



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Χρήστου Χαρ. Κολόσακα
ΜΠΠΛ 13039

**Σύστημα Διδασκαλίας για την εκμάθηση βασικών
εννοιών Πληροφορικής**
**A tutoring system for learning computing
fundamentals**

Επιβλέπων: Επίκουρος Καθηγητής Ευθύμιος Αλέπης

Πειραιάς Νοέμβριος 2015

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

ΠΕΡΙΛΗΨΗ

Η παρούσα μεταπτυχιακή διατριβή πραγματεύεται την κατασκευή ενός συστήματος διδασκαλίας για εκμάθηση βασικών εννοιών της πληροφορικής. Το συγκεκριμένο εκπαιδευτικό σύστημα δίνει τη δυνατότητα στους μαθητές να περιηγηθούν ελεύθερα στα κεφάλαια διδασκαλίας του συστήματος, να αξιολογήσουν τη γνώση τους και τους δίνει βοήθεια αναφορικά με πιθανές λανθασμένες απαντήσεις σε ένα περιβάλλον φιλικό προς το χρήστη. Επιπρόσθετα, ο καθηγητής (διαχειριστής) έχει τη δυνατότητα, μέσω του συστήματος, να προσθέσει, να αφαιρέσει ή να αλλάξει τη θεωρία και τις ασκήσεις του κάθε κεφαλαίου. Επιπλέον, το σύστημα παρέχει τη δυνατότητα ελέγχου της προόδου των μαθητών μέσω της εμφάνισης των στατιστικών στοιχείων τους. Συμπερασματικά, το σύστημα αυτό αποτελεί μία προσπάθεια για την ταχέως αναπτυσσόμενη ερευνητική περιοχή «Πληροφορική στην Εκπαίδευση».

ABSTRACT

This post-graduate dissertation deals with the construction of a tutoring system for learning the computing fundamentals. This educational system offers the possibility of students' navigating freely to the chapters, evaluating their level of knowledge and being assisted in case of wrong answers, in a user friendly environment. Furthermore, the instructor (administrator) can add, remove or edit the theory and the exercises of each chapter via the authoring tool provided. Moreover, the students' progress can easily monitored through the system. Conclusively, this dissertation is an effort towards the research area of "Informatics in Education".

ΠΕΡΙΕΧΟΜΕΝΑ

- 1^ο :ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ
 - 1.1 Εισαγωγή
 - 1.2 Η διαφορά CAI και ICAI (ή ITS)
 - 1.3 Στόχοι των Έξυπνων Διδακτικών Συστημάτων
 - 1.4 Ο ρόλος των μοντέλων
 - 2^ο: ΕΓΧΕΙΡΙΔΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΤΗ
 - 2.1 ΓΕΝΙΚΑ
 - I. Εισαγωγή
 - II. Στόχος
 - 2.2 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ
 - I. Για την Εφαρμογή
 - II. Για τον Χρήστη
 - III. Αρχικοί Περιορισμοί του Συστήματός μας
 - IV. Αρχικοί Στόχοι
 - 2.3 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ
 - 2.4 ΕΠΕΞΗΓΗΣΗ ΚΥΡΙΩΣ ΠΡΟΓΡΑΜΜΑΤΟΣ
 - I. Γενικά Στοιχεία
 - II. Λίγα λόγια για τη Visual Basic
 - III. Σκελετός του Προγράμματος
 - 3^ο: ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ
 - 3.1 ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ
 - I. Γενικά για το πρόγραμμα
 - II. Απαιτήσεις συστήματος
 - III. Χαρακτηριστικά της εφαρμογής
 - 3.2 ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ
 - 3.3 ΛΕΙΤΟΥΡΓΙΕΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ
 - I. Είσοδος μαθητή στο σύστημα
 - II. Είσοδος καθηγητή στο σύστημα
 - 4^ο :ΣΥΜΠΕΡΑΣΜΑΤΑ
 - 4.1 ΑΔΥΝΑΜΙΕΣ ΚΑΙ ΕΛΛΕΙΨΕΙΣ
 - 4.2 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΟΠΤΙΚΕΣ
- ΒΙΒΛΙΟΓΡΑΦΙΑ

1^ο : ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

1.1 Εισαγωγή

Όταν χρησιμοποιούμε την φράση πληροφορική στην εκπαίδευση εννοούμε τους διάφορους τρόπους που μπορούμε να χρησιμοποιήσουμε την τεχνολογία της πληροφορικής για να εξυπηρετηθεί ο στόχος της εκπαίδευσης. Στην ουσία δηλαδή μιλάμε για εκπαιδευτικό λογισμικό, δηλαδή για προγράμματα και περιβάλλοντα ανάπτυξης λογισμικού που έχουν να κάνουν με την εκπαίδευση. Με άλλα λόγια η ερώτηση που μας απασχολεί είναι πώς μπορεί ο υπολογιστής να διδάξει κάτι στον χρήστη. Οι πρωταρχικές έννοιες που σχετίζονται με το εκπαιδευτικό λογισμικό είναι:

- 1) Η αλληλεπίδραση ανθρώπου υπολογιστή
- 2) Το πεδίο της γνώσης που πρέπει να μεταδοθεί στον χρήστη από τον υπολογιστή.
- 3) Ο χρήστης που σ' αυτή την περίπτωση θεωρείται μαθητής, διότι διδάσκεται κάτι από τον υπολογιστή.
- 4) Η στρατηγική που ακολουθεί ο υπολογιστής για να μεταδώσει τη γνώση στον μαθητή-χρήστη.

Όλες οι έρευνες και τα αποτελέσματα που έχουν προκύψει μέχρι τώρα με θέμα το εκπαιδευτικό λογισμικό έχουν γίνει σε τομείς της Πληροφορικής που αναγνωρίζονται από τα εξής ονόματα:

- 1) CAI (Computer-Aided Instruction)
- 2) ICAI (Intelligent Computer-Aided Instruction) ή αλλιώς ITS (Intelligent Tutoring Systems).

Οι δύο τομείς CAI και ICAI έχουν σαν κοινό στόχο την ανάπτυξη μεθόδων και εφαρμογών για εκπαιδευτικό λογισμικό αλλά διαφέρουν στο γεγονός ότι ο πρώτος τομέας (CAI) χρησιμοποιεί παραδοσιακές μεθόδους προγραμματισμού ενώ ο δεύτερος τομέας (ICAI) εμπεριέχει τη χρήση τεχνικών Τεχνητής Νοημοσύνης. Συνήθως θα αναφερόμαστε στο εκπαιδευτικό λογισμικό που χρησιμοποιεί Τεχνητή Νοημοσύνη με τον όρο TS (Intelligent Tutoring Systems) δηλαδή Έξυπνα Διδακτικά Συστήματα.

1.2 Η διαφορά CAI και ICAI (ή ITS)

Τα παραδοσιακά προγράμματα CAI είναι οργανωμένα στατικά έτσι ώστε να ενσωματώνουν τη γνώση γύρω από το πεδίο (domain) και την παιδαγωγική γνώση ενός έμπειρου δάσκαλου. Μια παρομοίωση που χρησιμοποιεί ο Wenger για να δείξει τη δομή και χρησιμότητα κάποιου εκπαιδευτικού λογισμικού που είναι CAI, είναι αυτή με το βιβλίο. Το λογισμικό CAI παρομοιάζεται με ένα βιβλίο. Το κάθε βιβλίο περιέχει τη γνώση του συγγραφέα.

Τα βιβλία έχουν ευκολίες όπως:

- 1) γραμμές
- 2) σελίδες
- 3) παράγραφοι
- 4) κεφάλαια
- 5) περιεχόμενα
- 6) σχήματα, τα οποία είναι εργαλεία για την οργάνωση και παρουσίαση της ύλης.

Αντίστοιχα ένα πρόγραμμα μπορεί να χρησιμοποιεί εργαλεία για την παρουσίαση του περιεχομένου. Ένα βιβλίο μπορεί να είναι αρκετά σύνθετο στον τρόπο παρουσίασης. Δηλαδή πολλές φορές ο συγγραφέας μπορεί να προτείνει στον αναγνώστη να παραλείψει κάποιες ενότητες εάν το επίπεδο του είναι πιο προχωρημένο, δημιουργώντας έτσι μία διαφοροποίηση

μεταξύ αναγνωστών διαφορετικού επιπέδου. Όμως δεν μπορούμε να περιμένουμε από ένα βιβλίο να αναπροσαρμόζει δυναμικά την οργάνωση της ύλης του ούτε να απαντά σε ερωτήσεις που δεν αναμένονται πλήρως από πριν. (Μπορεί να υπάρχουν σωστές απαντήσεις σε ερωτήσεις που είναι διατυπωμένες μέσα στο βιβλίο). Αντίθετα, με τη χρήση Τεχνητής Νοημοσύνης (οπότε έχουμε την περίπτωση του ICAI ή ITS) ενδιαφερόμαστε να κωδικοποιήσουμε την ίδια τη γνώση του εμπειρογνώμονα και όχι τις αποφάσεις του εμπειρογνώμονα στις οποίες οδηγεί η γνώση του.

Τα όρια μεταξύ CAI και ICAI δεν είναι απολύτως σαφή γι' αυτό από εδώ και στο εξής θα αναφερόμαστε στις τεχνικές του εκπαιδευτικού λογισμικού στα πλαίσια των Έξυπνων Διδακτικών Συστημάτων (ITS) τα οποία προϋποθέτουν μια όσο το δυνατόν μεγαλύτερη προσαρμοστικότητα στις ανάγκες της διδασκαλίας κατά τη διάρκεια της εκτέλεσης του προγράμματος.

1.3 Στόχοι των Έξυπνων Διδακτικών Συστημάτων

Ο στόχος ενός ΕΔΣ δεν είναι να αντικαταστήσει τον άνθρωπο στη διδασκαλία ενός θέματος. (Αυτό άλλωστε δεν είναι δυνατόν να γίνει). Ένα ΕΔΣ μπορεί να χρησιμεύσει σαν ένα μό «ζωντανό» βιβλίο το οποίο μπορεί να επικοινωνεί και να αλληλεπιδρά με τον αναγνώστη, να θέτει ερωτήσεις, να απαντά σε ερωτήσεις, να προσαρμόζει το περιεχόμενό του στο επίπεδο του αναγνώστη κ.λ.π.

Τα μέσα τα οποία μπορεί να χρησιμοποιήσει ένα ΕΔΣ για να μεταδώσει τη γνώση περιλαμβάνουν :

- 1) Μία περιορισμένη γλώσσα αλληλεπίδρασης (η οποία δεν μπορεί ποτέ να φθάσει τη πολυπλοκότητα της φυσικής γλώσσας).
- 2) Κείμενο.
- 3) Γραφικά οθόνης.
- 4) Ασκήσεις και ερωτήσεις κ.λ.π.

1.4 Ο ρόλος των μοντέλων

Ένα μοντέλο παρέχει ένα γενικό καλούπι συμπεριφοράς και γνώσης. Στην προκειμένη περίπτωση στα ΕΔΣ χρειαζόμαστε:

- 1) μοντέλο του πεδίου (domain) γνώσης.
- 2) μοντέλο του μαθητή
- 3) μοντέλο της στρατηγικής διδασκαλίας.

1.5 Συναφείς τομείς και επιστήμες που παίζουν ρόλο

Η ανάπτυξη ενός ΕΔΣ χρειάζεται να «δανειστεί» γνώσεις και από άλλες επιστήμες εκτός από την Πληροφορική και χρειάζεται επίσης να δώσει ιδιαίτερη έμφαση σε κάποιους τομείς της Πληροφορικής, όπως είναι η Αλληλεπίδραση Ανθρώπου Υπολογιστή (Human Computer Interaction). Η Ψυχολογία και ειδικά η Παιδαγωγική Ψυχολογία, οι Γνωστικές Επιστήμες είναι πολύ συνδεδεμένες με τα ΕΔΣ. Επίσης η Γλωσσολογία, ειδικά όταν έχουμε την ανάπτυξη συστήματος διεπαφής το οποίο χρησιμοποιεί φυσική γλώσσα.

2° : ΕΓΧΕΙΡΙΔΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΤΗ

2.1 ΓΕΝΙΚΑ

I. Εισαγωγή

Η εργασία αυτή πραγματοποιήθηκε στα πλαίσια του μαθήματος «Εκπαιδευτικό Λογισμικό» του Τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς.

II. Στόχος

Θέμα της εργασίας είναι η κατασκευή μιας εφαρμογής η οποία θα χρησιμοποιείται από τους χρήστες για εκπαιδευτικούς σκοπούς, ενώ παράλληλα θα κρατάει και χρήσιμες πληροφορίες γι' αυτούς.

Το κυρίως πρόγραμμα δημιουργήθηκε σε Visual Basic και η Βάση Δεδομένων σε MS-Access.

2.2 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

I. Για την Εφαρμογή

Η εφαρμογή μας έγινε με σκοπό την διδασκαλία του μαθήματος της Πληροφορικής της Α' Γυμνασίου στους μαθητές της τάξης αυτής. Η δημιουργία του προγράμματος κατευθύνθηκε σε μεγάλο βαθμό από το αντίστοιχο βιβλίο της πρώτης γυμνασίου ενώ σε αρκετές περιπτώσεις στηρίχθηκε εξ' ολοκλήρου στο βιβλίο αυτό.

Επειδή το βιβλίο δεν ήταν αρκετό για το σχεδιασμό του προγράμματος καθώς ορισμένα από τα καθήκοντα ενός καθηγητή δεν αναγράφονται στο βιβλίο παράδοσης. Έτσι, λοιπόν, έγιναν αρκετές συζητήσεις με εκπαιδευτικούς, οι οποίοι μας έδωσαν μια καλύτερη και πιο λεπτομερή εικόνα για τα καθήκοντα τους απέναντι στους μαθητές και τελικά αποφασίστηκε ότι οι απαιτήσεις που θα έπρεπε να καλύπτονται από την εφαρμογή ήταν:

Στοιχεία Μαθητών

- ✓ Εγγραφή Μαθητή
- ✓ Ανάκτηση Στοιχείων Μαθητή

Βαθμολογίες

- ✓ Καταχώρηση Βαθμολογιών
- ✓ Παρακολούθηση Προόδου Μαθητών

Κεφάλαια

- ✓ Θεωρία Κεφαλαίου
- ✓ Ασκήσεις Κεφαλαίου

Ασκήσεις Μαθημάτων

- ✓ Καταγραφή Απαντήσεων
- ✓ Έλεγχος Ορθότητας

Εξυπηρέτηση Καθηγητών

- ✓ Εύρεση πληροφοριών για τους Μαθητές
- ✓ Παρακολούθηση πορείας Μαθητών
- ✓ Ανανέωση-Διαγραφή Ασκήσεων Μαθημάτων
- ✓ Ανανέωση- Διαγραφή Θεωρίας Μαθημάτων

Αναλυτικότερα :

Η εφαρμογή μας στοχεύει στην διδασκαλία του μαθήματος της πληροφορικής της Α' Γυμνασίου στους μαθητές της τάξης αυτής. Οι μαθητές την πρώτη φορά που έρχονται σε επαφή

με την εφαρμογή πρέπει να εγγραφούν, ώστε να μπορέσουν γίνουν χρήστες του προγράμματος. Κατά την εγγραφή, τους ζητούνται οι απαραίτητες πληροφορίες που χρειάζονται για τη χρήση του προγράμματος και τις οποίες κρίνει απαραίτητες και ο καθηγητής τους ώστε να βγάλει τα κατάλληλα συμπεράσματα.

Στη συνέχεια, δηλαδή στο τέλος κάθε μαθήματος όπως και στο σχολικό βιβλίο ο μαθητής καλείται να λύσει ασκήσεις, οι οποίες βασίζονται στο μάθημα το οποίο διδάχτηκε ο μαθητής. Μετά το τέλος των μαθημάτων και των ασκήσεων του κάθε μαθήματος, ο μαθητής μπορεί να δει τα αποτελέσματα των ασκήσεων.

Η εφαρμογή δεν δημιουργήθηκε μόνο για την εξυπηρέτηση των μαθητών, αλλά και για την εξυπηρέτηση των καθηγητών τους. Οι καθηγητές των μαθητών μπορούν να δουν την πρόοδο τους παρατηρώντας τις απαντήσεις στις ασκήσεις τις οποίες έλυσαν. Επιπλέον, ο καθηγητής μπορεί να δει και να αλλάξει τις ασκήσεις οι οποίες έχουν καταχωρηθεί στην εφαρμογή και τις οποίες καλείται ο μαθητής να λύσει, να προσθέσει νέες και να διαγράψει υπάρχουσες.

