



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



Πτυχιακή Εργασία

“Εργαλείο διδασκαλίας και εξέτασης μαθημάτων που απαιτούν τη χρήση γλώσσας προγραμματισμού”

Χατζηκυριάκου Γεώργιος (ΑΜ:1047)

Επιβλέπων καθηγητής : Δρ. Παπαδουράκης Γεώργιος

Επιτροπή Αξιολόγησης :

- 1.
- 2.
- 3.

Ημερομηνία παρουσίασης:

Ευχαριστίες

Abstract

The aim of this thesis was to study, design and develop of a development environment (IDE), aiming at helping new but also advanced users to learn programming languages in operating systems such as GNU/Linux and Windows.

The application's graphical interface simplifies the process of compiling of assessment's source code as well as the execution of application.

This application can be also used as examining tool, sending the code of assessment for evaluation in a MySQL database or straight forward in an e-mail address.

keywords: learning tool, evaluation tool, development environment, programming languages, computer network, Java, MySQL, PHP

Σύνοψη

Ο σκοπός αυτής της πτυχιακής ήταν η μελέτη, ο σχεδιασμός και η ανάπτυξη ενός περιβάλλοντος ανάπτυξης που θα έχει στόχο να βοηθήσει νέους αλλά και έμπειρους χρήστες να μάθουν γλώσσες προγραμματισμού σε λειτουργικά συστήματα GNU/Linux και Windows. Η γραφική διεπαφή χρήστη απλοποιεί την διαδικασία μεταγλώττισης και εκτέλεσης των εργασιών. Η εφαρμογή μπορεί επίσης να χρησιμοποιηθεί και σαν εξεταστικό εργαλείο, αποστέλλοντας τον κώδικα της εργασίας για αξιολόγηση σε βάση δεδομένων ή κατευθείαν σε διεύθυνση ηλεκτρονικού ταχυδρομείου.

Λέξεις κλειδιά: εργαλείο διδασκαλίας, εξεταστικό εργαλείο, περιβάλλον ανάπτυξης, γλώσσες προγραμματισμού, δίκτυο υπολογιστών, Java, MySQL, PHP

Πίνακας Περιεχομένων

Ευχαριστίες	2
Abstract	3
Σύνοψη	4
1. Εισαγωγή	9
1.1. Κίνητρο για τη διεξαγωγή της εργασίας	9
1.2. Σκοπός και στόχοι της εργασίας	9
1.3. Δομή εργασίας	9
2. Μεθοδολογία υλοποίησης	11
3. Τεχνολογία αιχμής	12
4 Κύριο μέρος	13
4.1 Ανάλυση του προβλήματος	13
4.1.1 Απαιτήσεις συστήματος	13
4.1.2 Περιπτώσεις χρήσης	13
4.1.3 Λειτουργικές απαιτήσεις	14
4.2 Διαγράμματα ροής δεδομένων	18
4.3 Σχεδιασμός Υλοποίησης	25
4.3.1 Δεδομένα εισόδου	25
4.3.2 Σχεδιασμός βάσης δεδομένων	26
4.3.3 Σχεδιασμός γραφικής διεπαφής χρήστη	28
4.4 Τεχνολογίες που χρησιμοποιήθηκαν	29
4.4.1 Java	29
4.4.1.1 Χρήσεις της Java	30
4.4.2 Apache web server	30
4.4.3 Αρχή λειτουργίας ενός web server	31
4.4.3.1 Από την πλευρά του Πελάτη.	31
4.4.3.2 Από την πλευρά του Διακομιστή.	32
4.4.4 PHP	32
4.4.4.1 Πλεονεκτήματα της PHP	33
4.4.5 MySQL	33
4.4.5.1 Πλεονεκτήματα της MySQL	33
4.5 Υλοποίηση	34
4.5.1 Γραφική διεπαφή χρήστη	35
4.5.2 Εκκίνηση εφαρμογής	36
4.5.3 Μενού	36
4.5.4 Άνοιγμα αρχείου πηγαίου κώδικα	36
4.5.5 Αποθήκευση αρχείου πηγαίου κώδικα	37
4.5.6 Εκτύπωση πηγαίου κώδικα	38
4.5.7 Έξοδος από την εφαρμογή	39
4.5.8 Εύρεση κειμένου	39
4.5.9 Μεταγλώττιση εργασίας	40
4.5.10 Εκτέλεση εργασίας	41
4.5.11 Αποσφαλμάτωση	42

4.5.12 Αποστολή κώδικα	43
4.5.13 Εγκατάσταση βάσης δεδομένων	45
4.5.14 Διαγραφή βάσης δεδομένων	45
4.5.15 Προσθήκη χρηστών	45
4.5.16 Διαγραφή χρηστών	46
4.5.17 Διαχείριση κώδικα από τη βάση δεδομένων	46
4.5.18 Ανάλυση κώδικα	47
4.5.19 Μορφοποίηση κώδικα	48
4.5.20 Άνοιγμα τερματικού	48
4.5.21 Δημιουργία αντίγραφου ασφαλείας	49
4.5.22 Εμφάνιση μηνυμάτων σφαλμάτων	50
4.5.23 Πληροφορίες κώδικα	50
4.5.24 Ρυθμίσεις	51
4.5.25 Η σελίδα των εξεταστών	53
4.5.26 Εγκατάσταση	53
4.5.27 Είσοδος χρήστη	53
4.5.28 Επιλογή ημερομηνίας	55
4.5.29 Επιλογή φοιτητή	55
4.5.30 Αξιολόγηση εργασίας	56
4.5.31 Απαιτήσεις συστήματος	58
4.5.32 Βήματα λειτουργίας εφαρμογής	58
5. Αποτελέσματα	59
5.1 Συμπεράσματα	60
5.2 Μελλοντική Εργασία και Επεκτάσεις	60
5.2.1 Επεκτάσεις	61
5.2.2 Ευφυή συστήματα διδασκαλίας	61
5.2.2.1 Βάση Πεδίου	62
5.2.2.2 Το μοντέλο μαθητή	62
5.2.2.3 Παιδαγωγικό Μοντέλο	63
5.2.2.4 Έμπειρο Σύστημα	63
5.2.2.5 Διεπιφάνεια Χρήστη	63
6. Ευφυή συστήματα διδασκαλίας προγραμματισμού υπολογιστών	65
6.1 Το σύστημα Bits	65
6.2 Το σύστημα CIMELS ITS	65
6.3 Το σύστημα C-Tutor	66
6.4 Το σύστημα CITS	67
6.5 Το σύστημα E-TCL	67
6.6 Το σύστημα ALEA	68
6.7 Το σύστημα ELM-ART	68
6.8 Το σύστημα ELM-ART II	69
6.9 Το σύστημα ISIS-Tutor	69
6.10 Το σύστημα Pl@tos	69
Πηγές	70
Παράρτημα	71

