπο με τον οποίο το λογισμικό ανταποκρίνεται σε προκαθορισμένα κριτήρια ποιότητας, κάτι το οποίο θα επιτύχουμε συλλέγοντας τόσο τα ποσοτικά όσο και τα ποιοτικά στοιχεία που θα προκύψουν από τη χρήση του.

Η αξιολόγηση θα επιτευχθεί με τη συμπλήρωση ερωτηματολογίων, στα οποία θα χρησιμοποιηθούν ερωτήσεις κλειστού τύπου.

Σε αντίθεση με τη συνέντευξη, η μέθοδος της συμπλήρωσης ερωτηματολογίων είναι λιγότερο ευέλικτη καθώς οι ερωτήσεις είναι καταγεγραμμένες και δε μπορούν να αλλάξουν κατά περίπτωση. Το γεγονός αυτό κάνει τη διαχείρισή τους αλλά και την εξαγωγή συμπερασμάτων από αυτά πολύ πιο εύκολη.

Προτού προχωρήσουμε στη δημιουργία του ερωτηματολογίου θα πρέπει να θέσουμε την κατάταξη των απαντήσεων:

0 = μη κατάλληλο	2 = φτωχό	4 = μέτριο
6 = καλό	8 = πολύ καλό	10 = εξαιρετικό

Για να δημιουργήσουμε μια λίστα με κριτήρια προκειμένου να αξιολογήσουμε το σχεδιασμό του συστήματος διεπαφής με τους χρήστες, θα βασιστούμε στη χρησιμοποίησιμότητα, καθώς αποτελεί τον κύριο στόχο του σχεδιαστή μιας εφαρμογής.

Γνωρίζοντας πως η χρησιμοποίησιμότητα διέπεται από κάποιες αρχές (ευκολία, εκμάθησης, ευκαμψία, ανθεκτικότητα) και βασιζόμενοι σε αυτές, θα προσπαθήσουμε να δημιουργήσουμε τη λίστα με τα κριτήρια αξιολόγησης.

5.1.1.1. Ευκολία στην εκμάθηση

Με τον όρο ευκολία στην εκμάθηση εννοούμε το πόσο εύκολο είναι για το χρήστη να μάθει να χρησιμοποιεί το λογισμικό. Η αρχή αυτή εξαρτάται με τη σειράς της, από κάποιες άλλες αρχές:

5.1.1.1.1. Προβλεψιμότητα

Η προβλεψιμότητα ορίζει το κατά πόσο μπορεί ο χρήστης να καθορίσει τα αποτελέσματα κάποιων λειτουργιών, καθώς και να είναι σε θέση να καταλάβει ποιες λειτουργίες

επιτρέπονται. Βάση της προβλεψιμότητας λοιπόν μπορούμε να δημιουργήσουμε το παρακάτω ερωτηματολόγιο:

1) Υπάρχει ένδειξη για το ποιες εντολές από το μενού της γραμμής μενού μπορεί να χρησιμοποιήσει ο χρήστης μια δεδομένη στιγμή;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
Σημείωση: Οι εντολές των μενού που δεν μπορεί να χρησιμοποιήσει ο χρήστης σε μια δεδομένη χρονική στιγμή, εμφανίζονται αχνά, δηλαδή είναι ανενεργές, σε αντίθεση με αυτές που μπορεί να επιλέξει ο χρήστης οι οποίες είναι ενεργές.		

5.1.1.1.2. Σύνθεση

Η σύνθεση ορίζει το κατά πόσο μπορεί ο χρήστης να δει τις αλλαγές που γίνονται όταν εκτελεί μια εντολή. Με βάση τη σύνθεση, δημιουργούμε τις παρακάτω ερωτήσεις:

1) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν η σωστή ή όχι;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
Σημείωση: Όταν ο χρήστης ολοκληρώσει ένα τεστ, μπει στο ιστορικό του και επιλέξει το τεστ που έκανε, μεταβαίνει στην ερώτηση που θέλει και την ελέγχει.		
2) Έχει τη δυνατότητα ο χρήστης να δει τη θεωρία που έχει εισαχθεί στην εφαρμογή;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
Σημείωση: Τη θεωρία μπορεί να τη δει μέσω των τεστ που έχει ολοκληρώσει.		

5.1.1.1.3. Οικειότητα





Η οικειότητα ορίζει το κατά πόσο η εμπειρία του χρήστη από άλλες εφαρμογές μπορεί να τον βοηθήσει στο να χρησιμοποιήσει αυτή την εφαρμογή. Βάση της οικειότητας, οι ερωτήσεις που δημιουργήθηκαν είναι οι παρακάτω:

1) Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

2) Οι ονομασίες των κουμπιών της εφαρμογής είναι ευνόητες για χρήστες οι οποίοι χρησιμοποιούν εφαρμογές;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

5.1.1.1.4. Συνέπεια

Η συνέπεια ορίζει το κατά πόσο χρησιμοποιούνται οι ίδιοι συμβολισμοί για να αναπαρασταθούν τα ίδια πράγματα. Με τη συνέπεια δημιουργούμε τις παρακάτω ερωτήσεις:

1) Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, το κουμπί «ακύρωσης» είναι ίδιο και παραμένει στην ίδια θέση σε όλα τα παράθυρα;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Το κουμπί ακύρωσης έχει την ίδια μορφή και παραμένει στην ίδια θέση σε όλα τα παράθυρα κι έτσι ο χρήστης το συνηθίζει.				
2) Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, τα κουμπιά «έξοδος», «ανανέωση» και «διαγραφή» είναι ίδια και παραμένουν ίδια σε όλα τα παράθυρα;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Τα κουμπιά αυτά έχουν την ίδια μορφή και παραμένουν στην ίδια θέση σε όλα τα παράθυρα.				
3) Ο συμβολισμός ενός αποτυχημένου αλλά κι ενός επιτυχημένου τεστ παραμένει πάντοτε ίδιος;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Τα αποτυχημένα τεστ συμβολίζονται πάντοτε με  και τα αποτυχημένα, πάντοτε με  .				
4) Ο συμβολισμός των σωστών απαντήσεων κατά τη δημιουργία των ερωτήσεων, όπως και των λάθος απαντήσεων, είναι ίδιος;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Οι λάθος απαντήσεις συμβολίζονται με  και οι σωστές με  .				
5) Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Οι εικόνες εμφανίζονται στο πάνω αριστερό μέρος του παραθύρου του τεστ.				

5.1.1.2. Ευκαμψία του συστήματος διεπαφής

Με τον όρο ευκαμψία του συστήματος διεπαφής νοούνται οι τρόποι επικοινωνίας του χρήστη με το σύστημα. Η ευκαμψία με τη σειρά της εξαρτάται από τα παρακάτω:

5.1.1.2.1. Πολλαπλός έλεγχος

Ο πολλαπλός έλεγχος ορίζει το κατά πόσο ο χρήστης έχει δυνατότητα διαλόγου για περισσότερες από μία εργασίες τη φορά. Οι ερωτήσεις που δημιουργούνται είναι:

1) Ο χρήστης μπορεί να ανοίξει πάνω από ένα παράθυρα τη φορά;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

5.1.1.2.2. Δυνατότητα αλλαγής εργασιών

Η δυνατότητα αλλαγής εργασιών ορίζει το κατά πόσο μπορεί ο χρήστης να περάσει από μια εργασία σε μια άλλη. Οι ερωτήσεις που δημιουργούνται είναι:

1) Δίνεται η δυνατότητα στο χρήστη να εκτελεί διαφορετικές εργασίες τη μια μετά την άλλη με ευκολία;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
Σημείωση: Για παράδειγμα μπορεί να κάνει ένα τεστ και αμέσως να δει την πρόοδό του από το ιστορικό.				

5.1.1.2.3. Προσαρμοστικότητα

Η προσαρμοστικότητα ορίζει το κατά πόσο ο χρήστης μπορεί να αλλάξει κάποιες παραμέτρους στο πρόγραμμα για τη δική του ευκολία. Βάση της προσαρμοστικότητας, δημιουργούμε τις παρακάτω ερωτήσεις:

1) Μπορεί ο διαχειριστής του συστήματος να δημιουργήσει μια ερώτηση να τροποποιήσει ή και να τη διαγράψει;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
2) Μπορεί ο διαχειριστής του συστήματος να εισάγει νέα θεωρία ή και να διαγράψει μια ήδη υπάρχουσα;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

3) Μπορεί ο διαχειριστής του συστήματος να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

5.1.1.3. Ανθεκτικότητα του συστήματος διεπαφής

Με τον όρο ανθεκτικότητα του συστήματος διεπαφής εννοούμε το κατά πόσο η εφαρμογή μπορεί να στηρίξει το χρήστη σε πιθανά λάθη. Η ανθεκτικότητα εξαρτάται με τη σειρά της από τους παρακάτω παράγοντες:

5.1.1.3.1. Δυνατότητα παρατήρησης

Η δυνατότητα παρατήρησης ορίζει το κατά πόσο μπορεί ο χρήστης να δει τα αποτελέσματα ενεργειών που εκτελεί. Με βάση τη δυνατότητα παρατήρησης δημιουργούνται διάφορα κριτήρια ανάλογα με τις ενέργειες που θέλει να κάνει ο χρήστης. Επιλέξαμε μερικές από αυτές που χρησιμοποιούνται συχνότερα και δημιουργήσαμε τις παρακάτω ερωτήσεις:

1) Αν ο διαχειριστής του συστήματος επιλέξει διαγραφή μιας ερώτησης, βλέπει το αποτέλεσμα της επιλογής του;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
2) Αν ο διαχειριστής του συστήματος δημιουργήσει ένα καινούριο χρήστη, βλέπει το αποτέλεσμα της επιλογής του;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
3) Αν ο διαχειριστής του συστήματος εισάγει νέα θεωρία, βλέπει το αποτέλεσμα της επιλογής του;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
4) Αν ο διαχειριστής του συστήματος επιλέξει διαγραφή μιας απάντησης, βλέπει το αποτέλεσμα της επιλογής του;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

5.1.1.3.2. *Ανανηψιμότητα*

Η ανανηψιμότητα ορίζει το κατά πόσο μπορεί να διορθωθεί ένα λάθος. Οι ερωτήσεις που δημιουργήθηκαν με βάση την ανανηψιμότητα είναι οι παρακάτω:

1) Έχει τη δυνατότητα ο χρήστης να ακυρώσει την εκτέλεση μιας εντολής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
Σημείωση: Δεν υπάρχει αναίρεση.		
2) Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Μπορεί ο διαχειριστής του συστήματος να αλλάξει τις λάθος απαντήσεις μιας ερώτησης;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Μπορεί ο διαχειριστής να αλλάξει τη θεωρία;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
5) Μπορεί ο απλός χρήστης να αλλάξει τον κωδικό πρόσβασής του;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.1.3.3. *Λειτουργικότητα*

Η λειτουργικότητα ορίζει το κατά πόσο υποστηρίζονται όλες οι ενέργειες που θα ήθελε να κάνει ο χρήστης. Οι ερωτήσεις με βάση τη λειτουργικότητα που δημιουργούνται είναι πολλές γι' αυτό κι έχουμε παραθέσει τις πιο σημαντικές:

1) Μπορεί ο απλός χρήστης να διαγράψει ένα από τα τεστ που έχει κάνει;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

2) Μπορεί ο διαχειριστής του συστήματος να διαγράψει ένα τεστ κάποιου χρήστη;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.1.4. Ερωτήσεις γενικού περιεχομένου

1) Η δομή των παραθύρων που εμφανίζονται είναι συνεπής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Η μορφή των παραθύρων που εμφανίζονται είναι συνεπής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Η μορφή των μηνυμάτων που εμφανίζονται είναι συνεπής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Δίνεται η δυνατότητα στο χρήστη να χρησιμοποιήσει συντομεύσεις για κάποιες ενέργειες;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5) Τα μηνύματα προς το χρήστη είναι περιεκτικά αλλά και πλήρη όσον αφορά την πληροφορία που χρειάζεται ο χρήστης;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
6) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
7) Το help είναι εύχρηστο;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
8) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
9) Προκαλείται το ενδιαφέρον του χρήστη;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
10) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
11) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>
12) Οι χρήστες όταν κάνουν λάθη ακυρώνουν εύκολα τις πράξεις τους;	10 <input type="checkbox"/> 8 <input type="checkbox"/> 6 <input type="checkbox"/> 4 <input type="checkbox"/> 2 <input type="checkbox"/> 0 <input type="checkbox"/>

13) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

5.1.2. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

Όπως έχει προαναφερθεί, για την αξιολόγηση θα χρησιμοποιηθεί το ερωτηματολόγιο με το οποίο αξιολογήσαμε και το πρόγραμμά μας. Εξαιτίας του ότι τα προγράμματα διαφέρουν πολύ, κάποιες ερωτήσεις δε θα χρησιμοποιηθούν ενώ θα προστεθούν και καινούριες.

Προκειμένου να ξεχωρίσουμε τις παλιές από τις καινούριες, έχουμε σκιάσει το κομμάτι του πίνακα που είναι οι παλιές, με γκρι χρώμα.

5.1.2.1. Ευκολία στην εκμάθηση

1) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν σωστή ή όχι;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
2) Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
3) Οι ονομασίες των κουμπιών της εφαρμογής είναι ευνόητες για χρήστες οι οποίοι χρησιμοποιούν εφαρμογές;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
4) Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>
5) Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, το κουμπί «Αρχική οθόνη» είναι ίδιο και παραμένει στην ίδια θέση σε όλα τα παράθυρα;	10	<input type="checkbox"/>	8	<input type="checkbox"/>
	6	<input type="checkbox"/>	4	<input type="checkbox"/>
	2	<input type="checkbox"/>	0	<input type="checkbox"/>

6) Η ονομασία μιας επιτυχημένης ή αποτυχημένης απάντησης σε ένα τεστ παραμένει ίδια για όλα τα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
7) Ο χαρακτηρισμός ενός επιτυχημένου ή αποτυχημένου τεστ παραμένει ίδιος για όλα τα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.2.2. Ευκαμψία του συστήματος διεπαφής

1) Ο χρήστης μπορεί να ανοίξει πάνω από ένα παράθυρο τη φορά;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Δίνεται η δυνατότητα στο χρήστη να εκτελεί διαφορετικές εργασίες τη μια μετά την άλλη με ευκολία;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Μπορεί ο διαχειριστής του συστήματος να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.2.3. Ανθεκτικότητα του συστήματος επαφής

1) Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

3) Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Μπορεί ο χρήστης να δει πόσες είναι οι σωστές και οι λάθος απαντήσεις;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
5) Μπορεί ο χρήστης να δει αν ολοκλήρωσε με επιτυχία το τεστ μόλις τελειώσει;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.2.4. Ερωτήσεις γενικού περιεχομένου

1) Η δομή των παραθύρων που εμφανίζονται είναι συνεπής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Η μορφή των παραθύρων που εμφανίζονται είναι συνεπής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Δίνεται η δυνατότητα στο χρήστη να χρησιμοποιήσει συντομεύσεις για κάποιες ενέργειες;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5) Το help είναι εύχρηστο;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
6) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
7) Προκαλείται το ενδιαφέρον του χρήστη;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
8) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
9) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
10) Οι χρήστες όταν κάνουν λάθη ακυρώνουν εύκολα τις πράξεις τους;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
11) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.3. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: //kok.artinoi.gr

Όπως και με το προηγούμενο πρόγραμμα έτσι και σε αυτό θα χρησιμοποιήσουμε το ερωτηματολόγιο που χρησιμοποιήσαμε και για το πρόγραμμά μας ενώ πάλι κάποιες ερωτήσεις δε θα χρησιμοποιηθούν και κάποιες καινούριες θα προστεθούν.

Προκειμένου να ξεχωρίσουμε τις παλιές από τις καινούριες, έχουμε σκιάσει το κομμάτι του πίνακα που είναι οι παλιές, με γκρι χρώμα.

5.1.3.1. Ευκολία στην εκμάθηση

1) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν σωστή ή όχι;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Ο συμβολισμός ενός αποτυχημένου αλλά κι ενός επιτυχημένου τεστ παραμένει πάντοτε ίδιος;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
5) Η ονομασία μιας επιτυχημένης ή αποτυχημένης απάντησης σε ένα τεστ παραμένει ίδια για όλα τα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
6) Ο χαρακτηρισμός ενός επιτυχημένου ή αποτυχημένου τεστ παραμένει ίδιος για όλα τα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.3.2. Ευκαμψία του συστήματος διεπαφής

1) Μπορεί ο χρήστης να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.3.3. Ανθεκτικότητα του συστήματος επαφής

1) Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
3) Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Μπορεί ο χρήστης να δει πόσες είναι οι σωστές και οι λάθος απαντήσεις;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
5) Μπορεί ο χρήστης να δει αν ολοκλήρωσε με επιτυχία το τεστ μόλις το τελειώσει;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.3.4. Ερωτήσεις γενικού περιεχομένου

1) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
2) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

3) Προκαλείται το ενδιαφέρον του χρήστη;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
4) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
5) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>
6) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	10 <input type="checkbox"/>	8 <input type="checkbox"/>
	6 <input type="checkbox"/>	4 <input type="checkbox"/>
	2 <input type="checkbox"/>	0 <input type="checkbox"/>

5.1.4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΑΞΙΟΛΟΓΗΣΗΣ

Στην αξιολόγηση του λογισμικού συμμετείχαν 20 άτομα εκ των οποίων οι 10 έχουν δώσει εξετάσεις για δίπλωμα οδήγησης με το παλιό σύστημα, οι 7 με το καινούριο και οι 3 δεν έχουν δώσει εξετάσεις.

Καθώς το δείγμα της αξιολόγησης δεν είναι μεγάλο, η εξαγωγή των συμπερασμάτων δεν μπορεί να είναι τόσο αξιόπιστη. Εν τούτοις ήταν μια πολύ καλή ευκαιρία για συλλογή πρώτων εντυπώσεων και παρατηρήσεων. Παρακάτω φαίνονται τα αποτελέσματα της αξιολόγησης αναλυτικά.

5.1.4.1. Σύστημα εκπαίδευσης και εξέτασης υποψηφίων οδηγών

5.1.4.1.1. Ευκολία στην εκμάθηση

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Υπάρχει ένδειξη για το ποιες εντολές από το μενού της γραμμής μενού μπορεί να χρησιμοποιήσει ο χρήστης μια δεδομένη στιγμή;	10
2) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν σωστή ή όχι;	7,7
3) Έχει τη δυνατότητα ο χρήστης να δει τη θεωρία που έχει	6,3

	εισαχθεί στην εφαρμογή;	
4)	Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	8,5
5)	Οι ονομασίες των κουμπιών της εφαρμογής είναι ευνόητες για χρήστες οι οποίοι χρησιμοποιούν εφαρμογές;	9,8
6)	Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, το κουμπί «ακύρωση» είναι ίδιο και παραμένει στην ίδια θέση σε όλα τα παράθυρα;	10
7)	Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, τα κουμπιά «έξοδος», «ανανέωση» και «διαγραφή» είναι ίδια και παραμένουν ίδια σε όλα τα παράθυρα;	10
8)	Ο συμβολισμός ενός αποτυχημένου αλλά κι ενός επιτυχημένου τεστ παραμένει πάντοτε ίδιος;	10
9)	Ο συμβολισμός των σωστών απαντήσεων κατά τη δημιουργία των ερωτήσεων, όπως και των λάθος απαντήσεων, είναι ίδιος;	10
10)	Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10

Σύνολο ερωτήσεων: 10

Μέσος όρος απαντήσεων: 9,2

Από το μέσο όρο των απαντήσεων που δόθηκαν οδηγούμαστε στο συμπέρασμα ότι πρόκειται για μια εφαρμογή, η εκμάθηση της οποίας είναι εξαιρετικά εύκολη.

Συγκεκριμένα, παρατηρούμε ότι στην εφαρμογή οι συμβολισμοί των εντολών που χρησιμοποιούνται είναι ίδιοι με αυτούς που χρησιμοποιούνται στις περισσότερες εφαρμογές. Παράλληλα τα κοινά κουμπιά που χρησιμοποιούνται στα παράθυρα έχουν πάντοτε την ίδια μορφή και την ίδια θέση. Τέλος, οι πληροφορίες που επιθυμεί να αντλήσει ο χρήστης από την εφαρμογή, σε γενικές γραμμές είναι εύκολα προσβάσιμες.

5.1.4.1.2. Ευκαμψία του συστήματος διεπαφής

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Ο χρήστης μπορεί να ανοίξει πάνω από ένα παράθυρο τη φορά;	0
2) Δίνεται η δυνατότητα στο χρήστη να εκτελεί διαφορετικές εργασίες τη μια μετά την άλλη με ευκολία;	8,3
3) Μπορεί ο διαχειριστής του συστήματος να δημιουργήσει μια ερώτηση να τροποποιήσει ή και να τη διαγράψει;	10
4) Μπορεί ο διαχειριστής του συστήματος να εισάγει νέα θεωρία ή και να διαγράψει μια ήδη υπάρχουσα;	10
5) Μπορεί ο διαχειριστής του συστήματος να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	10

Σύνολο ερωτήσεων: 5

Μέσος όρος απαντήσεων: 7,6

Όσον αφορά τις αλλαγές που μπορεί να κάνει ο χρήστης στην εφαρμογή αλλά και το πως μπορεί να επικοινωνήσει με το σύστημα, ο μέσος όρος υποδηλώνει ότι πρόκειται για ένα αρκετά εύκαμπτο σύστημα. Η ευκαμψία αυτή αφορά κυρίως τον διαχειριστή του συστήματος στον οποίο δίνεται η δυνατότητα αρκετών αλλαγών που αφορούν τα βασικά κομμάτια της εφαρμογής (θεωρία, ερωτήσεις, απαντήσεις).

Αντιθέτως δυσκαμψία της εφαρμογής παρατηρούμε στο γεγονός ότι ένας χρήστης δεν μπορεί να κάνει πολλές εργασίες ταυτόχρονα.

5.1.4.1.3. Ανθεκτικότητα του συστήματος διεπαφής

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Αν ο διαχειριστής του συστήματος επιλέξει διαγραφή μιας ερώτησης, βλέπει το αποτέλεσμα της επιλογής του;	10
2) Αν ο διαχειριστής του συστήματος δημιουργήσει ένα καινούριο χρήστη, βλέπει το αποτέλεσμα της επιλογής του;	10
3) Αν ο διαχειριστής του συστήματος εισάγει νέα θεωρία, βλέπει το αποτέλεσμα της επιλογής του;	8
4) Αν ο διαχειριστής του συστήματος επιλέξει διαγραφή μιας απάντησης, βλέπει το αποτέλεσμα της επιλογής του;	10
5) Έχει τη δυνατότητα ο χρήστης να ακυρώσει την εκτέλεση μιας εντολής;	0
6) Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	10
7) Μπορεί ο διαχειριστής του συστήματος να αλλάξει τις λάθος απαντήσεις μιας ερώτησης;	10
8) Μπορεί ο διαχειριστής να αλλάξει τη θεωρία;	10
9) Μπορεί ο απλός χρήστης να αλλάξει τον κωδικό πρόσβασής του;	0
10) Μπορεί ο απλός χρήστης να διαγράψει ένα από τα τεστ που έχει κάνει;	0
11) Μπορεί ο διαχειριστής του συστήματος να διαγράψει ένα τεστ κάποιου χρήστη;	0
12) Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	7,7
13) Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	10

Σύνολο ερωτήσεων: 13

Μέσος όρος απαντήσεων: 6,6

Βασιζόμενοι στις απαντήσεις που δόθηκαν και κατά συνέπεια του μέσου όρου που προέκυψε συμπεραίνουμε ότι η ανθεκτικότητα του συστήματος είναι καλή. Ο χρήστης

οποιαδήποτε αλλαγή κι αν κάνει έχει τη δυνατότητα να δει το αποτέλεσμα της. Παράλληλα, η εφαρμογή υποστηρίζει πάρα πολλές από τις ενέργειες που ο χρήστης θα ήθελε να εκτελεί.

Ένα μειονέκτημα σε εισαγωγικά της εφαρμογής είναι το ότι ο χρήστης σε περίπτωση που κάνει κάποιο λάθος, π.χ. αντιστοιχίσει μια ερώτηση σε λάθος κομμάτι θεωρίας, δεν έχει τη δυνατότητα να διορθώσει το λάθος αυτό με το πάτημα ενός κουμπιού (αναίρεση) αλλά πηγαίνοντας στο σημείο στο οποίο έγινε το λάθος και διορθώνοντας το χειροκίνητα.

5.1.4.1.4. Ερωτήσεις γενικού περιεχομένου

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Η δομή των παραθύρων που εμφανίζονται είναι συνεπής;	10
2) Η μορφή των παραθύρων που εμφανίζονται είναι συνεπής;	10
3) Η μορφή των μηνυμάτων που εμφανίζονται είναι συνεπής;	10
4) Δίνεται η δυνατότητα στο χρήστη να χρησιμοποιήσει συντομεύσεις για κάποιες ενέργειες;	0
5) Τα μηνύματα προς το χρήστη είναι περιεκτικά αλλά και πλήρη όσον αφορά την πληροφορία που χρειάζεται ο χρήστης;	8,3
6) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	7,7
7) Το help είναι εύχρηστο;	9,3
8) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10
9) Προκαλείται το ενδιαφέρον του χρήστη;	9,5
10) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	8,3
11) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	7,5
12) Οι χρήστες όταν κάνουν λάθη ακυρώνουν εύκολα τις πράξεις τους;	6,3
13) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	7,7

Σύνολο ερωτήσεων: 13

Μέσος όρος απαντήσεων: 8

Όσον αφορά τις ερωτήσεις γενικού περιεχομένου, μπορούμε να συμπεράνουμε ότι η μορφή και η δομή των παραθύρων αλλά και των μηνυμάτων της εφαρμογής είναι συνεπής κι έτσι δεν κουράζουν αλλά κυρίως δεν μπερδεύουν τον χρήστη.

Η ηλεκτρονική βοήθεια που παρέχεται στους χρήστες τις περισσότερες φορές είναι αρκετά ικανοποιητική με αποτέλεσμα όταν δυσκολεύονται να κατανοήσουν τον τρόπο λειτουργίας της εφαρμογής η βοήθεια τους διαφωτίζει για το πως θα πρέπει να ενεργήσουν.

5.1.4.2. Πρόγραμμα υπουργείου μεταφορών και επικοινωνιών

Όπως έχει προαναφερθεί, για την αξιολόγηση θα χρησιμοποιηθεί το ερωτηματολόγιο με το οποίο αξιολογήσαμε και το πρόγραμμά μας. Εξαιτίας του ότι τα προγράμματα διαφέρουν πολύ, κάποιες ερωτήσεις δε θα χρησιμοποιηθούν ενώ θα προστεθούν και καινούριες.