II. Για τον Χρήστη

Ο κύριος στόχος πάνω στον σχεδιασμό ενός συστήματος διεπαφής (interface) είναι η μέγιστη δυνατή χρησιμοποίησιμότητα. Οι κανόνες σχεδιασμού ενός συστήματος στηρίζονται πάνω στις τρεις βασικές αρχές της χρησιμοποίησιμότητας. Αυτές είναι:

- ✓ Ευκολία εκμάθησης
- ✓ Ευκαμψία
- ✓ Ανθεκτικότητα

Σύμφωνα λοιπόν με τα παραπάνω, το σύστημα πρέπει να εκπληρώνει τους κανόνες σχεδιασμού που ακολουθούν. Όσον αφορά την αρχή της ευκολίας εκμάθησης θα πρέπει να δώσουμε ιδιαίτερη προσοχή στους εξής κανόνες:

- ❖ Το σύστημα μας πρέπει να είναι συνεπές. Αυτό με άλλα λόγια, σημαίνει ότι το σύστημα πρέπει να είναι συνεπές στις δομές που χρησιμοποιεί καθ' όλη την έκταση του. Δηλαδή, η δομή δεν πρέπει να αλλάζουν εύκολα και σημαντικά. Επιπλέον, τα χρώματα, αλλά και τα εικονίδια πρέπει να διατηρούν την μορφή τους. Με αυτόν τον τρόπο η αίσθηση της όρασης συνηθίζει σε ένα μοντέλο που είναι μετά δύσκολο να ξεχάσει, και έτσι δεν δημιουργείται καμία σύγχυση στον χρήστη.
- ❖ Να δίνει απαντήσεις που βγάζουν νόημα. Τα μηνύματα, δηλαδή, που δέχεται ένας χρήστης μετά από κάποια ενέργεια πρέπει να είναι περιεκτικά και να έχουν τόσο, όλη την πληροφορία που χρειάζεται, όσο και τα βήματα που πρέπει να ακολουθήσει από κει και πέρα. Ιδιαίτερα αν το μήνυμα που δέχεται είναι μήνυμα λάθους.
- ❖ Να ελαττώνει τις πληροφορίες που χρειάζονται απομνημόνευση. Ο χρήστης δεν θα πρέπει να αναγκάζεται να θυμάται μια σειρά από στοιχεία για να ολοκληρώσει μια ενέργεια. Η μετάβαση από μία ενέργεια στο αποτέλεσμα της δεν πρέπει να γίνεται με έντονη χρήση της μνήμης μικρής διάρκειας και φυσικά είναι απαγορευτική η χρήση της μνήμης μεγάλης διάρκειας του ανθρώπου.
- ❖ Να οργανώνει με λογικό τρόπο την γεωγραφία της οθόνης. Η τυχαία σειρά κουμπιών, εικονιδίων και άλλων χρήσιμων αντικειμένων στην οθόνη, που αλληλεπιδρούν με τον χρήστη, ελαττώνει τόσο την ικανότητα διαχωρισμού όσο και την αφαιρετική ικανότητα του χρήστη. Αυτό το σημείο δεν βοηθά καθόλου στην ευκολία εκμάθησης.
- ❖ Να υπάρχει κείμενο ενεργής βοήθειας. Το εγχειρίδιο χρήσης και η on-line βοήθεια εδώ παίζουν το σημαντικότερο ρόλο στην αρχή ευκολίας στην εκμάθηση.

Η δεύτερη κατηγορία κανόνων που ακολουθούν αναφέρεται στην αρχή της ευκαμψίας. Οι κανόνες είναι οι παρακάτω:

- ❖ Καταρχήν ο χρήστης θα πρέπει να έχει την άνεση να επικοινωνήσει με πολλούς τρόπους με το σύστημα. Το ιδανικό θα ήταν να μπορεί να ενεργήσει όπως θέλει είτε με το ποντίκι είτε με το πληκτρολόγιο.
- ❖ Η εφαρμογή θα πρέπει να κατηγοριοποιεί τους χρήστες. Αυτό σημαίνει ότι οι αρχάριοι χρήστες θα πρέπει να δέχονται μεγαλύτερες ευκολίες από το πρόγραμμα με χρήση βοήθειας, ετικετών πάνω στα κουμπιά και παρουσίασης χρήσης της εφαρμογής. Οι μεσαίου επιπέδου χρήστες πρέπει να δέχονται βοήθεια όποτε αυτοί το επιθυμούν χωρίς αυτή να γίνεται ενοχλητική. Τέλος, οι έμπειροι χρήστες πρέπει να έχουν την δυνατότητα να χρησιμοποιήσουν συντομεύσεις για τις ενέργειες τους. Βέβαια, οι διακρίσεις αυτές δεν είναι τόσο εμφανείς σε μία εφαρμογή που θα χρησιμοποιηθεί από δύο ή τρεις (το πολύ) ανθρώπους.
- ❖ Επιπλέον, ο χρήστης θα πρέπει να έχει τη δυνατότητα να περάσει από μία εργασία σε μια άλλη με ευκολία και με πολλούς τρόπους. Αυτό δίνει την δυνατότητα για γρηγορότερα μονοπάτια μεταξύ των εργασιών και συνεπώς, εξοικονόμηση χρόνου εργασίας.

Η τρίτη κατηγορία κανόνων είναι οι κανόνες που ελέγχουν τον χρήστη και τον σταματούν πριν από μια καταστροφική πράξη που θα σβήσει πολλά αναγκαία δεδομένα. Εδώ κρίνεται η ικανότητα ανθεκτικότητας ενός συστήματος σύμφωνα με τους ακόλουθους κανόνες:

- ❖ Να ζητά επαλήθευση πριν από κάθε καταστροφική εντολή. Αυτή η συζήτηση χρήστη και συστήματος μέσω μηνυμάτων διαλόγου βοηθά τον χρήστη να αντιληφθεί πριν είναι πολύ αργά την καταστροφική ενέργεια του, το αντίστοιχο αποτέλεσμα της και να προσπαθήσει έστω και για τελευταία στιγμή να το αποτρέψει.
- ❖ Να επιτρέπει την αντιστροφή εντολών. Οι εντολές τύπου «Undo» παίζουν τα τελευταία χρόνια ένα σημαντικό παράγοντα για την ανθεκτικότητα μιας εφαρμογής. Ακόμα, και να εκτελεστεί μια καταστροφική πράξη για το σύστημα, μπορούν να το επαναφέρουν στην προηγούμενη κατάσταση του σώζοντας πολλές φορές πολύτιμα δεδομένα.
- ❖ Να δίνονται μηνύματα λάθους. Όταν χρήστης εκτελεί μια λάθος εντολή πρέπει να μπλοκάρεται και να μην αφήνεται από το σύστημα να προχωρήσει παρακάτω για να συνεχίσει την εκτέλεση της υπόλοιπης διαδικασίας που θα έχει λανθασμένο αποτέλεσμα. Το μήνυμα πρέπει να έχει νόημα για το χρήστη ώστε ο χρήστης να μπορεί ο ίδιος να καταλάβει τόσο το λάθος του, όσο και σε ποιο σημείο βρίσκεται.
- ❖ Τέλος να «συγχωρά» τα λάθη. Κάποια λάθη μπορούν να παίζουν σημαντικό ρόλο για το σύστημα εκείνη την στιγμή. Αυτά θα πρέπει να αγνοούνται ώστε να μην κουράζεται ο χρήστης και να θέτονται σε πρωταρχικό ρόλο όποτε αυτό είναι απολύτως αναγκαίο.

Αυτοί οι κανόνες μας βοηθούν να σχηματίσουμε στο νου μας το βασικό κορμό σχεδιασμού ενός συστήματος διεπαφής χρήστη. Στην συνέχεια θα δούμε την υλοποίηση του δικού μας συστήματος σε συνδυασμό με αυτούς τους κανόνες των τριών βασικών αρχών.

III. Αρχικοί Περιορισμοί του Συστήματός μας

Αρχικά, οι πρώτοι περιορισμοί που τέθηκαν ήταν ότι το υπό κατασκευή σύστημα δεν πρέπει κατά τη λειτουργία του να καταναλώνει όλους τους πόρους του συστήματος. Αυτό θα βοηθήσει στο να μην καταρρέει εύκολα το σύστημα και στο να μην περιορίζει το χρήστη. Για παράδειγμα, η αποθήκευση των δεδομένων δεν πρέπει να τον καθυστερεί από τις υπόλοιπες εργασίες του.

Επιπλέον, έπρεπε να θυμόμαστε σε κάθε στάδιο υλοποίησης της εφαρμογής ότι ο χρήστης δεν είχε ιδιαίτερες γνώσεις υπολογιστών. Για να καταφέρουμε να δημιουργήσουμε ένα καλό και ολοκληρωμένο σύστημα έπρεπε να ορίσουμε κάποιους αρχικούς στόχους, κάτι το οποίο παραθέτουμε πιο κάτω.

IV. Αρχικοί Στόχοι

Έτσι, λοιπόν, σύμφωνα με τα παραπάνω, ένας από τους πρώτους και βασικότερους στόχους για τον σχεδιασμό του συστήματος ήταν να είναι ένα σύστημα απλό και κυρίως φιλικό προς το χρήστη. Για αυτό τον λόγο έπρεπε να δημιουργηθεί ένα σύστημα σε περιβάλλον που είναι διαδεδομένο και που είναι εύκολο στην εκμάθηση. Το καταλληλότερο περιβάλλον για να καλύψει την απαίτηση αυτή, είναι αδιαμφισβήτητα αυτό των Windows XP. Έτσι λοιπόν αποφασίσαμε ότι το σύστημα θα αλληλεπιδρά με το χρήστη με οθόνες – παράθυρα.

Άλλοι στόχοι που τέθηκαν ήταν :

- ❖ Η μεταφερισιμότητα της εφαρμογής : Δηλαδή, η ευκολία με την οποία το λογισμικό μπορεί να μεταφερθεί από έναν υπολογιστή σε άλλο ή από ένα περιβάλλον σε άλλο.
- ❖ Η αξιοπιστία : Το πρόγραμμα πρέπει να είναι ικανό να εκτελεί τις λειτουργίες για τις οποίες σχεδιάστηκε πληρώνοντας ορισμένες προϋποθέσεις, με ασφάλεια για τα δεδομένα και χωρίς να ξεπερνά τον προδιαγεγραμμένο χρόνο.
- ❖ Η αποδοτικότητα : Το λογισμικό πέρα από την ταχύτητα θα πρέπει να είναι ικανό να εκτελέσει τις λειτουργίες του καταναλώνοντας όσο το δυνατόν λιγότερους πόρους. Εφόσον μας ενδιαφέρει η μεταφερισιμότητα θα πρέπει το πρόγραμμα να μπορεί να εκτελείται και σε υπολογιστές με λιγότερες δυνατότητες σε hardware.
- ❖ Η ακρίβεια : Σε αυτή περιλαμβάνονται η εκτίμηση της μη ύπαρξης λαθών και το ποσοτικό μέτρο του μεγέθους ενός λάθους. Το λογισμικό πρέπει να είναι ικανό να αναγνωρίζει μία ασυμβατότητα μεταξύ μίας τιμής ή συνθήκης όπως έχει προκύψει από τον υπολογιστή και της πραγματικής, καθορισμένης ή θεωρητικά σωστής τιμής ή συνθήκης.
- ❖ Η ευρωστία : δηλαδή ο βαθμός στον οποίο το λογισμικό μπορεί να λειτουργήσει σωστά παρά την εισαγωγή μη έγκυρων δεδομένων.
- ❖ Η ορθότητα : Η ορθότητα αποτελείται από τρία επιμέρους στοιχεία
 - τον βαθμό στον οποίο το λογισμικό είναι απαλλαγμένο από σχεδιαστικές ατέλειες και ατέλειες κωδικοποίησης,
 - τον βαθμό στον οποίο το λογισμικό επιτυγχάνει τις καθορισμένες απαιτήσεις και τέλος
 - τον βαθμό στον οποίο το λογισμικό ικανοποιεί τις προσδοκίες του χρήστη.
- ❖ Τέλος, το κόστος : Ο υπολογισμός του κόστους είναι μία από τις πιο βασικές και πιο δύσκολες δουλειές κατά το σχεδιασμό του έργου, για αυτό και θα πρέπει να είμαστε ιδιαίτερα προσεκτικοί. Ο υπολογισμός κόστους είναι αναγκαίος για να γίνει ανάληψη ενός έργου, για αυτό και αποτελεί βασικό κριτήριο αποδοχής του συστήματος. Οι παράγοντες που καθορίζουν το κόστος είναι :

- η ικανότητα των προγραμματιστών
- η πολυπλοκότητα του προϊόντος
- το μέγεθος του προϊόντος
- ο διαθέσιμος χρόνος
- η απαιτούμενη αξιοπιστία
- το επίπεδο τεχνολογίας

Επιπλέον θέλουμε να τονίσουμε ότι εξαρχής κύριο μέλημα μας ήταν η υλοποίηση ενός πληροφοριακού συστήματος το οποίο θα προσέφερε όσο το δυνατόν περισσότερα στους αποδέκτες του και φυσικά θα ξεπερνούσε τα προβλήματα του ήδη υπάρχοντος.

2.3 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

Ένα από τα σημαντικότερα κομμάτια, αν όχι το σημαντικότερο, κάθε συστήματος ή στην προκειμένη περίπτωση της εφαρμογής μας, είναι αναμφίβολα η βάση δεδομένων στην οποία καταχωρούνται όλα όσα χρειάζονται για την εύρυθμη λειτουργία της εφαρμογής μας.

Για το πρόγραμμα μας, τα δεδομένα είναι το ότι πιο πολύτιμο έχει! Είναι λοιπόν απαραίτητο, αυτά να οργανωθούν με το πιο σωστό τρόπο ώστε:

- ✓ Να μην υπάρχει ούτε καν υποψία απώλειας δεδομένων
- ✓ Να αποφεύγεται η ύπαρξη περιττών δεδομένων (π.χ. διπλοεγγραφές)
- ✓ Η αναζήτησή τους μέσα στη βάση να γίνεται με τον ταχύτερο τρόπο
- ✓ Και φυσικά να εμφανίζονται πάντα τα ζητούμενα από το χρήστη

Σε αυτή λοιπόν την κατεύθυνση μας βοήθησε σημαντικά η Θεωρία Σχεδίασης Σχεσιακών Βάσεων Δεδομένων.

Σημείωση : Για την ορθή λειτουργία της εφαρμογής, είναι απαραίτητη η τοποθέτηση της βάσης σε ένα φάκελο με το όνομα Database. Αυτός ο φάκελος πρέπει να τοποθετηθεί στον ίδιο φάκελο που έχει μέσα την εφαρμογή.

Οι βασικοί πίνακες της βάσης μας με τα χαρακτηριστικά τους είναι οι εξής:

- ANSWERS
 - ANS_ID, ANS_STU_ID, ANS_EXE_ID, ANS_RATE
- CHAPTERS
 - CHA_ID, CHA_NUMB_CHA_TITLE
- EXERCISE
 - EXE_ID, EXE_NUMBER, EXE_TEXT, EXE_ANSWER, EXE_CHAPTER, EXE_TYPE, EXE_CHOICES
- IMAGE
 - IMA_ID, IMA_ORDER, IMA_NUMBER, IMA_PATH, IMA_CHA_ID
- STUDENT
 - STU_ID, STU_NAME, STU_PASSWORD, STU_TEA_ID
- TEACHER
 - TEA_ID, TEA_NAME, TEA_PASSWORD.

Ακολουθούν screenshots με τους πίνακες :

| Πίνακες | | ANSWERS | | | |
|---------|--------|------------|------------|----------|----------|
| ANSWERS | ANS_ID | ANS_STU_IC | ANS_EXE_IC | ANS_DATE | ANS_RATE |
| | 1 | 7 | 14 | 0 | 1 |
| | 2 | 7 | 15 | 0 | 1 |
| | 3 | 7 | 24 | 0 | 1 |
| | 4 | 7 | 17 | 0 | 0 |
| | 5 | 7 | 18 | 0 | 0 |
| | 6 | 7 | 19 | 0 | 0 |
| | 7 | 7 | 20 | 0 | 0 |
| | 8 | 7 | 23 | 0 | 0 |
| | 9 | 7 | 22 | 0 | 1 |
| | 10 | 5 | 14 | 0 | 0 |
| | 11 | 8 | 25 | 0 | 1 |
| | 12 | 8 | 13 | 0 | 1 |
| | 13 | 8 | 12 | 0 | 0 |
| | 14 | 8 | 10 | 0 | 0 |
| | 15 | 8 | 11 | 0 | 0 |
| | 16 | 4 | 14 | 0 | 1 |
| | 17 | 4 | 20 | 0 | 0 |
| | 18 | 4 | 22 | 0 | 0 |
| | 19 | 4 | 18 | 0 | 1 |
| | 20 | 9 | 13 | 0 | 0 |

- Ο πίνακας ANSWERS αναφέρεται στις απαντήσεις που δίνονται στις εργασίες και συνδέεται με τους πίνακες STUDENT και EXERCISE.

| Πίνακες | | CHAPTERS | | |
|----------|--------|----------|-----------------|--|
| CHAPTERS | CHA_ID | CHA_NUMB | CHA_TITLE | |
| | 1 | | 1 EISAGOGH | |
| | 10 | | 2 Τα βασικά μερ | |
| | 11 | | 3 MS-OFFICE | |
| | * | (Νέο) | 0 | |

- Ο πίνακας CHAPTERS αναφέρεται στα διδασκόμενα κεφάλαια από τους καθηγητές.

| EXE_ID | EXE_N | EXE_TEXT | EXE_A | EXE_CHA | EXE_T | EXE_CHOISE |
|--------|-------|--|-------|---------|-------|---------------------|
| 10 | 6 | ΤΙ ΔΕΝ ΕΙΝΑΙ ΜΕΡΟΣ ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ | 1 | 9 | 0 | ΡΛΗΚΤΡΟΛΟΓΙΟ Τ |
| 11 | 7 | ΡΟΙΑ ΑΡΟ ΤΙΣ ΡΑΡΑΚΑΤΟ ΕΙΝΑΙ ΜΝΗΜΕΣ | 0 | 9 | 0 | ΡΑΜ ΔΙΣΚΕΤΑ ΚΑ |
| 12 | 2 | ΤΑ 1000 ΜΕΓΑ ΒΥΤΕ ΕΙΝΑΙ ΙΣΑ ΜΕ | 2 | 9 | 0 | 40 ΒΙΤ 23 ΓΙΓΑ ΒΥ |
| 13 | 1 | ΡΟΙΑ ΑΡΟ ΤΑ ΡΑΡΑΚΑΤΟ ΔΕΝ ΔΕΝ ΕΙΝΑΙ | 0 | 9 | 0 | ΗΧΕΙΟ ΜΙΚΡΟΦΟΝ |
| 14 | 1 | Ο σκληρός δίσκος είναι | 1 | 10 | 0 | συσκευή ήχου συ |
| 15 | 2 | το mega byte είναι | 1 | 10 | 0 | μονάδα ταχύτητα |
| 17 | 4 | το 1 GB είναι ίσο με 10 byte | 0 | 10 | 1 | |
| 18 | 5 | το 1 GB είναι ίσο με 1000 mb | 1 | 10 | 1 | |
| 19 | 6 | το ποντίκι είναι | 0 | 10 | 0 | συσκευή εισόδου |
| 20 | 7 | επιλέξτε τη σωστή ισοότητα | 3 | 10 | 0 | 1 gb=100mb=1000i |
| 22 | 9 | h annoula epezhse ths ekrikshs | 0 | 10 | 1 | |
| 23 | 8 | επιλέξτε το σωστό | 2 | 10 | 0 | το ποντίκι και ο σκ |
| 24 | 3 | το 1 GB είναι ίσο με 100 mb | 0 | 10 | 1 | |
| 25 | 3 | τι είναι i othoni? | 1 | 9 | 0 | siskeui eisodou si |
| 26 | 1 | to word einai meros tou ms office? | 1 | 11 | 1 | |
| 27 | 2 | ποιο απο ta parakatw den einai meros to | 3 | 11 | 0 | word excel acces |
| 28 | 4 | ti einai dedomeno? | 0 | 9 | 0 | kati pou isxiei kat |
| 29 | 5 | i pliroforia einai ayto pou zitame na vrou | 0 | 9 | 1 | |

- Ο πίνακας EXERCISE αναφέρεται στις ασκήσεις (πολλαπλή επιλογή -0- και σωστό-λάθος -1-) που δίνονται από τους καθηγητές στους μαθητές. Συνδέεται με τους πίνακες ANSWERS και CHAPTERS.

| IMA_ID | IMA_ORDER_NUMBER | IMA_PATH | IMA_CHA_ID |
|--------|------------------|----------------|------------|
| 33 | | 1 C:\Documents | 10 |
| 34 | | 1 C:\Documents | 11 |
| 35 | | 1 C:\Documents | 9 |
| * | (Νέο) | 0 | 0 |

- Ο πίνακας IMAGE αναφέρεται στις εικόνες που εισάγονται από τους καθηγητές για θεωρία και συνδέεται με τον πίνακα CHAPTERS.

| STU_ID | STU_NAME | STU_PASSW | STU_TEA_ID |
|--------|----------|-----------|------------|
| 9 | antreas | 123 | 5 |
| 10 | ioanna | 123 | 6 |
| * | (Νέο) | | 0 |

- Ο πίνακας STUDENT αναφέρεται στους μαθητές που παρακολουθούν το μάθημα της Πληροφορικής Α' Γυμνασίου και χρησιμοποιούν την εφαρμογή και συνδέεται με τον πίνακα TEACHER.

| TEA_ID | TEA_NAME | TEA_PASSW |
|--------|----------|-----------|
| 5 | xristos | 123 |
| 6 | nancy | 123 |
| * | (Νέο) | |

- Ο πίνακας TEACHER αναφέρεται στους καθηγητές που διδάσκουν το μάθημα της Πληροφορικής Α' Γυμνασίου.