Πίνακας Εικόνων

Εικόνα 1: UML διάγραμμα των περιπτώσεων χρήσης.	14
Εικόνα 2: Διάγραμμα ροής δεδομένων πρώτου επιπέδου.	19
Εικόνα 3: Διάγραμμα ροής δεδομένων μαθητή δεύτερου επιπέδου.	19
Εικόνα 4: Διάγραμμα ροής δεδομένων καθηγητή δεύτερου επιπέδου.	20
Εικόνα 5: Διάγραμμα ροής δεδομένων διαχειριστή δεύτερου επιπέδου.	20
Εικόνες 6-20: Διαγράμματα ροής δεδομένων τρίτου επιπέδου.	20-25
Εικόνα 21: Διάγραμμα συσχετίσεων της βάσης δεδομένων.	26
Εικόνα 22: Πρωτότυπο σχέδιο του γραφικού περιβάλλοντος χρήστη	29
Εικόνα 23: Γραφική διεπαφή χρήστη	35
Εικόνα 24: Άνοιγμα αρχείου πηγαίου κώδικα	37
Εικόνα 25: Εκτύπωση πηγαίου κώδικα	38
Εικόνα 26: Εύρεση κειμένου	40
Εικόνα 27: Εκτέλεση εφαρμογής	41
Εικόνα 28: Επιλογέας κλάσης	42
Εικόνα 29: Αποσφαλμάτωση	43
Εικόνα 30: Αποστολή κώδικα	43
Εικόνα 31: Μήνυμα ηλεκτρονικού ταχυδρομείου που περιλαμβάνει μια εργασία σε Java	44
Εικόνα 32: Εγκατάσταση βάσης δεδομένων	45
Εικόνα 33: Διαγραφή βάσης δεδομένων	45
Εικόνα 34: Προσθήκη χρηστών	46
Εικόνα 35: Διαγραφή χρηστών	46
Εικόνα 36: Διαχείριση κώδικα από τη βάση δεδομένων	47
Εικόνα 37: Ανάλυση κώδικα Java με το εργαλείο PMD	48
Εικόνα 38: Η εφαρμογή xterm	49
Εικόνα 39: Δημιουργία αντίγραφου ασφαλείας	49
Εικόνα 40: Εμφάνιση μηνυμάτων σφαλμάτων	50
Εικόνα 41: Πληροφορίες κώδικα	50
Εικόνα 42: Αρχείο ρυθμίσεων examj.properties	52
Εικόνα 43: Ρυθμίσεις	52
Εικόνα 44: Είσοδος χρήστη	54
Εικόνα 45: Επιλογή ημερομηνίας	55
Εικόνα 46: Επιλογή φοιτητή	56
Εικόνα 47: Αξιολόγηση εργασίας	57
Εικόνα 48: Έλεγχος της εφαρμογής σε τοπικό δίκτυο.	59
Εικόνα 49: Η Αρχιτεκτονική του Ευφυούς Συστήματος Διδασκαλίας	61
Εικόνα 50: Αρχιτεκτονική του Bits	65
Εικόνα 51: Αρχιτεκτονική του CIMEL ITS	66
Εικόνα 52: Αρχιτεκτονική του C-Tutor	66
Εικόνα 53: Αρχιτεκτονική του CITS	67
Εικόνα 54: Αρχιτεκτονική του E-TCS	68
Εικόνα 55: Αρχιτεκτονική του ALEA	68

Λίστα Πινάκων

Πίνακας 1: Συγκριτικό περιβάλλοντων ανάπτυξης	12
Πίνακας 2: Εξωτερικές βιβλιοθήκες/εργαλεία	34

1. Εισαγωγή

Σκοπός της εργασίας αυτής είναι η μελέτη, σχεδίαση και ανάπτυξη ενός δικτυακού εργαλείου ανάπτυξης το οποίο όταν εγκατασταθεί σε ένα εργαστήριο ηλεκτρονικών υπολογιστών να δίνει την δυνατότητα στους σπουδαστές να υλοποιούν και να αποστέλλουν για αξιολόγηση εργασίες οι οποίες απαιτούν τη χρήση κάποιας γλώσσας προγραμματισμού. Επίσης στα πλαίσια της εργασίας μελετήθηκαν υφιστάμενα συστήματα ανάπτυξης εφαρμογών.

Μέσα από τη μελέτη αυτή, καθορίστηκαν οι ανάγκες σχεδίασης νέων περιβαλλόντων ανάπτυξης εφαρμογών και δημιουργήθηκε εργαλείο που στοχεύει στην διδασκαλία των γλωσσών προγραμματισμού. Απευθύνεται σε άτομα χωρίς μεγάλη εμπειρία και δίνει έμφαση στην ευκολία εκμάθησης του.

Έτσι, το εργαλείο που αναπτύχθηκε, με κεντρικό θέμα την εκπαίδευση, δίνει την δυνατότητα σε κάποιο σπουδαστή να γράφει, να μεταγλωττίσει, να τρέξει την εργασία του και στην συνέχεια να την στείλει στον εξεταστή του μαθήματος για αξιολόγηση και βαθμολόγηση.

Η γραφική διεπαφή (Εικόνα 1) της εφαρμογής έγινε στη γλώσσα προγραμματισμού Java, ενώ το περιβάλλον αξιολόγησης (Εικόνα 2) των εργασιών αναπτύχθηκε στη γλώσσα προγραμματισμού PHP. Τέλος χρησιμοποιήθηκε η βάση δεδομένων MySQL για την αποθήκευση και οργάνωση των εργασιών των χρηστών.^[1]
[2][3][4][5][6][7][8]

1.1 Κίνητρο για τη διεξαγωγή της εργασίας

Καθε φορά που οι σπουδαστές είναι υποχρεωμένοι να μάθουν μια νέα γλώσσα προγραμματισμού δυσκολεύονται να προσαρμοστούν στο νέο περιβάλλον ανάπτυξης της γλώσσας. Από την άλλη πλευρά είναι επίσης δύσκολο για τον διδάσκοντα να αποθηκεύει και να οργανώνει τις εργασίες που του παραδίδονται.

Η παρούσα εργασία στοχεύει στην επίλυση των πιο πάνω δυσκολιών, με την κατασκευή ενός εργαλείου για εύκολη δημιουργία, μεταγλώττιση, εκτέλεση και αποστολή του πηγαίου κώδικα για αξιολόγηση.