Προκειμένου να ξεχωρίσουμε τις παλιές από τις καινούριες, έχουμε σκιάσει το κομμάτι του πίνακα που είναι οι παλιές, με γκρι χρώμα.

5.1.4.2.1. Ευκολία στην εκμάθηση

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν σωστή ή όχι;	6,5
2) Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	8,5
3) Οι ονομασίες των κουμπιών της εφαρμογής είναι ευνόητες για χρήστες οι οποίοι χρησιμοποιούν εφαρμογές;	9,6
4) Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10
5) Στα παράθυρα που εμφανίζονται όταν ο χρήστης επιλέγει μια εντολή, το κουμπί «Αρχική οθόνη» είναι ίδιο και παραμένει στην ίδια θέση σε όλα τα παράθυρα;	10
6) Η ονομασία μίας επιτυχημένης ή αποτυχημένης απάντησης σε ένα τεστ παραμένει ίδια για όλα τα τεστ;	10
7) Ο χαρακτηρισμός ενός επιτυχημένου ή αποτυχημένου τεστ παραμένει ίδιος για όλα τα τεστ;	10

Σύνολο ερωτήσεων: 7

Μέσος όρος απαντήσεων: 9,2

Με βάση τις απαντήσεις που δόθηκαν και κατ' επέκταση το μέσο όρο που προέκυψε, συμπεραίνουμε ότι η εκμάθηση της συγκεκριμένης εφαρμογής είναι πάρα πολύ εύκολη. Αναλυτικότερα τώρα, βλέπουμε ότι οι ονομασίες που χρησιμοποιούνται για συγκεκριμένες έννοιες παραμένουν ίδιες και παράλληλα τα κουμπιά που είναι ίδια, έχουν όμοια μορφή και θέση στα παράθυρα που εμφανίζονται. Τέλος, οι πληροφορίες που δίνει τη δυνατότητα η εφαρμογή να αντλήσει ο χρήστης είναι σχετικά εύκολα προσβάσιμες.

5.1.4.2.2. *Ευκαμψία του συστήματος διεπαφής*

Ερωτήσεις		Μέσος όρος απαντήσεων
1)	Ο χρήστης μπορεί να ανοίξει πάνω από ένα παράθυρο τη φορά;	0
2)	Δίνεται η δυνατότητα στο χρήστη να εκτελεί διαφορετικές εργασίες τη μια μετά την άλλη με ευκολία;	9,5
3)	Μπορεί ο διαχειριστής του συστήματος να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	0

Σύνολο ερωτήσεων: 3

Μέσος όρος απαντήσεων: 3

Κρίνοντας από το μέσο όρο που έχει προκύψει μπορούμε εύκολα να συμπεράνουμε ότι δεν πρόκειται για μια αρκετά εύκαμπτη εφαρμογή, καθώς ο χρήστης δεν έχει τη δυνατότητα να κάνει αλλαγές στην εφαρμογή.

5.1.4.2.3. *Ανθεκτικότητα του συστήματος επαφής*

Ερωτήσεις		Μέσος όρος απαντήσεων
1)	Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	0
2)	Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	0
3)	Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	0
4)	Μπορεί ο χρήστης να δει πόσες είναι οι σωστές και οι λάθος απαντήσεις;	10
5)	Μπορεί ο χρήστης να δει αν ολοκλήρωσε με επιτυχία το τεστ μόλις τελειώσει;	10

Σύνολο ερωτήσεων: 5

Μέσος όρος απαντήσεων: 4

Από το μέσο όρο των απαντήσεων που δόθηκαν καταλαβαίνουμε ότι η ανθεκτικότητα του συστήματος είναι μέτρια. Σημαντικό είναι το γεγονός ότι δε δίνεται η δυνατότητα στο χρήστη να αλλάξει την απάντηση σε μια ερώτηση εφόσον την έχει επιβεβαιώσει. Η πρόοδος του χρήστη είναι μια επίσης σημαντική παράμετρος αλλά δεν παρέχεται από την εφαρμογή. Από την άλλη πλευρά η δυνατότητα που δίδεται στο χρήστη να δει αν πέτυχε στο τεστ καθώς και πόσες απαντήσεις ήταν σωστές ή λάθος κατά κάποιο τρόπο αντισταθμίζει τα προηγούμενα.

5.1.4.2.4. Ερωτήσεις γενικού περιεχομένου

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Η δομή των παραθύρων που εμφανίζονται είναι συνεπής;	10
2) Η μορφή των παραθύρων που εμφανίζονται είναι συνεπής;	10
3) Δίνεται η δυνατότητα στο χρήστη να χρησιμοποιήσει συντομεύσεις για κάποιες ενέργειες;	0
4) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	10
5) Το help είναι εύχρηστο;	10
6) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10
7) Προκαλείται το ενδιαφέρον του χρήστη;	9,5
8) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	9,8
9) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	9,8
10) Οι χρήστες όταν κάνουν λάθη ακυρώνουν εύκολα τις πράξεις τους;	6,5
11) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	7,7

Σύνολο ερωτήσεων: 11

Μέσος όρος απαντήσεων: 8,5

Με βάση το μέσο όρο αυτών των ερωτήσεων, προκύπτει ότι η μορφή και η δομή των παραθύρων της εφαρμογής είναι εξαιρετική όπως και η συνέπεια στους όρους αλλά και τα σύμβολα που χρησιμοποιούνται. Η βοήθεια που δίνεται για τον τρόπο διεξαγωγής των τεστ είναι επίσης εξαιρετικά κατανοητή και κατ' επέκταση αρκετή για την ομαλή διεξαγωγή ενός τεστ από οποιαδήποτε χρήστη.

Τα «μελανά» σημεία που συναντάμε εδώ είναι ο χρήστης δε μπορεί με ευκολία να διορθώσει κάποιο λάθος που θα κάνει αλλά και το ότι η εφαρμογή θα μπορούσε να ικανοποιούσε και κάποιες επιπλέον «απαιτήσεις» του χρήστη.

5.1.4.3. Πρόγραμμα ιστοσελίδας: kok.artinoi.gr

Όπως και με το προηγούμενο πρόγραμμα έτσι και σε αυτό θα χρησιμοποιήσουμε το ερωτηματολόγιο που χρησιμοποιήσαμε και για το πρόγραμμά μας ενώ πάλι κάποιες ερωτήσεις δε θα χρησιμοποιηθούν και κάποιες καινούριες θα προστεθούν.

Προκειμένου να ξεχωρίσουμε τις παλιές από τις καινούριες, έχουμε σκιασεί το κομμάτι του πίνακα που είναι οι παλιές, με γκρι χρώμα.

5.1.4.3.1. Ευκολία στην εκμάθηση

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Έχει τη δυνατότητα ο χρήστης να δει αν η απάντηση που έχει δώσει σε κάποιο τεστ ήταν σωστή ή όχι;	10
2) Ο χρήστης μπορεί να βοηθηθεί στην εκμάθηση της εφαρμογής αν έχει χρησιμοποιήσει άλλη εφαρμογή που έχει τη μορφή ερωτήσεων – απαντήσεων;	10
3) Ο συμβολισμός ενός αποτυχημένου αλλά κι ενός επιτυχημένου τεστ παραμένει πάντοτε ίδιος;	10
4) Οι εικόνες που έχουν κάποιες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο;	10
5) Η ονομασία μιας επιτυχημένης ή αποτυχημένης απάντησης σε ένα τεστ παραμένει ίδια για όλα τα τεστ;	10
6) Ο χαρακτηρισμός ενός επιτυχημένου ή αποτυχημένου τεστ παραμένει ίδιος για όλα τα τεστ;	10

Σύνολο ερωτήσεων: 7

Μέσος όρος απαντήσεων: 10

Όπως προκύπτει από το μέσο όρο των απαντήσεων πρόκειται για μια εξαιρετικά εύκολη στην εκμάθηση της εφαρμογής.

5.1.4.3.2. Ευκαμψία του συστήματος διεπαφής

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Μπορεί ο χρήστης να αλλάξει τον αριθμό των ερωτήσεων ενός τεστ;	0

Σύνολο ερωτήσεων: 1

Μέσος όρος απαντήσεων: 0

Βασίζομενοι στο μέσο όρο καταλαβαίνουμε ότι σίγουρα δεν πρόκειται για μια εύκαμπτη εφαρμογή.

5.1.4.3.3. Ανθεκτικότητα του συστήματος επαφής

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Έχει τη δυνατότητα ο χρήστης να αλλάξει την απάντηση σε ένα τεστ εφόσον δεν έχει ολοκληρωθεί;	10
2) Μπορεί ο χρήστης να δει συνολικά την πρόοδό του;	0
3) Μπορεί ο χρήστης να δει τις απαντήσεις που έχει δώσει σε ένα τεστ;	10
4) Μπορεί ο χρήστης να δει πόσες είναι οι σωστές και οι λάθος απαντήσεις;	10
5) Μπορεί ο χρήστης να δει αν ολοκλήρωσε με επιτυχία το τεστ μόλις το τελειώσει;	10

Σύνολο ερωτήσεων: 5

Μέσος όρος απαντήσεων: 8

Όσον αφορά την ανθεκτικότητα του συστήματος κρίνουμε ότι είναι πολύ καλή. Η εφαρμογή δίνει τη δυνατότητα στο χρήστη να αλλάξει όσες φορές θέλει μια απάντηση που έχει δώσει σε μια ερώτηση εφόσον το τεστ δεν έχει ολοκληρωθεί. Επίσης, μπορεί να δει τις απαντήσεις που έχει δώσει σε ένα τεστ (σωστές και λάθος) και φυσικά μόλις ολοκληρώσει το τεστ να ενημερωθεί για το αν πέτυχε ή όχι.

5.1.4.3.4. Ερωτήσεις γενικού περιεχομένου

Ερωτήσεις	Μέσος όρος απαντήσεων
1) Ο χρήστης πρέπει να θυμάται πολλά στοιχεία για να ολοκληρώσει μια ενέργεια;	10
2) Υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται;	10
3) Προκαλείται το ενδιαφέρον του χρήστη;	10
4) Χρησιμοποιούν οι χρήστες την εφαρμογή με ελάχιστη εκπαίδευση;	10
5) Οι χρήστες ζητούν βοήθεια για τη λειτουργία της εφαρμογής;	10
6) Το πρόγραμμα καλύπτει τις ανάγκες του τελικού χρήστη;	5,5

Σύνολο ερωτήσεων: 6

Μέσος όρος απαντήσεων: 9,2

Όπως προκύπτει και από αυτές τις ερωτήσεις υπάρχει συνέπεια στους όρους και στα σύμβολα που χρησιμοποιούνται κι έτσι ο χρήστης δεν μπερδεύεται ενώ ταυτόχρονα η εύκολη

δομή του τεστ αλλά και η μορφή του, προκαλεί το ενδιαφέρον του χρήστη. Επίσης σημαντικό είναι το γεγονός ότι ο χρήστης δε χρειάζεται καμία βοήθεια προκειμένου να ολοκληρώσει το τεστ κάτι που οφείλεται καθαρά στη μορφή και τη δομή του.

Σαν μειονέκτημα θα μπορούσαμε να δούμε εδώ ότι η εφαρμογή δεν καλύπτει όλες τις ανάγκες του χρήστη καθώς περιορίζεται στα βασικά.

5.2. ΣΥΓΚΡΙΣΗ ΠΡΟΓΡΑΜΜΑΤΩΝ

Προκειμένου να συγκρίνουμε τα τρία αυτά προγράμματα μεταξύ τους καλό θα ήταν να καταγράψουμε τα πλεονεκτήματα και τα μειονεκτήματα του καθενός κι έτσι η εξαγωγή του συμπεράσματος για το ποιο είναι το πιο κατάλληλο για την θεωρητική προετοιμασία των υποψηφίων οδηγών να γίνει πιο εύκολη.

5.2.1. ΠΡΟΓΡΑΜΜΑ ΥΠΟΥΡΓΕΙΟΥ ΜΕΤΑΦΟΡΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

A) Πλεονεκτήματα

- 1) **Ξένες γλώσσες.** Η εφαρμογή δίνει τη δυνατότητα στο χρήστη να κάνει το τεστ είτε στα ελληνικά είτε στα Αγγλικά, είτε στα Ρωσικά είτε στα Αλβανικά ανάλογα με την εθνικότητά του.
- 2) **Κατηγορίες διπλώματος.** Ένα επίσης σημαντικό πλεονέκτημα της εφαρμογής είναι ότι καλύπτει τις ανάγκες για όλες τις κατηγορίες διπλώματος.
- 3) **Τυχαία εμφάνιση ερωτήσεων στα τεστ.** Κάθε τεστ είναι διαφορετικό καθώς οι ερωτήσεις που εμφανίζονται δεν είναι ίδιες. Όταν ο χρήστης λοιπόν επιλέξει την κατηγορία του διπλώματος που τον ενδιαφέρει δημιουργείται και το τεστ κι έτσι κάθε φορά είναι διαφορετικό.
- 4) **Χρήση ίδιων συμβολισμών και όρων.** Προκειμένου η εφαρμογή να μη κουράζει τους χρήστες αλλά κυρίως να μην τους μπερδεύει χρησιμοποιούνται ίδιοι συμβολισμοί και όροι για να αναπαράσταθούν τα ίδια πράγματα.
- 5) **Εμφάνιση εικόνων ερωτήσεων.** Οι εικόνες που αντιστοιχούν σε ορισμένες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο κι έτσι οι χρήστες δεν μπερδεύονται.
- 6) **Συνεπής δομή και μορφή παραθύρων.** Τα κουμπιά που είναι κοινά στα παράθυρα έχουν πάντοτε την ίδια μορφή και εμφανίζονται στο ίδιο σημείο. Το ίδιο επίσης συμβαίνει και με τις ερωτήσεις και τις εικόνες που αντιστοιχούν σε αυτές.
- 7) **Βοήθεια.** Η εφαρμογή προκειμένου να γίνει κατανοητή η λειτουργία της εμπεριέχει βοήθεια.
- 8) **Ευκολία στη χρήση της εφαρμογής.** Η εφαρμογή μπορεί να χρησιμοποιηθεί από οποιονδήποτε χωρίς ιδιαίτερη εκπαίδευση.

A) Μειονεκτήματα

- 1) **Τύποι τεστ.** Το πρόγραμμα δεν έχει κάποιο διαχωρισμό των τεστ με βάση το βαθμό δυσκολίας κι έτσι ένας χρήστης δεν μπορεί να επιλέξει με αποτέλεσμα σε κάποιο τεστ να υπάρχει η πιθανότητα ύπαρξης πολλών δύσκολων ή εύκολων ερωτήσεων.
- 2) **Θεωρία.** Στην εφαρμογή δεν εμπεριέχεται η θεωρία πάνω στην οποία βασίζονται οι ερωτήσεις κι έτσι όταν ο χρήστης κάνει λάθος σε μια ερώτηση δεν μπορεί να ανατρέξει στην αντίστοιχη θεωρία για να βρει και τη σωστή απάντηση.

- 3) **Αλλαγή απάντησης.** Σε περίπτωση που ο χρήστης επιθυμεί να αλλάξει την απάντηση που έχει δώσει σε κάποια ερώτηση, δεν μπορεί να το κάνει.
- 4) **Δημιουργία χρηστών.** Δεν δίδεται δυνατότητα δημιουργίας χρήστη.
- 5) **Πρόοδος χρήστη.** Ο χρήστης δε μπορεί να παρακολουθήσει την πρόοδό του καθώς τα αποτελέσματα των τεστ που έχει κάνει δεν αποθηκεύονται κάπου.

5.2.2. ΠΡΟΓΡΑΜΜΑ ΙΣΤΟΣΕΛΙΔΑΣ: <http://kok.artinoi.gr>

A) Πλεονεκτήματα

- 1) **Κατηγορίες διπλώματος.** Η εφαρμογή εμπεριέχει τις δυο πιο σημαντικές κατηγορίες διπλώματος, αυτοκινήτου και μηχανής.
- 2) **Χρήση ίδιων συμβολισμών και όρων.** Σε όλη την εφαρμογή για να αναπαρασταθούν ίδια πράγματα, χρησιμοποιούνται οι ίδιοι συμβολισμοί και όροι.
- 3) **Εμφάνιση εικόνων ερωτήσεων.** Οι εικόνες που αντιστοιχούν σε ορισμένες ερωτήσεις εμφανίζονται πάντοτε στο ίδιο σημείο κι έτσι οι χρήστες δεν μπερδεύονται.
- 4) **Αλλαγή απάντησης.** Ο χρήστης έχει τη δυνατότητα να αλλάξει την απάντηση σε μια ερώτηση όσες φορές θέλει εφόσον δεν έχει ολοκληρώσει το τεστ.
- 5) **Ευκολία χρήσης της εφαρμογής.** Η εφαρμογή μπορεί να χρησιμοποιηθεί από οποιονδήποτε χρήστη καθώς είναι πολύ απλή και κατανοητή.

A) Μειονεκτήματα

- 1) **Συγκεκριμένος αριθμός τεστ.** Τα τεστ που είναι διαθέσιμα προς χρήση είναι συγκεκριμένα και δεν αλλάζουν.
- 2) **Τύποι τεστ.** Ο βαθμός δυσκολίας των τεστ δεν διευκρινίζεται κι έτσι ένας χρήστης δεν έχει τη δυνατότητα να επιλέξει ένα εύκολο ή ένα δύσκολο τεστ.
- 3) **Θεωρία.** Παρόλο που στην εφαρμογή όταν ο χρήστης έχει δώσει λάθος απάντηση σε μια ερώτηση, σημειώνεται και η σωστή, δεν έχει τη δυνατότητα να δει τη θεωρία που αντιστοιχεί σε αυτή την ερώτηση κι έτσι να κατανοήσει το λάθος του.
- 4) **Ξένες γλώσσες.** Δε δίδεται η δυνατότητα σε χρήστες που δεν γνωρίζουν ελληνικά να προχωρήσουν στην εκτέλεση των τεστ.
- 5) **Δημιουργία χρηστών.** Δεν μπορούν να δημιουργηθούν χρήστες.
- 6) **Πρόοδος χρήστη.** Η πρόοδος του χρήστη δεν αποθηκεύεται κι έτσι ο χρήστης δεν μπορεί να ξέρει τον συνολικό αριθμό των επιτυχημένων και αποτυχημένων τεστ.

5.2.3. ΠΡΟΓΡΑΜΜΑ «ΣΥΣΤΗΜΑ ΕΚΠΑΙΔΕΥΣΗΣ ΚΑΙ ΕΞΕΤΑΣΗΣ ΥΠΟΨΗΦΙΩΝ ΟΔΗΓΩΝ»

A) Πλεονεκτήματα

- 1) **Τυχαία εμφάνιση ερωτήσεων.** Κάθε φορά που ο χρήστης επιλέγει να κάνει ένα τεστ, αυτό δημιουργείται εκείνη τη στιγμή κι έτσι όλα τα τεστ είναι διαφορετικά.
- 2) **Τύποι τεστ.** Η εφαρμογή δίνει τη δυνατότητα στον χρήστη να επιλέξει έναν από τους τύπους τεστ που υπάρχουν. Ο χρήστης λοιπόν μπορεί να επιλέξει ανάμεσα σε εύκολα, δύσκολα ή μέτρια, με χρόνο ή χωρίς χρόνο.
- 3) **Χρήση ίδιων συμβολισμών και όρων.** Οι συμβολισμοί και οι όροι που χρησιμοποιούνται στην εφαρμογή είναι παντού ίδιοι εφόσον αναπαρίστανται τα ίδια πράγματα. Με τον τρόπο αυτό η εφαρμογή γίνεται πολύ φιλική προς τον χρήστη.

- 4) **Θεωρία.** Η εφαρμογή εμπεριέχει τη θεωρία στην οποία βασίζονται οι ερωτήσεις που χρησιμοποιούνται στα τεστ. Σε περίπτωση λοιπόν που ο χρήστης έχει δώσει λάθος απάντηση σε κάποια ερώτηση και θέλει να δει τη σωστή, το πρόγραμμα του δίνει αυτή τη δυνατότητα αρκεί να επιλέξει το κουμπί «θεωρία». Μόλις πατήσει το κουμπί θα εμφανιστεί η θεωρία στην οποία βασίζεται η ερώτηση.
- 5) **Διαχείριση θεωρίας.** Ο διαχειριστής του συστήματος έχει τη δυνατότητα να αλλάξει μια ήδη υπάρχουσα θεωρία αλλά και να προσθέσει μια νέα εάν είναι απαραίτητο.
- 6) **Διαχείριση ερωτήσεων.** Ο διαχειριστής του συστήματος μπορεί να αλλάξει μια ήδη υπάρχουσα ερώτηση ή και να προσθέσει μια καινούρια.
- 7) **Διαχείριση τύπου τεστ.** Ο διαχειριστής του συστήματος μπορεί να αλλάξει τον τύπο ενός τεστ αλλάζοντας τον αριθμό των δύσκολων, μέτριων και εύκολων ερωτήσεων που αυτός περιέχει αλλά και να δημιουργήσει ένα νέο τύπο.
- 8) **Αλλαγή απάντησης.** Εφόσον ο χρήστης δεν έχει ολοκληρώσει το τεστ μπορεί να αλλάξει την απάντηση που έχει δώσει σε μια ερώτηση όσες φορές θέλει.
- 9) **Δημιουργία χρηστών.** Η εφαρμογή δίνει τη δυνατότητα δημιουργίας χρηστών είτε αυτοί είναι απλοί χρήστες είτε διαχειριστές του συστήματος.
- 10) **Πρόοδος χρηστών.** Ένας χρήστης έχει τη δυνατότητα ανά πάσα στιγμή να ελέγξει την πρόοδο του. Ο διαχειριστής του συστήματος πάλι μπορεί να ελέγξει την πρόοδο όλων των χρηστών.
- 11) **Συνεπής μορφή και δομή παραθύρων.** Η μορφή και η δομή των παραθύρων είναι όμοια σε όλη την εφαρμογή, κάτι που δεν κουράζει αλλά κυρίως δεν μπερδεύει το χρήστη. Με λίγα λόγια είναι φιλική προς το χρήστη.
- 12) **Βοήθεια.** Η βοήθεια που εμπεριέχεται στην εφαρμογή κρίνεται απαραίτητη για τη διευκόλυνση των χρηστών στον τρόπο λειτουργίας της.
- 13) **Ευκολία στη χρήση της εφαρμογής.** Πρόκειται για μια αρκετά εύκολη στη χρήση της εφαρμογή.

A) Μειονεκτήματα

- 1) **Ξένες γλώσσες.** Χρήστες που δεν γνωρίζουν ελληνικά αλλά μένουν στην Ελλάδα και επιθυμούν να χρησιμοποιήσουν την εφαρμογή, δεν μπορούν.
- 2) **Κατηγορίες διπλώματος.** Η εφαρμογή δίνει τη δυνατότητα να χρησιμοποιηθεί μόνο από χρήστες που επιθυμούν να προετοιμαστούν για το δίπλωμα αυτοκινήτου.

5.4. ΣΥΜΠΕΡΑΣΜΑΤΑ ΣΥΓΚΡΙΣΗΣ

Εφόσον αναγνωρίσαμε τα κυριότερα πλεονεκτήματα και μειονεκτήματα των προγραμμάτων μπορούμε να βγάλουμε κάποια συμπεράσματα για το καθένα κι έτσι να κρίνουμε ποιο είναι πιο κατάλληλο ως πρόγραμμα.

Ένα κοινό πλεονέκτημα των προγραμμάτων είναι η χρήση ιδίων συμβολισμών και όρων. Πρόκειται για ένα πολύ σημαντικό πλεονέκτημα καθώς από αυτό κρίνεται γενικά ένα πρόγραμμα το πόσο φιλικό είναι προς το χρήστη.

Άλλο κοινό πλεονέκτημα των εφαρμογών που εξετάζουμε, είναι η ευκολία στη χρήση τους. Το τρίτο αλλά και τελευταίο κοινό πλεονέκτημα των εφαρμογών είναι η εμφάνιση των εικόνων των ερωτήσεων η οποία γίνεται πάντοτε στο ίδιο σημείο.

Στη συνέχεια και όσον αφορά τα υπόλοιπα χαρακτηριστικά των εφαρμογών, βλέπουμε ότι η χρήση ξένων γλωσσών γίνεται μόνο από την εφαρμογή του υπουργείου καθιστώντας με τον τρόπο αυτό τις άλλες δύο σε μειονεκτικότερη θέση καθώς στην εποχή μας το

χαρακτηριστικό αυτό κρίνεται απαραίτητο. Οι κατηγορίες διπλωμάτων οι οποίες εμφανίζονται στα προγράμματα του υπουργείου αλλά και της ιστοσελίδας δίνουν ένα επιπλέον προβάδισμα σε σύγκριση με την εφαρμογή μας στην οποία δεν υπάρχει τέτοια δυνατότητα.

Οι ερωτήσεις που χρησιμοποιούνται στα τεστ της εφαρμογής του υπουργείου αλλά και της δικής μας εμφανίζονται τυχαία σε αντίθεση με το πρόγραμμα της ιστοσελίδας στο οποίο η εμφάνιση των ερωτήσεων είναι συγκεκριμένη καθώς έχουν δημιουργηθεί συγκεκριμένα τεστ.

Η δομή αλλά και η μορφή των παραθύρων της εφαρμογής του υπουργείου αλλά και της δικής μας είναι συνεπής με αποτέλεσμα ο χρήστης να μη κουράζεται αλλά ούτε και να μπερδεύεται προκειμένου να κατανοήσει τη λειτουργία του κάθε παραθύρου. Ένα επίσης πολύ σημαντικό χαρακτηριστικό των δυο αυτών εφαρμογών είναι η βοήθεια που δίνουν στο χρήστη για τη λειτουργία τους.

Η δυνατότητα αλλαγής της απάντησης που έχει δώσει ο χρήστης σε μια ερώτηση, η οποία δίνεται στη δική μας εφαρμογή αλλά και σε αυτή της ιστοσελίδας πρόκειται για ένα πολύ σημαντικό χαρακτηριστικό των εφαρμογών αυτών καθώς δίνουν τη δυνατότητα στους χρήστες να αλλάξουν γνώμη σε περίπτωση που κάνουν λάθος.