2.4 ΕΠΕΞΗΓΗΣΗ ΚΥΡΙΩΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

I. Γενικά Στοιχεία

Όπως προαναφέρθηκε, καθ' όλη την διάρκεια υλοποίησης του προγράμματος, ακολουθήθηκαν κάποιες κατευθυντήριες γραμμές. Αυτές ήταν :

Σαφής ονοματολογία μεταβλητών : Κάθε που χρησιμοποιήθηκε στο πρόγραμμα έχει όσο το δυνατόν πιο σαφές και αντιπροσωπευτικό όνομα της λειτουργίας που αυτό επιτελεί. Αυτό αποτελεί μεγάλη βοήθεια για όσους θέλουν να παρατηρήσουν τις διάφορες προγραμματιστικές τεχνικές που έχουν χρησιμοποιηθεί αλλά αποδείχθηκε επίσης σημαντικό εφόδιο στους ίδιους τους δημιουργούς του προγράμματος κυρίως στο μέρος της διόρθωσης ή βελτίωσης κάποιων λεπτομερειών.

Τα χρώματα, που χρησιμοποιήθηκαν στην εφαρμογή (και παρέμειναν σταθερά) επιλέχθηκαν με βάση του τι κρίναμε ότι ήταν περισσότερο ξεκούραστο στο μάτι για τον χρήστη.

II. Λίγα λόγια για τη Visual Basic

Η Visual Basic είναι σήμερα η πιο δημοφιλής *καθοδηγούμενη από συμβάντα (event driven language)* γλώσσα οπτικού προγραμματισμού και συνδυάζει τα χαρακτηριστικά της γνωστής γλώσσας προγραμματισμού Basic για εύκολη εκμάθηση όπως επίσης και πληθώρα εντολών και συναρτήσεων για να ικανοποιήσει και τους πιο απαιτητικούς επαγγελματίες προγραμματιστές. Μπορεί να χρησιμοποιηθεί ακόμη και για δημιουργία εφαρμογών πολυμέσων όπως επίσης και για επεξεργασία αρχείων (βάσεων δεδομένων).

Οι εφαρμογές σε γραφικό περιβάλλον αποτελούνται από μια ή περισσότερες οθόνες με αντικείμενα (objects), όπως πλήκτρα εντολών, λίστες, πλαίσια κειμένου κ.ά., τα οποία

ενεργοποιούνται από κάποιο συμβάν, όπως είναι το πάτημα ενός πλήκτρου από το πληκτρολόγιο ή από το ποντίκι.

Τα βήματα που πρέπει να ακολουθήσουμε για να δημιουργήσουμε μια εφαρμογή στην Visual Basic είναι τα εξής :

- Δημιουργία και ονομασία μιας νέας φόρμας (form).
- Σχεδίαση μέσα στη φόρμα και ονομασία των αντικειμένων της (objects).
- Προσάρτηση στα αντικείμενα του κώδικα για την απόκριση στα συμβάντα (events) του χρήστη ή του συστήματος.

Στη Visual Basic δημιουργούμε ένα σύνολο από υποπρογράμματα (υπορουτίνες), όπου το καθένα απ' αυτά ανταποκρίνεται σε κάποια ενέργεια του χρήστη.

Μια *ιδιότητα (property)* είναι ένα συγκεκριμένο γνώρισμα ενός αντικειμένου που ορίζει τα χαρακτηριστικά του, όπως το μέγεθος, το χρώμα, τη θέση του κ.ά. Μια *μέθοδος (method)* είναι μια ενσωματωμένη διαδικασία, δηλ. ένα τμήμα κώδικα ή προγράμματος, που ενεργοποιείται για να προσδώσει ορισμένη ενέργεια σ' ένα αντικείμενο. Οι ιδιότητες περιγράφουν ένα αντικείμενο ενώ οι μέθοδοι τού επιτρέπουν να κάνει κάτι. Δηλαδή, οι ιδιότητες είναι δεδομένα ενώ οι μέθοδοι είναι κώδικας προγράμματος.

Η γενική σύνταξη για την απόδοση τιμής σε μια ιδιότητα είναι η εξής :

Object.Property = Setting

Το περιβάλλον σχεδίασης της Visual Basic αποτελείται από ένα *παράθυρο έργου (Project)* και ένα *παράθυρο ιδιοτήτων (Properties)*. Το πρώτο εμφανίζει όλα τα αρχεία που έχουν σχέση με το ενεργό έργο και το δεύτερο αναφέρεται στο τρέχον επιλεγμένο αντικείμενο και μας δίνει τη δυνατότητα να δούμε ή να αλλάξουμε τις ιδιότητες του επιλεγμένου αντικειμένου. Με την επιλογή άλλου αντικειμένου, αλλάζουν και οι ιδιότητες που εμφανίζονται στο παράθυρο αυτό.

Στα αντικείμενα (χειριστήρια) της Visual Basic μπορούμε να δώσουμε δικά μας χαρακτηριστικά ονόματα για να μπορούμε να ξεχωρίζουμε αμέσως το είδος του αντικειμένου, δηλ. αν είναι πλήκτρο εντολής ή ετικέτα κλπ, αλλά και τον ρόλο που θέλουμε να έχει στην εφαρμογή. Τα τρία πρώτα γράμματα (πρόθεμα) είναι συνήθως μικρά και χαρακτηρίζουν το είδος του αντικειμένου. Η Visual Basic δίνει μόνη της ονόματα στα χειριστήρια, όπως Form1, Command2 κλπ, τα οποία μπορούμε και πρέπει να διορθώσουμε. Ακολουθεί ένας πίνακας με τα συνήθη αντικείμενα της Visual Basic, τα προθέματά τους και μερικά χαρακτηριστικά παραδείγματα.

| Αντικείμενο | Πρόθεμα | Παράδειγμα |
|----------------------------------|----------|-------------------------|
| Φόρμα (Form) | frm | frmFirst |
| Πλήκτρο Εντολής (Command Button) | cmd, btn | cmdExit, btnCloseWindow |
| Ετικέτα (Label) | lbl | lblOnomata |
| Πλαίσιο Κειμένου (Text Box) | txt | txtMessage01 |
| Οριζόντια Μπάρα Κύλισης | hsb | hsbRate |
| Κατακόρυφη Μπάρα Κύλισης | vsb | vsbTime |
| Μενού (Menu) | mnu | mnuMainMenu |
| Πλήκτρο Επιλογής (Option Button) | opt | optCountry |
| Πλαίσιο Ελέγχου (Check Box) | chk | chkLanguage |
| Πλαίσιο Λίστας (List) | lst | lstColors |
| Σύνθετο Πλαίσιο (Combo Box) | cbo | cboYpalliloi |

Τα πλήκτρα εντολής και οι φόρμες έχουν έναν *τίτλο* (*caption*) που εμφανίζεται στην οθόνη αλλά και ένα *όνομα* (*name*) που το χρησιμοποιούμε στον προγραμματισμό. Ο τίτλος και το όνομα μπορεί να έχουν διαφορετικές τιμές και πρέπει να έχουμε υπόψη μας ότι το όνομα θα πρέπει να έχει οπωσδήποτε λατινικούς χαρακτήρες και να είναι σύντομο, ενώ ο τίτλος καλό θα είναι να έχει ελληνικούς χαρακτήρες και να είναι δηλωτικός της χρήσης του αντικειμένου. Φυσικά δεν θα μπορούν δύο διαφορετικά αντικείμενα να έχουν το ίδιο όνομα ενώ θα μπορούν να έχουν τον ίδιο τίτλο.

Τα ονόματα των μεταβλητών της Visual Basic μπορούν να έχουν μήκος έως 255 χαρακτήρες, πρέπει να αρχίζουν από γράμμα, να μην περιέχουν τον χαρακτήρα της τελείας και να μην χρησιμοποιούν κάποιες από τις δεσμευμένες λέξεις (*reserved words*) της Visual Basic, όπως είναι για παράδειγμα οι λέξεις Sub και Function. Ακόμη, η Visual Basic δεν ξεχωρίζει τους πεζούς από τους κεφαλαίους χαρακτήρες στα ονόματα των μεταβλητών.

Στην Visual Basic υπάρχουν τέσσερα επίπεδα εμβέλειας των μεταβλητών :

- Επίπεδο διαδικασίας (τοπικές μεταβλητές).
- Στατικές μεταβλητές εντός διαδικασιών.
- Επίπεδο φόρμας και Επίπεδο ενότητας κώδικα.
- Καθολικές μεταβλητές.

Με την έμμεση δήλωση ή με την πρόταση Dim δηλώνουμε μια μεταβλητή σε επίπεδο διαδικασίας και η μεταβλητή είναι *τοπική* (*local*) μέσα στη διαδικασία, δηλ. δεν αναγνωρίζεται εκτός της διαδικασίας και δεν διατηρεί τις τιμές της ανάμεσα στις κλήσεις της διαδικασίας.

Με τη δήλωση Static μια μεταβλητή παραμένει τοπική μέσα στη διαδικασία όπου ορίζεται και δεν αναγνωρίζεται εκτός της διαδικασίας αλλά διατηρεί την τιμή της όταν επανέλθουμε στη διαδικασία (*στατική*).

Μια μεταβλητή επιπέδου φόρμας είναι διαθέσιμη σε κάθε διαδικασία αυτής της φόρμας και μια μεταβλητή επιπέδου ενότητας κώδικα είναι διαθέσιμη σε κάθε διαδικασία αυτής της ενότητας κώδικα. Για να δηλώσουμε μια μεταβλητή επιπέδου φόρμας, επιλέγουμε General από τη λίστα

Object και πάμε στην περιοχή General Declarations. Για να δηλώσουμε μια μεταβλητή επιπέδου ενότητας κώδικα, επιλέγουμε Module από το μενού Insert.

Το ευρύτερο επίπεδο εμβέλειας είναι το *καθολικό*, όπου όλες οι διαδικασίες και οι συναρτήσεις έχουν πρόσβαση στις καθολικές μεταβλητές. Για να δηλώσουμε μια μεταβλητή σαν καθολική, χρησιμοποιούμε την πρόταση Global στην περιοχή δηλώσεων :

```
Global DirectorName As String
Global EuroRate As Double
Global Temperature As Single
```

Στη Visual Basic μπορούμε να ορίσουμε και δικούς μας τύπους δεδομένων, κάτι αντίστοιχο δηλαδή με την δήλωση Type της Pascal. Οι οριζόμενοι από τον χρήστη τύποι δεδομένων μπορούν να περιέχουν ένα ή περισσότερα στοιχεία των βασικών τύπων. Ορίζουμε έναν δικό μας τύπο δεδομένων στην περιοχή δηλώσεων μιας ενότητας κώδικα, χρησιμοποιώντας τις λέξεις κλειδιά *Type* και *End Type*, ως εξής :

```
Type EmpData
  ID As Long
  Name As String*30
  Salary As Currency
End Type
```

Με τη χρήση του τελεστή τελείας (.), μπορούμε να δώσουμε τιμές στις μεταβλητές αυτές, ως εξής :

```
Dim Employee As EmpData
Employee.ID = 1234
Employee.Name = "Jonh Papadopoulos"
Employee.Salary = 1500.00
```

III. Σκελετός του Προγράμματος

Ακολουθεί η ανάλυση κάποιων βασικών φορμών της εφαρμογής.

ΦΟΡΜΑ ΛΑΘΟΥΣ

Με την ακόλουθη φόρμα , εμφανίζεται μήνυμα λάθους σε περίπτωση εισαγωγής λανθασμένων στοιχείων στη διαδικασία εισόδου.

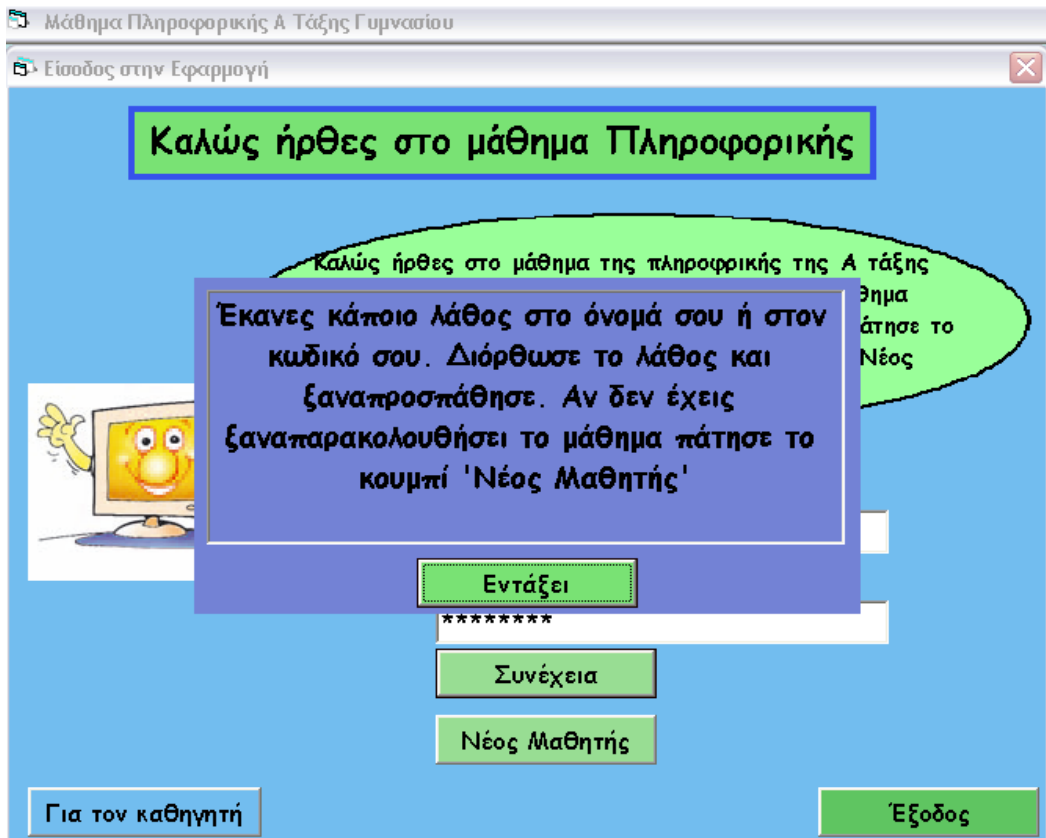
```
Private mstrMessage As String
Private mIngMsFondSize As Long
Public Property Let Message(ByVal strMessage As String)
  mstrMessage = strMessage
  Label6.Caption = mstrMessage
End Property
Public Property Let MsFondSize(ByVal IngMsFondSize As String)
  mIngMsFondSize = IngMsFondSize
  Label6.FontSize = mIngMsFondSize
End Property
```

```
Private Sub Command3_Click()
```

```
    Unload Me
```

```
End Sub
```

Το screenshot που ακολουθεί επεξηγεί τη φόρμα λάθους που εξηγήθηκε παραπάνω.



ΦΟΡΜΑ ΒΟΗΘΕΙΑΣ

Με την ακόλουθη φόρμα , εμφανίζεται η φόρμα βοήθειας στον καθηγητή κατά την εισαγωγή θεωρίας ή ασκήσεων.