1.2 Σκοπός και στόχοι της εργασίας

Η αρχική πρόταση της πτυχιακής εργασίας στοχεύει στην υλοποίηση των παρακάτω:

1. Να δημιουργηθεί ένα ενιαίο περιβάλλον ανάπτυξης για όλα τα μαθήματα, ανεξαρτήτως της γλώσσας προγραμματισμού που χρησιμοποιείται.
2. Να είναι εύκολο στην εκμάθηση. Ο σπουδαστής θα δίνει περισσότερη έμφαση στην εκμάθηση της γλώσσας προγραμματισμού παρά στην εκμάθηση και ρύθμιση του περιβάλλοντος ανάπτυξης.
3. Να προσφέρει την δυνατότητα εύκολης παράδοσης της εργασίας από τους φοιτητές.
4. Να είναι πολυπλατφορμικό. Η χρήση της γλώσσας προγραμματισμού Java του δίνει τη δυνατότητα να τρέχει σε πολλά λειτουργικά συστήματα (GNU/Linux, BSD, MacOS X, Windows κ.ά).

1.3 Δομή εργασίας

Στα κεφάλαια που ακολούθησαν έγινε ανάλυση της εφαρμογής, μέσω συγκεκριμένης μεθοδολογίας. Θα γίνει αναφορά σε σχετικές εφαρμογές που υπάρχουν. Στο κύριο μέρος της πτυχιακής θα γίνει ανάλυση και

υλοποίηση της εφαρμογής ακολουθώντας κάποια μοντέλα και θα παρουσιαστούν αποτελέσματα και συμπεράσματα από τη χρήση της εφαρμογής σε πραγματικό περιβάλλον.
Τέλος θα αναφερθούν ιδέες για περαιτέρω ανάπτυξη της εφαρμογής με έμφαση στα ευφυή συστήματα διδασκαλίας.

2. Μεθοδολογία υλοποίησης

Για την υλοποίηση της εφαρμογής ακολουθήθηκαν τα παρακάτω βήματα:

1. Προσδιορισμός απαιτήσεων: Στο στάδιο αυτό έγινε συστηματική ανάλυση για τον καθορισμό των απαιτήσεων του συστήματος.
2. Σχεδιασμός: Μετάφραση των λειτουργιών του εργαλείου σε ένα σύστημα που να ικανοποιεί τις απαιτήσεις.
3. Κωδικοποίηση: Υλοποίηση της εφαρμογής με τη χρήση της γλώσσας προγραμματισμού Java.
4. Έλεγχος: Η εφαρμογή εκτελεστηκε για να διαπιστωθεί αν ικανοποιεί τις απαιτήσεις που τέθηκαν.
5. Παράδοση συστήματος: Στο στάδιο αυτό ελέγχεται αν ο χρήστης είναι ικανοποιημένος από την εφαρμογή.
6. Λειτουργία/συντήρηση: Μετά από κάποιο χρονικό διάστημα χρήσης της εφαρμογής ελέγχεται αν υπάρχουν προβλήματα και καταβάλλεται προσπάθεια να επιλυθούν.

Τα βήματα 5 και 6 θα υλοποιηθούν όταν η εφαρμογή εγκατασταθεί σε πραγματικό περιβάλλον (π.χ εργαστήριο)

3. Τεχνολογία αιχμής

Σε αυτό το κεφάλαιο θα γίνει αναφορά σε σχετικά συστήματα που χρησιμοποιήθηκαν από τη σχολή μου για την διδασκαλία μαθημάτων που απαιτούσαν χρήση γλώσσας προγραμματισμού. Ποιο συγκεκριμένα αναφέρομαι στα εργαστηριακά μαθήματα:

1. Εισαγωγή στη πληροφορική (Pascal)
2. Προγραμματισμός (C)
3. Δομές δεδομένων και αλγόριθμοι (C)
4. Αντικειμενοστραφής προγραμματισμός (Java)
5. Μικροϋπολογιστές (προγραμματισμός μικροελεγκτών AVR με χρήση της γλώσσας C)
6. Ανάπτυξη εφαρμογών πολυμέσων (Visual Basic)
7. Θέματα προγραμματισμού διαδικτύου (HTML/Javascript)

Ακολουθεί συγκριτικός πίνακας των περιβάλλοντων ανάπτυξης που χρησιμοποιήθηκαν για την διδασκαλία αυτών των μαθημάτων. Ο πίνακας συγκρίνει:

- Τις πλατφόρμες που μπορεί να τρέξει η εφαρμογή (L=Linux, W= Windows, Mac OS X).
- Τις άδειες χρήσης της εφαρμογής.
- Τις γλώσσες προγραμματισμού που υποστηρίζουν.
- Αν υπάρχει εργαλείο σχεδιασμού γραφικών διεπαφών (GUI builder).
- Αν υπάρχει σύστημα για την αξιολόγηση εργασιών.

Για σκοπούς αναφοράς, η εφαρμογή που αναπτύχθηκε για αυτή τη πτυχιακή, ονομάστηκε “ExamJ”.

Περιβάλλον ανάπτυξης	Πλατφόρμες	Άδεια χρήσης	Pascal	C	C για AVR	Java	Visual Basic	HTML/ Java Script	Άλλες γλώσσες	Ανάπτυξη GUI	Σύστημα Αξιολόγησης
ExamJ IDE	L, W, M	Ανοικτού κώδικα	Ναι	Ναι	Ναι	Ναι	Όχι	Ναι	Ναι	Όχι	Ναι
Microsoft Visual Studio	W	Ιδιόκτητο	Όχι	Ναι	Όχι	Ναι	Ναι	Ναι	Ναι	Ναι	Όχι
Netbeans	L, W, M	Ανοικτού κώδικα	Όχι	Ναι	Όχι	Ναι	Όχι	Όχι	Ναι	Ναι	Όχι
BlueJ	L, W, M	Ανοικτού κώδικα	Όχι	Όχι	Όχι	Ναι	Όχι	Όχι	Όχι	Όχι	Όχι
IAR Embedded Workbench	W	Ιδιόκτητο	Όχι	Ναι	Ναι	Όχι	Όχι	Όχι	Ναι	Όχι	Όχι
Macromedia Dreamweaver	W, M	Ιδιόκτητο	Όχι	Όχι	Όχι	Όχι	Όχι	Ναι	Ναι	Όχι	Όχι

Πίνακας 1: Συγκριτικό περιβάλλοντων ανάπτυξης

4. Κύριο μέρος.

Το κεφάλαιο αυτό ασχολείται με το κύριο μέρος της πτυχιακής δηλαδή με την ανάλυση του προβλήματος, την σχεδίαση της λύσης και την υλοποίηση της λύσης.