Τα υπόλοιπα πλεονεκτήματα που καταγράφηκαν αφορούν μόνο τη δική μας εφαρμογή. Έτσι λοιπόν, η εφαρμογή δίνει τη δυνατότητα στους χρήστες να επιλέξουν τεστ με βάση το βαθμό δυσκολίας τους αλλά και να δημιουργήσουν και νέους τύπους τεστ. Η δυνατότητα διαχείρισης της θεωρίας και των ερωτήσεων είναι ένα πολύ σημαντικό πλεονέκτημα της εφαρμογής μας καθώς ακόμα και σε πιθανές αλλαγές του κ.ο.κ. η εφαρμογή μπορεί να προσαρμοστεί εύκολα. Τέλος, η δημιουργία χρηστών και η παρακολούθηση της προόδου τους δίνει τη δυνατότητα σωστής απόφασης για το αν οι χρήστες είναι πραγματικά έτοιμοι προκειμένου να δώσουν εξετάσεις.

Στον παρακάτω πίνακα φαίνονται συγκριτικά τα χαρακτηριστικά των τριών εφαρμογών.

Πρόγραμμα Υπουργείου Μεταφορών Και Επικοινωνιών	Πρόγραμμα Ιστοσελίδας: http://kok.artinoi.gr	Πρόγραμμα «Σύστημα Εκπαίδευσης Και Εξέτασης Υποψηφίων Οδηγών»
Πλεονεκτήματα		
Ξένες γλώσσες	Όχι	Όχι
Κατηγορίες διπλωμάτων	Ναι	Όχι
Τυχαία εμφάνιση ερωτήσεων	Όχι	Ναι
Χρήση ίδιων συμβόλων	Ναι	Ναι
Συνεπής δομή και μορφή παραθύρων	Όχι	Ναι
Βοήθεια	Όχι	Ναι
Ευκολία χρήσης	Ναι	Ναι
Εμφάνιση εικόνων ερωτήσεων	Ναι	Ναι
Όχι	Αλλαγή απάντησης	Ναι
Όχι	Όχι	Τύποι τεστ
Όχι	Όχι	Αλλαγή θεωρίας
Όχι	Όχι	Αλλαγή ερωτήσεων
Όχι	Όχι	Αλλαγή τύπου τεστ
Όχι	Όχι	Δημιουργία χρηστών
Όχι	Όχι	Πρόσδος χρηστών

Συμπερασματικά μπορούμε να πούμε ότι η εφαρμογή μας δίνει πολύ περισσότερες δυνατότητες στους χρήστες της σε σχέση με τις άλλες δύο. Η εφαρμογή μας θα μπορούσε να θεωρηθεί πιο ολοκληρωμένη αν προσθέταμε τη δυνατότητα χρήσης του προγράμματος και σε άλλες γλώσσες καθώς και τη δημιουργία κατηγοριών διπλώματος δίνοντας έτσι τη δυνατότητα και σε άλλους υποψηφίους οδηγούς να το χρησιμοποιήσουν.

Κ Ε Φ Α Λ Α Ι Ο 6

6. ΣΥΜΠΕΡΑΣΜΑΤΑ

Έχοντας ολοκληρώσει λοιπόν την ανάλυση τη σχεδίαση και την υλοποίηση του πληροφοριακού συστήματος, όπως επίσης και την αξιολόγησή του, μπορούμε να προβούμε σε διάφορα συμπεράσματα.

Οι δυνατότητες που δίνει η εφαρμογή είναι αρκετές και καλύπτουν ικανοποιητικά τις απαιτήσεις των χρηστών. Πρόκειται για τη μοναδική εφαρμογή θεωρητικής εκπαίδευσης υποψηφίων οδηγών αυτοκινήτου η οποία πέρα από τα τεστ που εμπεριέχει, περιλαμβάνει και τη θεωρία πάνω στην οποία βασίζονται οι ερωτήσεις. Είναι ένα από τα πολύ σημαντικά πλεονεκτήματα της εφαρμογής καθώς ο χρήστης μπορεί να μελετήσει τη θεωρία μέσα από την εφαρμογή. Επίσης, σε περίπτωση που δώσει λάθος απάντηση, η εφαρμογή του δίνει τη δυνατότητα να εμφανίσει το κομμάτι της θεωρίας στο οποίο αντιστοιχεί η ερώτηση που του έγινε.

Το γεγονός ότι η δομή και η μορφή των παραθύρων είναι όμοια, καθώς και το ότι χρησιμοποιούνται τα ίδια σύμβολα για κάποιες έννοιες συγκεκριμένες το κάνει φιλικό προς το χρήστη κάτι το οποίο φάνηκε και στην αξιολόγηση που έγινε.

Ένα άλλο επίσης σημαντικό στοιχείο της εφαρμογής είναι το ότι οι χρήστες μπορούν να δουν ανά πάσα στιγμή την πρόοδό τους. Με αυτό τον τρόπο μπορούν να κρίνουν το πότε θα είναι έτοιμοι να δώσουν τις θεωρητικές εξετάσεις.

Μέσα από την εφαρμογή ο administrator μπορεί να αλλάξει τη θεωρία αλλά και τις ερωτήσεις που θα περιλαμβάνονται στα τεστ. Σε περίπτωση λοιπόν που για κάποιο λόγο αλλάξουν κάποια κομμάτια του κ.ο.κ. ο διαχειριστής μπορεί να τα διαχειριστεί μέσα από το πρόγραμμα, όπως επίσης μπορεί να διαχειριστεί και τις ερωτήσεις που αφορούν τα κομμάτια αυτά της θεωρίας.

Αρχικά, το συγκεκριμένο λογισμικό προορίζεται για ακαδημαϊκή χρήση αλλά με κάποιες παρεμβάσεις θα μπορούσε να χρησιμοποιηθεί και για εμπορικούς σκοπούς.

Μια προσθήκη που θα μπορούσε να γίνει στο πρόγραμμα θα ήταν οι κατηγορίες διπλώματος. Η εφαρμογή, αυτή τη στιγμή μπορεί να εξυπηρετήσει μόνο υποψήφιους οδηγούς αυτοκινήτων. Θα μπορούσε όμως να χρησιμοποιηθεί και από τις υπόλοιπες κατηγορίες, για παράδειγμα φορτηγά, μοτοσικλέτες κ.λ.π. αρκεί να προσθέταμε τη θεωρία που αφορά αυτές τις κατηγορίες καθώς και τις αντίστοιχες ερωτήσεις.

Η δυνατότητα χρήσης της εφαρμογής και σε άλλες γλώσσες θα μπορούσε να κάνει την εφαρμογή μας πιο ολοκληρωμένη.

Αυτές είναι οι πιο σημαντικές και ίσως απαραίτητες επεκτάσεις που θα έπρεπε να γίνουν στην εφαρμογή. Στη συνέχεια θα μπορούσε να προστεθεί και η κατηγορία διπλώματος για ταχύπλοα και ιστιοφόρα. Όπως είναι κατανοητό θα έπρεπε να προστεθεί και η θεωρία αυτής της κατηγορίας και στη συνέχεια οι ερωτήσεις για τη δημιουργία των τεστ.

Επίσης, θα μπορούσε να φτιαχτεί και σε web based μορφή προκειμένου να είναι προσβάσιμη από περισσότερους χρήστες.

Τέλος, θα μπορούσε να αναπτυχθεί σε περιβάλλον android ή ios για χρήση σε tab τελευταίας τεχνολογίας.

Π Α Ρ Α Ρ Τ Η Μ Α Τ Α

Α. ΚΩΔΙΚΑΣ

Α.1. ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

Α.1.1. MODULES

MSGPROPERTIES

Κώδικας

'Module used to hold all message properties values

```
Public Const msg_header_cancel_init As String = "Ακύρωση εκκίνησης"
```

```
Public Const msg_header_system_error As String = "Σφάλμα εφαρμογής"
```

```
Public Const msg_header_system_info As String = "Ενημέρωση εφαρμογής"
```

```
Public Const msg_header_system_alert As String = "Επιλογή ενέργειας"
```

```
Public Const msg_frm_about As String = "Πληροφορίες"
```

```
Public Const msg_frm_about_text As String = "Drivers Test Copyright © 2009 - 2010 " + vbCrLf + vbCrLf + "Προσοχή: Το πρόγραμμα αυτό χρησιμοποιείται για εκπαιδευτικούς σκοπούς. Τυχόν αναπαραγωγή ή διανομή του λογισμικού, θα πρέπει να συνοδεύεται με έγγραφη συγκατάθεση του ιδιοκτήτη."
```

```
Public Const msg_text_cancel_init As String = "Εφαρμογή ήδη ανοιχτή!!!"
```

```
Public Const msg_text_terminate As String = "Επιθυμείτε τερματισμό της εφαρμογής?"
```

```
Public Const msg_init As String = "Έναρξη εφαρμογής..."
```

```
Public Const msg_init_prompt As String = "Η εφαρμογή δεν έχει εκκινήσει σωστά" + vbCrLf +
```

```
"Με την ενέργεια αυτή η εφαρμογή θα τερματιστεί." + vbCrLf + "Τερματισμός?"
```

```
Public Const msg_init_failure As String = "Σφάλμα κατά την έναρξη!!!" + vbCrLf + "Τερματισμός εφαρμογής."
```

```
Public Const msg_init_config As String = "Έλεγχος παραμέτρων εφαρμογής..."
```

```
Public Const msg_init_config_failure As String = "Σφάλμα κατά τον έλεγχο παραμέτρων εφαρμογής..."
```

```
Public Const msg_init_fs As String = "Έλεγχος αρχείων εφαρμογής..."
```

```
Public Const msg_init_fs_failure As String = "Σφάλμα κατά τον έλεγχο αρχείων εφαρμογής..."
```

```
Public Const msg_init_db As String = "Έλεγχος βάσης δεδομένων..."
```

```
Public Const msg_init_db_failure As String = "Σφάλμα κατά τον έλεγχο βάσης δεδομένων..."
```

```
Public Const msg_init_registry As String = "Έλεγχος μητρώου συστήματος..."
```

```
Public Const msg_init_registry_failure As String = "Σφάλμα κατά τον έλεγχο μητρώου συστήματος..."
```

```
Public Const msg_config_db_msaccess As String = "Έλεγχος αρχείου Βάσης Δεδομένων"
```

```
Public Const msg_config_db_msaccess_nofile As String = "Παρακαλώ επιλέξτε αρχείο Βάσης Δεδομένων"
```

```
Public Const msg_config_update As String = ""
```

Public Const msg_config_update_failure As String = "Σφάλμα κατά την ανανέωση των ρυθμίσεων της εφαρμογής." + vbCrLf + "Επαναλάβετε την ενέργεια, διαφορετικά επανεκκινήστε και προσπαθήστε ξανά."

Public Const frm_login As String = "Είσοδος χρήστη"
Public Const msg_login_failure As String = "Μη έγκυρος χρήστης" + vbCrLf + "Παρακαλώ προσπαθήστε ξανά"

Public Const msg_db_conn_failure As String = "Σφάλμα κατά την επικοινωνία με τη Βάση Δεδομένων" + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
Public Const msg_db_conn_warning As String = "Σφάλμα κατά τον τερματισμό επικοινωνίας με τη Βάση Δεδομένων." + vbCrLf + "Εάν το πρόβλημα επαναληφθεί, παρακαλώ επανεκκινήστε την εφαρμογή."
Public Const msg_db_execute_failure As String = "Σφάλμα κατά την διεκπεραίωση ενέργειας στη Βάση Δεδομένων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
Public Const msg_db_records_failure As String = "Σφάλμα κατά την προσπέλαση εγγραφών από τη Βάση Δεδομένων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
Public Const msg_db_records_warning As String = "Σφάλμα κατά τον καθαρισμό στιγμιότυπου εγγραφών από τη Βάση Δεδομένων." + vbCrLf + "Εάν το πρόβλημα επαναληφθεί, παρακαλώ επανεκκινήστε την εφαρμογή."
Public Const msg_db_general_no_records As String = "Δεν βρέθηκαν εγγραφές"
Public Const msg_db_no_record_correlation As String = "Δεν ήταν δυνατή η πρόσβαση στις πληροφορίες που ζητήσατε." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."

Public Const frm_userMgt As String = "Διαχείριση χρηστών"
Public Const msg_user_duplicate_check_error As String = "Το συνθηματικό χρήστη που επιλέξατε είναι σε χρήση." + vbCrLf + "Παρακαλώ επιλέξτε νέο και προσπαθήστε ξανά."
Public Const msg_user_create_check_error As String = "Παρακαλώ συμπληρώστε τα απαιτούμενα πεδία και προσπαθήστε ξανά."
Public Const msg_user_update_check_error As String = "Παρακαλώ συμπληρώστε τα απαιτούμενα πεδία και προσπαθήστε ξανά."
Public Const msg_user_delete_alert As String = "Επιθυμείτε διαγραφή του χρήστη?" + vbCrLf + "Κατά την ενέργεια αυτή θα διαγραφεί και το σχετικό ιστορικό του χρήστη."
Public Const msg_user_delete_cancel As String = "Ο χρήστης που επιθυμείτε να διαγραφεί είναι ενεργός." + vbCrLf + "Ακύρωση διαγραφής!!!"

Public Const frm_testMgtQ As String = "Διαχείριση ερωτήσεων"
Public Const msg_question_create_check_error As String = msg_user_create_check_error
Public Const msg_question_update_check_error As String = msg_user_update_check_error
Public Const msg_question_delete_alert As String = "Επιθυμείτε διαγραφή της ερώτησης?" + vbCrLf + "Κατά την ενέργεια αυτή θα διαγραφεί και όλο το σχετικό ιστορικό της ερώτησης."
Public Const msg_answer_create_check_error As String = "Παρακαλώ συμπληρώστε τα απαιτούμενα πεδία και προσπαθήστε ξανά." + vbCrLf + "Επίσης ενδέχεται να υπάρχει ήδη σωστή απάντηση για τη συγκεκριμένη ερώτηση."
Public Const msg_answer_delete_alert As String = "Επιθυμείτε διαγραφή της απάντησης?" + vbCrLf + "Κατά την ενέργεια αυτή θα διαγραφεί και όλο το σχετικό ιστορικό της απάντησης."
Public Const msg_answer_update_check_error As String = msg_answer_create_check_error
Public Const msg_question_duplicate_check_error As String = "Ο τίτλος που επιλέξατε είναι σε χρήση." + vbCrLf + "Παρακαλώ επιλέξτε νέο και προσπαθήστε ξανά."
Public Const msg_qimage_duplicate_check_error As String = "Η εικόνα που επιλέξατε είναι ήδη καταχωρημένη." + vbCrLf + "Παρακαλώ επιλέξτε νέα και προσπαθήστε ξανά."
Public Const msg_answer_limit_reached_error As String = "Δεν μπορούν να προστεθούν άλλες ερωτήσεις στην λίστα"
Public Const msg_image_limit_reached_error As String = "Δεν μπορούν να προστεθούν άλλες εικόνες στην λίστα."
Public Const msg_answer_image_browse As String = "Επιλογή εικόνας"

Public Const msg_no_correct_answer_alert As String = "Δεν υπάρχει σωστή απάντηση για την συγκεκριμένη ερώτηση." + vbCrLf + "Συνέχεια?"

Public Const frm_testMgtC As String = "Διαχείριση κατηγοριών"
 Public Const frm_testMgtC_timelimit_label As String = "Διάρκεια τεστ"
 Public Const frm_testMgtC_timelimit_nolimit As String = "Μη καθορισμένη"
 Public Const frm_testMgtC_timelimit_unit As String = "λεπτά"
 Public Const frm_testMgtC_num_msg As String = "Αριθμός Ερωτήσεων:"
 Public Const frm_testMgtC_num_pass_msg As String = "Επιτυχείς ερωτήσεις:"
 Public Const msg_type_duplicate_check_error As String = "Το όνομα κατηγορίας που επιλέξατε είναι σε χρήση." + vbCrLf + "Παρακαλώ επιλέξτε νέο και προσπαθήστε ξανά."
 Public Const msg_type_create_check_error As String = msg_user_create_check_error
 Public Const msg_type_update_check_error As String = msg_user_update_check_error
 Public Const msg_type_delete_alert As String = "Επιθυμείτε διαγραφή της κατηγορίας?" + vbCrLf + "Κατά την ενέργεια αυτή θα διαγραφεί και όλο το σχετικό ιστορικό της κατηγορίας."

Public Const frm_testRun As String = "Τεστ"
 Public Const frm_testRun_Exit As String = "Το τεστ δεν έχει ολοκληρωθεί!!!" + vbCrLf + "Επιθυμείτε έξοδο?"

Public Const frm_testRunPause_message As String = "Προσωρινή διακοπή τεστ..."
 Public Const frm_testRunFinish_message As String = "Υπάρχουν αναπάντητες ερωτήσεις!!!" + vbCrLf + "Επιθυμείτε ολοκλήρωση?"

Public Const frm_testRunCreateFailed_message As String = "Αποτυχία δημιουργίας λίστας ερωτήσεων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
 Public Const frm_testRunPrepareFailed_message As String = "Αποτυχία προετοιμασίας φόρμας ερωτήσεων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
 Public Const frm_testRunQuestionNotFound_message As String = "Αποτυχία εύρεσης ερώτησης."
 Public Const frm_testRunPleaseWait_message As String = "Λίστα ερωτήσεων σε επεξεργασία..."
 Public Const frm_testRunRetrievingAnswers_message As String = "Προσπέλαση απαντήσεων..."
 Public Const frm_testRunCalculatingResults_message As String = "Υπολογισμός αποτελεσμάτων..."
 Public Const frm_testRunLoggingResults_message As String = "Αποθήκευση αποτελεσμάτων..."
 Public Const frm_testRunResultsHeader_message As String = "Αποτελέσματα τεστ"
 Public Const frm_testRunResultsPrefixSuccess_message As String = "Συγχαρητήρια, επιτύχατε στο τεστ!!!" + vbCrLf
 Public Const frm_testRunResultsPrefixFailure_message As String = "Δυστυχώς αποτύχατε στο τεστ!!!" + vbCrLf
 Public Const frm_testRunResultsPrefixBreakdown_message As String = "Αποτελέσματα:" + vbCrLf
 Public Const frm_testRunResultsPrefixBreakdownPercent_message As String = vbCrLf + "Ποσοστό επιτυχίας:"
 Public Const frm_testRunResultsPrefixBreakdownPercentPass_message As String = vbCrLf + "Όριο:"
 Public Const frm_testRunResultsPrefix_message As String = vbCrLf + vbCrLf + "Για περισσότερες πληροφορίες χρησιμοποιήστε από το μενού την επιλογή 'Ιστορικό'."
 Public Const frm_testRunLogResultsFailed As String = "Αποτυχία αποθήκευσης αποτελεσμάτων."

Public Const frm_TestReports_admin_message As String = "Ιστορικό Χρηστών"
 Public Const frm_TestReports_user_message As String = "Ιστορικό"
 Public Const frm_TestReports_analysis_message As String = "Κάντε διπλό κλικ στο τεστ της επιλογής σας για περισσότερες λεπτομέρειες."


```

Public Const frm_TestReportsBD_message As String = "Τεστ"
Public Const frm_TestReportsBDPleaseWait_message As String = "Λίστα απαντήσεων σε
επεξεργασία..."
Public Const frm_TestReportsBDCreateFailed_message As String = "Αποτυχία προσπέλασης
λίστας απαντήσεων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
Public Const frm_TestReportsBDPrepareFailed_message As String = "Αποτυχία προβολής
λίστας απαντήσεων." + vbCrLf + "Παρακαλώ προσπαθήστε ξανά."
Public Const frm_TestReportsBDQuestionNotFound_message As String = "Αποτυχία εύρεσης
ερώτησης."
Public Const frm_TestReportsBDTheoryRefNotFound_message As String = "Αποτυχία εύρεσης
σχετικής θεωρίας."

Public Const frm_TheoryMgt As String = "Διαχείριση θεωρίας"
Public Const msg_theory_duplicate_check_error As String = "Ο τίτλος που επιλέξατε είναι σε
χρήση." + vbCrLf + "Παρακαλώ επιλέξτε νέο και προσπαθήστε ξανά."
Public Const msg_theory_create_check_error As String = "Παρακαλώ συμπληρώστε τα
απαιτούμενα πεδία και προσπαθήστε ξανά."
Public Const msg_theory_update_check_error As String = "Παρακαλώ συμπληρώστε τα
απαιτούμενα πεδία και προσπαθήστε ξανά."
Public Const msg_theory_file_browse As String = "Επιλογή αρχείου θεωρίας"
Public Const msg_theory_delete_alert As String = "Επιθυμείτε διαγραφή της σχετικής
θεωρίας?" + vbCrLf + "Μετά την ενέργεια αυτή η θεωρεία δεν θα είναι πλέον προσβάσιμη."

Public Const msg_GeneralDocNotFound_message As String = "Αποτυχία εύρεσης αρχείου."

```

MsgSUBMAIN

Κώδικας

```

Option Explicit
'Initialisations are located at this sub procedure

Public Sub Main()

'Detect if application is already open
If App.PreviousInstance = True Then
    Call infoMSG(msg_text_cancel_init, msg_header_cancel_init)

    'Terminate application instance
    End
End If

frmSplash.Show

End Sub

```

REGISTRYMODULE**Κώδικας**

```
'Module used for checking/updating OS registry
'with required controls
.....
.....
```

```
Public Function checkRegistry(formObject As Form)
```

```
    Dim isValid As Boolean
    isValid = True
```

```
    'Update status label
    formObject.lblSplashStatus.Caption = msg_init_registry
```

```
    'TODO: implementation of registry check
    .....
    .....
```

```
    'Check if valid to set progress bar value
    If (isValid) Then
        formObject.pgbSplashStatus.Value = _
            formObject.pgbSplashStatus.Value + 1
    End If
```

```
    checkRegistry = isValid
```

```
End Function
```

APPLICATIONPROPERTIES**Κώδικας**

```
'Module used to hold general application property values
.....
.....
```

```
'Configuration file
Public Const configFile As String = "/settings.ini"
Public Const configContents As String = "db_path = " + vbCrLf + "db_type = msaccess" +
vbCrLf + "timer_on = true" + vbCrLf + "max_q = 50" + vbCrLf + "images_path = /images" +
vbCrLf
```

```
Public Const configContentsNew As String = "db_path = #####" + vbCrLf + "db_type = #####"
+ vbCrLf + "timer_on = #####" + vbCrLf + "max_q = #####" + vbCrLf + "images_path = #####"
+ vbCrLf
```

```
Public Const maxAllowedQImages As Integer = 1
Public Const maxAllowedQAnswers As Integer = 4
```

MsgAccessSQL**Κώδικας**

```
'Module used to hold all ms access sql statements used
Public Const msaccess_sql_test_1 As String = "select answer_id, answer_description,
answer_question_id, answer_question_id" + " FROM tbl_answers"
Public Const msaccess_sql_test_2 As String = "select question_id, question_title,
question_description, question_theory_id, question_category_id, question_single_answer,
question_images, question_active, question_extra, question_created_ts" + " FROM
tbl_questions"
Public Const msaccess_sql_test_3 As String = "select testlog_entry_id,
testlog_entry_testlog_id, testlog_entry_question_id, testlog_entry_answer_id,
testlog_entry_answer_correct_id" + " FROM tbl_testlog_entries"
Public Const msaccess_sql_test_4 As String = "select testlog_id, testlog_user_id,
testlog_user_id, testlog_time_taken, testlog_passed, testlog_created_ts" + " FROM tbl_testlogs"
Public Const msaccess_sql_test_5 As String = "select theory_id, theory_title,
theory_description, theory_doc, theory_extra, theory_created_ts" + " FROM tbl_theory"
Public Const msaccess_sql_test_6 As String = "select type_id, type_name, type_description,
type_questions_num, type_questions_num_pass, type_categories_mix, type_active,
type_time_limit, type_extra, type_created_ts" + " FROM tbl_types"
Public Const msaccess_sql_test_7 As String = "select user_id, user_name, user_password,
user_isAdmin, user_fname, user_lname, user_extra, user_created_ts" + " FROM tbl_users"
'Public Const msaccess_sql_test_8 As String = "select category_id, category_name,
category_extra, category_created_ts" + " FROM tbl_categories"
```

PUBLIC VARIABLES**Κώδικας**

```
'Module used to hold all public variables used in application
.....
.....

Public Const config_num As Integer = 5
Public config_db_path As String
Public config_db_type As String
Public config_timer_on As Boolean
Public config_max_q As Integer
Public config_images_path As String

'User variables used to hold session info
Public user_name As String
Public user_fname As String
Public user_type As String
Public user_id As Integer

'Default user account
Public Const def_user_name As String = "letmein"
Public Const def_user_pwd As String = "letmein"
Public Const def_user_fname As String = "ἡρώδης Ἀούάάβὰδ"

'Data paths
Public Const config_path_imgs As String = "\images"
Public Const config_path_docs As String = "\docs"
```

```
'Help file name
Public Const config_help_filename As String = "DriversTest.chm"
```

MsgBoxMODULE

Κώδικας

```
'Module used to hold public message box functions
.....
.....
```

```
Public Sub errorMsg(message As String, header As String)
```

```
    msg = MsgBox(message, vbOKOnly & vbCritical, header)
```

```
End Sub
```

```
Public Function alertMSG(message As String, header As String)
```

```
    alertMSG = MsgBox(message, vbQuestion & vbOKCancel, header)
```

```
End Function
```

```
Public Sub infoMSG(message As String, header As String)
```

```
    msg = MsgBox(message, vbInformation & vbOKOnly, header)
```

```
End Sub
```

FILESYSTEMMODULE

Κώδικας

```
'Module used to hold methods for checking/accessing
'machine's file system
.....
.....
```

```
Option Explicit
```

```
Const configFileBak As String = configFile + "_bak"
```

```
Public Function getPathRelFile(strFilename)
```

```
    getPathRelFile = App.Path + strFilename
```

```
End Function
```

```
Public Function checkFS(formObject As Form)
```

```
    On Error GoTo CheckFSError
```

```

Dim isValid As Boolean
isValid = True

'Update status label
formObject.lblSplashStatus.Caption = msg_init_fs

'PERFORM ALL FS CHECKS HERE
.....

'Check DB file
isValid = fileExists(config_db_path)

'Check Images directory
If (Not dirExists(getPathRelFile(config_images_path))) Then
    Mkdir (App.Path + "images")
End If

.....

'Check if valid to set progress bar value
If (isValid) Then
    formObject.pgbSplashStatus.Value = _
        formObject.pgbSplashStatus.Value + 1
End If

checkFS = isValid

Exit Function

CheckFSError:
checkFS = False

End Function

Public Function checkConfig(formObject As Form)

    Dim isValid As Boolean
    isValid = False

    'Update status label
    formObject.lblSplashStatus.Caption = msg_init_config

    'Check if file exists
    If (fileExists(getPathRelFile(configFile))) Then
        isValid = loadConfig(configFile)
    Else
        'Load default config
        isValid = saveConfig(configContents)
    End If

    'Check if valid to set progress bar value
    If (isValid) Then
        formObject.pgbSplashStatus.Value = _
            formObject.pgbSplashStatus.Value + 1
    End If