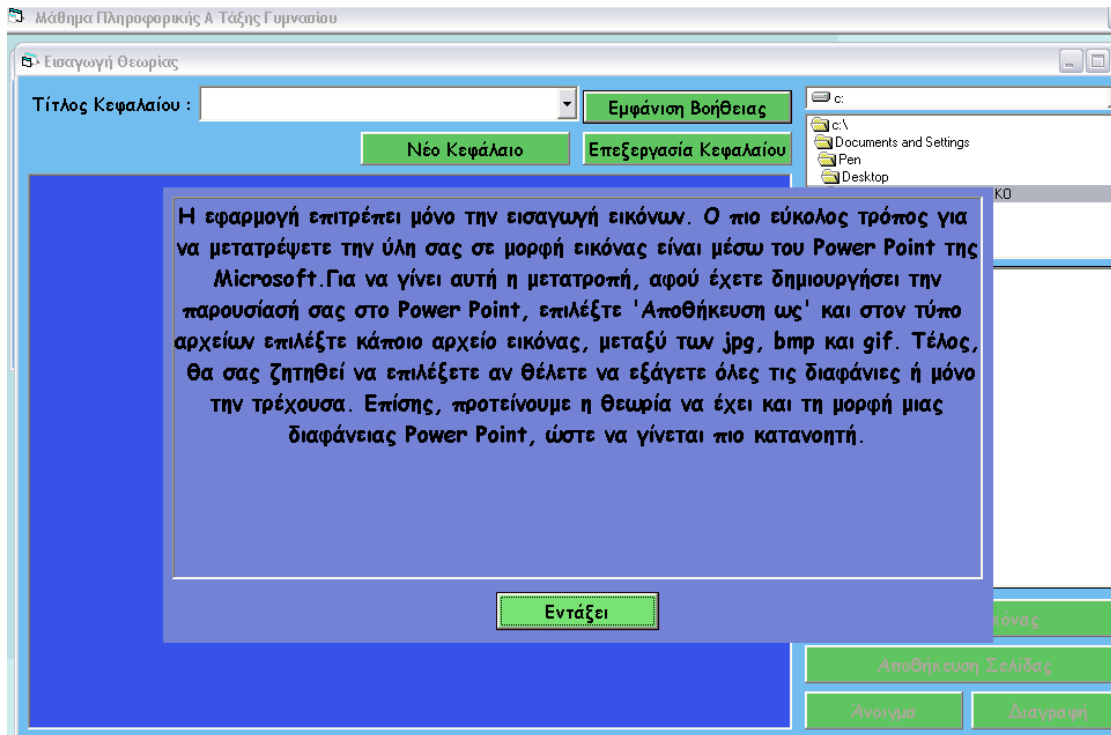
```
Private mstrMessage As String
Private mIngMsFondSize As Long
Public Property Let Message(ByVal strMessage As String)
    mstrMessage = strMessage
    Label6.Caption = mstrMessage
End Property
Public Property Let MsFondSize(ByVal IngMsFondSize As String)
    mIngMsFondSize = IngMsFondSize
    Label6.FontSize = mIngMsFondSize
```

```

End Property
Private Sub Command3_Click()
    Unload Me
End Sub

```

Το screenshot που ακολουθεί επεξηγεί τη φόρμα βοήθειας που εξηγήθηκε παραπάνω.



ΦΟΡΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΚΕΦΑΛΑΙΩΝ

Με τη συγκεκριμένη φόρμα δίνεται η δυνατότητα στον καθηγητή να αλλάξει ή να διαγράψει τα κεφάλαια του μαθήματος.

```

Private Sub FillCombo()
    Dim strSql As String
    Dim rs As Recordset

    strSql = "SELECT * FROM CHAPTERS"
    Set rs = modBas.LoadRecordset(strSql)

    Combo1.Clear

    If Not rs Is Nothing Then
        If Not rs.EOF Then

```

```

        Do While Not rs.EOF
            Combo1.AddItem rs!CHA_TITLE
            Combo1.ItemData(Combo1.NewIndex) = rs!CHA_ID
            rs.MoveNext
        Loop
    End If
End Sub
Private Sub EnableButtons()
    If Combo1.ListIndex >= 0 Then
        Command3.Enabled = True
        Command1.Enabled = True
    Else
        Command3.Enabled = False
        Command1.Enabled = False
    End If
End Sub
Private Sub Combo1_Change()
    EnableButtons
End Sub
Private Sub Combo1_Click()
    EnableButtons
End Sub
Private Sub Command1_Click()
    Frame1.Visible = True
    txtName.Text = Combo1.Text
End Sub
Private Sub Command2_Click()
    Dim strSql As String
    If ChapterExists Then
        MsgBox "I aneeiuo eaoaeUeio ?io Y?aoa ?eceonieia?oae o?Un?ae ?ac", vbCritical, ""
    End If
    If Text1.Text = vbNullString Then
        strSql = "UPDATE CHAPTERS SET CHA_TITLE=" & txtName.Text & " WHERE
CHA_ID=" & Combo1.ItemData(Combo1.ListIndex)
    Else
        strSql = "UPDATE CHAPTERS SET CHA_TITLE=" & txtName.Text & ",CHA_NUMBER="
& Text1.Text & " WHERE CHA_ID=" & Combo1.ItemData(Combo1.ListIndex)
    End If
    modBas.LoadRecordset (strSql)
    Frame1.Visible = False
    FillCombo

```

```
End Sub
Private Function ChapterExists() As Boolean
    Dim strSql As String
    Dim rs As Recordset
    strSql = "SELECT CHA_ID FROM CHAPTERS WHERE CHA_NUMBER=" & Text1.Text
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            If rs!CHA_ID <> Combo1.ItemData(Combo1.ListIndex) Then
                ChapterExists = True
            End If
        End If
    End If
End Function
Private Sub Command3_Click()
    Dim strSql As String
    Dim fso As New Scripting.FileSystemObject
    strSql = "DELETE FROM CHAPTERS WHERE CHA_ID=" &
    Combo1.ItemData(Combo1.ListIndex)
    modBas.LoadRecordset (strSql)
    On Error Resume Next
    fso.DeleteFolder CurDir & "\Chapters\" & Combo1.ItemData(Combo1.ListIndex)
    On Error GoTo 0
    FillCombo
End Sub
Private Sub Command4_Click()
    Unload Me
End Sub
Private Sub Command5_Click()
    Unload Me
End Sub
Private Sub Form_Load()
    FillCombo
    Frame1.Visible = False
    Command3.Enabled = False
    Command1.Enabled = False
End Sub
Private Sub Text1_KeyPress(KeyAscii As Integer)
    If Not IsNumeric(Chr(KeyAscii)) Then
        If KeyAscii <> 32 And KeyAscii <> 8 And KeyAscii <> 127 Then
            KeyAscii = 0
        End If
    End If
End Sub
```

```
End If
End Sub
```

ΦΟΡΜΑ ΑΞΙΟΛΟΓΗΣΗΣ

Με την ακόλουθη φόρμα, ο καθηγητής μπορεί να ελέγχει την πρόοδο των μαθητών του, βλέποντας το αποτέλεσμα του καθένα στις ασκήσεις.

```
Private mIngTeacherID As Long
Public Property Let TeacherID(ByVal lngTeacherID As Long)
    mIngTeacherID = lngTeacherID
End Property
Public Property Get TeacherID() As Long
    TeacherID = mIngTeacherID
End Property
Private Sub Command2_Click()
    frmTeacherSelections.TeacherID = mIngTeacherID
    frmTeacherSelections.Show
    Unload Me
End Sub
Private Sub Command3_Click()
    Unload Me
End Sub
Private Sub cmbChapter_Change()
    ChapterChanged
End Sub
Private Sub cmbChapter_Click()
    ChapterChanged
End Sub
Private Sub ChapterChanged()
    With cmbChapter
        If .ListIndex >= 0 Then
            LoadEvalsPerChapter
        End If
    End With
End Sub
Private Sub LoadEvalsPerChapter()
    Dim strSql As String
    Dim rs As Recordset
    Dim l As Long
    strSql = "SELECT * FROM EXERCISE WHERE EXE_CHA_ID=" &
    cmbChapter.ItemData(cmbChapter.ListIndex) & " ORDER BY EXE_NUMBER"
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs.EOF Then
```



```

With grdResults
    'fill exercise numbers
    I = 0
    Do While Not rs.EOF
        I = I + 1
        rs.MoveNext
    Loop
    .Cols = I + 1
    I = 1
    rs.MoveFirst
    Do While Not rs.EOF
        .ColData(I) = rs!EXE_ID
        .TextMatrix(0, I) = rs!EXE_NUMBER
        I = I + 1
        rs.MoveNext
    Loop
    'fillStudents
    FillStudentNames
    'FILL RATES
    FillRates
End With
Else
    InitializeGrid
    frmError.Message = "ΑαΙ Υ?αοα εαοα?un?οαε αοε?οαεο αεα αοου οι εαοUεαει οοοι
αοανιια?"
    frmError.Show vbModal
End If
End Sub
Private Sub FillRates()
    Dim rs As Recordset
    Dim lngCol As Long
    Dim lngRow As Long
    With grdResults
        For lngCol = .FixedCols To .Cols - 1
            For lngRow = .FixedRows To .Rows - 1
                Set rs = modBas.LoadRecordset("SELECT ans_rate FROM ANSWERS WHERE
ANS_EXE_ID=" & .ColData(lngCol) & " and ans_stu_id=" & .RowData(lngRow))
                If Not rs.EOF Then
                    If rs!ans_rate = eaCorrect Then
                        .TextMatrix(lngRow, lngCol) = "Ouoou!"
                    Else
                        .TextMatrix(lngRow, lngCol) = "EUeio"
                    End If
                End If
            Next lngRow
        Next lngCol
    End With
End Sub

```

```

        End If
    End If
    Next lngRow
    Next lngCol
End With
End Sub
Private Sub FillStudentNames()
    Dim rs As Recordset
    Set rs = modBas.LoadRecordset("SELECT STU_NAME, STU_ID FROM STUDENT WHERE
STU_TEA_ID=" & mlngTeacherID & " ORDER BY STU_NAME")

    With grdResults
        If Not rs.EOF Then
            Do While Not rs.EOF
                .AddItem ""
                .TextMatrix(.Rows - 1, 0) = rs!STU_NAME
                .RowData(.Rows - 1) = rs!STU_ID
                rs.MoveNext
            Loop
        Else
            .Rows = .FixedRows
            frmError.Message = "Αα ο?Un?ioi aaaaanaiiYiie iaecoYo oao ooci aoaniia?"
            frmError.Show vbModal
        End If
    End With
End Sub
Private Sub Form_Load()
    FillChapters
    InitializeGrid
    Me.Width = 15105
    Me.Height = 9300
End Sub
Private Sub InitializeGrid()
    With grdResults
        .Cols = 2
        .Rows = 2
        .FixedCols = 1
        .FixedRows = 1
        .TextMatrix(0, 1) = ""
        .TextMatrix(0, 0) = "Iaeco?o / ?oecoc"
        .TextMatrix(1, 0) = ""
        .TextMatrix(1, 1) = ""
    End With
End Sub

```

```

        .RowHeightMin = 300
        .Font.Size = 10
        .Font.Bold = True
        .Rows = 1
        .ColWidth(0) = 3000
        .AllowUserResizing = flexResizeColumns
    End With
End Sub
Private Sub FillChapters()
    Dim strSql As String
    Dim rs As Recordset

    strSql = "SELECT * FROM CHAPTERS"
    Set rs = modBas.LoadRecordset(strSql)
    cmbChapter.Clear
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            Do While Not rs.EOF
                cmbChapter.AddItem rs!CHA_TITLE
                cmbChapter.ItemData(cmbChapter.NewIndex) = rs!CHA_ID
                rs.MoveNext
            Loop
        End If
    End If
End Sub

```

ΦΟΡΜΑ ΕΙΣΑΓΩΓΗΣ ΣΤΟΙΧΕΙΩΝ

Με την ακόλουθη φόρμα, δίνεται η δυνατότητα στους καθηγητές και μαθητές να εισάγουν τα στοιχεία τους ή να εγγραφούν.

```

Private mblnContinue As Boolean
Private mlngStudentID As Long
Private Property Let ContinueLesson(ByVal blnContinue As Boolean)
    mblnContinue = blnContinue
    If mblnContinue Then
        frmSelectLesson.StudentID = mlngStudentID
        frmSelectLesson.Show
        Unload Me
    End If
End Property
Public Property Get StudentID() As Long

```

```

    StudentID = mIngStudentID
End Property
Private Sub Command1_Click()
    frmNewStudent.Show
    Unload Me
End Sub
Private Sub cmdContinue_Click()
    ContinueLesson = CheckStudent
End Sub
Private Function CheckStudent() As Boolean
    Dim strSql As String
    Dim rs As Recordset

    strSql = "SELECT STU_ID FROM STUDENT WHERE UCASE(TRIM(STU_NAME))=" &
Trim(UCase(txtName.Text)) & " AND STU_PASSWORD=" & Trim(UCase(txtPassword.Text)) &
""

    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            mIngStudentID = rs!STU_ID
            CheckStudent = True
        End If
    End If
    Set rs = Nothing
    If Not CheckStudent Then
        frmError.Message = "?eaiio eU?iei eUeio ooi uiiiU oio ? ooiu euaeuu oio. Aeuneuoia oi
eUeio eae iaia?nio?Ueoa. Ai aai Y?aeo iaia?anaeieioe?oae oi iUecia ?Uocoa oi eioi?? 'IYio
laeco?o"
        frmError.Show vbModal
    End If
End Function
Private Sub Command3_Click()
    Unload Me
End Sub
Private Sub Command4_Click()
    frmTeacher.Show
    Unload Me
End Sub
Private Sub Form_Load()
    cmdContinue.Enabled = False
    Me.Top = 0
    Me.Left = 0
    txtPassword.PasswordChar = "*"

```

```

txtPassword.MaxLength = 8
lblDiractions.Caption = "Εαε?ο ?neao ooi iUecia oco ?ecnionee?ο oco A οUico aoiiiao?io. Ai
Y?aeo iaia?anaaeieioe?oae oi iUecia ooi??nuoa oi uiiiiU oio eae oii euaeuu oio eae ?Uocoa oi
eioi?? 'OoiY?aea'. Aeee?ο ?Uocoa oi eioi?? 'IYio laeco?ο' aea ia ooia??oaeo oi iUecia."
End Sub
Private Sub txtName_Change()
    EnableContinue
End Sub
Private Sub EnableContinue()
    If txtName.Text <> vbNullString And txtPassword.Text <> vbNullString Then
        cmdContinue.Enabled = True
    Else
        cmdContinue.Enabled = False
    End If
End Sub
Private Sub txtPassword_Change()
    EnableContinue
End Sub

```

ΦΟΡΜΑ ΔΗΜΙΟΥΡΓΙΑΣ ΝΕΩΝ ΚΕΦΑΛΑΙΩΝ

Με τη συγκεκριμένη φόρμα δίνεται η δυνατότητα στον καθηγητή να δημιουργήσει νέα κεφάλαια για το μάθημα.

```

Private Sub Command1_Click()
    Unload Me
End Sub
Private Sub Command2_Click()
    Unload Me
End Sub
Private Sub Command3_Click()
    Dim strSql As String
    If ChapterExists Then
        MsgBox "I aneeiuo eaoaeUeio ?io Y?aoa ?eceonieia?oae ο?Un?ae ?ac",
vbCritical, ""
    End If
    strSql = "INSERT INTO CHAPTERS (CHA_NUMBER,CHA_TITLE) VALUES (" &
Text1.Text & "," & txtName.Text & ")"
    modBas.LoadRecordset strSql
    Unload Me
End Sub
Private Function ChapterExists() As Boolean
    Dim strSql As String
    Dim rs As Recordset

```

```

strSql = "SELECT 1 FROM CHAPTERS WHERE CHA_NUMBER=" & Text1.Text
Set rs = modBas.LoadRecordset(strSql)
If Not rs Is Nothing Then
    If Not rs.EOF Then
        ChapterExists = True
    End If
End If
End Function
Private Sub Text1_KeyPress(KeyAscii As Integer)
    If Not IsNumeric(Chr(KeyAscii)) Then
        If KeyAscii <> 32 And KeyAscii <> 8 And KeyAscii <> 127 Then
            KeyAscii = 0
        End If
    End If
End Sub
Private Sub Form_Load()
    Command3.Enabled = False
End Sub
Private Sub Text1_Change()
    EnableOK
End Sub

Private Sub EnableOK()
    If Text1.Text <> vbNullString And txtName.Text <> vbNullString Then
        Command3.Enabled = True
    Else
        Command3.Enabled = False
    End If
End Sub
Private Sub txtName_Change()
    EnableOK
End Sub

```

ΦΟΡΜΑ ΕΠΙΛΟΓΗΣ ΚΕΦΑΛΑΙΩΝ

Με την ακόλουθη φόρμα, δίνεται η δυνατότητα επιλογής κεφαλαίου για ανεύρεση αντίστοιχης θεωρίας και ασκήσεων.

```

Private mIngStudentID As Long
Private mblnShowInrto As Boolean
Public Property Let StudentID(ByVal lngStudentID As Long)
    mIngStudentID = lngStudentID
End Property