4.1 Ανάλυση του προβλήματος

Στη παράγραφο αυτή θα κάνουμε την ανάλυση απαιτήσεων της τελικής εφαρμογής. Κατά την ανάλυση των απαιτήσεων εντοπίζονται για πρώτη φορά οι απαιτήσεις από το λογισμικό και ακολουθούν έναν κύκλο ταξινόμησης, ιεράρχησης και επαλήθευσης. Αποτέλεσμα των εργασιών που εκτελούνται στη φάση αυτή είναι ένα σύνολο απαιτήσεων από το λογισμικό οι οποίες περιγράφονται με μορφή διαγραμμάτων. Η περιγραφή αυτή αποτελεί την είσοδο στο επόμενο βήμα, αυτό της διάκρισης και προδιαγραφής των απαιτήσεων από το λογισμικό.

4.1.1 Απαιτήσεις συστήματος

Η εφαρμογή πρέπει να δίνει την δυνατότητα στο χρήστη να:

1. Να πληκτρολογεί τον πηγαίο κώδικα.
2. Να αποθηκεύει και να ανοίγει αρχεία πηγαίου κώδικα.
3. Να μεταγλωττίζει, να τρέχει και να αποσφαλματώνει την εργασία του.
4. Να στέλνει την εργασία του για αξιολόγηση.
5. Η εφαρμογή πρέπει να δίνει την δυνατότητα στο διδάσκοντα να εισέρχεται στο σύστημα και να αξιολογεί τις εργασίες.
6. Η εφαρμογή πρέπει να δίνει την δυνατότητα στο διδάσκοντα και στο διαχειριστή να διαχειρίζονται τις εργασίες που υπάρχουν στη βάση δεδομένων.
7. Η εφαρμογή πρέπει να δίνει την δυνατότητα στο διαχειριστή να εγκαθιστά και να διαγράφει την βάση δεδομένων.
8. Η εφαρμογή πρέπει να δίνει την δυνατότητα στο διαχειριστή να προσθέτει ή να αφαιρεί χρήστες από το σύστημα.

4.1.2 Περιπτώσεις χρήσης

Αφού προσδιορίσαμε τις απαιτήσεις του συστήματος, κάναμε χρήση διαγράμματος UML ορίσαμε τις περιπτώσεις χρήσης για τον κάθε χρήστη στο σύστημα.

Η UML είναι ένα σύνολο γραφικών παραστάσεων που αποσκοπούν στην περιγραφή και τον σχεδιασμό συστημάτων λογισμικού.

Για την εξαγωγή του διαγράμματος χρησιμοποιήθηκε η ανοικτού κώδικα εφαρμογή Umbrello UML modeller, σε

λειτουργικό σύστημα GNU/Linux.



Εικόνα 1: UML διάγραμμα των περιπτώσεων χρήσης.

4.1.3 Λειτουργικές Απαιτήσεις

Σε αυτή την ενότητα κάναμε μία εκτενέστερη περιγραφή των απαιτήσεων, δίνοντας έμφαση στη είσοδο, την επεξεργασία και την έξοδο που θα έχει η κάθε λειτουργία:

Αριθμός	1
Τίτλος	Πληκτρολόγηση πηγαίου κώδικα.
Χρήστης	Student
Περιγραφή	Η εφαρμογή εμφανίζει πεδίο για την πληκτρολόγηση του πηγαίου κώδικα.
Είσοδος	Πληκτρολόγηση πηγαίου κώδικα
Επεξεργασία	Επεξεργασία πηγαίου κώδικα
Έξοδος	Πηγαίος κώδικας έτοιμος προς αποθήκευση.

Αριθμός	2
Τίτλος	Αποθήκευση αρχείου πηγαίου κώδικα
Χρήστης	Student
Περιγραφή	Η εφαρμογή εμφανίζει πεδίο για την εισαγωγή του ονόματος του αρχείου πηγαίου κώδικα που πρόκειται να αποθηκευτεί.
Είσοδος	Όνομα αρχείου από το πληκτρολόγιο.
Επεξεργασία	Έλεγχος ύπαρξης αρχείου με το ίδιο όνομα, επαλήθευση
Έξοδος	Ενημερωμένο αρχείο πηγαίου κώδικα ή μήνυμα λάθους.

Αριθμός	3
Τίτλος	Άνοιγμα αρχείου πηγαίου κώδικα
Χρήστης	Student
Περιγραφή	Η εφαρμογή εμφανίζει πεδίο για την επιλογή του αρχείου πηγαίου κώδικα που πρόκειται να αποθηκευτεί.
Είσοδος	Όνομα αρχείου πηγαίου κώδικα
Επεξεργασία	Επαλήθευση, δυνατότητα αποθήκευσης της προηγούμενης εργασίας ή μήνυμα λάθους..
Έξοδος	Παρουσίαση της λίστας του πηγαίου κώδικα.

Αριθμός	4
Τίτλος	Μεταγλώττιση πηγαίου κώδικα
Χρήστης	Student
Περιγραφή	Στην εφαρμογή υπάρχει διαδικασία μεταγλώττισης του πηγαίου κώδικα βάση του ονόματος του αρχείου και της γλώσσας προγραμματισμού που επιλέχτηκε
Είσοδος	Όνομα αρχείου πηγαίου κώδικα
Επεξεργασία	Αποθήκευση αρχείου πηγαίου κώδικα, εκτέλεση διεργασίας μεταγλώττισης λαμβάνοντας τις απαιτούμενες παραμέτρους
Έξοδος	Εκτελέσιμο αρχείο ή μήνυμα λάθους.

Αριθμός	5
Τίτλος	Εκτέλεση/τρέξιμο εφαρμογής

Χρήστης	Student
Περιγραφή	Στην εφαρμογή υπάρχει διαδικασία εκτέλεσης του εκτελέσιμου αρχείου βάση του ονόματος του αρχείου και της γλώσσας προγραμματισμού που επιλέχτηκε
Είσοδος	Όνομα εκτελέσιμου αρχείου
Επεξεργασία	Εκτέλεση διεργασίας εκτέλεσης της εφαρμογής λαμβάνοντας υπόψη τις απαιτούμενες παραμέτρους
Έξοδος	Εμφάνιση τερματικού και εκτέλεση του εκτελέσιμου αρχείου ή μήνυμα λάθους.

Αριθμός	6
Τίτλος	Αποσφαλμάτωση εκτελέσιμου αρχείου
Χρήστης	Student
Περιγραφή	Στην εφαρμογή υπάρχει διαδικασία αποσφαλμάτωσης του εκτελέσιμου αρχείου βάση του ονόματος του αρχείου και της γλώσσας προγραμματισμού που επιλέχτηκε
Είσοδος	Όνομα εκτελέσιμου αρχείου
Επεξεργασία	Εκτέλεση διεργασίας αποσφαλμάτωσης της εφαρμογής λαμβάνοντας υπόψη τις απαιτούμενες παραμέτρους
Έξοδος	Εμφάνιση τερματικού και αποσφαλμάτωση του εκτελέσιμου αρχείου ή μήνυμα λάθους.