```

```
    checkConfig = isValid  
End Function
```

```
Public Function fileExists(strFilenamePath As String) As Boolean
```

```
    Dim isFile As Boolean  
    Dim tmpFile As String  
    isFile = False
```

```
    tmpFile = Dir(strFilenamePath)
```

```
    If (tmpFile <> "") Then isFile = True
```

```
    fileExists = isFile
```

```
End Function
```

```
Public Function dirExists(strDirectoryPath As String) As Boolean
```

```
    Dim isDirectory As Boolean  
    Dim tmpDir As String  
    isDirectory = False
```

```
    tmpDir = Dir(strDirectoryPath, vbDirectory)
```

```
    If (tmpDir <> "") Then isDirectory = True
```

```
    dirExists = isDirectory
```

```
End Function
```

```
Public Function getConfigArray()
```

```
    Dim configArray(0 To 4) As String
```

```
    configArray(0) = config_db_path  
    configArray(1) = config_db_type  
    configArray(2) = config_timer_on  
    configArray(3) = config_max_q  
    configArray(4) = config_images_path
```

```
    getConfigArray = configArray
```

```
End Function
```

```
Public Function formConfigContents(ByRef strSampleConfigContents, configArray() As String)
```

```
    On Error GoTo FormConfigError
```

```
    Dim isFormed As Boolean  
    Dim configArraySize As Integer  
    Dim Counter As Integer
```

```
    isFormed = False  
    configArraySize = config_num - 1
```

```
    'Check that configuration is valid
```

```

If (UBound(configArray) = configArraySize) Then
  'Start updating configuration contents
  For Counter = 0 To UBound(configArray) Step 1
    strSampleConfigContents = _
      Replace(strSampleConfigContents, "#####", configArray(Counter), 1, 1)
  Next Counter

  isFormed = True
End If

formConfigContents = isFormed

Exit Function

FormConfigError:
  formConfigContents = False

End Function

Public Function saveConfig(strConfigContents)

  On Error GoTo ErrorSaveConfig

  Dim isSaved As Boolean
  isSaved = False

  'Open and save to backup file
  Open getPathRelFile(configFileBak) For Output As #1

  Print #1, strConfigContents

  Close #1

  'Replace original (if exists) with backup
  If (fileExists(getPathRelFile(configFile))) Then
    'Delete original
    Kill getPathRelFile(configFile)
  End If

  'Rename backup to original filename
  FileCopy getPathRelFile(configFileBak), _
    getPathRelFile(configFile)

  'Clean by deleting backup file
  Kill getPathRelFile(configFileBak)

  'Load config
  isSaved = loadConfig(configFile)

  saveConfig = isSaved

  Exit Function

ErrorSaveConfig:
  saveConfig = False

End Function

```

```
Public Function loadConfig(strConfigFile)
```

```
    On Error GoTo ErrorLoadConfig
```

```
    Dim isLoaded As Boolean
    Dim strTMP As String
    Dim strPropertiesArray() As String
    Dim boolArrayDimensioned As Boolean
    isLoaded = False
    boolArrayDimensioned = False
```

```
    Open getPathRelFile(strConfigFile) For Input As #2
```

```
    Seek #2, 1
```

```
    'Iterate and load every line
```

```
    While Not EOF(2)
```

```
        'Redimension array
```

```
        If (boolArrayDimensioned) Then
```

```
            ReDim Preserve strPropertiesArray(0 To UBound(strPropertiesArray) + 1) As String
```

```
        Else
```

```
            ReDim strPropertiesArray(0 To 0) As String
```

```
            boolArrayDimensioned = True
```

```
        End If
```

```
        Line Input #2, strTMP
```

```
        'Check if line not empty and then add to properties array
```

```
        If (Trim(strTMP) <> "") Then
```

```
            strPropertiesArray(UBound(strPropertiesArray)) = Trim(strTMP)
```

```
        End If
```

```
    Wend
```

```
    Close #2
```

```
    'Validate configuration loading
```

```
    isLoaded = validateConfig(strPropertiesArray)
```

```
    'Erase array
```

```
    Erase strPropertiesArray
```

```
    loadConfig = isLoaded
```

```
    Exit Function
```

```
ErrorLoadConfig:
```

```
    loadConfig = False
```

```
End Function
```

```
Public Function validateConfig(strConfigArray() As String)
```

```
    On Error GoTo ErrorValidateConfig
```

```
    Dim isValidConfig As Boolean
```

```
    isValidConfig = True
```



```
'DB_PATH CHECK
If (InStr(1, strConfigArray(0), "db_path") > 0) Then
    config_db_path = _
        Trim(Replace(strConfigArray(0), "db_path =", ""))
Else
    isValidConfig = False
End If

'DB_TYPE CHECK
If (InStr(1, strConfigArray(1), "db_type") > 0) Then
    config_db_type = _
        Trim(Replace(strConfigArray(1), "db_type =", ""))
Else
    isValidConfig = False
End If

'TIMER_ON CHECK
If (InStr(1, strConfigArray(2), "timer_on") > 0) Then
    config_timer_on = _
        CBool(Trim(Replace(strConfigArray(2), "timer_on =", "")))
Else
    isValidConfig = False
End If

'MAXIMUM QUESTIONS CHECK
If (InStr(1, strConfigArray(3), "max_q") > 0) Then
    config_max_q = _
        CInt(Trim(Replace(strConfigArray(3), "max_q =", "")))
Else
    isValidConfig = False
End If

'IMAGES_PATH CHECK
If (InStr(1, strConfigArray(4), "images_path") > 0) Then
    config_images_path = _
        Trim(Replace(strConfigArray(4), "images_path =", ""))
Else
    isValidConfig = False
End If

validateConfig = isValidConfig

Exit Function

ErrorValidateConfig:
validateConfig = False

End Function
```

DBMODULE**Κώδικας**

'Module used for holding methods used to access DB

Option Explicit

Public Function checkDB(formObject As Form)

Dim isValid As Boolean
 isValid = False

'Update status label
 formObject.lblSplashStatus.Caption = msg_init_db

'Check if db has been set. If not popup DB config
 If (Trim(config_db_path) = "" And Trim(config_db_type) = "msaccess") Then
 frmRegDBMsaccess.Show vbModal
 ElseIf (Trim(config_db_type) <> "msaccess") Then 'Force MSAccess DB type
 config_db_type = "msaccess"
 frmRegDBMsaccess.Show vbModal
 ElseIf (Trim(config_db_type) = "msaccess" And Not fileExists(config_db_path)) Then
 frmRegDBMsaccess.Show vbModal
 End If

'Check if proper configuration preformed and also db properly access
 Select Case config_db_type
 Case "msaccess"
 If (Trim(config_db_path) <> "") Then
 'Run SQL tests
 isValid = testMSAccess(msaccess_sql_test_1, msaccess_sql_test_2, _
 msaccess_sql_test_3, msaccess_sql_test_4, _
 msaccess_sql_test_5, msaccess_sql_test_6, _
 msaccess_sql_test_7)
 End If
 End Select

'Check if valid to set progress bar value
 If (isValid) Then
 formObject.pgbSplashStatus.Value = _
 formObject.pgbSplashStatus.Value + 1
 End If

checkDB = isValid

End Function

Public Function testMSAccess(ParamArray msaccessSQLArray() As Variant)

Dim testPassed As Boolean
 Dim Counter As Integer
 testPassed = True

'Go through params and check SQL queries
 For Counter = 0 To UBound(msaccessSQLArray)

```

    If (Not testMSAccessSQL(msaccessSQLArray(Counter))) Then
        testPassed = False
        Exit For 'On test error break immediately
    End If
Next Counter

testMSAccess = testPassed

End Function

Public Function testMSAccessSQL(strSQL)

    On Error GoTo MSAccessSQLError

    Dim dbConn As ADODB.Connection
    Dim rsTemp As ADODB.Recordset
    Dim sqlTestPassed As Boolean
    sqlTestPassed = True

    'Open db connection
    Set dbConn = New ADODB.Connection
    dbConn.Open getMSAccessConnString

    'Open recordset
    Set rsTemp = New ADODB.Recordset
    rsTemp.Open strSQL, dbConn, adOpenStatic, adLockReadOnly, adCmdText
    'Debug.Print "SQL: " + strSQL + vbCrLf
    'Debug.Print "Record Count: " + CStr(rsTemp.RecordCount)

    'Close recordset
    rsTemp.Close

    'Close db connection
    dbConn.Close

    testMSAccessSQL = sqlTestPassed

    Exit Function

MSAccessSQLError:
    testMSAccessSQL = False

End Function

Public Function openDBConnection(ByRef dbConn As ADODB.Connection, strDBType As
String) As Boolean

    On Error GoTo openDBConnectionError

    Set dbConn = New ADODB.Connection

    Select Case strDBType
        Case "msaccess"
            dbConn.Open getMSAccessConnString
    End Select

    openDBConnection = True

```

Exit Function

```
openDBConnectionError:  
    openDBConnection = False
```

End Function

```
Public Function getDBRecordset(ByRef dbConn As ADODB.Connection, ByRef dbRecs As  
ADODB.Recordset, strSQL As String) As Boolean
```

```
    On Error GoTo getDBRecordsetError
```

```
    Set dbRecs = New ADODB.Recordset
```

```
    dbRecs.Open strSQL, dbConn, adOpenStatic, adLockReadOnly, adCmdText
```

```
    getDBRecordset = True
```

Exit Function

```
getDBRecordsetError:  
    getDBRecordset = False
```

End Function

```
Public Function executeDBSQLQuery(ByRef dbConn As ADODB.Connection, ByRef dbRecs  
As ADODB.Recordset, strSQL As String) As Boolean
```

```
    Dim querySuccess As Boolean  
    Dim lngRecordsAffected As Long
```

```
    querySuccess = False
```

```
    On Error GoTo executeDBSQLQueryError
```

```
    Set dbRecs = dbConn.Execute(strSQL, lngRecordsAffected)
```

```
    If lngRecordsAffected > 0 Then querySuccess = True
```

```
    executeDBSQLQuery = querySuccess
```

Exit Function

```
executeDBSQLQueryError:  
    executeDBSQLQuery = False
```

End Function

```
Public Function getDBRecorsetCount(ByRef dbRecs As ADODB.Recordset, strDBType As  
String) As Integer
```

```
    Dim recordsCounter As Integer
```

```
    On Error GoTo getDBRecorsetCountError
```

```
    Select Case strDBType
```

```
Case "msaccess"
    recordsCounter = dbRecs.RecordCount
End Select

getDBRecorsetCount = recordsCounter

Exit Function

getDBRecorsetCountError:
    getDBRecorsetCount = 0

End Function

Public Function getDBRecorsetAffectedStatus(ByRef dbRecs As ADODB.Recordset, strDBType
As String) As Boolean

    Dim recordsAffected As Boolean
    recordsAffected = False

    On Error GoTo getDBRecorsetAffectedStatusError

    Select Case strDBType
        Case "msaccess"
            If dbRecs.EditMode = adEditNone Then recordsAffected = True
    End Select

    getDBRecorsetAffectedStatus = recordsAffected

    Exit Function

getDBRecorsetAffectedStatusError:
    getDBRecorsetAffectedStatus = False

End Function

Public Function closeDBConnection(ByRef dbConn As ADODB.Connection) As Boolean

    On Error GoTo closeDBConnectionError

    If dbConn.State Then dbConn.Close

    closeDBConnection = True

    Exit Function

closeDBConnectionError:
    closeDBConnection = False

End Function

Public Function closeDBRecordset(ByRef dbRecs As ADODB.Recordset) As Boolean

    On Error GoTo closeDBRecordsetError

    If dbRecs.State Then dbRecs.Close

    closeDBRecordset = True
```

Exit Function

```
closeDBRecordsetError:  
    closeDBRecordset = False
```

End Function

```
Public Function getMSAccessConnString() As String
```

'Function that generates and returns connection string

```
getMSAccessConnString = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" _  
    & config_db_path _  
    & ";Jet OLEDB:Database Password=letmein;" 'HARDCODED MSACCESS DB PASSWORD
```

End Function

```
Public Function performDBAction(ByRef dbConnection As ADODB.Connection, ByRef  
dbRecords As ADODB.Recordset, strSQLQuery As String) As Boolean
```

```
    Dim actionSuccess As Boolean  
    actionSuccess = False
```

```
    If (openDBConnection(dbConnection, config_db_type)) Then  
        If (executeDBSQLQuery(dbConnection, dbRecords, strSQLQuery)) Then  
            actionSuccess = True
```

```
                If (closeDBRecordset(dbRecords)) Then  
                    If (Not closeDBConnection(dbConnection)) Then  
                        Call infoMSG(msg_db_conn_warning, msg_header_system_info)  
                    End If
```

```
                Else  
                    Call infoMSG(msg_db_records_warning, msg_header_system_info)  
                End If
```

```
            Else  
                Call errorMSG(msg_db_execute_failure, msg_header_system_error)  
            End If
```

```
        Else  
            Call errorMSG(msg_db_conn_failure, msg_header_system_error)  
        End If
```

```
        performDBAction = actionSuccess
```

End Function

A.1.2. ΦΟΡΜΕΣ**FRMSPLASH****Κώδικας**

Option Explicit

Dim checkStep As Integer

Private Sub resetForm()

'Reset progress bar
pgbSplashStatus.Max = 4
pgbSplashStatus.Min = 0
pgbSplashStatus.Value = 0

'Reset status label
lblSplashStatus.Caption = ""

End Sub

Private Sub Form_Load()

'Clear form
resetForm

checkStep = 1
lblSplashStatus.Caption = msg_init

'Perform checks
'If (Not checkRegistry(Me)) Then
'Elseif (Not checkConfig(Me)) Then
'Elseif (Not checkDB(Me)) Then
'Elseif (Not checkFS(Me)) Then
'End If

End Sub

Private Sub tmrLoader_Timer()

Select Case checkStep

Case 1:

If (Not checkRegistry(Me)) Then
Call errorMsg(msg_init_registry_failure, msg_header_system_error)

'Terminate application

End

End If

Case 2:

If (Not checkConfig(Me)) Then
Call errorMsg(msg_init_config_failure, msg_header_system_error)

'Terminate application

End

End If

Case 3:

If (Not checkDB(Me)) Then

```
        Call errorMsg(msg_init_db_failure, msg_header_system_error)

        'Terminate application
        End
    End If
Case 4:
    If (Not checkFS(Me)) Then
        Call errorMsg(msg_init_fs_failure, msg_header_system_error)

        'Terminate application
        End
    End If

Case 5: 'Step used to load main form
    MDIMainForm.Show

    'Unload splash form
    Unload Me
End Select

'Increment check step
checkStep = checkStep + 1

DoEvents

End Sub
```

FRMLOGIN

Κώδικας

```
Option Explicit
Dim dbConnection As ADODB.Connection
Dim dbRecords As ADODB.Recordset
Dim strSQLQuery As String

Private Sub resetForm()

    txtUsername.Text = ""
    txtPwd.Text = ""

End Sub

Private Sub cmdCancel_Click()

    'Set bypass value
    Call MDIMainForm.setLoginBypass(True)

    'Close login popup
    Unload Me

End Sub

Private Sub cmdOK_Click()
```



```

If checkLogin Then
    'Close popup form
    Unload Me
Else
    Call errorMsg(msg_login_failure, msg_header_system_error)
    Call Form_Activate
End If

```

```
End Sub
```

```
Private Sub Form_Activate()
```

```
    'Reset form
    resetForm
```

```
    txtUsername.SetFocus
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
    'Set form's caption
    Me.Caption = frm_login
```

```
End Sub
```

```
Private Function checkLogin()
```

```
    Dim isValidLogin As Boolean
    Dim strUsername As String
    Dim strUserpwd As String
```

```
    isValidLogin = False
```

```
    strUsername = Trim(txtUsername.Text)
    strUserpwd = Trim(txtPwd.Text)
```

```
    'First check if default login
```

```
    If strUsername = def_user_name And strUserpwd = def_user_pwd Then
```

```
        'Set user session variables
```

```
        user_name = def_user_name
```

```
        user_fname = def_user_fname
```

```
        user_type = "default" 'Hardcoded since default user type
```

```
        user_id = 0 'Hardcoded since default user id
```

```
        'Set return value to true
```

```
        isValidLogin = True
```

```
    Else
```

```
        'Set sql query string
```

```
        strSQLQuery = "select * from tbl_users " _
            + "where user_name = '" + strUsername + "' " _
            + "and user_password = '" + strUserpwd + "'" _
```

```
        If (openDBConnection(dbConnection, config_db_type)) Then
```

```
            If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
```

```
                'Check if records exist
```

```
                If (getDBRecorsetCount(dbRecords, config_db_type) > 0) Then
```

```

'Set user session info
user_name = dbRecords("user_name")

user_fname = dbRecords("user_fname") + " " _
            + dbRecords("user_lname")

If (CBool(dbRecords("user_isAdmin"))) Then
    user_type = "admin"
Else
    user_type = "user"
End If

user_id = CInt(dbRecords("user_id"))

'DEBUG
'Debug.Print "### DEBUG USER SESSION START ###"
'Debug.Print "user_name: " + user_name
'Debug.Print "user_fname: " + user_fname
'Debug.Print "user_type: " + user_type
'Debug.Print "user_id: " + CStr(user_id)
'Debug.Print "### DEBUG USER SESSION END ###"

'Set validate flag to true
isValidLogin = True
End If

If (closeDBRecordset(dbRecords)) Then
    If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
    End If
Else
    Call infoMSG(msg_db_records_warning, msg_header_system_info)
End If
Else
    Call errorMSG(msg_db_records_failure, msg_header_system_error)
End If
Else
    Call errorMSG(msg_db_conn_failure, msg_header_system_error)
End If
End If

checkLogin = isValidLogin

End Function

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)

    If (UnloadMode = vbFormControlMenu) Then
        Cancel = 1
        Call cmdCancel_Click
    End If

End Sub

```

MDIMAINFORM**Κώδικας**

Option Explicit

Private boolLoginBypass As Boolean

```
Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hWnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long
```

```
Public Sub setLoginBypass(boolValue)
```

```
    boolLoginBypass = boolValue
```

```
End Sub
```

```
Private Sub clearUserSession()
```

```
    user_name = ""
    user_fname = ""
    user_type = ""
    user_id = 0
```

```
    'Reset login bypass value
    Call setLoginBypass(False)
```

```
End Sub
```

```
Private Sub resetMenus()
```

```
    mnuGeneral.Visible = True
    mnuGeneral.Enabled = True
    mnuGeneralLogin.Visible = True
    mnuGeneralLogin.Enabled = True
    mnuGeneralLogout.Visible = True
    mnuGeneralLogout.Enabled = True
    mnuGeneralExit.Visible = True
    mnuGeneralExit.Enabled = True
```

```
    mnuTests.Visible = True
    mnuTests.Enabled = True
    mnuTestsRun.Visible = True
    mnuTestsRun.Enabled = True
    mnuTestsReports.Visible = True
    mnuTestsReports.Enabled = True
    mnuTestsReportsAll.Visible = True
    mnuTestsReportsAll.Enabled = True
```

```
    mnuAdmin.Visible = True
    mnuAdmin.Enabled = True
    mnuAdminUsers.Visible = True
    mnuAdminUsers.Enabled = True
    mnuAdminTests.Visible = True
    mnuAdminTests.Enabled = True
    mnuAdminTheory.Visible = True
    mnuAdminTheory.Enabled = True
```

```
mnuApplication.Visible = True
mnuApplication.Enabled = True
mnuApplicationHelp.Visible = True
mnuApplicationHelp.Enabled = True
mnuApplicationAbout.Visible = True
mnuApplicationAbout.Enabled = True
```

```
stbMain.Panels(1).Text = ""
```

```
End Sub
```

```
Private Sub setMenus()
```

```
'First reset menus
resetMenus
```

```
Select Case user_type
```

```
Case "default"
```

```
    mnuGeneralLogin.Visible = False
```

```
    mnuTests.Enabled = False
```

```
    mnuAdminTests.Enabled = False
```

```
    mnuAdminTheory.Enabled = False
```

```
Case "admin"
```

```
    mnuGeneralLogin.Visible = False
```

```
    mnuTestsRun.Visible = False
```

```
    mnuTestsReports.Visible = False
```

```
Case "user"
```

```
    mnuGeneralLogin.Visible = False
```

```
    mnuTestsReportsAll.Visible = False
```

```
    mnuAdmin.Enabled = False
```

```
Case Else
```

```
    mnuGeneralLogout.Visible = False
```

```
    mnuTests.Enabled = False
```

```
    mnuAdmin.Enabled = False
```

```
End Select
```

```
stbMain.Panels(1).Text = user_fname
```

```
End Sub
```

```
Private Sub MDIForm_Activate()
```

```
'Check if user logged in and if not popup login form
```

```
If Trim(user_name) = "" And Not boolLoginBypass Then frmLogin.Show vbModal
```

```
'Set menus
```

```
Call setMenus
```

```
End Sub
```

```
Private Sub MDIForm_Load()
```

```
    'Set Main form caption  
    Me.Caption = App.Title + " v" _  
        + CStr(App.Major) _  
        + "." + CStr(App.Minor) _  
        + "." + CStr(App.Revision)
```

```
End Sub
```

```
Private Sub MDIForm_QueryUnload(Cancel As Integer, UnloadMode As Integer)
```

```
    'Inform user that is about to terminate application  
    If alertMSG(msg_text_terminate, msg_header_system_info) = vbOK Then  
        'Terminate application  
        End  
    Else  
        'Cancel termination  
        Cancel = 1  
    End If
```

```
End Sub
```

```
Private Sub mnuAdminTestsCategories_Click()
```

```
    frmTestMgtCat.Show vbModal
```

```
End Sub
```

```
Private Sub mnuAdminTestsQuestions_Click()
```

```
    frmTestMgt.Show vbModal
```

```
End Sub
```

```
Private Sub mnuAdminTheory_Click()
```

```
    frmTheoryMgt.Show vbModal
```

```
End Sub
```

```
Private Sub mnuAdminUsers_Click()
```

```
    frmUserMgt.Show vbModal
```

```
End Sub
```

```
Private Sub mnuApplicationAbout_Click()
```

```
    frmAbout.Show vbModal
```

```
End Sub
```

```
Private Sub mnuApplicationHelp_Click()
```

```
    Dim strHelpDocName As String
```

```
'Get full path
strHelpDocName = getPathRelFile("\\" + config_help_filename)

'Check if file exists and if not show error
If fileExists(strHelpDocName) Then
    'Open file
    ShellExecute hWnd, "open", strHelpDocName, vbNullString, vbNullString, 1
Else
    Call errorMsg(msg_GeneralDocNotFound_message, msg_header_system_error)
End If

End Sub

Private Sub mnuGeneralExit_Click()

    DoEvents

    'Terminate application
    Unload Me

End Sub

Private Sub mnuGeneralLogin_Click()

    frmLogin.Show vbModal

End Sub

Private Sub mnuGeneralLogout_Click()

    'Clear user session info
    Call clearUserSession

    'Reset menus
    Call resetMenus

    'Refresh MDI form
    Call MDIForm_Activate

End Sub

Private Sub mnuTestsReports_Click()

    frmTestReports.Show vbModal

End Sub

Private Sub mnuTestsReportsAll_Click()

    frmTestReports.Show vbModal

End Sub

Private Sub mnuTestsRun_Click()

    frmTestRun.Show vbModal
```

End Sub

FRMTESTRUN

Κώδικας

```

Option Explicit
Private boolTestInProgress As Boolean
Public boolFinishTest As Boolean
Dim dblSeconds As Double
Dim dblSecondsTotal As Double
Dim dbConnection As ADODB.Connection
Dim dbRecords As ADODB.Recordset
Dim strSQLQuery As String
Dim catDifficultyIndexArray() As String
Dim catDetails(0 To 2) As Integer
Dim qEasyArray() As Integer
Dim qMediumArray() As Integer
Dim qHardArray() As Integer
Dim qTestArray() As Integer
Dim qAnswerNumArray() As Integer
Dim randSessionArray() As Integer
Dim intCurrentQuestionIndex As Integer
Private Const prefixFrame As String = "frQ"
Private Const prefixImage As String = "imgQ"
Private Const prefixDescription As String = "txtDQ"
Private Const prefixAnswer As String = "txtAQ"
Private Const prefixAnswerID As String = "lblAIDQ"
Private Const prefixOption As String = "optQ"
Dim finalResultsArray(0 To 2) As Integer

Private Sub cmbCategory_Click()

    cmbCategoryID.ListIndex = cmbCategory.ListIndex
    Call GetCategoryDetails(CInt(cmbCategoryID.Text))
    cmdRun.Enabled = True

End Sub

Private Sub cmdExit_Click()

    Unload Me

End Sub

Private Sub cmdFinish_Click()

    Call FinishTest

End Sub

Private Sub cmdNext_Click()

```

```

If intCurrentQuestionIndex < catDetails(1) Then
  'Set current question index and display current frame
  Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = False
  intCurrentQuestionIndex = intCurrentQuestionIndex + 1
  Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

  'Check if last question to show finish button
  If intCurrentQuestionIndex = catDetails(1) Then
    cmdFinish.Visible = True
  Else
    cmdFinish.Visible = False
  End If
End If

End Sub

Private Sub cmdPause_Click()

  If tmrCountdown.Enabled Then frmTestRunPause.Show vbModal

End Sub

Private Sub cmdPrevious_Click()

  'Always hide finish button
  cmdFinish.Visible = False

  If intCurrentQuestionIndex < 2 Then Exit Sub

  'Set current question index and display current frame
  Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = False
  intCurrentQuestionIndex = intCurrentQuestionIndex - 1
  Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

End Sub

Private Sub cmdRun_Click()

  'Prepare question list and display question area
  If getQTestArray(qEasyArray, qMediumArray, qHardArray, qTestArray) Then
    'Redimension answer array
    ReDim qAnswerNumArray(0 To UBound(qTestArray)) As Integer

    If Not PrepareQuestions(qTestArray) Then
      Call errorMsg(frm_testRunPrepareFailed_message, msg_header_system_error)
      Exit Sub
    End If
  Else
    Call errorMsg(frm_testRunCreateFailed_message, msg_header_system_error)
    Exit Sub
  End If

  'Reset current question index and display current frame
  intCurrentQuestionIndex = 1
  Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