```

```
Public Property Get StudentID() As Long
    StudentID = mInqStudentID
End Property
Private Sub FillCombo()
    Dim strSql As String
    Dim rs As Recordset
    strSql = "SELECT * FROM CHAPTERS ORDER BY CHA_NUMBER"
    Set rs = modBas.LoadRecordset(strSql)
    Combo1.Clear
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            Do While Not rs.EOF
                Combo1.AddItem rs!CHA_NUMBER & " : " & rs!CHA_TITLE
                Combo1.ItemData(Combo1.NewIndex) = rs!CHA_ID
                rs.MoveNext
            Loop
        End If
    End If
End Sub
Private Sub Combo1_Change()
    EnableLabelels
End Sub
Private Sub Combo1_Click()
    EnableLabelels
End Sub
Private Sub Command3_Click()
    Unload Me
End Sub
Private Sub Form_Load()
    FillCombo
    lblTheory.Visible = False
    lblExe.Visible = False
End Sub
Private Sub EnableLabelels()
    If Combo1.ListIndex >= 0 Then
        lblTheory.Visible = True
        lblExe.Visible = True
    Else
        lblTheory.Visible = False
        lblExe.Visible = False
    End If
End Sub
```

```

Private Sub Form_Unload(Cancel As Integer)
    If Not mbInShowInrto Then
        frmIntro.Show
    End If
End Sub
Private Sub lblExe_Click()
    Dim rs As Recordset
    Set rs = modBas.LoadRecordset("SELECT 1 FROM EXERCISE WHERE
EXE_CHA_ID=" & Combo1.ItemData(Combo1.ListIndex))
    If rs.EOF Then
        frmError.Message = "I eaecaco?o oio aai Y?ae aeoUaae aeuia aoe?oaeo aea oi
a?eeaaaiYii eaoUeaei."
        frmError.Show vbModal
    Else
        frmSolveExercise.ChapterID = Combo1.ItemData(Combo1.ListIndex)
        frmSolveExercise.StudentID = mIngStudentID
        frmSolveExercise.ChapterName = Combo1.Text
        frmSolveExercise.Show
        mbInShowInrto = True
        Unload Me
    End If
End Sub
Private Sub lblTheory_Click()
    Dim rs As Recordset
    Set rs = modBas.LoadRecordset("SELECT 1 FROM IMAGE WHERE
IMA_CHA_ID=" & Combo1.ItemData(Combo1.ListIndex))
    If rs.EOF Then
        frmError.Message = "I eaecaco?o oio aai Y?ae aeoUaae aeuia eaun?a aea oi
a?eeaaaiYii eaoUeaei."
        frmError.Show vbModal
    Else
        frmLesson.StudentID = mIngStudentID
        frmLesson.ChapterID = Combo1.ItemData(Combo1.ListIndex)
        frmLesson.ChapterName = Combo1.Text
        frmLesson.Show
        mbInShowInrto = True
        Unload Me
    End If
End Sub

```

ΦΟΡΜΑ ΕΠΙΛΟΓΗΣ ΚΑΘΗΓΗΤΗ

Με την παρακάτω φόρμα, ο μαθητής κατά την εγγραφή του επιλέγει τον καθηγητή, στου οποίου το τμήμα βρίσκεται και του οποίου τη διδασκαλία (κεφαλαίων) παρακολουθεί.


```

Private mIngTeacherID As Long
Public Property Let TeacherID(ByVal lngTeacherID As Long)
    mIngTeacherID = lngTeacherID
End Property
Public Property Get TeacherID() As Long
    TeacherID = mIngTeacherID
End Property
Private Sub Command3_Click()
    frmIntro.Show
    Unload Me
End Sub
Private Sub Form_Unload(Cancel As Integer)
    frmIntro.Show
End Sub
Private Sub Label1_Click()
    frmEnterEx.TeacherID = mIngTeacherID
    frmEnterEx.Show
    frmEnterEx.Height = 8580
    frmEnterEx.Width = 9615
End Sub
Private Sub Label2_Click()
    frmEvaluations.TeacherID = mIngTeacherID
    frmEvaluations.Show
End Sub
Private Sub Label3_Click()
    frmInserImage.TeacherID = mIngTeacherID
    frmInserImage.Show
End Sub

```

ΦΟΡΜΑ ΕΙΣΑΓΩΓΗΣ ΑΣΚΗΣΕΩΝ

```

Private Const cGREEN_COLOR As Long = &H93DD98
Private Const cBLUE_COLOR As Long = &HEFBD72
Private mIngTeacherID As Long
Private menmCurrentView As ExerciseModeEnumeration
Private mIngExeID As Long
Private mstrExeNumbers As String
Private Sub MakeExeNumberString()
    Dim rs As Recordset
    Dim strSql As String
    Dim lngNumber As Long

```

```
strSql = "SELECT EXE_NUMBER FROM EXERCISE WHERE EXE_CHA_ID=" &  
Combo1.ItemData(Combo1.ListIndex) & " ORDER BY EXE_NUMBER"  
Set rs = modBas.LoadRecordset(strSql)  
mstrExeNumbers = vbNullString  
If Not rs Is Nothing Then  
    If Not rs.EOF Then  
        Do While Not rs.EOF  
            mstrExeNumbers = mstrExeNumbers & "," & rs!EXE_NUMBER & ","  
            rs.MoveNext  
        Loop  
    End If  
End If  
If mstrExeNumbers = vbNullString Then  
    txtNumber.Text = 1  
Else  
    lngNumber = 1  
    Do  
        If InStr(mstrExeNumbers, "," & CStr(lngNumber) & ",") <= 0 Then  
            txtNumber.Text = lngNumber  
            Exit Do  
        End If  
        lngNumber = lngNumber + 1  
    Loop  
End If  
End Sub  
Private Sub EnableButtons()  
    Dim blnDelete As Boolean  
    Dim blnSave As Boolean  
    Dim blnOpen As Boolean  
    Dim lngAnswerCounter As Long  
    Dim blnHasAnswer As Boolean  
    Dim lngIndex As Long  
    blnDelete = True  
    blnSave = True  
    blnOpen = True  
  
    If mlngExeID <= 0 Then blnDelete = False  
    If Combo1.ListIndex < 0 Then  
        blnOpen = False  
        blnSave = False  
    End If
```

```

    If (txtQuestion(menmCurrentView).Text = vbNullString) Or (txtNumber.Text =
vbNullString) Then blnSave = False
    If menmCurrentView = emChoise Then
        For lngIndex = 0 To 3
            If txtAnswer(lngIndex).Text = vbNullString Then
                lngAnswerCounter = lngAnswerCounter + 1
            End If
            If lngAnswerCounter > 2 Then
                blnSave = False
                Exit For
            End If
        Next lngIndex
        For lngIndex = 0 To 3
            If optCorrect(lngIndex).Value = True Then
                blnHasAnswer = True
                Exit For
            End If
        Next lngIndex
        blnSave = blnSave And blnHasAnswer
    Else
        If cmbAnswer.ListIndex < 0 Then
            blnSave = False
        End If
    End If
    cmdOpen.Enabled = blnOpen
    cmdDelete.Enabled = blnDelete
    cmdSave.Enabled = blnSave
End Sub
Private Property Let CurrentView(ByVal enmCurrentView As
ExerciseModeEnumeration)
    Dim enmOhterLaber As ExerciseModeEnumeration
    menmCurrentView = enmCurrentView
    If menmCurrentView = emChoise Then
        enmOhterLaber = emCorrectFalse
    Else
        enmOhterLaber = emChoise
    End If
    lblTab(menmCurrentView).BackColor = cBLUE_COLOR
    lblTab(enmOhterLaber).BackColor = cGREEN_COLOR
    lblTab(menmCurrentView).FontBold = True
    lblTab(enmOhterLaber).FontBold = False
    fraExe(menmCurrentView).Visible = True

```

```

        fraExe(enmOhterLaber).Visible = False
    End Property

    Public Property Let TeacherID(ByVal lngTeacherID As Long)
        mlngTeacherID = lngTeacherID
    End Property
    Public Property Get TeacherID() As Long
        TeacherID = mlngTeacherID
    End Property
    Private Sub SaveExercise()
        Dim rs As Recordset
        Dim strSql As String
        Dim strAnswers As String
        Dim strChoises As String
        Dim l As Long
        If menmCurrentView = emChoise Then
            For l = 0 To 3
                If optCorrect(l).Value = True Then
                    strAnswers = l
                End If
            Next l
            strChoises = txtAnswer(0) & "|" & txtAnswer(1) & "|" & txtAnswer(2) & "|" &
txtAnswer(3)
        Else
            strAnswers = cmbAnswer.ItemData(cmbAnswer.ListIndex)
        End If
        If mlngExeID = 0 Then
            strSql = "INSERT INTO EXERCISE
(EXE_TYPE,EXE_NUMBER,EXE_TEXT,EXE_ANSWER,EXE_CHA_ID,EXE_CHOISES)
VALUES ("
                strSql = strSql & menmCurrentView & ","
                strSql = strSql & txtNumber.Text & ","
                strSql = strSql & "" & txtQuestion(menmCurrentView).Text & ","
                strSql = strSql & "" & strAnswers & ","
                strSql = strSql & Combo1.ItemData(Combo1.ListIndex) & ","
                strSql = strSql & "" & strChoises & ")"
        Else
            strSql = "UPDATE EXERCISE SET EXE_TEXT=" &
txtQuestion(menmCurrentView).Text & ",EXE_ANSWER=" & strAnswers & " WHERE
EXE_ID=" & mlngExeID
        End If
        Set rs = modBas.LoadRecordset(strSql)
        Set rs = Nothing
    End Sub

```

```

    MsgBox "C Uoecoc o?ecea ia a?eoo??a", vbOKOnly, ""
    strSql = "SELECT EXE_ID FROM EXERCISE WHERE EXE_NUMBER=" &
txtNumber.Text & " AND EXE_CHA_ID=" & Combo1.ItemData(Combo1.ListIndex)
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs.EOF Then
        mIngExeID = rs!EXE_ID
    End If
End Sub
Private Function NumberExists() As Boolean
    Dim strSql As String
    Dim rs As Recordset
    strSql = "SELECT 1 FROM EXERCISE WHERE EXE_CHA_ID=" &
Combo1.ItemData(Combo1.ListIndex) & " AND EXE_NUMBER=" & txtNumber.Text
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            NumberExists = True
        End If
    End If
End Function
Private Sub cmbAnswer_Change()
    EnableButtons
End Sub
Private Sub cmbAnswer_Click()
    EnableButtons
End Sub
Private Sub cmbClear_Click()
    ClearForm
    EnableButtons
End Sub
Private Sub cmbLoadExe_Change()
    cmdShow.Enabled = True
End Sub
Private Sub cmbLoadExe_Click()
    cmdShow.Enabled = True
End Sub
Private Sub cmdCancel_Click()
    CurrentView = emChoise
    HideForOpen True
End Sub
Private Sub cmdDelete_Click()
    If mIngExeID > 0 Then

```

```

        DeleteExercise
    End If
    MakeExeNumberString
End Sub
Private Sub DeleteExercise()
    Dim rs As Recordset
    Dim strSql As String
    strSql = "DELETE FROM EXERCISE WHERE EXE_ID=" & mlngExeID
    Set rs = modBas.LoadRecordset(strSql)
    Set rs = Nothing
    MsgBox "C Uoecoc aeaanUocea ia a?eoo??a"
    ClearForm
End Sub
Private Sub ClearForm()
    mlngExeID = 0
    txtAnswer(1).Text = ""
    txtAnswer(0).Text = ""
    txtAnswer(2).Text = ""
    txtAnswer(3).Text = ""
    txtQuestion(1).Text = ""
    txtQuestion(0).Text = ""
    txtNumber.Text = 0
    optCorrect(0).Value = False
    optCorrect(1).Value = False
    optCorrect(2).Value = False
    optCorrect(3).Value = False
    cmbAnswer.ListIndex = -1
    MakeExeNumberString
End Sub
Private Sub cmdOpen_Click()
    HideForOpen False
    modBas.LoadExerciseCombo Combo1.ItemData(Combo1.ListIndex), cmbLoadExe
    cmdShow.Enabled = False
End Sub
Private Sub HideForOpen(ByVal blnVisile As Boolean)
    fraExe(0).Visible = blnVisile
    fraExe(1).Visible = blnVisile
    lblTab(1).Visible = blnVisile
    lblTab(0).Visible = blnVisile
    fraOpen.Visible = Not blnVisile
End Sub
Private Sub cmdSave_Click()

```

```

Dim strSql As String
Dim strNewFilename As String
If mlngExeID = 0 Then
    If InStr(mstrExeNumbers, "," & txtNumber.Text & ",") > 0 Then
        frmError.Message = "I aneeiuo Y?ae a?ieceaooa? ?ac aea oi a?eeaiYii
eaoUeaei"
        frmError.Show vbModal
        Exit Sub
    End If
End If
SaveExercise
mlngExeID = 0
MakeExeNumberString
End Sub
Private Sub cmdShow_Click()
    HideForOpen True
    FillWithExercise
End Sub
Private Sub FillWithExercise()
    Dim rs As Recordset
    Dim strSql As String
    Dim vrnAnswers As Variant
    Dim I As Long
    strSql = "SELECT * FROM EXERCISE WHERE EXE_ID=" &
cmbLoadExe.ItemData(cmbLoadExe.ListIndex)
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            CurrentView = rs!EXE_TYPE
            txtQuestion(menmCurrentView).Text = rs!EXE_TEXT
            txtNumber.Text = rs!EXE_NUMBER
            mlngExeID = rs!EXE_ID
            If menmCurrentView = emChoise Then
                vrnAnswers = Split(rs!EXE_CHOISES, "|")
                For I = LBound(vrnAnswers) To UBound(vrnAnswers)
                    txtAnswer(I) = vrnAnswers(I)
                Next I
                optCorrect(rs!EXE_ANSWER).Value = True
            Else
                With cmbAnswer
                    For I = 0 To 1
                        If .ItemData(I) = rs!EXE_ANSWER Then

```

```
                .ListIndex = I
            Exit For
        End If
    Next I
End With
End If
End If
End If
EnableButtons
End Sub
Private Sub Combo1_Change()
    ChapterChanged
End Sub
Private Sub Combo1_Click()
    ChapterChanged
End Sub
Private Sub ChapterChanged()
    EnableButtons
    MakeExeNumberString
End Sub
Private Sub Form_Load()
    CurrentView = emChoise
    FillCombo
    With cmbAnswer
        .AddItem "Ouoou"
        .ItemData(.NewIndex) = 1
        .AddItem "EUeio"
        .ItemData(.NewIndex) = 0
    End With
    fraOpen.Visible = False
    cmdDelete.Enabled = False
    cmdOpen.Enabled = False
    cmdSave.Enabled = False
End Sub
Private Sub FillCombo()
    Dim strSql As String
    Dim rs As Recordset
    strSql = "SELECT * FROM CHAPTERS"
    Set rs = modBas.LoadRecordset(strSql)
    Combo1.Clear
    If Not rs Is Nothing Then
        If Not rs.EOF Then
```



```

        Do While Not rs.EOF
            Combo1.AddItem rs!CHA_TITLE
            Combo1.ItemData(Combo1.NewIndex) = rs!CHA_ID
            rs.MoveNext
        Loop
    End If
End Sub
Private Sub lblTab_Click(Index As Integer)
    CurrentView = Index
End Sub
Private Function ExNrExists() As Boolean
    Dim strSql As String
    Dim rs As Recordset
    strSql = "SELECT 1 FROM EXERCISE WHERE EXE_NUMBER=" & txtNumber.Text
    & " AND EXE_CHA_ID=" & Combo1.ItemData(Combo1.ListIndex)
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            ExNrExists = True
        End If
    End If
End Function
Private Sub optCorrect_Click(Index As Integer)
    If txtAnswer(Index).Text = vbNullString Then
        optCorrect(Index).Value = False
    End If
    EnableButtons
End Sub
Private Sub txtAnswer_KeyPress(Index As Integer, KeyAscii As Integer)
    EnableButtons
    If Len(txtAnswer(Index).Text) >= 255 Then KeyAscii = 0
End Sub
Private Sub txtNumber_Change()
    EnableButtons
End Sub
Private Sub txtQuestion_Change(Index As Integer)
    EnableButtons
End Sub
Private Sub txtQuestion_KeyPress(Index As Integer, KeyAscii As Integer)
    If Len(txtQuestion(Index).Text) >= 255 Then KeyAscii = 0
End Sub

```

ΦΟΡΜΑ ΜΑΘΗΜΑΤΟΣ

Με τη φόρμα αυτή, ο μαθητής μπορεί να δει τη θεωρία των κεφαλαίων και να τη διαβάσει.

```
Private mIngStudentID As Long
Private mIngChapterID As Long
Private mstrChapterName As String
Private mIngCurrentPage As Long
Private mIngFirstPage As Long
Private mIngLastPage As Long
Private mblnEnterResize As Boolean
Private mIngWidth As Long
Private mIngHeight As Long
Public Property Let StudentID(ByVal lngStudentID As Long)
    mIngStudentID = lngStudentID
End Property
Public Property Get StudentID() As Long
    StudentID = mIngStudentID
End Property
Public Property Let ChapterID(ByVal lngChapterID As Long)
    mIngChapterID = lngChapterID
End Property
Public Property Get ChapterID() As Long
    ChapterID = mIngChapterID
End Property
Public Property Let ChapterName(ByVal strChapterName As String)
    mstrChapterName = strChapterName
End Property
Public Property Get ChapterName() As String
    ChapterName = mstrChapterName
End Property
Private Sub Form_Load()
    Me.Width = 15885
    Me.Height = 10485
    Label7.Caption = mstrChapterName
    FillImageList
    Image6.Picture = imgList.ListImages.Item(mIngFirstPage).Picture
    mIngCurrentPage = mIngFirstPage
    mIngWidth = Me.Width
    mIngHeight = Me.Height
    mblnEnterResize = True
End Sub
```

```

Private Sub Form_Resize()
    DoEvents
    If Not mblnEnterResize Then Exit Sub
    Frame1.Left = Frame1.Left + (Me.Width - mlngWidth) / 2
    Frame2.Left = Frame2.Left + (Me.Width - mlngWidth) / 2
    Frame1.Top = Frame1.Top + Me.Height - mlngHeight
    Image6.Width = Image6.Width + (Me.Width - mlngWidth)
    Image6.Height = Image6.Height + (Me.Height - mlngHeight)
    mlngWidth = Me.Width
    mlngHeight = Me.Height
End Sub
Private Sub FillImageList()
    Dim strSql As String
    Dim rs As Recordset
    Dim I As Long
    strSql = "SELECT * FROM IMAGE WHERE IMA_CHA_ID=" & mlngChapterID & "
ORDER BY IMA_ORDER_NUMBER"
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs Is Nothing Then
        If Not rs.EOF Then
            I = 0
            With imgList
                Do While Not rs.EOF
                    If rs!IMA_ORDER_NUMBER = I + 1 Then
                        .ListImages.Add rs!IMA_ORDER_NUMBER,
LoadPicture(rs!IMA_PATH)
                        If mlngFirstPage = 0 Then
                            mlngFirstPage = rs!IMA_ORDER_NUMBER
                        End If
                    Else
                        .ListImages.Add I + 1
                    End If
                    I = I + 1
                    rs.MoveNext
                Loop
                mlngLastPage = I
            End With
        End If
    End Sub
Private Sub Image1_Click()
    Dim rs As Recordset