Αριθμός	7
Τίτλος	Αποστολή πηγαίου κώδικα
Χρήστης	Student
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο σπουδαστής συμπληρώνει τα στοιχεία του και επιλέγει τρόπο αποστολής της εργασίας.
Είσοδος	Πηγαίος κώδικας, στοιχεία σπουδαστή
Επεξεργασία	Εκτέλεση διεργασίας αποστολής κώδικα λαμβάνοντας υπόψη την επιλογή του σπουδαστή ή μήνυμα λάθους..
Έξοδος	Προσθήκη εργασίας της βάσης δεδομένων ή αποστολή της σε διεύθυνση ηλεκτρονικού ταχυδρομείου ή μήνυμα λάθους..

Αριθμός	8
Τίτλος	Εισαγωγή στο σύστημα
Χρήστης	Teacher
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο καθηγητής εισάγει το όνομα

	χρήστη και τον κωδικό για εισαγωγή του στο σύστημα της web εφαρμογής.
Είσοδος	Όνομα χρήστη, κωδικός
Επεξεργασία	Ταυτοποίηση χρήστη
Έξοδος	Εισαγωγή στο σύστημα ή μήνυμα λάθους.

Αριθμός	9
Τίτλος	Αξιολόγηση εργασίας
Χρήστης	Teacher
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο καθηγητής εισάγει τον βαθμό και το κείμενο αξιολόγησής της εργασίας
Είσοδος	Βαθμός, αξιολόγηση
Επεξεργασία	Εκτέλεση διεργασίας αποστολής των δεδομένων στη βάση δεδομένων για ενημέρωση των πεδίων.
Έξοδος	Εμφάνιση ενημερωμένων πεδίων ή μήνυμα λάθους.

Αριθμός	10
Τίτλος	Διαχείριση των εργασιών από τη βάση δεδομένων
Χρήστης	Teacher, Administrator
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο καθηγητής ή ο διαχειριστής μπορούν να φορτώσουν ή να διαγράψουν κώδικα από την βάση δεδομένων.
Είσοδος	Όνομα χρήστη, κωδικός
Επεξεργασία	Επιλογή στοιχείου από την βάση δεδομένων
Έξοδος	Εμφάνιση ή διαγραφή πηγαίου κώδικα από τη βάση δεδομένων ή μήνυμα λάθους.

Αριθμός	11
Τίτλος	Εγκατάσταση της βάσης δεδομένων
Χρήστης	Administrator
Περιγραφή	Στην εφαρμογή υπάρχει διαδικασία για την δημιουργία της βάσης δεδομένων και των πινάκων που απαιτούνται για την αποστολή εργασιών στη βάση δεδομένων.
Είσοδος	Όνομα χρήστη, κωδικός
Επεξεργασία	Εκτέλεση των κατάλληλων ερωτημάτων προς την βάση δεδομένων.
Έξοδος	Δημιουργία της βάσης δεδομένων

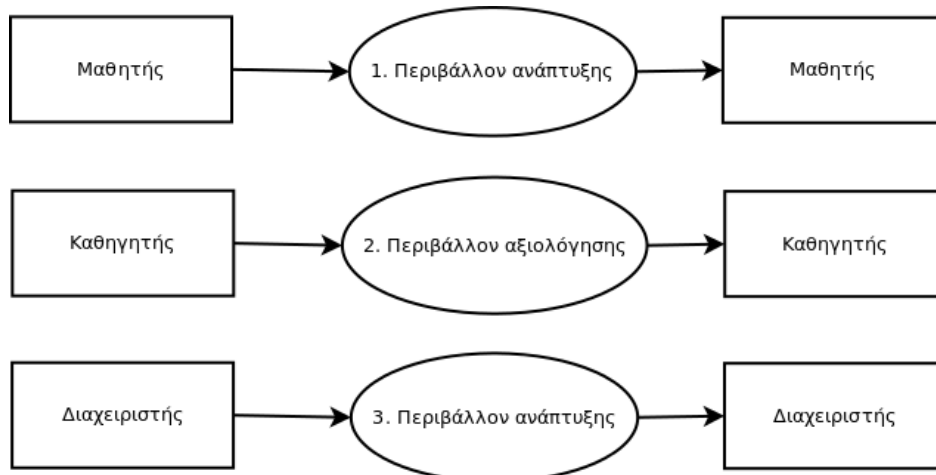
Αριθμός	12
Τίτλος	Διαγραφή της βάσης δεδομένων
Χρήστης	Administrator
Περιγραφή	Στην εφαρμογή υπάρχει διαδικασία για την διαγραφή της βάσης δεδομένων και των πινάκων που χρησιμοποιούνταν από την εφαρμογή.
Είσοδος	Όνομα χρήστη, κωδικός
Επεξεργασία	Εκτέλεση των κατάλληλων ερωτημάτων προς την βάση δεδομένων.
Έξοδος	Διαγραφή της βάσης δεδομένων

Αριθμός	13
Τίτλος	Προσθήκη χρήστη
Χρήστης	Administrator
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο διαχειριστής της βάσης δεδομένων εισάγει τα στοιχεία του καθηγητή.
Είσοδος	Όνομα χρήστη, κωδικός, στοιχεία καθηγητή
Επεξεργασία	Εκτέλεση των κατάλληλων ερωτημάτων προς την βάση δεδομένων.
Έξοδος	Εγγραφή του καθηγητή στη βάση δεδομένων

Αριθμός	14
Τίτλος	Διαγραφή χρήστη
Χρήστης	Administrator
Περιγραφή	Η εφαρμογή εμφανίζει φόρμα μέσω της οποίας ο διαχειριστής της βάσης δεδομένων εισάγει το όνομά χρήστη του καθηγητή που πρόκειται να διαγραφεί από το σύστημα.
Είσοδος	Όνομα χρήστη, κωδικός, όνομα χρήστη καθηγητή.
Επεξεργασία	Εκτέλεση των κατάλληλων ερωτημάτων προς την βάση δεδομένων.
Έξοδος	Διαγραφή του καθηγητή από βάση δεδομένων