```



```
'Disable run command button
cmdRun.Enabled = False

'Disable category combo box
cmbCategory.Enabled = False

'Disable check box
chkTimerActive.Enabled = False

'Hide pause button (if timer active button will be activated in following lines)
cmdPause.Visible = False

'Hide finish buttons
cmdFinish.Visible = False

'ONLY TO TEST TIMER
frQuestions.Visible = True

'Start countdown if selected
If (chkTimerActive.Value = vbChecked) Then
    lblCountDown.Caption = GetTimestamp
    lblCountDown.Visible = True
    tmrCountdown.Enabled = True
    cmdPause.Visible = True
End If

boolTestInProgress = True

End Sub

Private Sub Form_Load()

    Me.Caption = frm_testRun
    boolTestInProgress = False
    Call HideQuestionArea
    Call ClearCategoryDetails
    Call PrepareCategoryCombo

End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)

    If boolTestInProgress Then
        'Stop timer
        tmrCountdown.Enabled = False

        'Alert user that test is in progress
        If alertMSG(frm_testRun_Exit, msg_header_system_info) = vbCancel Then
            Cancel = 1

            'Start timer (if active)
            If chkTimerActive.Value = vbChecked Then tmrCountdown.Enabled = True
        End If
    End If

End Sub

End Sub
```

```
Private Sub Form_Activate()
```

```
    'Activate timer if checked on form activation  
    If boolTestInProgress And chkTimerActive.Value = vbChecked Then tmrCountdown.Enabled  
    = True 'START timer
```

```
End Sub
```

```
Private Sub Form_Deactivate()
```

```
    If boolTestInProgress And chkTimerActive.Value = vbChecked Then tmrCountdown.Enabled  
    = False 'STOP timer
```

```
End Sub
```

```
Private Function GetTimestamp() As String
```

```
    Dim intHours, intMinutes, intSeconds As Integer
```

```
    If dblSeconds = 0 Then  
        GetTimestamp = "00:00:00"  
        Exit Function  
    End If
```

```
    intHours = Fix(dblSeconds / 3600)  
    intMinutes = Fix((dblSeconds - CDbI(intHours * 3600)) / 60)  
    intSeconds = dblSeconds - (intHours * 3600) - (intMinutes * 60)
```

```
    GetTimestamp = Format(intHours, "00") _  
        + ":" + Format(intMinutes, "00") _  
        + ":" + Format(intSeconds, "00")
```

```
End Function
```

```
Private Sub HideQuestionArea()
```

```
    frQuestions.Visible = False  
    lblCountDown.Caption = ""  
    lblCountDown.Visible = False  
    lblCountDown.ForeColor = &H80000012
```

```
End Sub
```

```
Private Sub ClearCategoryDetails()
```

```
    chkTimerActive.Value = vbUnchecked  
    chkTimerActive.Enabled = False  
    lblCategoryDetails.Caption = ""
```

```
End Sub
```

```
Private Sub PrepareCategoryCombo()
```

```
    'Set sql query string  
    strSQLQuery = "select * from tbl_types order by type_id asc"
```

```

If (openDBConnection(dbConnection, config_db_type)) Then
  If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
    'Clear combos
    cmbCategoryID.Clear
    cmbCategory.Clear

    'Check if records exist
    If (getDBRecorsetCount(dbRecords, config_db_type) > 0) Then

      'Move recordset cursor to first element
      dbRecords.MoveFirst

      'Iterate through records and add to combo
      While Not dbRecords.EOF
        cmbCategoryID.AddItem CStr(dbRecords("type_id"))
        cmbCategory.AddItem CStr(dbRecords("type_name"))

        'Move cursor to next record
        dbRecords.MoveNext
      Wend
    End If

    If (closeDBRecordset(dbRecords)) Then
      If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
      End If
    Else
      Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If
  Else
    Call errorMSG(msg_db_records_failure, msg_header_system_error)
  End If
Else
  Call errorMSG(msg_db_conn_failure, msg_header_system_error)
End If

cmdRun.Enabled = False

cmbCategoryID.Enabled = False
cmbCategory.Enabled = True

'Reset category combo selections
cmbCategoryID.ListIndex = -1
cmbCategory.ListIndex = -1

End Sub

Private Sub GetCategoryDetails(intCategoryID As Integer)

  'Set sql query string
  strSQLQuery = "select * from tbl_types where type_id = " + CStr(intCategoryID)

  If (openDBConnection(dbConnection, config_db_type)) Then
    If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
      'Check if records exist
      If (getDBRecorsetCount(dbRecords, config_db_type) > 0) Then
        'Move recordset cursor to first element

```

```

dbRecords.MoveFirst

Call ClearCategoryDetails

catDifficultyIndexArray = Split(dbRecords("type_categories_mix"), "/")
catDetails(0) = dbRecords("type_id")
catDetails(1) = dbRecords("type_questions_num")
catDetails(2) = dbRecords("type_questions_num_pass")

'NOTE: No need to iterate, since only one record should be returned
Dim strTimeLimit As String
strTimeLimit = frm_testMgtC_timelimit_nolimit

If dbRecords("type_time_limit") > 0 Then
    chkTimerActive.Enabled = True
    chkTimerActive.Value = vbChecked
    strTimeLimit = CStr(dbRecords("type_time_limit")) + " " +
frm_testMgtC_timelimit_unit
End If

lblCategoryDetails.Caption = frm_testMgtC_timelimit_label + ": " + strTimeLimit _
    + vbCrLf + Trim(dbRecords("type_description"))

dblSeconds = dbRecords("type_time_limit") * 60
dblSecondsTotal = dblSeconds
Else
    'Alert that no record was found and return to questions list
    Call errorMsg(msg_db_no_record_correlation, msg_header_system_error)
End If

If (closeDBRecordset(dbRecords)) Then
    If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
    End If
Else
    Call infoMSG(msg_db_records_warning, msg_header_system_info)
End If
Else
    Call errorMsg(msg_db_records_failure, msg_header_system_error)
End If
Else
    Call errorMsg(msg_db_conn_failure, msg_header_system_error)
End If

End Sub

Private Sub tmrCountdown_Timer()

If dblSeconds = 0 Then
    'STOP timer
    tmrCountdown.Enabled = False

    'Set test in progress to false
    boolTestInProgress = False

    'Invoke finish test method
    Call FinishTest

```

```

Exit Sub
End If

'Subtract 1 second
dblSeconds = dblSeconds - 1

'Make countdown label red to alert
If dblSeconds < 60 Then lblCountDown.ForeColor = &HFF&

'Refresh countdown area
lblCountDown.Caption = GetTimestamp

End Sub

Private Function getQArray(qArray() As Integer, categoryID As Integer) As Boolean
On Error GoTo GetQArrayError

'Reset question array
ReDim qArray(0 To 0) As Integer

'Set sql query string
strSQLQuery = "select * from tbl_questions where question_active = true and
question_category_id = " _
+ CStr(categoryID) + " order by question_id asc"

If (openDBConnection(dbConnection, config_db_type)) Then
If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
'Check if records exist
If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
'Redimension question array
ReDim qArray(0 To (getDBRecordsetCount(dbRecords, config_db_type) - 1)) As
Integer

'Move recordset cursor to first element
dbRecords.MoveFirst

Dim intCounter As Integer
intCounter = 0

'Iterate through records and add to array
While Not dbRecords.EOF
qArray(intCounter) = dbRecords("question_id")

'Move cursor to next record
dbRecords.MoveNext
intCounter = intCounter + 1
Wend
End If

If (closeDBRecordset(dbRecords)) Then
If (Not closeDBConnection(dbConnection)) Then
Call infoMSG(msg_db_conn_warning, msg_header_system_info)
End If
Else
Call infoMSG(msg_db_records_warning, msg_header_system_info)
End If

```

```

Else
    Call errorMsg(msg_db_records_failure, msg_header_system_error)
    getQArray = False
    Exit Function
End If
Else
    Call errorMsg(msg_db_conn_failure, msg_header_system_error)
    getQArray = False
    Exit Function
End If

getQArray = True
Exit Function

GetQArrayError:
    Call errorMsg(Err.Description, msg_header_system_error)
    getQArray = False

End Function

Private Function getQTestArray(easyArray() As Integer, mediumArray() As Integer, hardArray()
As Integer, testArray() As Integer) As Boolean
    On Error GoTo GetQTestArrayError

    'All three category types mix numbers should be present in array
    If UBound(catDifficultyIndexArray) <> 2 Then
        getQTestArray = False
        Exit Function
    Else
        'Get easy questions array only if needed
        If CInt(catDifficultyIndexArray(0)) > 0 Then
            If Not getQArray(easyArray, 1) Then
                getQTestArray = False
                Exit Function
            End If
        End If

        'Get medium questions array only if needed
        If CInt(catDifficultyIndexArray(1)) > 0 Then
            If Not getQArray(mediumArray, 2) Then
                getQTestArray = False
                Exit Function
            End If
        End If

        'Get hard questions array only if needed
        If CInt(catDifficultyIndexArray(2)) > 0 Then
            If Not getQArray(hardArray, 3) Then
                getQTestArray = False
                Exit Function
            End If
        End If

        'Redimension test array
        ReDim testArray(0 To catDetails(1) - 1, 0 To 2) As Integer
    End Function

```

```

Dim intQCounter, intStartQCounter As Integer
Dim intRandomNumber As Integer
Dim strDebug As String

intQCounter = 1
intStartQCounter = 0
'START FORMING QUESTION LIST

'Check if easy questions are to be added
If CInt(catDifficultyIndexArray(0)) > 0 Then
'Reset random number session array
ReDim randSessionArray(0 To 0) As Integer

strDebug = ""

For intQCounter = 1 To CInt(catDifficultyIndexArray(0))
'Get random number
intRandomNumber = Rand(1, CLng(UBound(easyArray) + 1),
CInt(catDifficultyIndexArray(0)))

'Check if valid number retrieved and if not end process
If intRandomNumber = 0 Then GoTo GetQTestArrayPrematureEnd

'If valid get value from array and add it to current entry in test array
testArray(intQCounter + intStartQCounter - 1, 0) = easyArray(intRandomNumber - 1)
strDebug = strDebug + CStr(testArray(intQCounter + intStartQCounter - 1, 0)) + ","
Next

intStartQCounter = intStartQCounter + intQCounter - 1

Debug.Print "--Easy list: " + strDebug
End If

'Check if medium questions are to be added
If CInt(catDifficultyIndexArray(1)) > 0 Then
'Reset random number session array
ReDim randSessionArray(0 To 0) As Integer

strDebug = ""

For intQCounter = 1 To CInt(catDifficultyIndexArray(1))
'Get random number
intRandomNumber = Rand(1, CLng(UBound(mediumArray) + 1),
CInt(catDifficultyIndexArray(1)))

'Check if valid number retrieved and if not end process
If intRandomNumber = 0 Then GoTo GetQTestArrayPrematureEnd

'If valid get value from array and add it to current entry in test array
testArray(intQCounter + intStartQCounter - 1, 0) = mediumArray(intRandomNumber - 1)
strDebug = strDebug + CStr(testArray(intQCounter + intStartQCounter - 1, 0)) + ","
Next

intStartQCounter = intStartQCounter + intQCounter - 1

Debug.Print "--Medium list: " + strDebug
End If

```

```

'Check if hard questions are to be added
If CInt(catDifficultyIndexArray(2)) > 0 Then
    'Reset random number session array
    ReDim randSessionArray(0 To 0) As Integer

    strDebug = ""

    For intQCounter = 1 To CInt(catDifficultyIndexArray(2))
        'Get random number
        intRandomNumber = Rand(1, CLng(UBound(hardArray) + 1),
        CInt(catDifficultyIndexArray(2)))

        'Check if valid number retrieved and if not end process
        If intRandomNumber = 0 Then GoTo GetQTestArrayPrematureEnd

        'If valid get value from array and add it to current entry in test array
        testArray(intQCounter + intStartQCounter - 1, 0) = hardArray(intRandomNumber - 1)
        strDebug = strDebug + CStr(testArray(intQCounter + intStartQCounter - 1, 0)) + ","
    Next

    Debug.Print "--Hard list: " + strDebug
End If

getQTestArray = True
Exit Function

GetQTestArrayPrematureEnd:
getQTestArray = False
Exit Function

GetQTestArrayError:
Call errorMsg(Err.Description, msg_header_system_error)
getQTestArray = False

End Function

Public Function Rand(ByVal Low As Long, ByVal High As Long, intTotalNumsToDeliver As
Integer) As Long

    On Error GoTo RandError

    Dim lngLow, lngHigh As Long
    Dim intRandom As Integer

    lngLow = Low
    lngHigh = High

    'Not allowed zero values for total numbers
    If intTotalNumsToDeliver < 1 Then
        Rand = 0
        Exit Function
    End If

    If lngLow <> lngHigh Then
        If lngLow > lngHigh Then
            lngLow = High

```



```

    lngHigh = Low
End If

Dim intRetriesCounter As Integer
intRetriesCounter = 0

Do
    Randomize
    intRandom = Int((lngHigh - lngLow + 1) * Rnd) + lngLow
    intRetriesCounter = intRetriesCounter + 1

    If intRetriesCounter > intTotalNumsToDeliver Then
        intRandom = 0
        Exit Do
    End If
Loop While Not IsNumberNew(intRandom, randSessionArray)

    Rand = intRandom
Else
    Rand = 0
End If

Exit Function

RandError:
    Call errorMSG(Err.Description, msg_header_system_error)
    Rand = 0

End Function

Public Function IsNumberNew(numToCheck As Integer, sessionArray() As Integer) As Boolean

    Dim isNew As Boolean
    Dim intCounter As Integer

    isNew = True

    'Check if new exists
    For intCounter = 0 To UBound(sessionArray)
        If sessionArray(intCounter) = numToCheck Then
            isNew = False
            Exit For
        End If
    Next

    'Add new entry if new
    If isNew Then
        If UBound(sessionArray) = 0 And sessionArray(0) = 0 Then
            sessionArray(0) = numToCheck
        Else
            ReDim Preserve sessionArray(0 To UBound(sessionArray) + 1) As Integer
            sessionArray(UBound(sessionArray)) = numToCheck
        End If
    End If

    IsNumberNew = isNew

```

End Function

Private Function PrepareQuestions(questionsArray() As Integer) As Boolean

On Error GoTo PrepareQuestionsError

Dim intQuestionsCounter As Integer

PrepareQuestions = False

'Proceed only if questions exist
If UBound(questionsArray) = 0 Then
 PrepareQuestions = False
 Exit Function
End If

'Start process
IblPleaseWait.Visible = True
IblPleaseWait.Caption = frm_testRunPleaseWait_message
Me.MousePointer = vbHourglass
DoEvents

For intQuestionsCounter = 0 To UBound(questionsArray)
 If Not RenderNewQuestion(intQuestionsCounter, questionsArray) Then
 PrepareQuestions = False
 Exit Function
 End If
Next

'Stop process
IblPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

PrepareQuestions = True

Exit Function

PrepareQuestionsError:
IblPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

Call errorMsg(Err.Description, msg_header_system_error)
PrepareQuestions = False

End Function

Private Function RenderNewQuestion(intQIndex As Integer, qArray() As Integer) As Boolean

Dim ctrlframe As Control

'Set sql query string
strSQLQuery = "select * from tbl_questions, tbl_answers where question_id =
answer_question_id and question_id = " + CStr(qArray(intQIndex, 0))

If (openDBConnection(dbConnection, config_db_type)) Then

```

If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
    'Check if records exist
    If (getDBRecorsetCount(dbRecords, config_db_type) > 0) Then
        'Move recordset cursor to first element
        dbRecords.MoveFirst

        'Set question frame
        Set ctlframe = frmTestRun.Controls.Add("VB.Frame", prefixFrame + "_" +
CStr(intQIndex + 1), frQuestions)
        ctlframe.Visible = False
        ctlframe.Top = frQ.Top
        ctlframe.Left = frQ.Left
        ctlframe.Width = frQ.Width
        ctlframe.Height = frQ.Height
        ctlframe.Caption = CStr(intQIndex + 1) + " / " + CStr(UBound(qArray) + 1)

        'Set image control (if not null)
        If Trim(dbRecords("question_images")) <> "" Then
            Call SetImageControl(intQIndex, dbRecords("question_images"), ctlframe)
        End If

        'Set description control
        Call SetDescriptionControl(intQIndex, dbRecords("question_description"), ctlframe)

        Dim intAnswerNum As Integer
        intAnswerNum = 1

        'Iterate through records and add controls
        While Not dbRecords.EOF
            'Set question controls
            Call SetOptionControls(intQIndex, dbRecords("answer_description"),
dbRecords("answer_id"), intAnswerNum, ctlframe)

            'Check if correct answer and if yes add it to question array
            If dbRecords("answer_correct") Then
                qArray(intQIndex, 2) = dbRecords("answer_id")
            End If

            'Move cursor to next record
            dbRecords.MoveNext
            intAnswerNum = intAnswerNum + 1
        Wend

        'Set total answers number for question
        qAnswerNumArray(intQIndex) = intAnswerNum - 1
    Else
        Call infoMSG(frm_testRunQuestionNotFound_message, msg_header_system_info)
        RenderNewQuestion = False
        Exit Function
    End If

    If (closeDBRecordset(dbRecords)) Then
        If (Not closeDBConnection(dbConnection)) Then
            Call infoMSG(msg_db_conn_warning, msg_header_system_info)
        End If
    Else
        Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If

```

```

        End If
    Else
        Call errorMsg(msg_db_records_failure, msg_header_system_error)
        RenderNewQuestion = False
        Exit Function
    End If
Else
    Call errorMsg(msg_db_conn_failure, msg_header_system_error)
    RenderNewQuestion = False
    Exit Function
End If

RenderNewQuestion = True

End Function

Private Sub SetImageControl(intIndex As Integer, strImageName As String, frFrame As Control)

    Dim ctlImage As Control

    Set ctlImage = frmTestRun.Controls.Add("VB.Image", prefixImage + "_" + CStr(intIndex + 1),
frFrame)
    ctlImage.Top = imgQ.Top
    ctlImage.Left = imgQ.Left
    ctlImage.Width = imgQ.Width
    ctlImage.Height = imgQ.Height
    ctlImage.Stretch = imgQ.Stretch

    strImageName = Trim(Replace(strImageName, ";", ""))
    Set ctlImage.Picture = LoadPicture(App.Path + config_images_path + config_path_imgs + "\"
+ strImageName)

    ctlImage.Visible = True

End Sub

Private Sub SetDescriptionControl(intIndex As Integer, strDescription As String, frFrame As
Control)

    Dim ctlDescription As Control

    Set ctlDescription = frmTestRun.Controls.Add("VB.Label", prefixDescription + "_" +
CStr(intIndex + 1), frFrame)
    ctlDescription.Top = txtDQ.Top
    ctlDescription.Left = txtDQ.Left
    ctlDescription.Width = txtDQ.Width
    ctlDescription.Height = txtDQ.Height
    ctlDescription.Font = txtDQ.Font
    ctlDescription.BorderStyle = txtDQ.BorderStyle
    ctlDescription.BackStyle = txtDQ.BackStyle
    ctlDescription.Caption = strDescription
    ctlDescription.Visible = True

End Sub

Private Sub SetOptionControls(intIndex As Integer, strAnswer As String, intAnswerID As
Integer, intAnswerNum As Integer, frFrame As Control)

```

```

Dim ctlOption, ctlOptionSample, ctlAnswer, ctlAnswerID, ctlAnswerSample As Control

Select Case intAnswerNum
    Case 1
        Set ctlOptionSample = optQ1
        Set ctlAnswerSample = txtAQ1
    Case 2
        Set ctlOptionSample = optQ2
        Set ctlAnswerSample = txtAQ2
    Case 3
        Set ctlOptionSample = optQ3
        Set ctlAnswerSample = txtAQ3
    Case 4
        Set ctlOptionSample = optQ4
        Set ctlAnswerSample = txtAQ4
    Case 5
        Set ctlOptionSample = optQ5
        Set ctlAnswerSample = txtAQ5
End Select

Set ctlOption = frmTestRun.Controls.Add("VB.OptionButton", prefixOption +
CStr(intAnswerNum) + "_" + CStr(intIndex + 1), frFrame)
ctlOption.Caption = ctlOptionSample.Caption
ctlOption.Top = ctlOptionSample.Top
ctlOption.Left = ctlOptionSample.Left
ctlOption.Width = ctlOptionSample.Width
ctlOption.Height = ctlOptionSample.Height
ctlOption.Visible = True
Set ctlAnswer = frmTestRun.Controls.Add("VB.Label", prefixAnswer + CStr(intAnswerNum) +
"_" + CStr(intIndex + 1), frFrame)
ctlAnswer.Top = ctlAnswerSample.Top
ctlAnswer.Left = ctlAnswerSample.Left
ctlAnswer.Width = ctlAnswerSample.Width
ctlAnswer.Height = ctlAnswerSample.Height
ctlAnswer.Font = ctlAnswerSample.Font
ctlAnswer.BorderStyle = ctlAnswerSample.BorderStyle
ctlAnswer.BackStyle = ctlAnswerSample.BackStyle
ctlAnswer.Caption = strAnswer
ctlAnswer.Visible = True
'Hidden control to hold answer id
Set ctlAnswerID = frmTestRun.Controls.Add("VB.Label", prefixAnswerID +
CStr(intAnswerNum) + "_" + CStr(intIndex + 1), frFrame)
ctlAnswerID.Caption = CStr(intAnswerID)
ctlAnswerID.Visible = False

End Sub

Private Function GetAnswers(questionArray() As Integer, answerNumArray() As Integer) As
Boolean

    On Error GoTo GetAnswersError

    Dim intArrayCounter, intArrayCounter2 As Integer

    'Check that question and answer array have the same size
    If UBound(questionArray) <> UBound(answerNumArray) Then

```

```

    GetAnswers = False
    Exit Function
End If

'Start process
IbIPleaseWait.Visible = True
IbIPleaseWait.Caption = frm_testRunRetrievingAnswers_message
Me.MousePointer = vbHourglass
DoEvents

For intArrayCounter = 0 To UBound(questionArray)
    'Set -1 as default value
    questionArray(intArrayCounter, 1) = -1

    'Iterate through answers and overwrite -1 value with actual answer id
    For intArrayCounter2 = 1 To answerNumArray(intArrayCounter)
        If Me.Controls(prefixOption + CStr(intArrayCounter2) + "_" + CStr(intArrayCounter +
1)).Value Then
            questionArray(intArrayCounter, 1) = CInt(Me.Controls(prefixAnswerID +
CStr(intArrayCounter2) + "_" + CStr(intArrayCounter + 1)).Caption)
        Exit For
    End If
    Next
Next

'Stop process
IbIPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

GetAnswers = True
Exit Function

GetAnswersError:
IbIPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

Call errorMsg(Err.Description, msg_header_system_error)
GetAnswers = False

End Function

Private Function AllQuestionsAnswered(questionArray() As Integer) As Boolean

    Dim intQuestionCounter As Integer

    For intQuestionCounter = 0 To UBound(questionArray)
        If questionArray(intQuestionCounter, 1) = -1 Then
            AllQuestionsAnswered = False
            Exit Function
        End If
    Next

    AllQuestionsAnswered = True

End Function

```

Private Sub FinishTest()

```

'Get all question answers
If GetAnswers(qTestArray, qAnswerNumArray) Then
    'Check if test in progress
    If boolTestInProgress Then
        'Check if any questions not answered and display alert
        If Not AllQuestionsAnswered(qTestArray) Then
            boolFinishTest = False
            frmTestRunFinish.Show vbModal

            'Cancel test finish if user select to continue
            If Not boolFinishTest Then
                If chkTimerActive.Value = vbChecked Then tmrCountdown.Enabled = True 'START
timer
                Exit Sub
            End If
        End If

        'Stop timer if active
        If tmrCountdown.Enabled Then tmrCountdown.Enabled = False

        boolTestInProgress = False
    End If

    'Calculate results
    Call CalculateResults(qTestArray, finalResultsArray)

    Dim strResultBuffer As String
    Dim testIsSuccess As Boolean
    strResultBuffer = ""

    If finalResultsArray(2) >= finalResultsArray(1) Then
        strResultBuffer = strResultBuffer + frm_testRunResultsPrefixSuccess_message
        testIsSuccess = True
    Else
        strResultBuffer = strResultBuffer + frm_testRunResultsPrefixFailure_message
        testIsSuccess = False
    End If

    strResultBuffer = strResultBuffer + frm_testRunResultsPrefixBreakdown_message
    strResultBuffer = strResultBuffer + frm_testRunResultsPrefixBreakdownPercent_message
    -
    + CStr(finalResultsArray(2)) + "/" + CStr(finalResultsArray(0))
    strResultBuffer = strResultBuffer +
    frm_testRunResultsPrefixBreakdownPercentPass_message _
    + CStr(finalResultsArray(1)) + "/" + CStr(finalResultsArray(0))
    strResultBuffer = strResultBuffer + frm_testRunResultsPrefix_message

    Call infoMSG(strResultBuffer, frm_testRunResultsHeader_message)

    'Log test results
    Call LogResults(user_id, catDetails(0), dblSeconds, dblSecondsTotal, qTestArray,
    testIsSuccess)

    Unload Me

```

End If

End Sub

Private Sub CalculateResults(questionArray() As Integer, resultsArray() As Integer)

On Error GoTo CalculateResultsError

'Dim resultsArray(0 To 2), intQuestionCounter, intCorrentAnswersCounter As Integer
Dim intQuestionCounter, intCorrentAnswersCounter As Integer

intCorrentAnswersCounter = 0

'Start process
IblPleaseWait.Visible = True
IblPleaseWait.Caption = frm_testRunCalculatingResults_message
Me.MousePointer = vbHourglass
DoEvents

'Set test's total questions
resultsArray(0) = catDetails(1)

'Set tests's pass number
resultsArray(1) = catDetails(2)

'Get correct answers
For intQuestionCounter = 0 To UBound(questionArray)
 'Check if correct answer and increase counter
 If questionArray(intQuestionCounter, 1) = questionArray(intQuestionCounter, 2) Then
 intCorrentAnswersCounter = intCorrentAnswersCounter + 1
 End If
Next

'Set test's correct answers
resultsArray(2) = intCorrentAnswersCounter

'Stop process
IblPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

Exit Sub

CalculateResultsError:

IblPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

Call errorMsg(Err.Description, msg_header_system_error)

End Sub

Private Sub LogResults(userId As Integer, catTypeId As Integer, timeLeft As Double,
timeAvailable As Double, testQArray() As Integer, boolSuccess As Boolean)