```

```

Set rs = modBas.LoadRecordset("SELECT 1 FROM EXERCISE WHERE
EXE_CHA_ID=" & mIngChapterID)
If rs.EOF Then
    frmError.Message = "I eaecaco?o oio aai Y?ae aeoUaae aeuia aoe?oaeo aea oi
a?eeaaaiYii eaoUeaei."
    frmError.Show vbModal
Else
    frmSolveExercise.ChapterID = mIngChapterID
    frmSolveExercise.StudentID = mIngStudentID
    frmSolveExercise.ChapterName = mstrChapterName
    frmSolveExercise.Show
    Unload Me
End If
End Sub

```

```
Private Sub Image2_Click()
```

```
    frmIntro.Show
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub Image3_Click()
```

```
    frmSelectLesson.StudentID = mIngStudentID
```

```
    frmSelectLesson.Show
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub Image4_Click()
```

```
    If mIngCurrentPage = mIngFirstPage Then Exit Sub
```

```
    mIngCurrentPage = mIngCurrentPage - 1
```

```
    Image6.Picture = imgList.ListImages.Item(mIngCurrentPage).Picture
```

```
End Sub
```

```
Private Sub Image5_Click()
```

```
    If mIngCurrentPage = mIngLastPage Then Exit Sub
```

```
    mIngCurrentPage = mIngCurrentPage + 1
```

```
    Image6.Picture = imgList.ListImages.Item(mIngCurrentPage).Picture
```

```
End Sub
```

ΦΟΡΜΑ ΓΙΑ ΝΕΟ ΜΑΘΗΤΗ

Με τη φόρμα αυτή, γίνεται η καταχώρηση των στοιχείων ενός νέου μαθητή.

```
Private Sub Command1_Click()
```

```

        Unload Me
    End Sub
    Private Sub Command2_Click()
        Unload Me
    End Sub
    Private Sub Command3_Click()
        Dim strSql As String
        If ChapterExists Then
            MsgBox "I aneeiuo eaoaeUeio ?io Y?aoa ?eceonieia?oae o?Un?ae ?ac",
vbCritical, ""
        End If
        strSql = "INSERT INTO CHAPTERS (CHA_NUMBER,CHA_TITLE) VALUES (" &
Text1.Text & "," & txtName.Text & ")"
        modBas.LoadRecordset strSql
        Unload Me
    End Sub
    Private Function ChapterExists() As Boolean
        Dim strSql As String
        Dim rs As Recordset
        strSql = "SELECT 1 FROM CHAPTERS WHERE CHA_NUMBER=" & Text1.Text
        Set rs = modBas.LoadRecordset(strSql)
        If Not rs Is Nothing Then
            If Not rs.EOF Then
                ChapterExists = True
            End If
        End If
    End Function
    Private Sub Text1_KeyPress(KeyAscii As Integer)
        If Not IsNumeric(Chr(KeyAscii)) Then
            If KeyAscii <> 32 And KeyAscii <> 8 And KeyAscii <> 127 Then
                KeyAscii = 0
            End If
        End If
    End Sub
    Private Sub Form_Load()
        Command3.Enabled = False
    End Sub
    Private Sub Text1_Change()
        EnableOK
    End Sub
    Private Sub EnableOK()
        If Text1.Text <> vbNullString And txtName.Text <> vbNullString Then

```

```

        Command3.Enabled = True
    Else
        Command3.Enabled = False
    End If
End Sub
Private Sub txtName_Change()
    EnableOK
End Sub

```

ΦΟΡΜΑ ΕΠΙΛΥΣΗΣ ΑΣΚΗΣΕΩΝ

Με τη φόρμα αυτή, ο μαθητής μπορεί να επιλύσει τις ασκήσεις που έχει βάλει ο καθηγητής.

```

Private mlngStudentID As Long
Private mlngChapterID As Long
Private mstrChapterName As String
Private mlngCurrentPage As Long
Private mlngFirstPage As Long
Private mlngLastPage As Long
Private mlngWidth As Long
Private mlngHeight As Long
Private menmCurrentView As ExerciseModeEnumeration
Private mlngCorrectAnswer As Long
Private mlngExelD As Long
Private Sub EnableCheckButton()
    Dim I As Integer
    Dim blnEnable As Boolean
    If menmCurrentView = emChoise Then
        For I = 0 To 3
            If optCorrect(I).Value = True Then
                blnEnable = True
            Exit For
        End If
    Next I
    Else
        If cmbAnswer.ListIndex >= 0 Then blnEnable = True
    End If
    cmdCheck.Enabled = blnEnable
End Sub
Private Property Let CurrentView(ByVal enmCurrentView As
ExerciseModeEnumeration)
    Dim enmOhterLaber As ExerciseModeEnumeration

```

```
menmCurrentView = enmCurrentView
If menmCurrentView = emChoise Then
    enmOhterLaber = emCorrectFalse
Else
    enmOhterLaber = emChoise
End If
fraExe(menmCurrentView).Visible = True
fraExe(enmOhterLaber).Visible = False
End Property
Public Property Let StudentID(ByVal lngStudentID As Long)
    mlngStudentID = lngStudentID
End Property

Public Property Get StudentID() As Long
    StudentID = mlngStudentID
End Property
Public Property Let ChapterID(ByVal lngChapterID As Long)
    mlngChapterID = lngChapterID
End Property
Public Property Get ChapterID() As Long
    ChapterID = mlngChapterID
End Property
Public Property Let ChapterName(ByVal strChapterName As String)
    mstrChapterName = strChapterName
End Property
Public Property Get ChapterName() As String
    ChapterName = mstrChapterName
End Property

Private Sub cmbAnswer_Change()
    EnableCheckBox
End Sub
Private Sub cmbAnswer_Click()
    EnableCheckBox
End Sub
Private Sub cmbLoadExe_Change()
    LoadExercise
End Sub
Private Sub cmbLoadExe_Click()
    LoadExercise
End Sub
Private Sub LoadExercise()
```

```

Dim rs As Recordset
Dim varChoises As Variant
Dim I As Long
If cmbLoadExe.ListIndex < 0 Then Exit Sub
Set rs = modBas.LoadRecordset("SELECT * FROM EXERCISE WHERE EXE_ID="
& cmbLoadExe.ItemData(cmbLoadExe.ListIndex))
If Not rs Is Nothing Then
    If Not rs.EOF Then
        CurrentView = rs!EXE_TYPE
        lblExeNumber(menmCurrentView).Caption = "?οεοο " & rs!EXE_NUMBER & "
:"

        lblQuestion(menmCurrentView).Caption = rs!EXE_TEXT
        If menmCurrentView = emChoise Then
            varChoises = Split(rs!EXE_CHOISES, "|")
            For I = LBound(varChoises) To UBound(varChoises)
                lblAnswer(I).Caption = varChoises(I)
                optCorrect(I).Visible = True
                optCorrect(I).Value = False
            Next I
            For I = I To 3
                lblAnswer(I).Caption = ""
                optCorrect(I).Visible = False
                optCorrect(I).Value = False
            Next I
        End If
        mlngCorrectAnswer = rs!EXE_ANSWER
        mlngExeID = rs!EXE_ID
    End If
End If
Set rs = Nothing
cmdCheck.Enabled = False
End Sub
Private Sub cmdCheck_Click()
    Dim I As Long
    Dim blnCorrect As Boolean
    Dim strMessage As String

    If menmCurrentView = emChoise Then
        For I = 0 To 3
            If optCorrect(I).Value = True Then
                If mlngCorrectAnswer = I Then
                    blnCorrect = True
                End If
            End If
        Next I
    End If
End Sub

```



```

        Else
            blnCorrect = False
        End If
    End If
Next I
Else
    If cmbAnswer.ItemData(cmbAnswer.ListIndex) = mlngCorrectAnswer Then
        blnCorrect = True
    Else
        blnCorrect = False
    End If
End If
If blnCorrect Then
    strMessage = "?iey eaeU! C a?Uioco? oio a?iae ouoo?!"
Else
    strMessage = "Aoooo??o, c a?Uioco? oio a?iae eaieaoiYic."
End If
SaveAnswer (blnCorrect)
frmError.Message = strMessage
frmError.Show vbModal
cmdCheck.Enabled = False
End Sub
Private Sub SaveAnswer(ByVal blnCorrect As Boolean)
    Dim strSql As String
    Dim rs As Recordset
    Dim enmAnswer As ExerciseAnswersEnumeration
    If blnCorrect Then
        enmAnswer = eaCorrect
    Else
        enmAnswer = eaFalse
    End If
    strSql = "SELECT ANS_ID FROM ANSWERS WHERE ANS_STU_ID=" &
    mlngStudentID & " AND ANS_EXE_ID=" & mlngExeID
    Set rs = modBas.LoadRecordset(strSql)
    If Not rs.EOF Then
        strSql = "UPDATE ANSWERS SET ANS_RATE=" & enmAnswer & " WHERE
ANS_EXE_ID=" & mlngExeID & " AND ANS_STU_ID=" & mlngStudentID
    Else
        strSql = "INSERT INTO ANSWERS
(ANS_STU_ID,ANS_EXE_ID,ANS_DATE,ANS_RATE) values (" & mlngStudentID & "," &
mlngExeID & "," & Date & "," & enmAnswer & ")"
    End If
    modBas.LoadRecordset strSql

```

```
End Sub
Private Sub Form_Load()

    Me.Width = 15885
    Me.Height = 10485
    lblTitle.Caption = mstrChapterName
    With cmbAnswer
        .AddItem "Ouoou"
        .ItemData(.NewIndex) = 1
        .AddItem "EUeio"
        .ItemData(.NewIndex) = 0
    End With
    modBas.LoadExerciseCombo mlngChapterID, cmbLoadExe
    cmbLoadExe.ListIndex = 0
    mlngWidth = Me.Width
    mlngHeight = Me.Height
End Sub
Private Sub imgExit_Click()
    frmIntro.Show
    Unload Me
End Sub
Private Sub imgHome_Click()
    frmSelectLesson.StudentID = mlngStudentID
    frmSelectLesson.Show
    Unload Me
End Sub
Private Sub imgNext_Click()
    With cmbLoadExe
        If .ListIndex < .ListCount - 1 Then
            .ListIndex = .ListIndex + 1
        End If
    End With
End Sub
Private Sub imgPrevious_Click()
    With cmbLoadExe
        If .ListIndex > 0 Then
            .ListIndex = .ListIndex - 1
        End If
    End With
End Sub
Private Sub imgTheory_Click()
    Dim rs As Recordset
```

```
Set rs = modBas.LoadRecordset("SELECT 1 FROM IMAGE WHERE  
IMA_CHA_ID=" & mlngChapterID)  
If rs.EOF Then  
    frmError.Message = "I eaecaco?o oio aai Y?ae aeoUaae aeuia eaun?a aea oi  
a?eeaaaiYii eaoUeaei."  
    frmError.Show vbModal  
Else  
    frmLesson.StudentID = mlngStudentID  
    frmLesson.ChapterID = mlngChapterID  
    frmLesson.ChapterName = mstrChapterName  
    frmLesson.Show  
    Unload Me  
End If  
End Sub  
Private Sub optCorrect_Click(Index As Integer)  
    EnableCheckBox  
End Sub
```

3^ο : ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ

3.1 ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ

I. Γενικά για το πρόγραμμα

Η εφαρμογή μας έχει αναπτυχθεί με σκοπό να βοηθήσει τους μαθητές και τις μαθήτριες της Α' τάξης των ελληνικών γυμνασίων στην πληροφορική που διδάσκεται στο σχολείο στην Α' τάξη Γυμνασίου. Η εφαρμογή μπορεί να αναπτυχθεί εύκολα περαιτέρω για να συμπεριλάβει και όλες τις υπόλοιπες ενότητες των πληροφορικής της Α' Γυμνασίου και να αποτελέσει έτσι ένα πλήρες και χρήσιμο ηλεκτρονικό εγχειρίδιο για κάθε μαθητή που διαθέτει έστω και στοιχειώδεις γνώσεις χρήσης ηλεκτρονικού υπολογιστή.

Η ανάπτυξη της εφαρμογής έγινε σε Visual Basic, η βάση δεδομένων με τη βοήθεια του προγράμματος Microsoft Access 2003 και έχει σχεδιασθεί για να τρέχει σε λειτουργικά συστήματα της Microsoft. Οι επιλογές αυτές έγιναν για πολλούς λόγους. Πιο συγκεκριμένα, η επιλογή της Visual Basic για την ανάπτυξη λογισμικού έγινε λόγω της ευκολίας της, της μεταφερισιμότητας των εφαρμογών που αναπτύσσονται με αυτή καθώς και της προσφοράς ανάπτυξης παραθυρικών εφαρμογών με αυτή (πλέον οι χρήστες είναι εξοικειωμένοι περισσότερο με προϊόντα της Microsoft). Η επιλογή της Microsoft Access 2003 έγινε λόγω της ευκολίας χρήσης αυτής, καθώς και για λόγους πλήρους συνεργασίας με το περιβάλλον της Visual Basic. Σε γενικές γραμμές, η λειτουργία της εφαρμογής έχει ως εξής:

Ο κάθε μαθητής εισάγεται στο σύστημα με την εισαγωγή του ονοματεπωνύμου του και ενός κωδικού που έχει απομνημονεύσει. Η κάθε διδακτική ενότητα ξεκινάει με την αντίστοιχη θεωρία. Στη συνέχεια ακολουθούν ασκήσεις πολλαπλής επιλογής και σωστού-λάθους πάνω στη θεωρία που διδάχθηκε ο μαθητής. Το πρόγραμμα εντοπίζει τα λάθη που έγιναν στις ασκήσεις και επιστρέφει το τελικό σκορ ου επιτευχθεί.

Επίσης, υποστηρίζεται λειτουργία και για τους καθηγητές – διαχειριστές του συστήματος. Ο καθηγητής εισάγεται στο σύστημα με τρόπο παρόμοιο με των μαθητών. Στη συνέχεια, ο καθηγητής μπορεί να ανανεώσει ή να διαγράψει τις ασκήσεις ή να δει τα σκορ όλων των μαθητών που έκαναν τα τεστ.

II. Απαιτήσεις συστήματος

Όπως προαναφέρθηκε, η εφαρμογή έχει σχεδιασθεί και υλοποιηθεί για να τρέχει σε λειτουργικά συστήματα της Microsoft. Η υλοποίησή του έγινε με τέτοιο τρόπο ώστε να είναι ένα «ελαφρύ» και εύκολο στη χρήση πρόγραμμα. Πιο συγκεκριμένα, οι ελάχιστες απαιτήσεις συστήματος είναι οι εξής:

- Επεξεργαστής της Intel Pentium II ή επεξεργαστής παρόμοιας αρχιτεκτονικής και ταχύτητας.
- Μνήμη RAM 64 MB.
- 1 MB ελεύθερος χώρος στον σκληρό δίσκο.
- Λειτουργικό σύστημα Microsoft Windows 98.
- Να υπάρχει εγκατεστημένη η Microsoft Access 98 (ή νεότερη έκδοση) στον υπολογιστή του χρήστη.
- Να υπάρχει εγκατεστημένη η Visual Basic στον υπολογιστή του χρήστη.

III. Χαρακτηριστικά της εφαρμογής

Ο σχεδιασμός και η υλοποίηση της εφαρμογής βασίστηκε πάνω σε αρχές εργονομίας και αρχές αλληλεπίδρασης ανθρώπου και υπολογιστή. Ακόμη και ένας όχι και τόσο έμπειρος χρήστης υπολογιστή (για παράδειγμα ένας μαθητής της Α' τάξης του Γυμνασίου) δε θα δυσκολευτεί να τη χρησιμοποιήσει, αφού το περιβάλλον του προγράμματος του είναι οικείο και είναι παρόμοιο με το περιβάλλον οποιασδήποτε από τις γνωστές εφαρμογές που τρέχουν σε λειτουργικά συστήματα της Microsoft. Επίσης, παρέχεται πλήρης υποστήριξη σε αυτόν, μέσω της βοήθειας.

Κατά τις φάσεις του σχεδιασμού και της υλοποίησης της εφαρμογής δόθηκε ιδιαίτερη προσοχή στις εξής τρεις παραμέτρους:

- Διαφορετικοί τύποι χρηστών: Στην εφαρμογή υπάρχουν δύο διαφορετικοί τύποι χρηστών που επιτελούν και διαφορετικές εργασίες με τη βοήθεια της εφαρμογής και έχουν διαφορετικά δικαιώματα: οι καθηγητές που είναι και οι διαχειριστές του συστήματος και οι μαθητές που είναι οι απλοί χρήστες της εφαρμογής.

- «Αρχείο» για κάθε χρήστη: Η Εφαρμογή κρατάει την απαραίτητη πληροφορία για κάθε χρήστη, έτσι ώστε να «θυμάται» σε ποια κατάσταση βρισκόταν ο κάθε λογαριασμός χρήστη πριν τερματιστεί η εφαρμογή και όταν ο ίδιος χρήστης εισάγεται εκ νέου στο σύστημα αυτή η κατάσταση να ανακτάται.

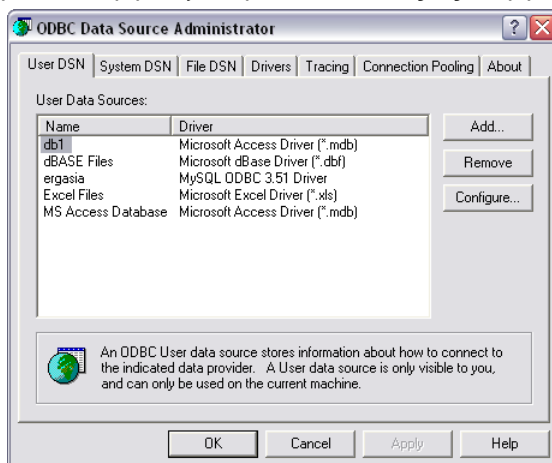
3.2 ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Αφού ολοκληρωθεί με επιτυχία η εγκατάσταση όλων των απαραίτητων αρχείων της εφαρμογής δηλαδή της Visual Basic, το επόμενο βήμα για να δουλεύει ορθά αυτή είναι η εγκατάσταση της Βάσης Δεδομένων στον Η/Υ του χρήστη. Όπως έχει ήδη αναφερθεί και παραπάνω, η Βάση Δεδομένων του συστήματος έχει υλοποιηθεί/αναπτυχθεί στο περιβάλλον της Microsoft Access 2003. Το αρχείο της Βάσης Δεδομένων είναι το "pliroforikiAgimn.mdb".