4.2 Διαγράμματα ροής δεδομένων

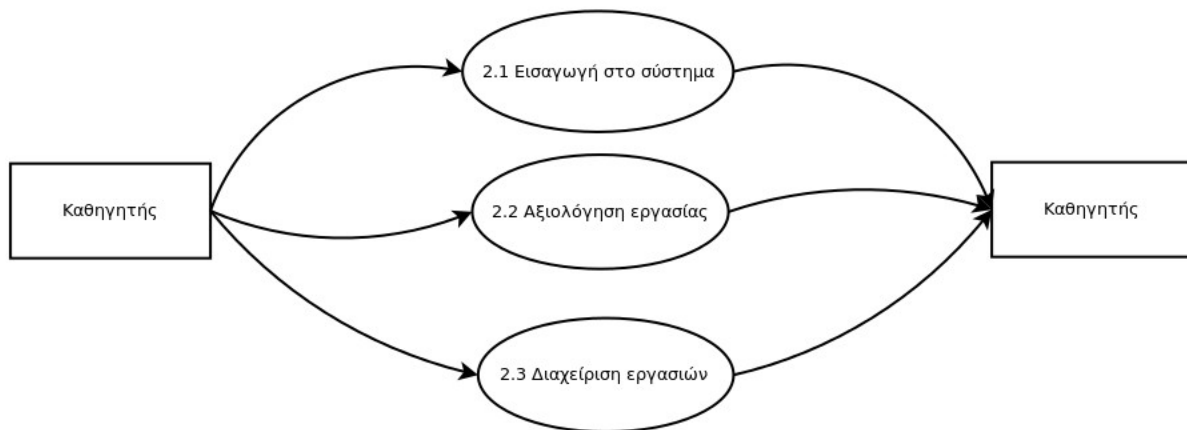
Σύμφωνα με τις απαιτήσεις που καταγράφηκαν, δημιουργήθηκαν τα παρακάτω διαγράμματα ροής δεδομένων. Για τη δημιουργία τους χρησιμοποιήθηκε η εφαρμογή Dia.



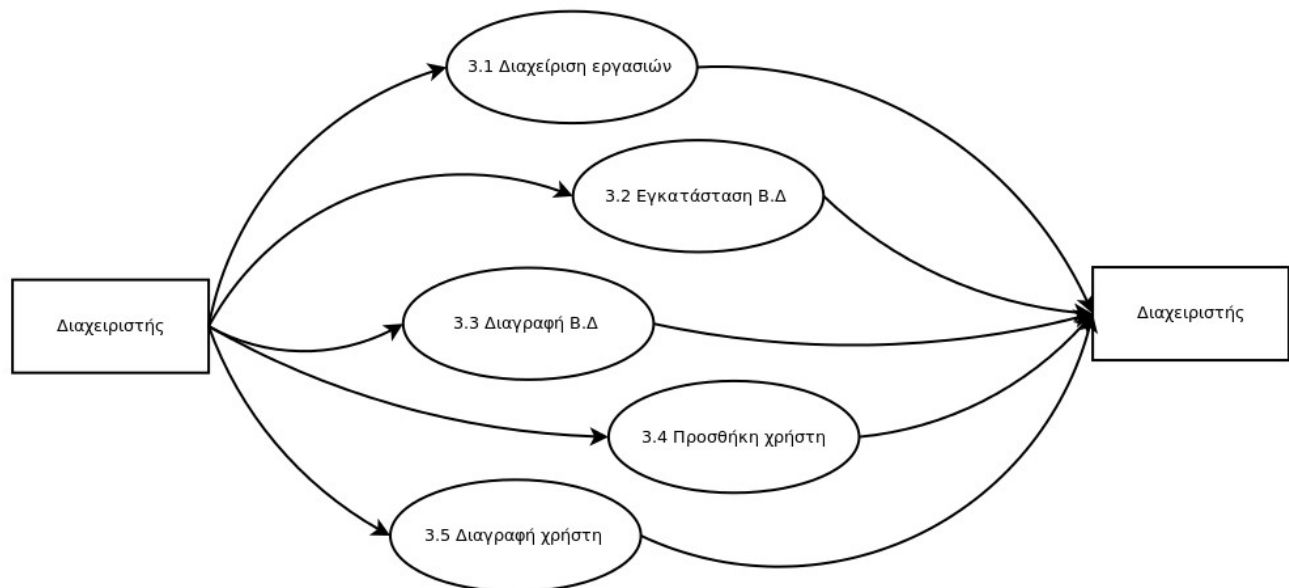
Εικόνα 2: Διάγραμμα ροής δεδομένων πρώτου επιπέδου.



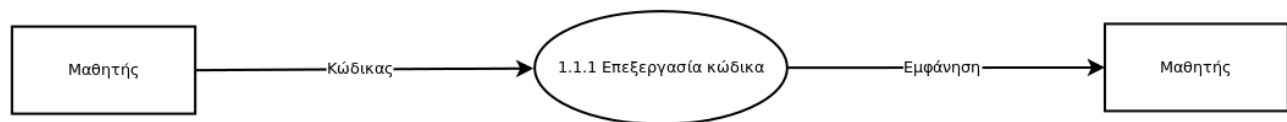
Εικόνα 3: Διάγραμμα ροής δεδομένων μαθητή δεύτερου επιπέδου.

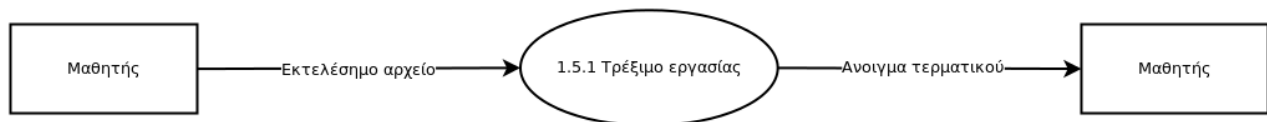
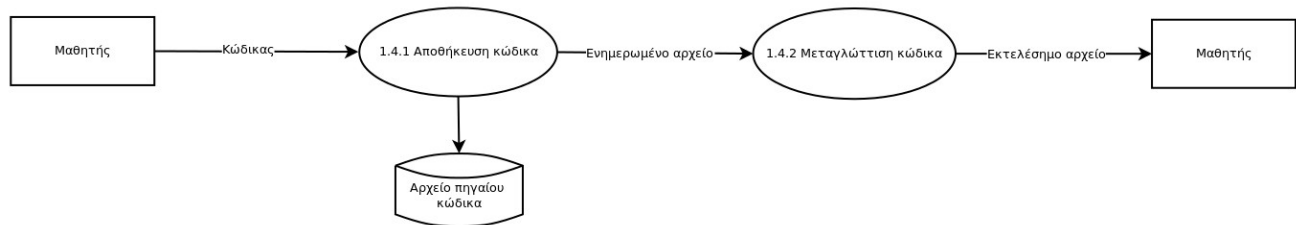
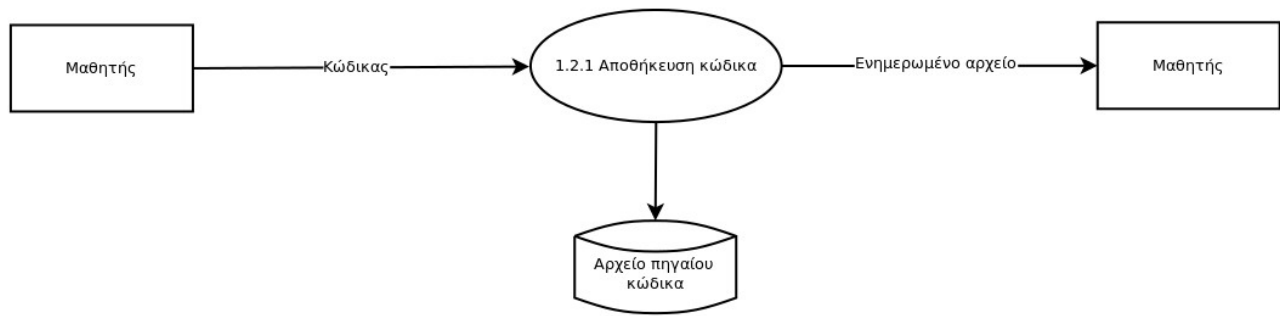