On Error GoTo LogResultsError


```

Dim strSQLQuery As String
Dim dblTime As Double
Dim boolSuccessResult As Boolean
Dim intCounter As Integer

'Start process
lblPleaseWait.Visible = True
lblPleaseWait.Caption = frm_testRunLoggingResults_message
Me.MousePointer = vbHourglass
DoEvents

'Create logresult query string

'Check first if time limit enabled
If chkTimerActive.Value = vbChecked Then
    dblTime = timeAvailable - timeLeft
Else
    dblTime = -1
End If

strSQLQuery = CreateLogResultSQLString(userId, catTypeId, dblTime, boolSuccess)

'Log result
If performDBAction(dbConnection, dbRecords, strSQLQuery) Then
    'Log entries
    For intCounter = 0 To UBound(testQArray)
        strSQLQuery = CreateLogEntrySQLString(GetInsertedRecordID, _
            testQArray(intCounter, 0), _
            testQArray(intCounter, 1), _
            testQArray(intCounter, 2))

        If Not performDBAction(dbConnection, dbRecords, strSQLQuery) Then
            'Stop process
            lblPleaseWait.Visible = False
            Me.MousePointer = vbDefault
            DoEvents

            Call errorMsg(frm_testRunLogResultsFailed, msg_header_system_error)
            Exit Sub
        End If
    Next
Else
    'Stop process
    lblPleaseWait.Visible = False
    Me.MousePointer = vbDefault
    DoEvents

    Call errorMsg(frm_testRunLogResultsFailed, msg_header_system_error)
End If

Exit Sub

LogResultsError:
'Stop process
lblPleaseWait.Visible = False
Me.MousePointer = vbDefault
DoEvents

```

```
Call errorMsg(Err.Description, msg_header_system_error)
```

```
End Sub
```

```
Private Function GetInsertedRecordID() As Integer
```

```
Dim returnValue As Integer
```

```
returnValue = -1
```

```
'Set sql query string
```

```
strSQLQuery = "select max(testlog_id) as maxID from tbl_testlogs"
```

```
If (openDBConnection(dbConnection, config_db_type)) Then
```

```
  If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
```

```
    'Check if records exist
```

```
    If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then _
      returnValue = CInt(dbRecords("maxID"))
```

```
  If (closeDBRecordset(dbRecords)) Then
```

```
    If (Not closeDBConnection(dbConnection)) Then
```

```
      Call infoMSG(msg_db_conn_warning, msg_header_system_info)
```

```
    End If
```

```
  Else
```

```
    Call infoMSG(msg_db_records_warning, msg_header_system_info)
```

```
  End If
```

```
Else
```

```
  Call errorMsg(msg_db_records_failure, msg_header_system_error)
```

```
End If
```

```
Else
```

```
  Call errorMsg(msg_db_conn_failure, msg_header_system_error)
```

```
End If
```

```
GetInsertedRecordID = returnValue
```

```
End Function
```

```
Private Function CreateLogResultSQLString(userId As Integer, typeId As Integer, timeTaken As Double, boolSuccess As Boolean) As String
```

```
Dim strSQLString As String
```

```
strSQLString = "INSERT INTO tbl_testlogs
(testlog_user_id,testlog_type_id,testlog_time_taken,testlog_passed,testlog_created_ts) _
+ " VALUES(" _
+ CStr(userId) + "," _
+ CStr(typeId) + "," _
+ CStr(timeTaken) + "," _
+ CStr(boolSuccess) + "," _
+ "" + Format(Now(), "yyyy/mm/dd") + "" _
+ ")"
```

```
CreateLogResultSQLString = strSQLString
```

```
End Function
```

```
Private Function CreateLogEntrySQLString(logResultId As Integer, qId As Integer, aid As Integer, aidCorrect As Integer) As String
```

```

Dim strSQLString As String

strSQLString = "INSERT INTO tbl_testlog_entries
(testlog_entry_testlog_id,testlog_entry_question_id,testlog_entry_answer_id,testlog_entry_ans
wer_correct_id)" _
+ " VALUES(" _
+ CStr(logResultId) + "," _
+ CStr(qId) + "," _
+ CStr(ald) + "," _
+ CStr(aldCorrect) _
+ ")"

CreateLogEntrySQLString = strSQLString

End Function

```

FRMTESTREPORTS

Κώδικας

```

Option Explicit
Dim dbConnection As ADODB.Connection
Dim dbRecords As ADODB.Recordset
Dim strSQLQuery As String
Dim intSelectedTestID As Integer
Public intTestID As Integer

Private Sub cmdExit_Click()

Unload Me

End Sub

Private Sub Form_Load()

lblAnalysis.Caption = ""

'Set caption
Call SetFormCation(user_type)

'Show tests list frame
Call ShowTestList

End Sub

Private Sub SetFormCation(userType As String)

If userType = "admin" Then
Me.Caption = frm_TestReports_admin_message
Elseif userType = "user" Then
Me.Caption = frm_TestReports_user_message
End If

End Sub

```

```
Private Sub ShowTestList()
```

```
    'Get question list data
    Call GetTestList(user_type, user_id)
```

```
End Sub
```

```
Private Sub GetTestList(userType As String, userID As Integer)
    Dim X As ListItem 'Variant used to hold list sub-items
```

```
    'Set sql query string
    strSQLQuery = "select testlog_id, testlog_time_taken, testlog_passed, testlog_created_ts," _
        + " type_name, type_time_limit," _
        + " user_fname, user_lname" _
        + " from tbl_testlogs, tbl_types, tbl_users" _
        + " where testlog_type_id = type_id" _
        + " and testlog_user_id = user_id"
```

```
    'Add user_id filter if not administrator
    If (userType <> "admin") Then
        strSQLQuery = strSQLQuery _
            + " and user_id = " + CStr(userID)
    End If
```

```
    strSQLQuery = strSQLQuery _
        + " order by testlog_created_ts desc"
```

```
    If (openDBConnection(dbConnection, config_db_type)) Then
        If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
            'Check if records exist
            If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
                'Prepare list view
                Call PrepareTestList(userType)
```

```
                'Move recordset cursor to first element
                dbRecords.MoveFirst
```

```
                'Iterate through records and add to list
                While Not dbRecords.EOF
```

```
                    If CBool(dbRecords("testlog_passed")) Then
                        Set X = lstTests.ListItems.Add(, , "", , 1)
                    Else
                        Set X = lstTests.ListItems.Add(, , "", , 2)
                    End If
```

```
                    X.ToolTipText = "#" + _
                        Format(Trim(dbRecords("testlog_id")), "0000")
```

```
                'Add sub items
                With Me.lstTests
```

```
                    If userType = "admin" Then
                        X.SubItems(1) = Trim(dbRecords("user_fname")) _
                            & " " & Trim(dbRecords("user_lname")) & ""
                        X.SubItems(2) = Trim(dbRecords("type_name")) & ""
```

```
                    If (CInt(dbRecords("type_time_limit")) > 0) _
```

```

        And (CInt(dbRecords("testlog_time_taken")) >= 0) Then
            X.SubItems(3) = Secs2Time(CDbI(dbRecords("testlog_time_taken"))) & ""
        Else
            X.SubItems(3) = "--" & ""
        End If

        X.SubItems(4) = Trim(dbRecords("testlog_created_ts")) & ""
    ElseIf userType = "user" Then
        X.SubItems(1) = Trim(dbRecords("type_name")) & ""

        If (CInt(dbRecords("type_time_limit")) > 0) _
            And (CInt(dbRecords("testlog_time_taken")) >= 0) Then
            X.SubItems(2) = Secs2Time(CDbI(dbRecords("testlog_time_taken"))) & ""
        Else
            X.SubItems(2) = "--" & ""
        End If

        X.SubItems(3) = Trim(dbRecords("testlog_created_ts")) & ""
    End If
End With

'Move cursor to next record
dbRecords.MoveNext
Wend

IblAnalysis.Caption = frm_TestReports_analysis_message
Else
    IblAnalysis.Caption = ""

'Move cursor to next record
Call ClearTestList
End If

If (closeDBRecordset(dbRecords)) Then
    If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
    End If
Else
    Call infoMSG(msg_db_records_warning, msg_header_system_info)
End If
Else
    Call errorMSG(msg_db_records_failure, msg_header_system_error)
End If
Else
    Call errorMSG(msg_db_conn_failure, msg_header_system_error)
End If
End Sub

Private Sub PrepareTestList(userType As String)

'Sub routine used to prepare list prior loading

'Clear column headers
lstTests.ColumnHeaders.Clear

'Clear column items

```

```

IstTests.ListItems.Clear

'Set columnheaders
IstTests.ColumnHeaders.Add , , "", 300, lvwColumnLeft

If userType = "admin" Then
    IstTests.ColumnHeaders.Add , , "×ñΠόδοçò", 1200, lvwColumnLeft
    IstTests.ColumnHeaders.Add , , "Óýðïò", 2300, lvwColumnLeft
ElseIf userType = "user" Then
    IstTests.ColumnHeaders.Add , , "Óýðïò", 3500, lvwColumnLeft
End If

IstTests.ColumnHeaders.Add , , "ÄéÜñêâéá", 1100, lvwColumnLeft
IstTests.ColumnHeaders.Add , , "Çìàñïïçíßá", 1600, lvwColumnLeft

'Show column headers
IstTests.HideColumnHeaders = False

End Sub

Private Sub ClearTestList()

    'Sub routine used to clear list on no records

    'Clear column headers
    IstTests.ColumnHeaders.Clear

    'Clear column items
    IstTests.ListItems.Clear

    'Set columnheaders
    IstTests.ColumnHeaders.Add , , "", IstTests.Width, lvwColumnLeft

    'Hide column headers
    IstTests.HideColumnHeaders = True

End Sub

Private Function GetSelectedTestID() As Integer

    Dim strTMP As String

    strTMP = Split(IstTests.SelectedItem.ToolTipText, ",")(0)

    GetSelectedTestID = CInt(Replace(strTMP, "#", ""))

End Function

Private Sub IstTests_DblClick()

    If IstTests.ListItems.Count > 0 Then
        If IstTests.SelectedItem.Index > 0 Then
            'Set selected text id
            intTestID = GetSelectedTestID

            'Open breakdown form
            frmTestReportsBD.Show vbModal
        End If
    End If
End Sub

```

```

    End If
    End If

End Sub

Private Function Secs2Time(dbfTime As Double) As String

    Dim intHours, intMinutes, intSeconds As Integer
    Dim strTimestamp As String

    intHours = Fix(dbfTime / 3600)
    intMinutes = Fix((dbfTime - CDBl(intHours * 3600)) / 60)
    intSeconds = dbfTime - (intHours * 3600) - (intMinutes * 60)

    strTimestamp = Format(intHours, "00") _
        + ":" + Format(intMinutes, "00") _
        + ":" + Format(intSeconds, "00")

    'Remove hour part if hours = 0
    If intHours = 0 Then
        strTimestamp = Mid(strTimestamp, 4)
    End If

    Secs2Time = strTimestamp

End Function

```

FRMTESTREPORTSBD

Κώδικας

```

Option Explicit
Dim intSelectedTestID As Integer
Dim intCurrentQuestionIndex As Integer
Dim reportArray() As Integer
Dim dbConnection As ADODB.Connection
Dim dbRecords As ADODB.Recordset
Dim strSQLQuery As String
Private Const prefixFrame As String = "frQ"
Private Const prefixImage As String = "imgQ"
Private Const prefixDescription As String = "txtDQ"
Private Const prefixAnswer As String = "txtAQ"
Private Const prefixTheoryID As String = "lblAIDQ"

Private Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hWnd As Long, ByVal lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory As String, ByVal nShowCmd As Long) As Long

Private Sub cmdExit_Click()

    Unload Me

End Sub

```

```
Private Sub cmdNext_Click()
```

```
    If intCurrentQuestionIndex < UBound(reportArray) + 1 Then
        'Set current question index and display current frame
        Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = False
        intCurrentQuestionIndex = intCurrentQuestionIndex + 1
        Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

        'Show help button if question id found
        If Trim(Me.Controls(prefixTheoryID + "_" + CStr(intCurrentQuestionIndex)).Caption) <> ""
Then
            cmdViewHelp.Visible = True
        Else
            cmdViewHelp.Visible = False
        End If
    End If
```

```
End Sub
```

```
Private Sub cmdPrevious_Click()
```

```
    If intCurrentQuestionIndex < 2 Then Exit Sub

    'Set current question index and display current frame
    Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = False
    intCurrentQuestionIndex = intCurrentQuestionIndex - 1
    Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

    'Show help button if question id found
    If Trim(Me.Controls(prefixTheoryID + "_" + CStr(intCurrentQuestionIndex)).Caption) <> ""
Then
        cmdViewHelp.Visible = True
    Else
        cmdViewHelp.Visible = False
    End If
```

```
End Sub
```

```
Private Sub cmdViewHelp_Click()
```

```
    Dim strTheoryDocName As String

    'Get corresponding theory doc name
    strTheoryDocName = GetTheoryRefDoc(CInt(Trim(Me.Controls(prefixTheoryID + "_" +
CStr(intCurrentQuestionIndex)).Caption)))

    'Check if theory ref exists and if not show error
    If strTheoryDocName <> "" Then
        'Get full path
        strTheoryDocName = getPathRelFile(config_images_path + config_path_docs + "\" +
strTheoryDocName)

        'Check if file exists and if not show error
        If fileExists(strTheoryDocName) Then
            'Open file
            ShellExecute hWnd, "open", strTheoryDocName, vbNullString, vbNullString, 1
```



```

    Else
        Call errorMsg(msg_GeneralDocNotFound_message, msg_header_system_error)
    End If
Else
    Call errorMsg(frm_TestReportsBDTheoryRefNotFound_message,
msg_header_system_error)
End If

End Sub

Private Sub Form_Load()

    intSelectedTestID = frmTestReports.intTestID

    Me.Caption = frm_TestReportsBD_message

    lblStatus.Caption = ""
    intCurrentQuestionIndex = 1

    'Prepare answer list and display report area
    If getReportArray(reportArray, intSelectedTestID) Then
        If Not PrepareReport(reportArray) Then
            Call errorMsg(frm_TestReportsBDPrepareFailed_message,
msg_header_system_error)
            Exit Sub
        End If
    Else
        Call errorMsg(frm_TestReportsBDCreateFailed_message, msg_header_system_error)
        Exit Sub
    End If

    'Reset current question index and display current frame
    intCurrentQuestionIndex = 1
    Me.Controls(prefixFrame + "_" + CStr(intCurrentQuestionIndex)).Visible = True

    'Show help button if question id found
    If Trim(Me.Controls(prefixTheoryID + "_" + CStr(intCurrentQuestionIndex)).Caption) <> ""
Then
        cmdViewHelp.Visible = True
    Else
        cmdViewHelp.Visible = False
    End If

End Sub

Private Function getReportArray(reportArray() As Integer, intSelectedLogEntry As Integer) As
Boolean

    On Error GoTo GetReportArrayError

    Dim isReportArrayCreated As Boolean
    Dim intInternalCounter As Integer

    'Set sql query string
    strSQLQuery = "select * from tbl_testlog_entries" _
        & " where testlog_entry_testlog_id = " + CStr(intSelectedLogEntry) _
        & " order by testlog_entry_id asc"

```

```

isReportArrayCreated = True
intInternalCounter = 0

If (openDBConnection(dbConnection, config_db_type)) Then
  If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
    'Check if records exist
    If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
      'Redimension report array
      ReDim reportArray(0 To (getDBRecordsetCount(dbRecords, config_db_type) - 1), 0 To
2) As Integer

      'Move recordset cursor to first element
      dbRecords.MoveFirst

      'Iterate through records and add to combo
      While Not dbRecords.EOF
        'Question ID
        reportArray(intInternalCounter, 0) = _
          dbRecords("testlog_entry_question_id")
        'Selected Answer ID
        reportArray(intInternalCounter, 1) = _
          dbRecords("testlog_entry_answer_id")
        'Correct Answer ID
        reportArray(intInternalCounter, 2) = _
          dbRecords("testlog_entry_answer_correct_id")

        'Increase counter
        intInternalCounter = intInternalCounter + 1

        'Move cursor to next record
        dbRecords.MoveNext
      Wend
    End If

    If (closeDBRecordset(dbRecords)) Then
      If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
      End If
    Else
      Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If
  Else
    Call errorMSG(msg_db_records_failure, msg_header_system_error)
    isReportArrayCreated = False
  End If
Else
  Call errorMSG(msg_db_conn_failure, msg_header_system_error)
  isReportArrayCreated = False
End If

getReportArray = isReportArrayCreated

Exit Function

GetReportArrayError:
  Call errorMSG(Err.Description, msg_header_system_error)

```

```

    getReportArray = False

End Function

Private Function PrepareReport(questionsArray() As Integer) As Boolean

    On Error GoTo PrepareReportError

    Dim intReportAnswersCounter As Integer

    PrepareReport = False

    'Proceed only if report entries exist
    If UBound(questionsArray) = 0 Then
        PrepareReport = False
        Exit Function
    End If

    'Start process
    lblStatus.Visible = True
    lblStatus.Caption = frm_TestReportsBDPleaseWait_message
    Me.MousePointer = vbHourglass
    DoEvents

    For intReportAnswersCounter = 0 To UBound(questionsArray)
        If Not RenderNewAnswer(intReportAnswersCounter, questionsArray) Then
            PrepareReport = False
            Exit Function
        End If
    Next

    'Stop process
    lblStatus.Visible = False
    Me.MousePointer = vbDefault
    DoEvents

    PrepareReport = True

    Exit Function

PrepareReportError:
    lblStatus.Visible = False
    Me.MousePointer = vbDefault
    DoEvents

    Call errorMsg(Err.Description, msg_header_system_error)
    PrepareReport = False

End Function

Private Function RenderNewAnswer(intQIndex As Integer, qArray() As Integer) As Boolean

    Dim ctrlframe As Control

    'Set sql query string
    strSQLQuery = "select * from tbl_questions, tbl_answers where question_id =
answer_question_id and question_id = " + CStr(qArray(intQIndex, 0))

```

```

If (openDBConnection(dbConnection, config_db_type)) Then
  If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
    'Check if records exist
    If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
      'Move recordset cursor to first element
      dbRecords.MoveFirst

      'Set question frame
      Set ctlframe = frmTestReportsBD.Controls.Add("VB.Frame", prefixFrame + "_" +
CStr(intQIndex + 1), frContainer)
      ctlframe.Visible = False
      ctlframe.Top = frQ.Top
      ctlframe.Left = frQ.Left
      ctlframe.Width = frQ.Width
      ctlframe.Height = frQ.Height
      ctlframe.Caption = CStr(intQIndex + 1) + " / " + CStr(UBound(qArray) + 1)

      'Set image control (if not null)
      If Trim(dbRecords("question_images")) <> "" Then
        Call SetImageControl(intQIndex, dbRecords("question_images"), ctlframe)
      End If

      'Set description control
      Call SetDescriptionControl(intQIndex, dbRecords("question_description"), ctlframe)

      'Set theory id control
      Call SetTheoryIDControl(intQIndex, qArray, dbRecords("question_theory_id"),
ctlframe)

      Dim intAnswerNum As Integer
      intAnswerNum = 1

      'Iterate through records and add controls
      While Not dbRecords.EOF
        'Set question controls
        Call SetOptionControls(intQIndex, dbRecords("answer_description"),
dbRecords("answer_id"), qArray, intAnswerNum, ctlframe)

        'Move cursor to next record
        dbRecords.MoveNext
        intAnswerNum = intAnswerNum + 1
      Wend
    Else
      Call infoMSG(frm_TestReportsBDQuestionNotFound_message,
msg_header_system_info)
      RenderNewAnswer = False
      Exit Function
    End If

    If (closeDBRecordset(dbRecords)) Then
      If (Not closeDBConnection(dbConnection)) Then
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
      End If
    Else
      Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If
  End If
End If

```

```

Else
    Call errorMsg(msg_db_records_failure, msg_header_system_error)
    RenderNewAnswer = False
    Exit Function
End If
Else
    Call errorMsg(msg_db_conn_failure, msg_header_system_error)
    RenderNewAnswer = False
    Exit Function
End If

RenderNewAnswer = True

End Function

Private Sub SetImageControl(intIndex As Integer, strImageName As String, frFrame As Control)

    Dim ctlImage As Control

    Set ctlImage = frmTestReportsBD.Controls.Add("VB.Image", prefixImage + "_" +
CStr(intIndex + 1), frFrame)
    ctlImage.Top = imgQ.Top
    ctlImage.Left = imgQ.Left
    ctlImage.Width = imgQ.Width
    ctlImage.Height = imgQ.Height
    ctlImage.Stretch = imgQ.Stretch

    strImageName = Trim(Replace(strImageName, ",", ""))
    Set ctlImage.Picture = LoadPicture(App.Path + config_images_path + config_path_imgs + "\"
+ strImageName)

    ctlImage.Visible = True

End Sub

Private Sub SetDescriptionControl(intIndex As Integer, strDescription As String, frFrame As
Control)

    Dim ctlDescription As Control

    Set ctlDescription = frmTestReportsBD.Controls.Add("VB.Label", prefixDescription + "_" +
CStr(intIndex + 1), frFrame)
    ctlDescription.Top = txtDQ.Top
    ctlDescription.Left = txtDQ.Left
    ctlDescription.Width = txtDQ.Width
    ctlDescription.Height = txtDQ.Height
    ctlDescription.Font = txtDQ.Font
    ctlDescription.BorderStyle = txtDQ.BorderStyle
    ctlDescription.BackStyle = txtDQ.BackStyle
    ctlDescription.Caption = strDescription
    ctlDescription.Visible = True

End Sub

Private Sub SetOptionControls(intIndex As Integer, strAnswer As String, intAnswerID As
Integer, answerArray() As Integer, intAnswerNum As Integer, frFrame As Control)

```

```

Dim ctlAnswer, ctlAnswerSample As Control

Select Case intAnswerNum
    Case 1
        Set ctlAnswerSample = txtAQ1
    Case 2
        Set ctlAnswerSample = txtAQ2
    Case 3
        Set ctlAnswerSample = txtAQ3
    Case 4
        Set ctlAnswerSample = txtAQ4
    Case 5
        Set ctlAnswerSample = txtAQ5
End Select

Set ctlAnswer = frmTestReportsBD.Controls.Add("VB.Label", prefixAnswer +
CStr(intAnswerNum) + "_" + CStr(intIndex + 1), frFrame)
ctlAnswer.Top = ctlAnswerSample.Top
ctlAnswer.Left = ctlAnswerSample.Left
ctlAnswer.Width = ctlAnswerSample.Width
ctlAnswer.Height = ctlAnswerSample.Height
ctlAnswer.Font = ctlAnswerSample.Font
ctlAnswer.BorderStyle = ctlAnswerSample.BorderStyle
ctlAnswer.BackStyle = ctlAnswerSample.BackStyle
ctlAnswer.Caption = strAnswer

'Red color answer
If intAnswerID = answerArray(intIndex, 1) Then
    ctlAnswer.BackColor = &HFF&
End If

'Green color correct answer
If intAnswerID = answerArray(intIndex, 2) Then
    ctlAnswer.BackColor = &HC000&
End If

ctlAnswer.Visible = True

End Sub

Private Sub SetTheoryIDControl(intIndex As Integer, answerArray() As Integer, theoryID As
Integer, frFrame As Control)

    Dim ctlTheoryID As Control

    'Hidden control to hold theory id if not correct
    Set ctlTheoryID = frmTestReportsBD.Controls.Add("VB.Label", prefixTheoryID + "_" +
CStr(intIndex + 1), frFrame)

    'ALTRNATIVE 1
    'If answerArray(intIndex, 1) = answerArray(intIndex, 2) Then
    '    ctlTheoryID.Caption = ""
    'Else
    '    'Hold only valid theory IDs
    '    If theoryID <> -1 Then
    '        ctlTheoryID.Caption = CStr(theoryID)
    '    Else

```

```

'    ctlTheoryID.Caption = ""
' End If
'End If

'ALTERNATIVE 2
'Hold only valid theory IDs
If theoryID <> -1 Then
    ctlTheoryID.Caption = CStr(theoryID)
Else
    ctlTheoryID.Caption = ""
End If

    ctlTheoryID.Visible = False

End Sub

Private Function GetTheoryRefDoc(intTheoryID As Integer) As String

    Dim strTheoryFilename As String

    strTheoryFilename = ""

    'Set sql query string
    strSQLQuery = "select theory_doc from tbl_theory where theory_id = " + CStr(intTheoryID)

    If (openDBConnection(dbConnection, config_db_type)) Then
        If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
            'Check if records exist
            If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
                'Move recordset cursor to first element
                dbRecords.MoveFirst

                'NOTE: No need to iterate since there should be only one record
                strTheoryFilename = Trim(dbRecords("theory_doc"))
            End If

            If (closeDBRecordset(dbRecords)) Then
                If (Not closeDBConnection(dbConnection)) Then
                    'Call infoMSG(msg_db_conn_warning, msg_header_system_info)
                End If
            Else
                'Call infoMSG(msg_db_records_warning, msg_header_system_info)
            End If
        Else
            Call errorMSG(msg_db_records_failure, msg_header_system_error)
        End If
    Else
        Call errorMSG(msg_db_conn_failure, msg_header_system_error)
    End If

    GetTheoryRefDoc = strTheoryFilename

End Function

```

FRMUSERMGT**Κώδικας**

Option Explicit

Dim dbConnection As ADODB.Connection

Dim dbRecords As ADODB.Recordset

Dim strSQLQuery As String

Dim intSelectedUserID As Integer

Private Sub cmdCancel_Click()

'show user list frame

Call ShowUserList

End Sub

Private Sub cmdCreate_Click()

'Reset user details form

Call ClearUserDetailsForm

'Set frame's proceed button caption

cmdProceed.Caption = "Άξιέπιδείξιμα"

'Enable username input field

txtUsername.Enabled = True

txtUsername.BackColor = &H80000005

'Enable user admin option

chkUserAdmin.Enabled = True

'Show user details frame

Call ShowUserDetails

'Set focus to username field

txtUsername.SetFocus

End Sub

Private Sub cmdDelete_Click()

'Check if items exist

If lstUsers.ListItems.Count > 0 Then

'Check if valid selection

If lstUsers.SelectedItem.Index > 0 Then

'Check if active user is selected to be deleted

If (user_id <> GetSelectedUserID) Then

'Prompt user for deletion

If alertMSG(msg_user_delete_alert, msg_header_system_alert) = vbOK Then

'Make insert SQL string

strSQLQuery = sqlStringForDelete(GetSelectedUserID)