Η διαδικασία της εγκατάστασης της Βάσης Δεδομένων του συστήματος στον Η/Υ του χρήστη έχει συνοπτικά ως εξής:

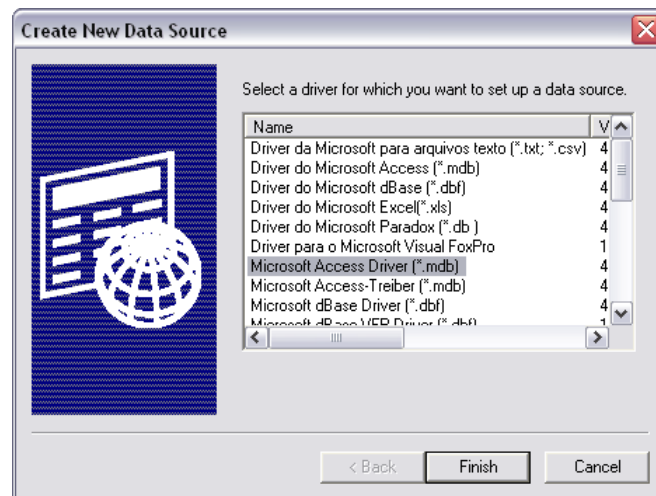
Ο χρήστης ανοίγει τον Πίνακα Ελέγχου (Control Panel) του υπολογιστή του πατώντας "Start → Control Panel". Στη συνέχεια, επιλέγει και ανοίγει "Administrative Tools → Data Sources (ODBC)". Εφόσον είναι εγκατεστημένη η Microsoft Access στον υπολογιστή του χρήστη, δε χρειάζεται κάποια αναβάθμιση στους ODBC drivers του συστήματός του (οι drivers αυτοί εγκαθίστανται αυτόματα με την εγκατάσταση της Microsoft Access). Εάν η Microsoft Access δεν είναι εγκατεστημένη στον υπολογιστή του χρήστη, τότε αυτός μπορεί να κατεβάσει δωρεάν τους drivers αυτούς από τον επίσημο δικτυακό τόπο της Microsoft.

Το παράθυρο διαλόγου που εμφανίζεται μετά από αυτές τις ενέργειες είναι το εξής:



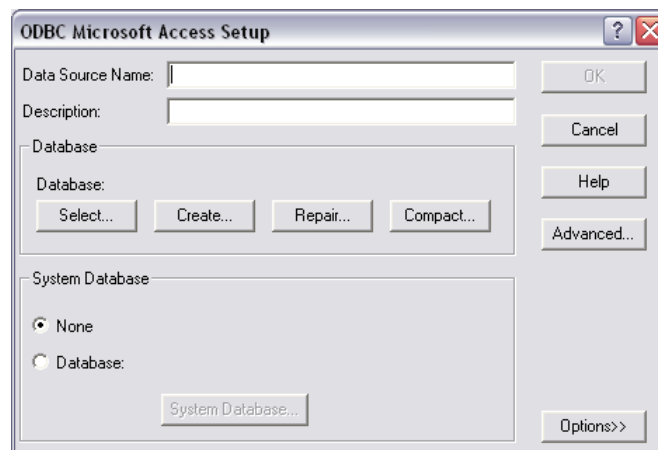
Παράθυρο διαλόγου του Data Sources (ODBC).

Σε παράθυρο διάλογο αυτό ο χρήστης πατάει πάνω στο πλήκτρο "add...". Στη συνέχεια πρέπει να ορίσει τη Βάση Δεδομένων που θα συνδεθεί με τον υπολογιστή του, δηλαδή τη Βάση Δεδομένων της εφαρμογής (pliroforikiAgimn.mdb).



Επιλογή τύπου Βάσης Δεδομένων που θα συνδεθεί με τον Η/Υ του χρήστη.

Ο τύπος της Βάσης Δεδομένων που πρέπει να επιλεγεί είναι ο “Microsoft Access Driver (*.mdb)”. Ο χρήστης στη συνέχεια πατάει το πλήκτρο “Finish” και ανοίγει ένας Browser συστήματος για να επιλεγεί το αρχείο της Βάσης Δεδομένων.



Browser συστήματος για επιλογή της Βάσης Δεδομένων.

Στο πεδίο “Data Source Name:” ο χρήστης πληκτρολογεί “ pliroforikiAgimn ” και στο panel ονόματι “database” ο χρήστης πατάει το πλήκτρο “Select...”.


Στο panel που εμφανίζεται ο χρήστης επιλέγει το φάκελο “Desktop”, οπότε εμφανίζεται και το αρχείο της Βάσης Δεδομένων στα αριστερά του συγκεκριμένου panel. Πατώντας το πλήκτρο “OK” η σύνδεση της Βάσης Δεδομένων με το σύστημα του χρήστη ολοκληρώνεται. Πλέον, η εφαρμογή είναι έτοιμη να «τρέξει».

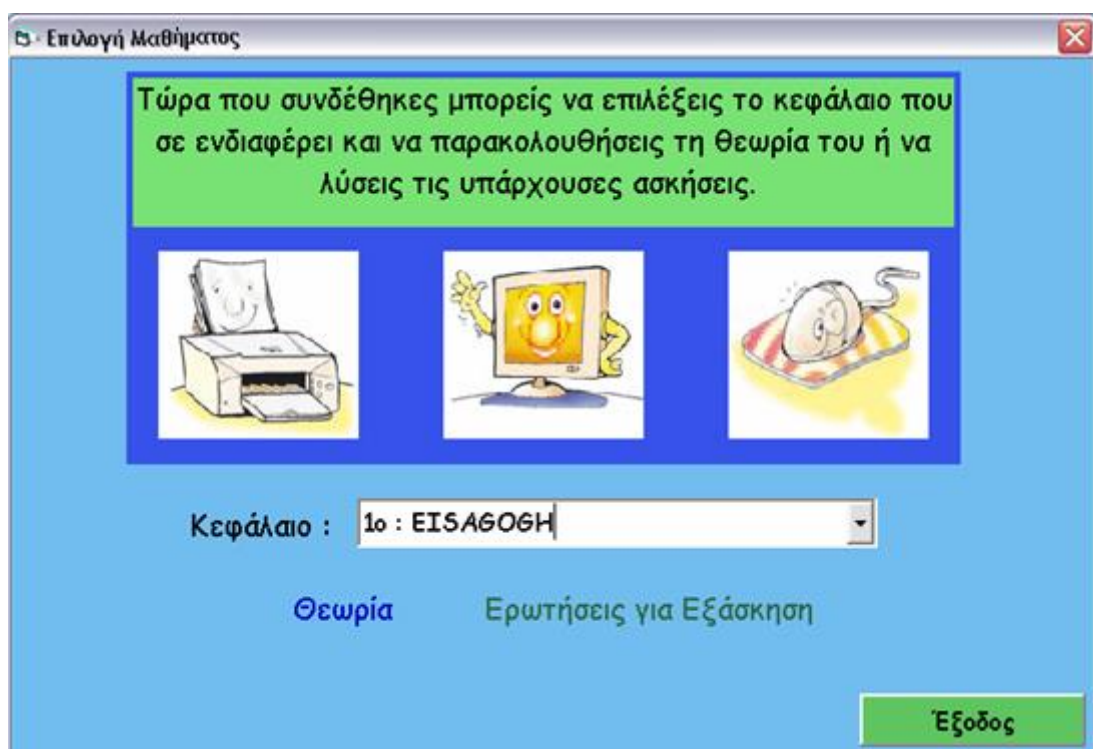
3.3 ΛΕΙΤΟΥΡΓΙΕΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

I. Είσοδος μαθητή στο σύστημα

Μετά την εγκατάσταση και της Βάσης Δεδομένων, η εφαρμογή είναι έτοιμη να τρέξει. Για να ξεκινήσει η εφαρμογή αρκεί ένα διπλό κλικ στο εικονίδιο της εφαρμογής. Η φόρμα που εμφανίζεται έχει ως εξής:

Η αρχική φόρμα της εφαρμογής

Στη φόρμα αυτή, αν ο μαθητής έχει εγγραφεί, κάνει login εισάγοντας το ονοματεπώνυμό του και τον κωδικό του και έπειτα πατάει το κουμπί . Επίσης υπάρχει η δυνατότητα εξόδου από την εφαρμογή πατώντας το αντίστοιχο κουμπί της εξόδου.



Επιλογή κεφαλαίου

Στη συνέχεια ο χρήστης μεταφέρεται στην παραπάνω οθόνη του συστήματος, όπου επιλέγει το κεφάλαιο του οποίου τη θεωρία και τις ασκήσεις θα μελετήσει.

Μάθημα Πληροφορικής Α Τάξης Γυμνασίου
Παρακολούθηση Μαθήματος

1ο : ΕΙΣΑΓΟΓΗ

ΕΝΟΤΗΤΑ 1 – Κεφάλαιο 1: Ο Υπολογιστής και η Επεξεργασία των Δεδομένων

Λέξεις Κλειδιά:
Δεδομένα (Data),
Πληροφορία (Information),
Επεξεργασία (Processing),
Υπολογιστής (Computer),
Πληροφορική (Informatics)


Από τα Δεδομένα στην Πληροφορία

Κωστής: Χρύσα, πόσα χρήματα είπε ο κύριος Πέτρος ότι θα κοστίσει συνολικά η εκδρομή;

Χρύσα: 200 € είναι το κόστος ενοικίασης του λεωφορείου. Νομίζω, όμως, ότι πρέπει να ρωτήσουμε τους συμμαθητές μας για να μάθουμε πόσοι θα έρθουν.

Κωστής: Ναι, βέβαια, όταν μάθουμε κι αυτό θα έχουμε όλα τα στοιχεία, για να βρούμε πόσα χρήματα πρέπει να ζητήσουμε από τον καθένα.

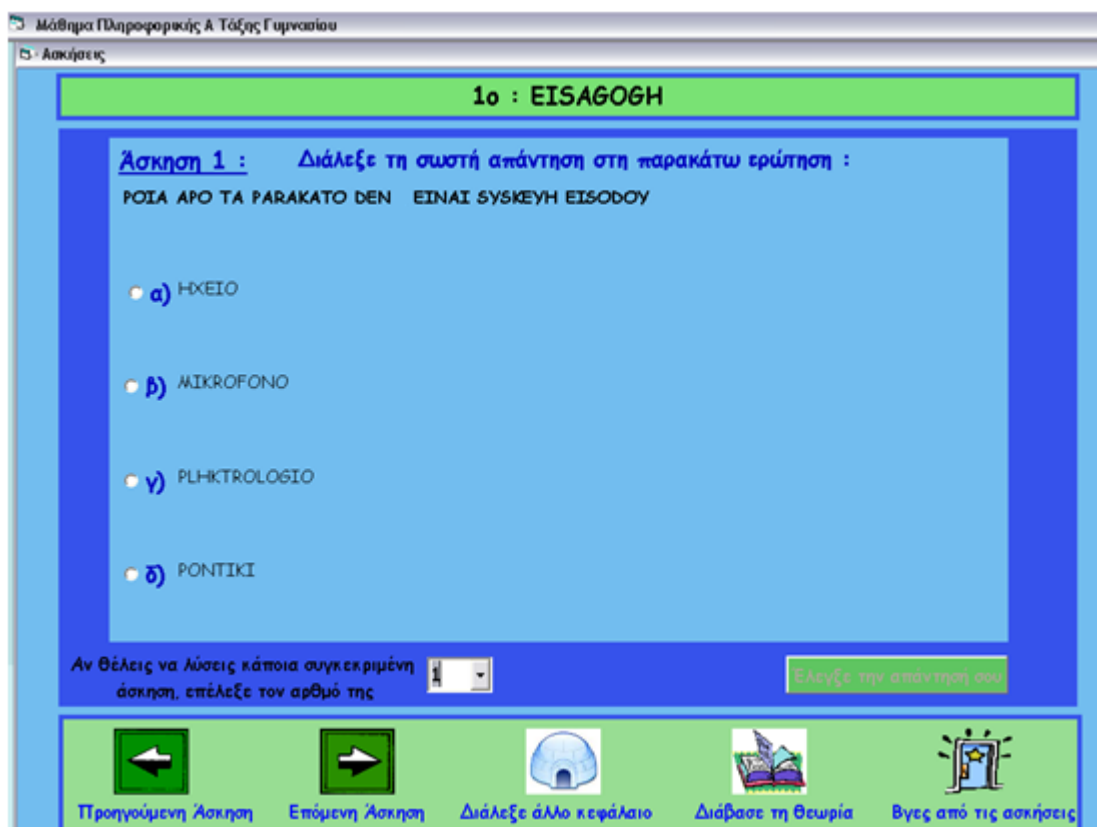
Χρύσα: Πάμε να τους ρωτήσουμε και στο διάλειμμα το ξανασυζητάμε.



Προηγούμενη σελίδα Επόμενη σελίδα Διάλεξε άλλο κεφάλαιο Λύσε τις ασκήσεις Βγες από το μάθημα

Θεωρία Επιλεγμένου Κεφαλαίου

Αφού ο μαθητής μελετήσει τη θεωρία μπορεί να λύσει ασκήσεις του αντίστοιχο κεφαλαίου.



Άσκηση Επιλεγμένου Κεφαλαίου

Επιλέγοντας τις ασκήσεις, εμφανίζεται το παραπάνω screenshot που εμφανίζεται η άσκηση 1 και επιπλέον οι επιλογές πλοήγησης στην προηγούμενη και επόμενη άσκηση, στο μενού επιλογής άλλου κεφαλαίου, στο μενού θεωρίας ή στην έξοδο από τις ασκήσεις, επιλέγοντας το αντίστοιχο εικονίδιο.

Μάθημα Πληροφορικής Α Τάξης Γυμνασίου

Ασκήσεις

1ο : ΕΙΣΑΓΟΓΗ

Άσκηση 1 : Διάλεξε τη σωστή απάντηση στη παρακάτω ερώτηση :

ΡΟΙΑ ΑΠΟ ΤΑ ΠΑΡΑΚΑΤΟ ΔΕΝ ΕΙΝΑΙ ΣΥΣΚΕΥΗ ΕΙΣΟΔΟΥ

- α) ΗΧΕΙΟ
- β) ΜΙΚΡΟΦΟΝΟ
- γ) ΠΛΗΚΤΡΟΛΟΓΙΟ
- δ) ΠΟΝΤΙΚΙ

Πολύ καλά!! Η απάντησή σου είναι σωστή και θα καταχωρηθεί στη βαθμολογία σου

Εντάξει

Αν θέλεις να λύσεις κάποια συγκεκριμένη άσκηση, επέλεξε τον αριθμό της

Έλεγε την απάντησή σου

Προηγούμενη Άσκηση
 Επόμενη Άσκηση
 Διάλεξε άλλο κεφάλαιο
 Διάβασε τη θεωρία
 Βγες από τις ασκήσεις

Σωστή Επίλυση Άσκησης

Ο μαθητής επιλέγει μία από τις απαντήσεις και έπειτα πατά το πλήκτρο ελέγχου και αν είναι σωστή, εμφανίζεται η παραπάνω οθόνη. Αν είναι λάθος, εμφανίζεται η οθόνη που ακολουθεί.

Μάθημα Πληροφορικής Α Τάξης Γυμνασίου

Άσκίσεις

1ο : ΕΙΣΑΓΩΓΗ

Άσκηση 1 : Διάλεξε τη σωστή απάντηση στη παρακάτω ερώτηση :
ΡΟΙΑ ΑΡΟ ΤΑ ΡΑΡΑΚΑΤΟ ΔΕΝ ΕΙΝΑΙ ΣΥΣΚΕΥΗ ΕΙΣΟΔΟΥ

- α) ΗΧΕΙΟ
- β) ΜΙΚΡΟΦΟΝΟ
- γ) ΠΛΗΚΤΡΟΛΟΓΙΟ
- δ) ΠΟΝΤΙΚΙ

Δυστυχώς, η απάντησή σου είναι λανθασμένη. Θα καταχωρηθεί στη βαθμολογία σου, αλλά αν θες μπορείς να τη διορθώσεις

Εντάξει

Αν θέλεις να λύσεις κάποια συγκεκριμένη άσκηση, επέλεξε τον αριθμό της

Έλεγε την απάντησή σου

Προηγούμενη Άσκηση
 Επόμενη Άσκηση
 Διάλεξε άλλο κεφάλαιο
 Διάβασε τη θεωρία
 Βγες από τις ασκήσεις

Λανθασμένη Επιλογή

Έπειτα, ο μαθητής συνεχίζει λύνοντας άλλη άσκηση ή ξαναδιαβάζει τη θεωρία του ίδιου κεφαλαίου ή συνεχίζει στο επόμενο κεφάλαιο.

Αν ο μαθητής δεν έχει εγγραφεί (χρησιμοποιεί πρώτη φορά την εφαρμογή) μπορεί να το κάνει πατώντας στην πρώτη οθόνη το κουμπί και μεταφέρεται στην παρακάτω οθόνη, όπου συμπληρώνει τα απαραίτητα πεδία (Όνομα & Επίθετο, Κωδικός, Επιβεβαίωση Κωδικού και Καθηγητής) και πατάει το πλήκτρο *Συνέχεια* για να ολοκληρωθεί η εγγραφή του.