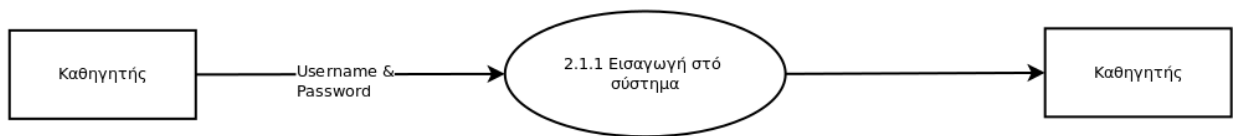
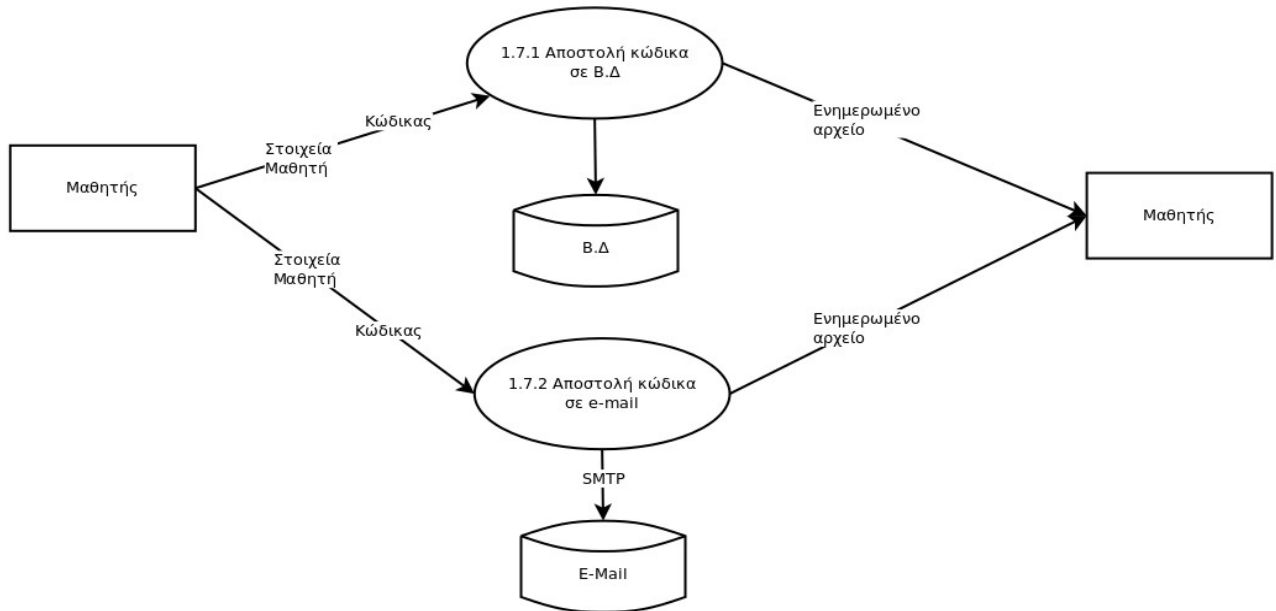
Εικόνα 4: Διάγραμμα ροής δεδομένων καθηγητή δεύτερου επιπέδου.

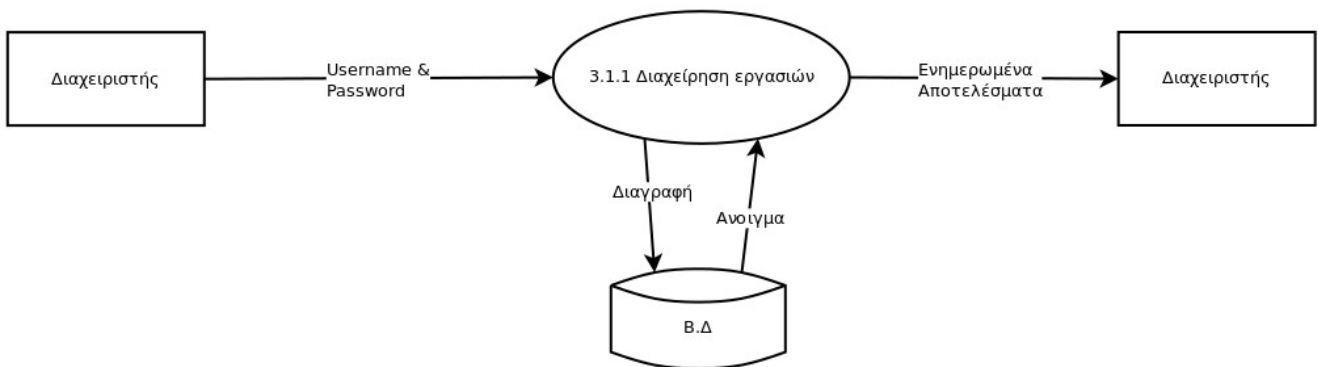
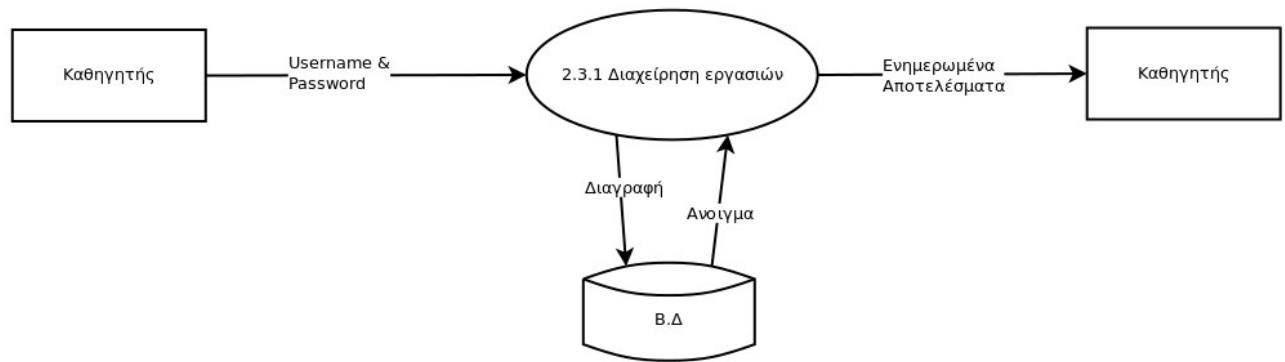
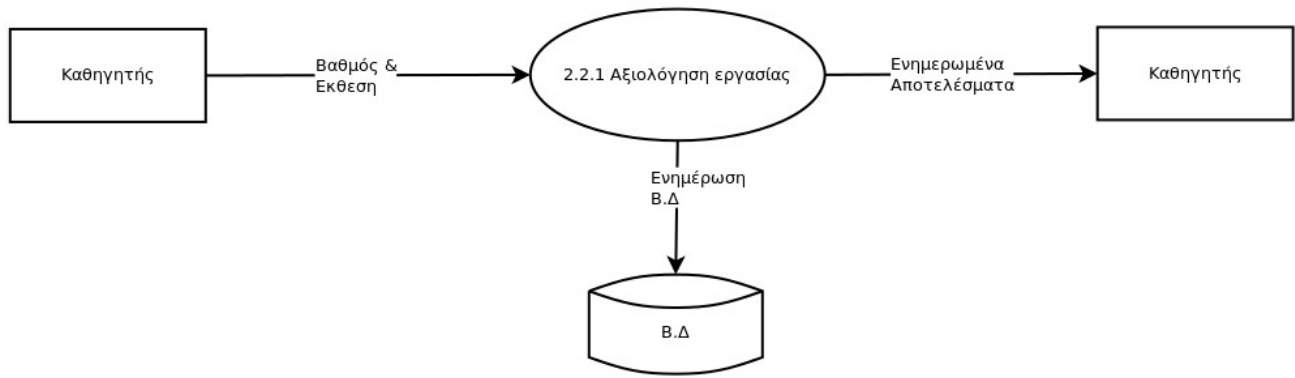