If (performDBAction(dbConnection, dbRecords, strSQLQuery)) Then

Call GetUserList

End If

End If


```
        Else
            Call infoMSG(msg_user_delete_cancel, msg_header_system_info)
        End If
    End If
End If
```

```
lstUsers.SetFocus
```

```
End Sub
```

```
Private Sub cmdExit_Click()
```

```
    'Close user management popup form
    Unload Me
```

```
End Sub
```

```
Private Sub ClearUserDetailsForm()
```

```
    txtUsername.Text = ""
    txtPassword.Text = ""
    txtPasswordRep.Text = ""
    txtFname.Text = ""
    txtLname.Text = ""
    txtExtra.Text = ""
    chkUserAdmin.Value = vbUnchecked
```

```
End Sub
```

```
Private Sub ActivateActionButtons(boollActive As Boolean)
```

```
    cmdUpdate.Enabled = boollActive
    cmdDelete.Enabled = boollActive
```

```
End Sub
```

```
Private Sub ShowUserList()
```

```
    'Hide user details frame
    frUserDetails.Visible = False

    'Enable appropriate form related buttons
    cmdCreate.Enabled = True
    cmdUpdate.Enabled = False
    cmdDelete.Enabled = False

    'Get user list data
    Call GetUserList

    'Enable users list
    lstUsers.Enabled = True
```

```
End Sub
```

```
Private Sub ShowUserDetails()
```

```
    'Show user details frame
```

```

frUserDetails.Visible = True

'Disable all form related buttons
cmdCreate.Enabled = False
cmdUpdate.Enabled = False
cmdDelete.Enabled = False

'Disable users list
lstUsers.Enabled = False

End Sub

Private Sub cmdProceed_Click()

'Check whether update or create
If txtUsername.Enabled Then 'CREATE USER
'Check for duplicated username value
If (CheckForUserDuplicates(Trim(txtUsername.Text))) Then
Call errorMsg(msg_user_duplicate_check_error, msg_header_system_error)
Else
If (checkFieldsForUserCreate) Then
'Make insert SQL string
strSQLQuery = sqlStringForCreate

If (performDBAction(dbConnection, dbRecords, strSQLQuery)) Then Call
ShowUserList
Else
Call errorMsg(msg_user_create_check_error, msg_header_system_error)
End If
End If
Else 'UPDATE USER
If (checkFieldsForUserUpdate) Then
'Make update SQL string
strSQLQuery = sqlStringForUpdate(intSelectedUserID)

If (performDBAction(dbConnection, dbRecords, strSQLQuery)) Then Call ShowUserList
Else
Call errorMsg(msg_user_update_check_error, msg_header_system_error)
End If
End If

End Sub

Private Sub cmdUpdate_Click()

'Check if items exist
If lstUsers.ListItems.Count > 0 Then
'Check if valid selection
If lstUsers.SelectedItem.Index > 0 Then
'Reset user details form
Call ClearUserDetailsForm

'Set frame's proceed button caption
cmdProceed.Caption = "ΆίάΎύόç"

'Disable username input field
txtUsername.Enabled = False

```

```
txtUsername.BackColor = &H80000011

'Get selected user id
intSelectedUserID = GetSelectedUserID

'Get user details data
Call GetUserDetails

'Show user details frame
Call ShowUserDetails
End If
End If

End Sub

Private Sub Form_Load()

'set form's caption
Me.Caption = frm_userMgt

'Show user list frame
Call ShowUserList

End Sub

Private Sub PrepareUserList()

'Sub routine used to prepare list prior loading

'Clear column headers
lstUsers.ColumnHeaders.Clear

'Clear column items
lstUsers.ListItems.Clear

'Set columnheaders
lstUsers.ColumnHeaders.Add , , "", 300, lvwColumnLeft
lstUsers.ColumnHeaders.Add , , "Όδιεçιαόέëü", 1700, lvwColumnLeft
lstUsers.ColumnHeaders.Add , , "¼ííá", 1800, lvwColumnLeft
lstUsers.ColumnHeaders.Add , , "Άðbíðí", 1800, lvwColumnLeft
lstUsers.ColumnHeaders.Add , , "Çi/íßá áãñáöðò", 1400, lvwColumnLeft

'Show column headers
lstUsers.HideColumnHeaders = False

End Sub

Private Sub ClearUserList()

'Sub routine used to clear list on no records

'Clear column headers
lstUsers.ColumnHeaders.Clear

'Clear column items
lstUsers.ListItems.Clear
```

```
'Set columnheaders
lstUsers.ColumnHeaders.Add , , "", lstUsers.Width, lvwColumnLeft
```

```
'Hide column headers
lstUsers.HideColumnHeaders = True
```

```
End Sub
```

```
Private Sub GetUserList()
```

```
Dim X As ListItem 'Variant used to hold list sub-items
```

```
'Set sql query string
strSQLQuery = "select * from tbl_users order by user_id asc"
```

```
If (openDBConnection(dbConnection, config_db_type)) Then
If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
```

```
'Check if records exist
```

```
If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
```

```
'Prepare list view
```

```
Call PrepareUserList
```

```
'Move recordset cursor to first element
```

```
dbRecords.MoveFirst
```

```
'Iterate through records and add to list
```

```
While Not dbRecords.EOF
```

```
    If CBool(dbRecords("user_isAdmin")) Then
```

```
        Set X = lstUsers.ListItems.Add(, , "", 3)
```

```
        X.ToolTipText = "#" + _
```

```
            Format(Trim(dbRecords("user_id")), "0000") + ",Άέά÷άέήέόδρò"
```

```
    Else
```

```
        Set X = lstUsers.ListItems.Add(, , "")
```

```
        X.ToolTipText = "#" + _
```

```
            Format(Trim(dbRecords("user_id")), "0000")
```

```
    End If
```

```
'Add sub items
```

```
With Me.lstUsers
```

```
    X.SubItems(1) = Trim(dbRecords("user_name")) & ""
```

```
    X.SubItems(2) = Trim(dbRecords("user_fname")) & ""
```

```
    X.SubItems(3) = Trim(dbRecords("user_lname")) & ""
```

```
    X.SubItems(4) = Trim(dbRecords("user_created_ts")) & ""
```

```
End With
```

```
'Move cursor to next record
```

```
dbRecords.MoveNext
```

```
Wend
```

```
Else
```

```
'Clear list view
```

```
Call ClearUserList
```

```
'Add no records message
```

```
'Set X = lstUsers.ListItems.Add(, , msg_db_general_no_records)
```

```
End If
```

```

    If (closeDBRecordset(dbRecords)) Then
        If (Not closeDBConnection(dbConnection)) Then
            Call infoMSG(msg_db_conn_warning, msg_header_system_info)
        End If
    Else
        Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If
Else
    Call errorMSG(msg_db_records_failure, msg_header_system_error)
End If
Else
    Call errorMSG(msg_db_conn_failure, msg_header_system_error)
End If

End Sub

Private Sub GetUserDetails()

    'Set sql query string
    strSQLQuery = "select * from tbl_users where user_id = " + CStr(intSelectedUserID)

    If (openDBConnection(dbConnection, config_db_type)) Then
        If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
            'Check if records exist
            If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then
                'Move recordset cursor to first element
                dbRecords.MoveFirst

                'NOTE: No need to iterate, since only one record should be returned
                txtUsername.Text = Trim(dbRecords("user_name"))
                txtPassword.Text = ""
                txtPasswordRep.Text = ""

                If CBool(dbRecords("user_isAdmin")) Then
                    chkUserAdmin.Value = vbChecked
                Else
                    chkUserAdmin.Value = vbUnchecked
                End If

                'Disable user admin option if current user details are accessed
                If user_id <> GetSelectedUserID Then
                    chkUserAdmin.Enabled = True
                Else
                    chkUserAdmin.Enabled = False
                End If

                txtFname.Text = Trim(dbRecords("user_fname"))
                txtLname.Text = Trim(dbRecords("user_lname"))
                txtExtra.Text = Trim(dbRecords("user_extra"))
            Else
                'Alert that no record was found and return to users list
                Call errorMSG(msg_db_no_record_correlation, msg_header_system_error)
                Call ShowUserList
            End If

            If (closeDBRecordset(dbRecords)) Then
                If (Not closeDBConnection(dbConnection)) Then

```

```
        Call infoMSG(msg_db_conn_warning, msg_header_system_info)
    End If
    Else
        Call infoMSG(msg_db_records_warning, msg_header_system_info)
    End If
    Else
        Call errorMSG(msg_db_records_failure, msg_header_system_error)
    End If
    Else
        Call errorMSG(msg_db_conn_failure, msg_header_system_error)
    End If

End Sub

Private Sub lstUsers_Click()

    Call lstUsers_GotFocus

End Sub

Private Sub lstUsers_DbClick()

    Call cmdUpdate_Click

End Sub

Private Sub lstUsers_GotFocus()

    If lstUsers.ListItems.Count > 0 Then
        If lstUsers.SelectedItem.Index > 0 Then
            Call ActivateActionButtons(True)
        Else
            Call ActivateActionButtons(False)
        End If
    Else
        Call ActivateActionButtons(False)
    End If

End Sub

Private Function checkFieldsForUserCreate() As Boolean

    Dim isChecked As Boolean
    isChecked = True

    If Trim(txtUsername.Text) = "" Then isChecked = False

    If Trim(txtPassword.Text) = "" _
        And Trim(txtPasswordRep.Text) = "" Then isChecked = False

    If Trim(txtPassword.Text) <> Trim(txtPasswordRep.Text) Then isChecked = False

    If Trim(txtFname.Text) = "" _
        And Trim(txtLname.Text) = "" Then isChecked = False

    checkFieldsForUserCreate = isChecked

End Function
```

End Function

Private Function checkFieldsForUserUpdate() As Boolean

```
Dim isChecked As Boolean
isChecked = True
```

```
If Trim(txtPassword.Text) <> Trim(txtPasswordRep.Text) Then isChecked = False
```

```
If Trim(txtFname.Text) = "" _
  And Trim(txtLname.Text) = "" Then isChecked = False
```

```
checkFieldsForUserUpdate = isChecked
```

End Function

Private Function sqlStringForCreate() As String

```
Dim strSQL As String
```

```
strSQL = "INSERT INTO tbl_users
(user_name,user_password,user_isAdmin,user_fname,user_lname,user_extra,user_created_ts
) VALUES("
```

```
strSQL = strSQL + "" + Trim(txtUsername.Text) + ","
strSQL = strSQL + "" + Trim(txtPassword.Text) + ","
strSQL = strSQL + CStr(CBool(chkUserAdmin.Value)) + ","
strSQL = strSQL + "" + Trim(txtFname.Text) + ","
strSQL = strSQL + "" + Trim(txtLname.Text) + ","
strSQL = strSQL + "" + Trim(txtExtra.Text) + ","
strSQL = strSQL + "" + Format(Now(), "yyyy/mm/dd") + ""
```

```
strSQL = strSQL + ")"
```

```
sqlStringForCreate = strSQL
```

End Function

Private Function sqlStringForUpdate(intUserID As Integer) As String

```
Dim strSQL As String
```

```
strSQL = "UPDATE tbl_users SET "
```

```
strSQL = strSQL + "user_isAdmin = " + CStr(CBool(chkUserAdmin.Value))
```

```
If Trim(txtPassword.Text) <> "" Then _
  strSQL = strSQL + ",user_password = " + Trim(txtPassword.Text) + ""
```

```
If Trim(txtFname.Text) <> "" Then _
  strSQL = strSQL + ",user_fname = " + Trim(txtFname.Text) + ""
```

```
If Trim(txtLname.Text) <> "" Then _
  strSQL = strSQL + ",user_lname = " + Trim(txtLname.Text) + ""
```

```
If Trim(txtExtra.Text) <> "" Then _
  strSQL = strSQL + ",user_extra = " + Trim(txtExtra.Text) + ""
```

```

    strSQL = strSQL + " WHERE user_id = " + CStr(intUserID)

    sqlStringForUpdate = strSQL

End Function

Private Function sqlStringForDelete(intUserID As Integer) As String
    sqlStringForDelete = "delete from tbl_users where user_id = " + CStr(intUserID)

End Function

Private Function GetSelectedUserID() As Integer

    Dim strTMP As String

    strTMP = Split(lstUsers.SelectedItem.ToolTipText, ",")(0)

    GetSelectedUserID = CInt(Replace(strTMP, "#", ""))

End Function

Private Function CheckForUserDuplications(strUsernameToCheck As String) As Boolean

    Dim hasDuplications As Boolean
    hasDuplications = False

    'Set sql query string
    strSQLQuery = "select * from tbl_users where user_name =" & _
        + strUsernameToCheck & """"

    If (openDBConnection(dbConnection, config_db_type)) Then
        If (getDBRecordset(dbConnection, dbRecords, strSQLQuery)) Then
            'Check if records exist
            If (getDBRecordsetCount(dbRecords, config_db_type) > 0) Then hasDuplications = True

            If (closeDBRecordset(dbRecords)) Then
                If (Not closeDBConnection(dbConnection)) Then
                    Call infoMSG(msg_db_conn_warning, msg_header_system_info)
                End If
            Else
                Call infoMSG(msg_db_records_warning, msg_header_system_info)
            End If
        Else
            Call errorMSG(msg_db_records_failure, msg_header_system_error)
        End If
    Else
        Call errorMSG(msg_db_conn_failure, msg_header_system_error)
    End If

    CheckForUserDuplications = hasDuplications

End Function

```


FRMABOUT**Κώδικας**

```
Private Sub Form_Load()
```

```
    Me.Caption = msg_frm_about  
    txtAbout.Text = msg_frm_about_text _  
        + vbCrLf + "v " + CStr(App.Major) + "." + CStr(App.Minor) + " " + CStr(App.Revision)
```

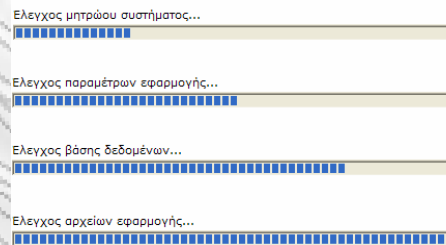
```
End Sub
```

Β. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ

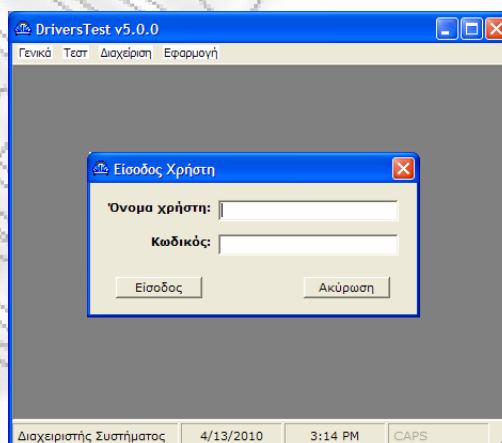
ΓΕΝΙΚΑ

Η εφαρμογή ξεκινάει με την παρακάτω φόρμα, κατά τη διάρκεια εμφάνισης της οποίας εκτελούνται οι απαραίτητοι έλεγχοι για τη σωστή λειτουργία της. Όπως φαίνεται και στις παρακάτω εικόνες, οι έλεγχοι που πραγματοποιούνται είναι:

- α) έλεγχος μητρώου συστήματος,
- β) έλεγχος παραμέτρων εφαρμογής,
- γ) έλεγχος βάσης δεδομένων και
- δ) έλεγχος αρχείων εφαρμογής.



Με την ολοκλήρωση των ελέγχων εμφανίζεται η φόρμα «Drivers test» και το παράθυρο διαλόγου «Είσοδος χρήστη». Υπάρχουν δύο ειδών χρήστες: Ο administrator και ο απλός χρήστης.



ΔΙΑΧΕΙΡΙΣΤΗΣ ΣΥΣΤΗΜΑΤΟΣ

Για την είσοδο του administrator στην εφαρμογή, στο παράθυρο διαλόγου «Είσοδος χρήστη», στα πεδία «όνομα χρήστη» και «κωδικός» πληκτρολογούμε «admin». Μόλις επιλέξουμε «είσοδο» ενεργοποιείται η φόρμα «Drivers Test» καθώς και τα μενού με τις αντίστοιχες εντολές που αφορούν τις ενέργειες που μπορεί να εκτελεί ο administrator. Στη γραμμή κατάστασης της φόρμας αυτής βλέπουμε ότι εμφανίζεται ο χρήστης, η ημερομηνία, η ώρα καθώς φαίνεται και το αν είναι ενεργοποιημένα τα κεφαλαία γράμματα.

Μενού «Γενικά»

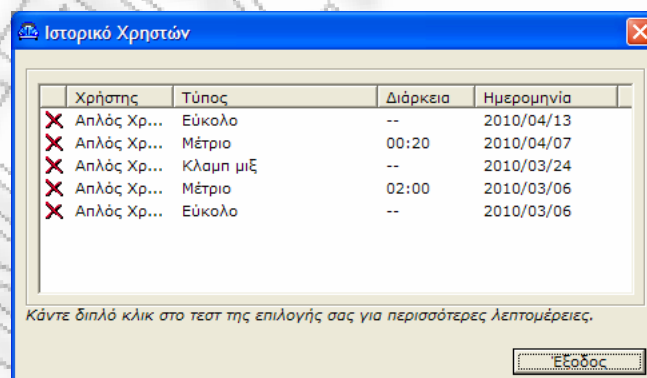
Στο μενού αυτό υπάρχουν δύο εντολές:

1. Έξοδος χρήστη. Αν επιλέξουμε αυτή την εντολή, ο administrator αποσυνδέεται και εμφανίζεται και πάλι το παράθυρο διαλόγου «Είσοδος χρήστη».
2. Τερματισμός. Έξοδος από την εφαρμογή.

Μενού «Τεστ»

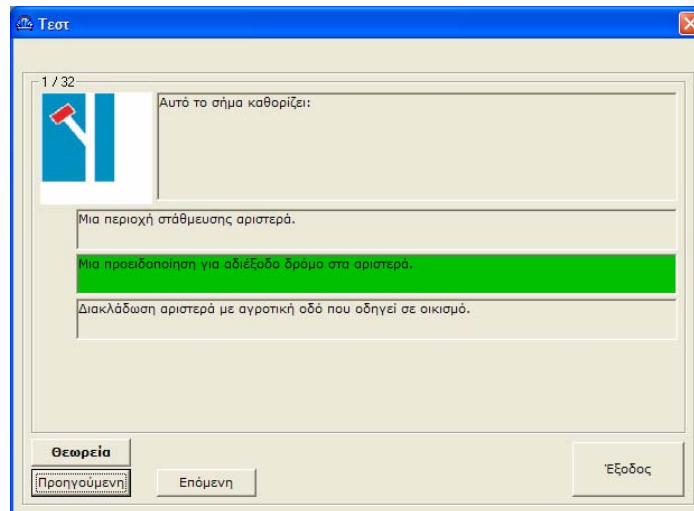
Στο μενού αυτό υπάρχει μία εντολή:

1. Ιστορικό χρηστών. Με αυτή την εντολή ο administrator μπορεί να ελέγξει τα τεστ που έχουν κάνει οι χρήστες. Μόλις επιλεγεί αυτή η εντολή εμφανίζεται το παρακάτω παράθυρο. Ο administrator επιλέγει το τεστ του χρήστη το οποίο θέλει να ελέγξει με βάση τον τύπο, τη διάρκεια και την ημερομηνία διεξαγωγής του και πατάει διπλό κλικ πάνω του.

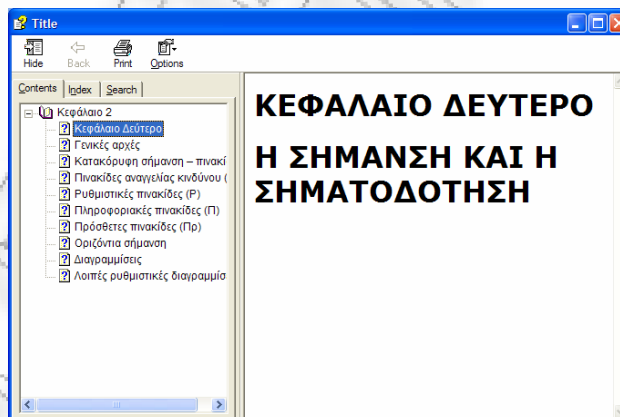


Το τεστ ανοίγει και ο administrator χρησιμοποιώντας τα κουμπιά «επόμενη» και «προηγούμενη» μπορεί να πλοηγηθεί και να ελέγξει. Όσον αφορά τις απαντήσεις αυτές, μπορεί να είναι σωστές, λάθος ή και να μην έχουν απαντηθεί οι ερωτήσεις.

Σωστές απαντήσεις: εφόσον ο χρήστης έχει δώσει σωστή απάντηση σε μια ερώτηση, το πεδίο της απάντησης είναι πράσινο, ενώ ταυτόχρονα θα εμφανιστεί και το κουμπί «Θεωρεία».

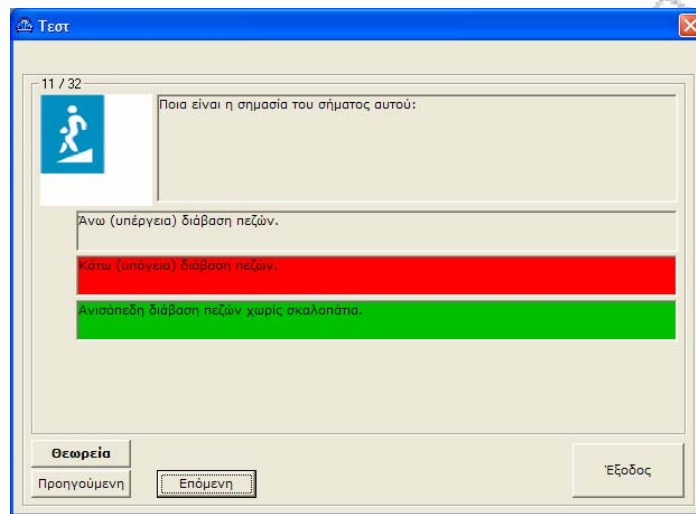


Πατώντας το κουμπί αυτό εμφανίζεται το κομμάτι της θεωρίας που αντιστοιχεί σε αυτή την ερώτηση.



Το κουμπί αυτό εμφανίζεται και σε περίπτωση που η απάντηση είναι λάθος ή και αν δεν έχει δοθεί απάντηση σε μια ερώτηση.

Λάθος απαντήσεις: εάν ο χρήστης έχει δώσει λάθος απάντηση, το πεδίο της απάντησής του είναι κόκκινο ενώ το πεδίο της σωστής απάντησης είναι πράσινο.

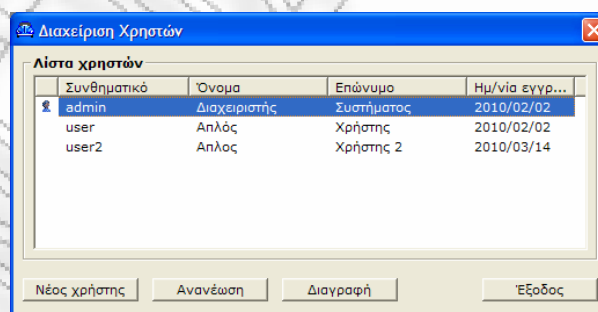


Αναπάντητες ερωτήσεις: εάν ο χρήστης δεν απαντήσει σε μια ερώτηση, το πεδίο της σωστής απάντησης θα είναι πράσινο.

Μενού «Διαχείριση»

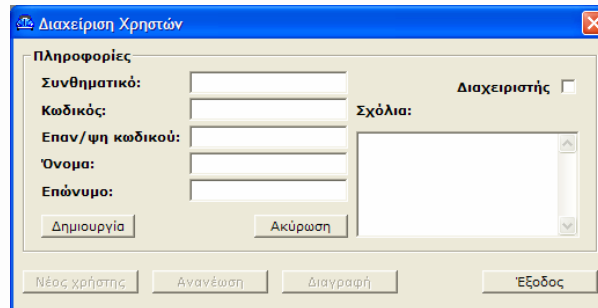
Στο μενού αυτό υπάρχουν δύο εντολές:

1. **Χρήστες.** Αν επιλέξουμε αυτή την εντολή, εμφανίζεται το παρακάτω παράθυρο, μέσω του οποίου ο administrator μπορεί να διαχειριστεί τους λογαριασμούς των χρηστών.

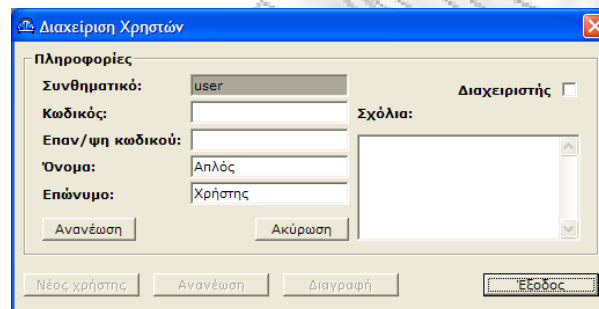


Κουμπί «Νέος χρήστης». Πατώντας αυτό το κουμπί, εμφανίζεται το παράθυρο που ακολουθεί. Στο πεδίο «Συνθηματικό» πληκτρολογούμε το όνομα του χρήστη που θέλουμε να δημιουργήσουμε και στο επόμενο πεδίο τον κωδικό του. Οι ονομασίες αυτές θα χρησιμοποιούνται κάθε φορά που ο χρήστης θα θέλει να εισαχθεί στο πρόγραμμα. Στο επόμενο πεδίο επαναλαμβάνεται ο κωδικός του χρήστη για να αποφευχθούν τα λάθη και στα δύο τελευταία πεδία εισάγουμε τον όνομα και το επώνυμο του χρήστη. Εάν θέλουμε ο χρήστης να μην είναι απλός αλλά να μπορεί να διαχειρίζεται όσα και ο administrator, τότε τσεκάρουμε την εντολή «διαχειριστής». Στο πεδίο «σχόλια» μπορούμε να πληκτρολογήσουμε λίγες

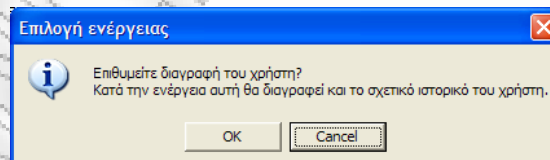
πληροφορίες που αφορούν το χρήστη. Μόλις τελειώσουμε την εισαγωγή των δεδομένων πατάμε το κουμπί «Δημιουργία». Σε περίπτωση που δε θέλουμε να δημιουργηθεί ο συγκεκριμένος χρήστης πατάμε το κουμπί «Ακύρωση».