Νέος Μαθητής

Γράψε το όνομα και το επίθετό σου, και έναν κωδικό, μέχρι 6 γράμματα ή αριθμούς, που θα θυμάσαι για να μπορέσεις να ξανασυνδεθείς. Πρέπει να τον πληκτρολογήσεις 2 φορές για να βεβαιωθείς ότι δεν θα αποθηκευτεί λάθος κωδικός. Τέλος, διάλεξε από τη λίστα το όνομα του καθηγητή σου και πάτησε το κουμπί 'Συνέχεια' για να συνεχίσεις το μάθημα

Όνομα και Επίθετο :

Κωδικός : Επιβεβαίωση Κωδικού

Καθηγητής :

Συνέχεια Πίσω

Εγγραφή Νέου Φοιτητή

II. Είσοδος καθηγητή στο σύστημα

Για να χρησιμοποιήσει ένας καθηγητής το πρόγραμμα αρκεί να επιλέξει το πλήκτρο **Για τον καθηγητή** στην αρχική οθόνη και μεταφέρεται στην επόμενη οθόνη. Αντίστοιχα με το μαθητή, ο καθηγητής μπορεί να κάνει login ή να εγγραφεί, αν χρησιμοποιεί για πρώτη φορά την εφαρμογή.

Για τον καθηγητή

Καλώς ήρθατε στην εφαρμογή για το Ηλεκτρονικό Μάθημα της Πληροφορικής της Α Γυμνασίου. Εισάγετε τα στοιχεία σας ή πατήστε το κουμπί 'Νέος Καθηγητής' για να εγγραφείτε ώστε να μπορείτε να εισάγετε ύλη για το μάθημα, ασκήσεις και να παρακολουθείτε την επίδοση των μαθητών που χρησιμοποιούν την εφαρμογή.

Όνομα και επίθετο :

Κωδικός :

Συνέχεια

Νέος Καθηγητής

Πίσω

Εξοδος

Login Καθηγητή

Κατά την εγγραφή του καθηγητή ο κωδικός της εφαρμογής που πρέπει να εισαχθεί, φαίνεται στο παρακάτω screenshot. Ο κωδικός ζητείται για ασφάλεια, έτσι ώστε να μη μπορεί κάποιος μαθητής να εισαχθεί ως καθηγητής.

Νέος Καθηγητής

Συμπληρώστε τα στοιχεία σας και γράψτε τον κωδικό σας 2 φορές για επιβεβαίωση. Για τον κωδικό της εφαρμογής επικοινωνείτε με τον καθηγητή που είναι υπεύθυνος για την εγκατάσταση της εφαρμογής.

Όνομα και Επίθετο :
nancy

Κωδικός : *** Επιβεβαίωση Κωδικού ***

Κωδικός Εφαρμογής :
ΠΛΗΡΟΦΟΡΙΚΙ Α ΓΥΜΝΑΣΙΟΥ

Συνέχεια Έξοδος

Εγγραφή Καθηγητή

Στη συνέχεια, πατώντας το πλήκτρο *Συνέχεια* εμφανίζεται η ακόλουθη οθόνη με τις επιλογές για τον καθηγητή

Επιλογές Καθηγητή

Επιλογές για τον Καθηγητή

Εισαγωγή Θεωρίας

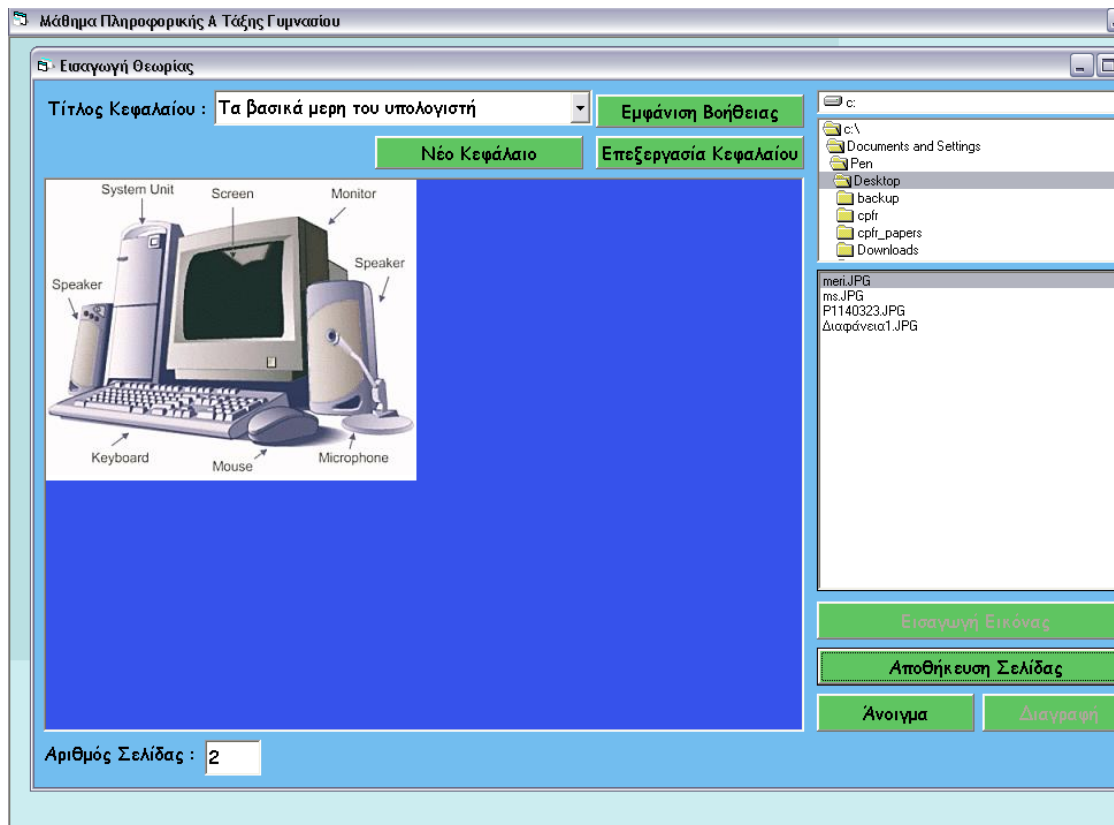
Εισαγωγή Ασκήσεων

Εμφάνιση Βαθμολογιών

Αρχική

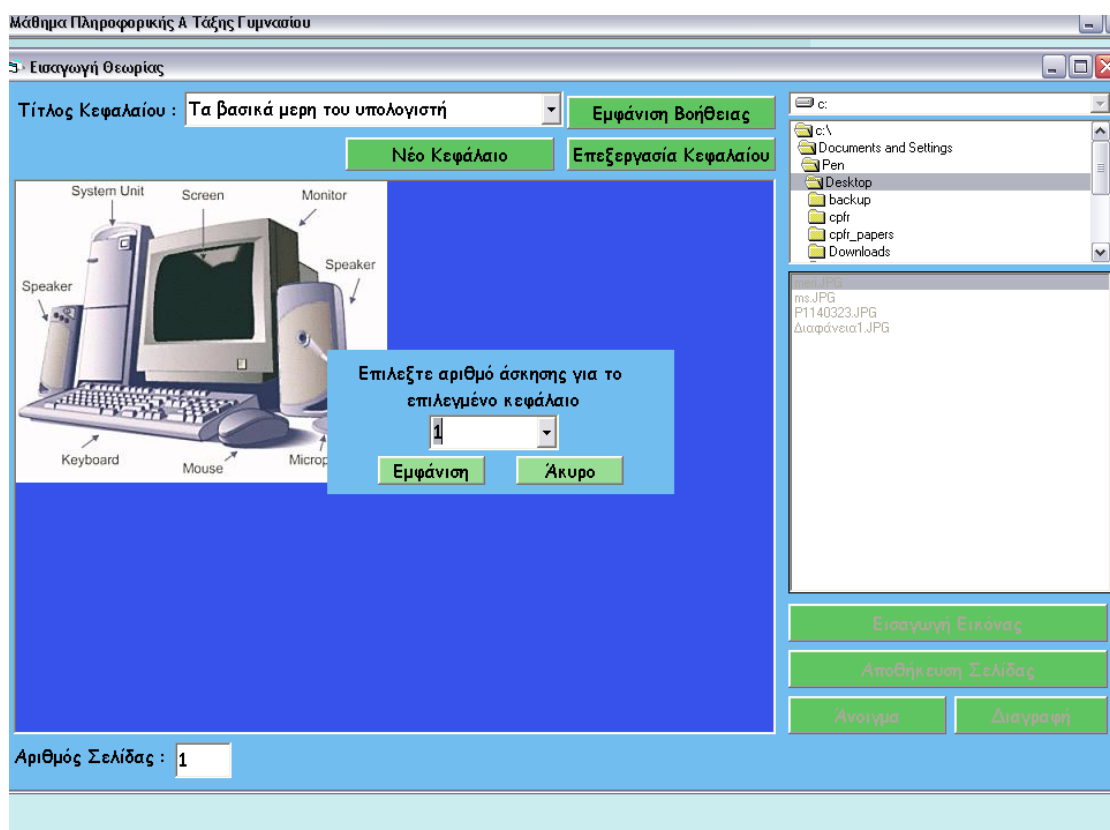
Επιλογές Καθηγητή

Επιλέγοντας την *Εισαγωγή Θεωρίας*, ο καθηγητής έχει τη δυνατότητα να εισάγει τη θεωρία για το κάθε κεφάλαιο από τον υπολογιστή του. Επίσης, έχει τη δυνατότητα να εισάγει νέο κεφάλαιο, να επεξεργαστεί ήδη υπάρχοντα κεφάλαια καθώς επίσης και να βοηθηθεί σε περίπτωση κωλύματος. Η διαδικασία έχει ως εξής: αφού ο καθηγητής επιλέξει το αρχείο προς εισαγωγή, πατάει το πλήκτρο *Εισαγωγή Εικόνας* (αυτό έχει ήδη γίνει στην οθόνη που ακολουθεί) και έπειτα το πλήκτρο *Αποθήκευση Σελίδας*.



Εισαγωγή Θεωρίας

Τέλος, το πλήκτρο *Άνοιγμα*, μπορεί να γίνει επιλογή κάποιας άσκησης, την οποία ο καθηγητής μπορεί να αλλάξει ή και να διαγράψει.



Άνοιγμα Άσκησης

Έχοντας επιλέξει την *Εισαγωγή Ασκήσεων*, ο καθηγητής μπορεί να επιλέξει το κεφάλαιο, το είδος των ασκήσεων (Πολλαπλής Επιλογής, Σωστό-Λάθος), καταχωρεί την εκφώνηση και τις επιλογές αν είναι πολλαπλής επιλογής (κάνει click στη σωστή απάντηση) ή την εκφώνηση και την απάντηση (σωστό λάθος), όπως φαίνονται στα δύο παρακάτω screenshots. Επιπρόσθετα, έχει τη δυνατότητα να ανοίξει παλαιότερα καταχωρημένες ασκήσεις και να τις διορθώσει ή να τις διαγράψει.

Εισαγωγή Ασκήσεων

Κεφάλαιο :

Πολλαπλής Επιλογής Σωστό - Λάθος Καθαρισμός

Εκφώνηση :

Σωστή
Απάντηση : **Επιλογές :**

α)

β)

γ)

δ)

Αριθμός Άσκησης : Καταχώρηση Άνοιγμα Διαγραφή

Εισαγωγή Άσκησης Πολλαπλής Επιλογής

Εισαγωγή Ασκήσεων

Κεφάλαιο :

Πολλαπλής Επιλογής Σωστό - Λάθος Καθαρισμός

Εκφώνηση :

Σωστό
Λάθος

Αριθμός Άσκησης : Καταχώρηση Άνοιγμα Διαγραφή

Εισαγωγή Άσκησης Σωστού-Λάθους

Στην παρακάτω οθόνη, βλέπουμε τη δυνατότητα του καθηγητή να βλέπει την πρόοδο των μαθητών του σε καθεμιά άσκηση ξεχωριστά του προεπιλεγμένου κεφαλαίου. Ο κάθε καθηγητής όμως μπορεί να ελέγξει την πρόοδο μόνο των δικών του μαθητών.

Μάθημα Πληροφορικής Α Τάξης Γυμνασίου

Προβολή Αξιολόγησης Μαθητών

Επιλέξτε Κεφάλαιο :

EΙΣΑΓΟΓΗ

| Μαθητής / Άσκηση | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|------------------|-------|--------|--------|-------|-------|-------|-------|
| antreas | Λάθος | Σωστό! | Σωστό! | Λάθος | Λάθος | Λάθος | Λάθος |

Έξοδος

Βαθμολογία μαθητή

4^ο : ΣΥΜΠΕΡΑΣΜΑΤΑ

4.1 ΑΔΥΝΑΜΙΕΣ ΚΑΙ ΕΛΛΕΙΨΕΙΣ

Η συγκεκριμένη πλατφόρμα που υλοποιήθηκε στα πλαίσια της μεταπτυχιακής αυτής διατριβής αποτελεί ένα συστήματος διδασκαλίας για εκμάθηση βασικών εννοιών της πληροφορικής. Δίνει τη δυνατότητα στους μαθητές να περιηγηθούν στα κεφάλαια διδασκαλίας του συστήματος, να αξιολογήσουν τη γνώση τους μέσω ασκήσεων και τους βοηθά αναφορικά με πιθανές λανθασμένες απαντήσεις.

Σε πραγματικό περιβάλλον, το σύστημα είναι απαραίτητο να είναι εύκολο στην εκμάθηση, εύκαμπτο και ανθεκτικό. Επιπρόσθετα, δεν πρέπει κατά τη λειτουργία του να καταναλώνει όλους τους πόρους του συστήματος γιατί θα καταρρέει εύκολα και θα περιορίζει το χρήστη. Αυτός είναι και ο λόγος της απλοϊκής αλλά πρακτικής σχεδίασης.

Συμπερασματικά, αν και το επίπεδο της τεχνικής υποδομής παίζει ένα σημαντικό ρόλο, υπάρχει ένας μεγάλος αριθμός μη τεχνικών θεμάτων που μπορεί να οδηγήσουν ένα σύστημα σε επιτυχία ή αποτυχία. Για να ξεπεραστούν οποιαδήποτε προκύπτοντα εμπόδια, απαιτείται έρευνα και προσοχή στη λεπτομέρεια.

4.2 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΟΠΤΙΚΕΣ

Η κατασκευή τέτοιου είδους λογισμικών αποτελεί μια ερευνητική περιοχή η οποία μελετά νέες διδακτικές προσεγγίσεις για τη διδασκαλία της Πληροφορικής και άλλων μαθημάτων. Το σύστημα εσπεύει τον μαθητή και προσαρμόζεται στις ανάγκες, κάνοντας την εκμάθηση πιο ευχάριστη και πιο αποδοτική. Προτείνεται διαμορφωτική αξιολόγηση του συστήματος και του γενικότερου σχεδιασμού, ώστε τα συμπεράσματα να συμβάλλουν στη βελτίωση και την ολοκλήρωση του συστήματος.

Η διαδικασία ανάπτυξης ενός τέτοιου συστήματος εμπρικλείει συνιστώσες τεχνικού, εκπαιδευτικού, οικονομικού και κοινωνικού χαρακτήρα που δεν πρέπει να θεωρούνται μεμονωμένες. Τα άτομα που είναι ειδικευμένα στην πληροφορική πρέπει να αποδίδουν ιδιαίτερη σημασία σε μη τεχνικά θέματα, ενώ τα λοιπά άτομα πρέπει να αναγνωρίζουν τη σπουδαιότητα των τεχνικών θεμάτων στην όλη διαδικασία. Όταν όλα τα άτομα που εμπλέκονται στη διαδικασία ανάπτυξης ενός τέτοιου συστήματος κατανοούν ότι υπάρχουν διαφορετικές θεωρήσεις γι' αυτό, η εκτέλεση του έργου γίνεται με επιτυχία.

Ένα εκπαιδευτικό λογισμικό, όπως το συγκεκριμένο, αποτελεί σημαντική επένδυση για έναν εκπαιδευτικό οργανισμό. Έτσι, πρέπει να αξιολογείται περιοδικά η αποδοτικότητα της επένδυσης αυτής. Για κάθε τέτοια αξιολόγηση πρέπει να εξετάζεται η καταλληλότητα του υπάρχοντος συστήματος και η ανάγκη τροποποίησης ή επέκτασής του. Σε μια τέτοια περίπτωση, το αντίστοιχο έργο πρέπει να εκτελείται κατά τρόπο ώστε το νέο σύστημα να αποτελεί ένα συνεκτικό, αποτελεσματικό και αποδοτικό σύνολο των επιμέρους συστατικών του.

Η χρήση της τεχνολογίας στην εκπαίδευση δημιουργεί τεράστιες επιχειρηματικές ευκαιρίες ανάπτυξης. Οι εκπαιδευτικοί φορείς πρέπει να μετασχηματίσουν τις παραδοσιακές, δύσκαμπτες δομές της εκπαίδευσης σε ευέλικτες, ώστε να προαχθεί η όλη διαδικασία και επιπλέον να επικεντρωθεί στον μαθητή.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Virvou, M. & Kabassi K. (2000), An Empirical Study Concerning Graphical User Interfaces that Manipulate Files'. Proceedings of ED-MEDIA 2000, World Conferences on Educational Multimedia and Educational Telecommunications.
- [2] Fischer, S., Steinmetz, R. (2000). Automatic Creation of Exercises in Adaptive Hypermedia Learning Systems. ACM Conference on Hypertext and Hypermedia, 2000, 49-55.
- [3] Brusilovsky, P. (1996). Methods and Techniques for Adaptive Hypermedia. User Modeling and User-Adapted Interaction, 4, 21-45.
- [4] Brusilovsky, P., Kobsa, A., Vassileva, J. (1998) (Eds.). Adaptive Hypertext and Hypermedia. Kluwer Academic Publishers.
- [5] Τρούσσας Χ. (2010), Αλληλεπιδραστικό πρόγραμμα εκμάθησης ξένων γλωσσών με προσαρμοστικές λειτουργίες υποβοήθησης μάθησης, Μεταπτυχιακή διατριβή στο Π.Μ.Σ. «Προηγμένα συστήματα Πληροφορικής» Τμήμα Πληροφορικής Πανεπιστημίου Πειραιώς.