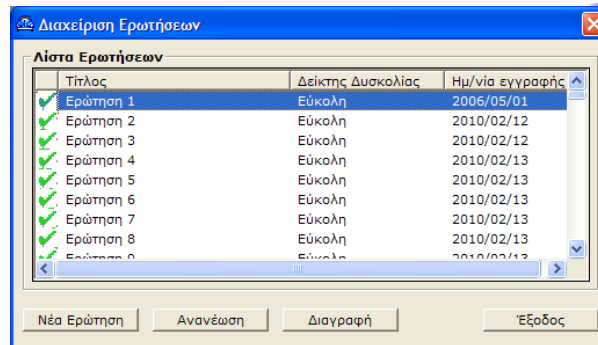
Κουμπί «Ανανέωση». Για να ενεργοποιηθεί αυτό το κουμπί, θα πρέπει να έχουμε επιλέξει το χρήστη του οποίου τα δεδομένα θέλουμε να αλλάξουμε. Πατώντας λοιπόν αυτό το κουμπί, εμφανίζεται το παράθυρο «Διαχείριση χρηστών» με τα στοιχεία του χρήστη που έχουμε επιλέξει και κάνουμε τις αλλαγές που θέλουμε.



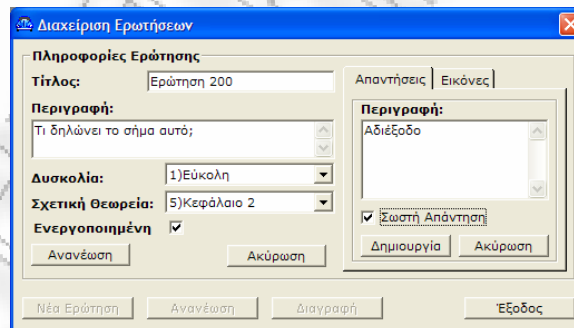
Κουμπί «Διαγραφή». Το κουμπί αυτό το χρησιμοποιούμε όταν θέλουμε να διαγράψουμε ένα χρήστη. Επιλέγουμε λοιπόν πρώτα το χρήστη που θέλουμε και στη συνέχεια πατάμε αυτό το κουμπί. Εάν διαγράψουμε ένα χρήστη θα διαγραφεί και το ιστορικό του, όπως μας ενημερώνει και η ερώτηση που γίνεται για επιβεβαίωση της διαγραφής.



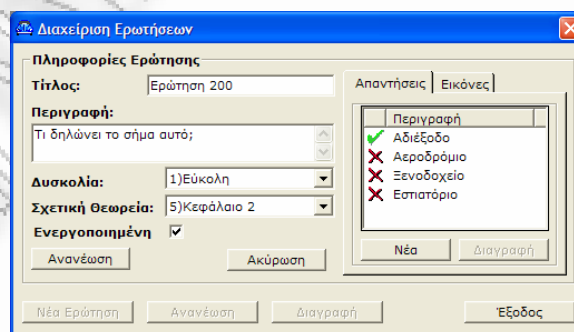
2. *Τεστ*: α) Ερωτήσεις. Επιλέγοντας «Ερωτήσεις» μπορούμε να εισάγουμε, να αλλάξουμε ή και να διαγράψουμε μια ερώτηση.



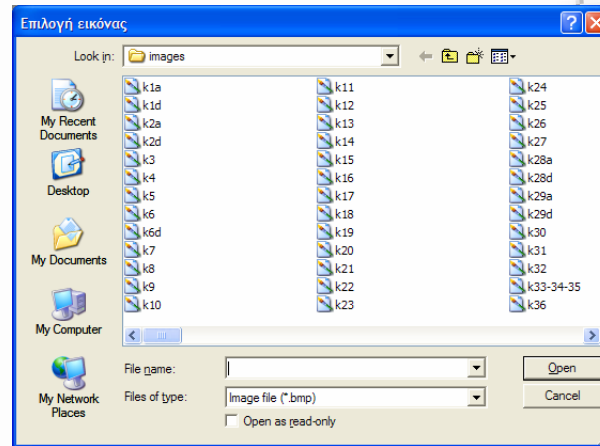
Κουμπί «Νέα ερώτηση». Πατώντας αυτό το κουμπί εμφανίζεται το παράθυρο «Διαχείριση ερωτήσεων». Στο πεδίο «Τίτλος» πληκτρολογούμε ένα διακριτικό για την ερώτηση, π.χ. ερώτηση 1. Στο επόμενο πεδίο πληκτρολογούμε την ερώτηση. Στη συνέχεια επιλέγουμε την κατηγορία δυσκολίας στην οποία ανήκει, καθώς και τη σχετική θεωρία. Αν τσεκάρουμε την εντολή «ενεργοποιημένη», τότε η ερώτηση αυτή θα μπορεί να επιλεγεί στα τεστ. Μόλις ολοκληρώσουμε αυτές τις επιλογές, πατάμε το κουμπί «δημιουργία» και έτσι ενεργοποιείται και το κομμάτι των απαντήσεων. Πατάμε λοιπόν «νέα», και στην περιγραφή πληκτρολογούμε την απάντηση που θέλουμε. Αν η απάντηση που μόλις γράψαμε είναι και η σωστή, τσεκάρουμε την εντολή «Σωστή απάντηση».



Οι περισσότερες απαντήσεις που μπορούμε να δώσουμε είναι τέσσερις, εκ των οποίων μία σωστή, όπως φαίνεται και στη φόρμα που ακολουθεί.



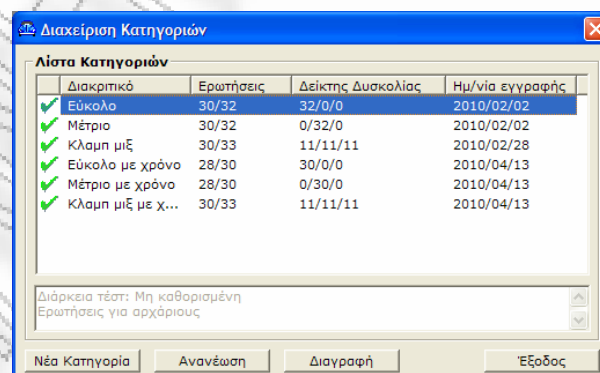
Εφόσον ολοκληρώσουμε το κομμάτι των απαντήσεων, μπορούμε να προχωρήσουμε στην εισαγωγή εικόνας Εάν απαιτείται από την ερώτηση. Επιλέγουμε λοιπόν, την καρτέλα «Εικόνες» και πατάμε το κουμπί «Νέα». Εμφανίζεται το διπλανό παράθυρο από το οποίο και θα επιλεγεί η εικόνα. Εφόσον γίνει η εισαγωγή πατάμε το κουμπί «Ολοκλήρωση».



Κουμπί «Ανανέωση». Για να ενεργοποιηθεί αυτό το κουμπί, θα πρέπει να έχουμε επιλέξει την ερώτηση της οποίας τα δεδομένα θέλουμε να αλλάξουμε. Πατώντας λοιπόν αυτό το κουμπί, εμφανίζεται το παράθυρο «Διαχείριση ερωτήσεων» με τα στοιχεία της ερώτησης που έχουμε επιλέξει και κάνουμε τις αλλαγές που θέλουμε.

Κουμπί «Διαγραφή». Το κουμπί αυτό το χρησιμοποιούμε όταν θέλουμε να διαγράψουμε μια ερώτηση. Επιλέγουμε λοιπόν πρώτα την ερώτηση που θέλουμε και στη συνέχεια πατάμε αυτό το κουμπί. Εάν διαγράψουμε μια ερώτηση θα διαγραφεί και το ιστορικό της, όπως μας ενημερώνει και η ερώτηση που γίνεται για επιβεβαίωση της διαγραφής.

2. Τεστ: β) Κατηγορίες. Επιλέγοντας «Κατηγορίες» μπορούμε να εισάγουμε, να αλλάξουμε ή και να διαγράψουμε μια κατηγορία.



Κουμπί «Νέα κατηγορία». Πατώντας αυτό το κουμπί εμφανίζεται το παράθυρο «Διαχείριση κατηγοριών». Στο πεδίο «Διακριτικό» πληκτρολογούμε ένα διακριτικό για την κατηγορία, π.χ. εύκολο. Στο επόμενο πεδίο πληκτρολογούμε μια περιγραφή της κατηγορίας προκειμένου να διακρίνεται από τις υπόλοιπες. Στη συνέχεια επιλέγουμε τον αριθμό των ερωτήσεων που θα περιλαμβάνει το συγκεκριμένο τεστ, καθώς και το πόσες από αυτές τις ερωτήσεις θα πρέπει να είναι σωστές για να είναι επιτυχές το τεστ. Αν-τσεκάρουμε την εντολή «ενεργοποιημένη», τότε η κατηγορία αυτή θα μπορεί να επιλεγεί στα τεστ. Εάν θέλουμε χρονομέτρηση του τεστ, τότε επιλέγουμε τις ώρες και τα λεπτά διάρκειάς του από το πεδίο «Χρόνος». Τέλος, επιλέγουμε πόσες εύκολες, μέτριες ή δύσκολες ερωτήσεις θα περιέχει το συγκεκριμένο τεστ. Την επιλογή αυτή την κάνουμε μέσω των δεικτών δυσκολίας και συγκεκριμένα μετακινώντας τις αντίστοιχες μπάρες δυσκολίας. Μόλις ολοκληρώσουμε τις επιλογές μας, πατάμε το κουμπί «δημιουργία».

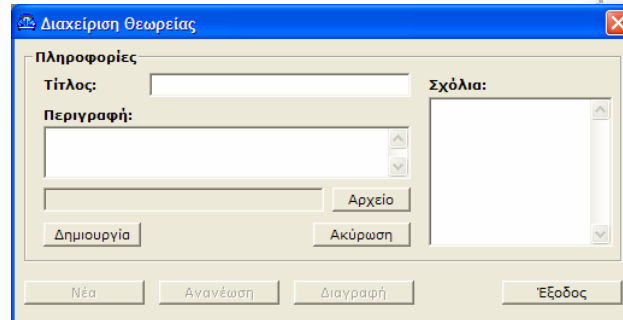
Κουμπί «Ανανέωση». Για να ενεργοποιηθεί αυτό το κουμπί, θα πρέπει να έχουμε επιλέξει την κατηγορία της οποίας τα δεδομένα θέλουμε να αλλάξουμε. Πατώντας λοιπόν αυτό το κουμπί, εμφανίζεται το παράθυρο «Διαχείριση κατηγοριών» με τα στοιχεία της κατηγορίας που έχουμε επιλέξει και κάνουμε τις αλλαγές που θέλουμε.

Κουμπί «Διαγραφή». Το κουμπί αυτό το χρησιμοποιούμε όταν θέλουμε να διαγράψουμε μια κατηγορία. Επιλέγουμε λοιπόν πρώτα την κατηγορία που θέλουμε και στη συνέχεια πατάμε αυτό το κουμπί. Εάν διαγράψουμε μια κατηγορία θα διαγραφεί και το ιστορικό της, όπως μας ενημερώνει και η ερώτηση που γίνεται για επιβεβαίωση της διαγραφής.

2. Τεστ: γ) Θεωρία. Επιλέγοντας «Θεωρία» μπορούμε να εισάγουμε, να αλλάξουμε ή και να διαγράψουμε μια κατηγορία.

Τίτλος	Ημ/νία εγγραφής
Κεφάλαιο 1	2010/04/13
Κεφάλαιο 2	2010/04/13
Κεφάλαιο 3	2010/04/13
Κεφάλαιο 4	2010/04/13
Κεφάλαιο 5	2010/04/13

Κουμπί «Νέα». Πατώντας αυτό το κουμπί εμφανίζεται το παράθυρο «Διαχείριση θεωρίας». Στο πεδίο «Τίτλος» πληκτρολογούμε ένα τίτλο για τη θεωρία π.χ. κεφάλαιο 1. Στο επόμενο πεδίο πληκτρολογούμε μια περιγραφή της θεωρίας που θα περιέχει το συγκεκριμένο κεφάλαιο. Στη συνέχεια επιλέγουμε το αρχείο της θεωρίας που θέλουμε να εισάγουμε και πατάμε «Δημιουργία».



Κουμπί «Ανανέωση». Για να ενεργοποιηθεί αυτό το κουμπί, θα πρέπει να έχουμε επιλέξει τη θεωρία της οποίας τα δεδομένα θέλουμε να αλλάξουμε. Πατώντας λοιπόν αυτό το κουμπί, εμφανίζεται το παράθυρο «Διαχείριση θεωρίας» με τα στοιχεία της θεωρίας που έχουμε επιλέξει και κάνουμε τις αλλαγές που θέλουμε.

Κουμπί «Διαγραφή». Το κουμπί αυτό το χρησιμοποιούμε όταν θέλουμε να διαγράψουμε μια θεωρία. Επιλέγουμε λοιπόν πρώτα τη θεωρία που θέλουμε και στη συνέχεια πατάμε αυτό το κουμπί. Εάν διαγράψουμε μια θεωρία θα διαγραφεί και το ιστορικό της, όπως μας ενημερώνει και η ερώτηση που γίνεται για επιβεβαίωση της διαγραφής.

ΧΡΗΣΤΕΣ

Για την είσοδο ενός χρήστη στην εφαρμογή, στο παράθυρο διαλόγου «Είσοδος χρήστη», στα πεδία «όνομα χρήστη» και «κωδικός» πληκτρολογούμε τα αντίστοιχα του χρήστη. Μόλις επιλέξουμε «είσοδο» ενεργοποιείται η φόρμα «Drivers Test» καθώς και τα μενού με τις αντίστοιχες εντολές που αφορούν τις ενέργειες που μπορεί να εκτελεί ο απλός χρήστης. Στη γραμμή κατάστασης της φόρμας αυτής βλέπουμε ότι εμφανίζεται ο χρήστης, η ημερομηνία, η ώρα καθώς φαίνεται και το αν είναι ενεργοποιημένα τα κεφαλαία γράμματα.

Μενού «Γενικά»

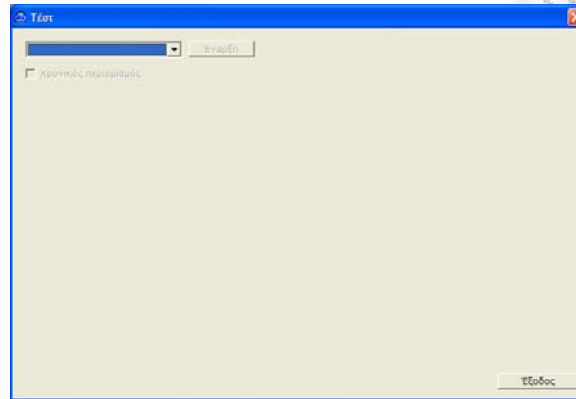
Στο μενού αυτό υπάρχουν δύο εντολές:

1. Έξοδος χρήστη. Αν επιλέξουμε αυτή την εντολή, ο χρήστης αποσυνδέεται και εμφανίζεται και πάλι το παράθυρο διαλόγου «Είσοδος χρήστη».
2. Τερματισμός. Έξοδος από την εφαρμογή.

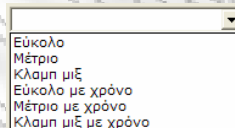
Μενού «Τεστ»

Στο μενού αυτό υπάρχουν δύο εντολές:

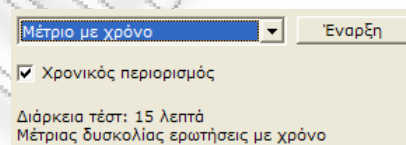
1. Έναρξη. Αν επιλέξουμε αυτή την εντολή εμφανίζεται το παρακάτω παράθυρο, από το οποίο και μπορεί να επιλέξει ο χρήστης το τεστ που θέλει να κάνει.



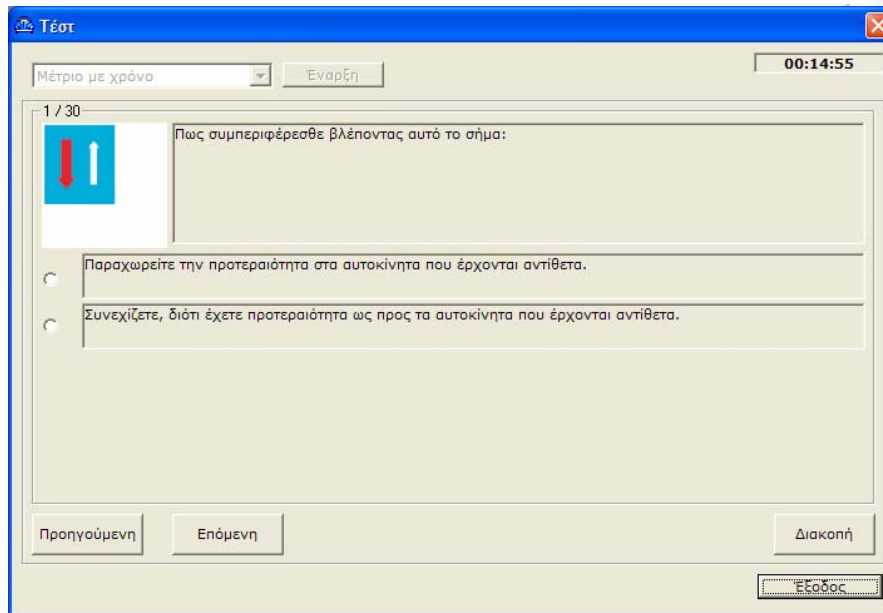
Πατάει λοιπόν, το βελάκι στο τέλος του πεδίου και από το μενού με τα τεστ που εμφανίζονται, επιλέγει το τεστ που θέλει.



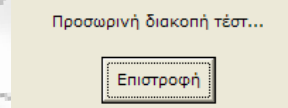
Όταν επιλέξει το τεστ θα εμφανιστεί μήνυμα το οποίο θα ενημερώνει για το είδος του τεστ που έχει επιλεγεί καθώς και το αν είναι με χρόνο. Στη συνέχεια ο χρήστης μπορεί να επιλέξει «Έναρξη».



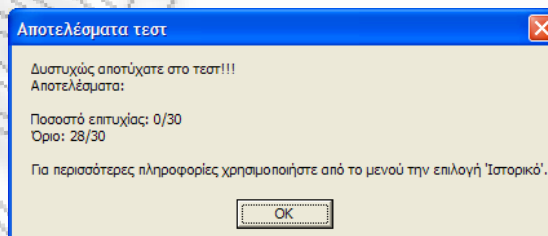
Εμφανίζεται το παρακάτω παράθυρο «Τεστ». Πάνω δεξιά βλέπουμε να εμφανίζεται ο χρόνος που απομένει για την ολοκλήρωση του τεστ. Στο αριστερό μέρος φαίνεται η εικόνα που αντιστοιχεί στη συγκεκριμένη ερώτηση, δίπλα η ερώτηση και κάτω οι απαντήσεις. Εφόσον ο χρήστης επιλέξει την απάντηση που θεωρεί σωστή, προχωράει στην επόμενη, πατώντας το αντίστοιχο κουμπί.



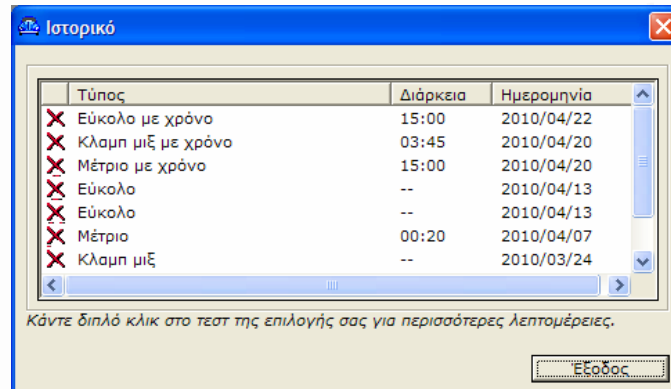
Αν για κάποιο λόγο θέλει να σταματήσει για λίγο τη διεξαγωγή του τεστ, επιλέγει «Διακοπή». Αν το τεστ είναι με χρόνο, τότε σταματάει να μετράει ο χρόνος. Για να συνεχίσει το τεστ, πατάει το κουμπί «Επιστροφή».



Όταν φτάσει και στη τελευταία ερώτηση, θα εμφανιστεί και το κουμπί «Ολοκλήρωση». Πατώντας αυτό θα ολοκληρωθεί η διεξαγωγή του τεστ. Αν ο χρήστης δεν έχει απαντήσει όλες τις ερωτήσεις, θα του γίνει ερώτηση για το αν θέλει να ολοκληρώσει το τεστ ή όχι. Σε περίπτωση που ο χρήστης δεν έχει ολοκληρώσει το τεστ, αλλά ο χρόνος διεξαγωγής του τεστ έχει τελειώσει, το τεστ θα σταματήσει. Όταν ολοκληρωθεί το τεστ, είτε λίγο χρόνου είτε από τον χρήστη, θα εμφανιστεί το παρακάτω μήνυμα στο οποίο θα ενημερώνεται ο χρήστης για το αν έχει επιτύχει ή όχι στο τεστ.

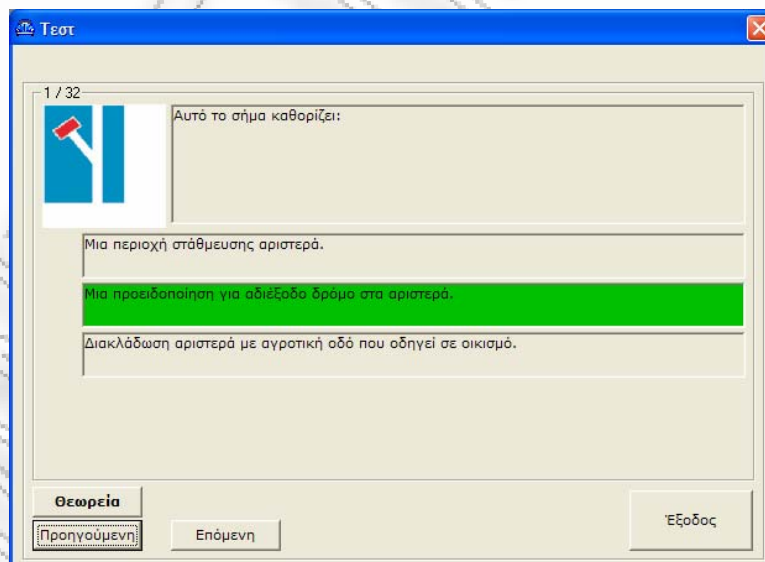


2. Ιστορικό. Επιλέγοντας αυτή την εντολή εμφανίζεται το παρακάτω παράθυρο μέσω του οποίου ο χρήστης μπορεί να δει όλα τα τεστ που έχει κάνει μέχρι τώρα. Επιλέγει λοιπόν το τεστ που θέλει με βάση τον τύπο, τη διάρκεια και την ημερομηνία διεξαγωγής του και πατάει διπλό κλικ πάνω του.

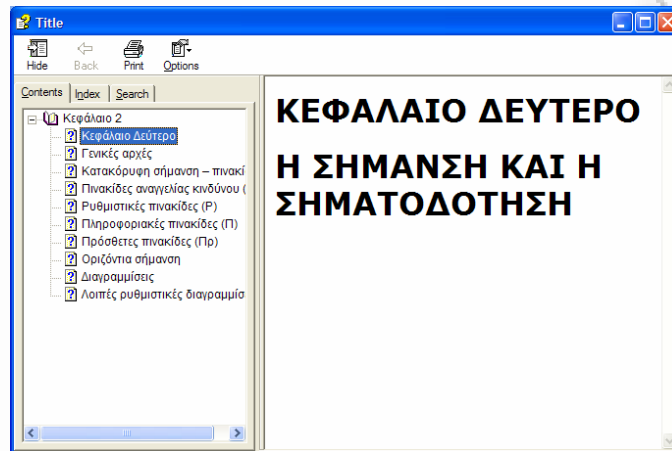


Το τεστ ανοίγει και ο χρήστης χρησιμοποιώντας τα κουμπιά «επόμενη» και «προηγούμενη» μπορεί να πλοηγηθεί και να ελέγξει. Όσον αφορά τις απαντήσεις αυτές, μπορεί να είναι σωστές, λάθος ή και να μην έχουν απαντηθεί οι ερωτήσεις.

Σωστές απαντήσεις: εφόσον ο χρήστης έχει δώσει σωστή απάντηση σε μια ερώτηση, το πεδίο της απάντησης είναι πράσινο, ενώ ταυτόχρονα θα εμφανιστεί και το κουμπί «Θεωρεία».

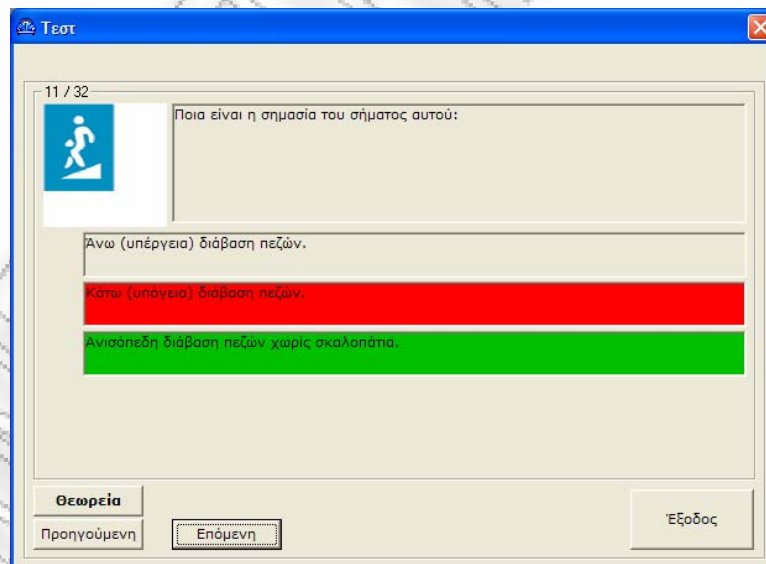


Πατώντας το κουμπί αυτό εμφανίζεται το κομμάτι της θεωρίας που αντιστοιχεί σε αυτή την ερώτηση.



Το κουμπί αυτό εμφανίζεται και σε περίπτωση που η απάντηση είναι λάθος ή και αν δεν έχει δοθεί απάντηση σε μια ερώτηση.

Λάθος απαντήσεις: εάν ο χρήστης έχει δώσει λάθος απάντηση, το πεδίο της απάντησής του είναι κόκκινο ενώ το πεδίο της σωστής απάντησης είναι πράσινο.



Αναπάντητες ερωτήσεις: εάν ο χρήστης δεν απαντήσει σε μια ερώτηση, το πεδίο της σωστής απάντησης θα είναι πράσινο.