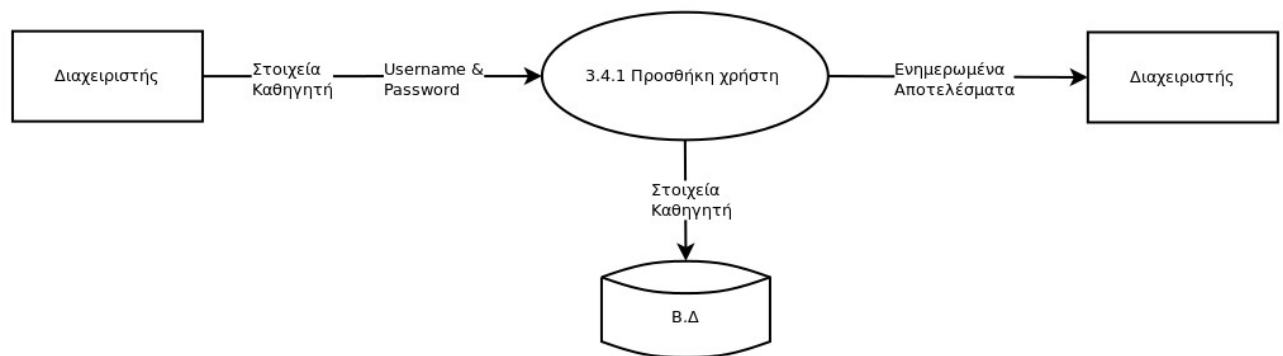
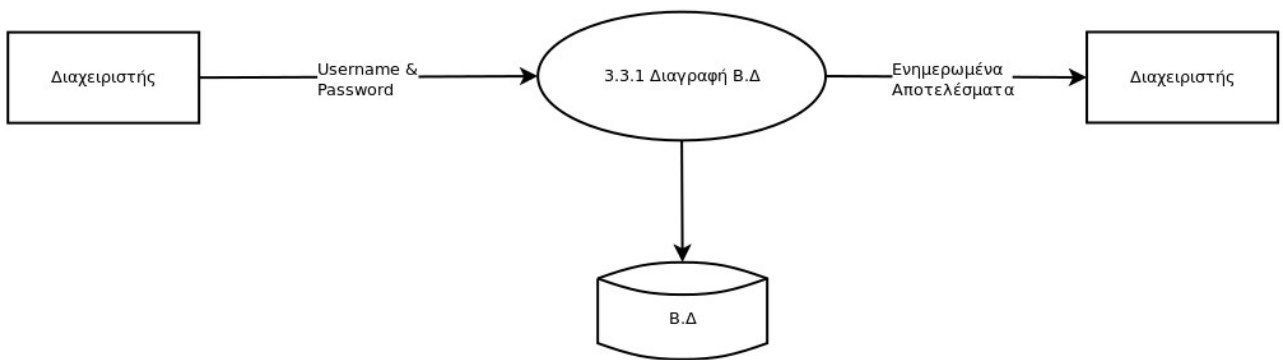
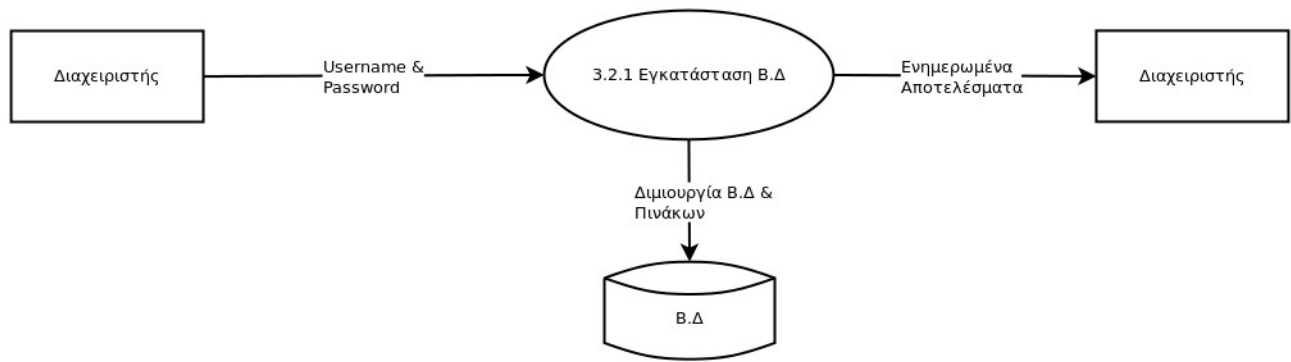
Εικόνα 5: Διάγραμμα ροής δεδομένων διαχειριστή δεύτερου επιπέδου.

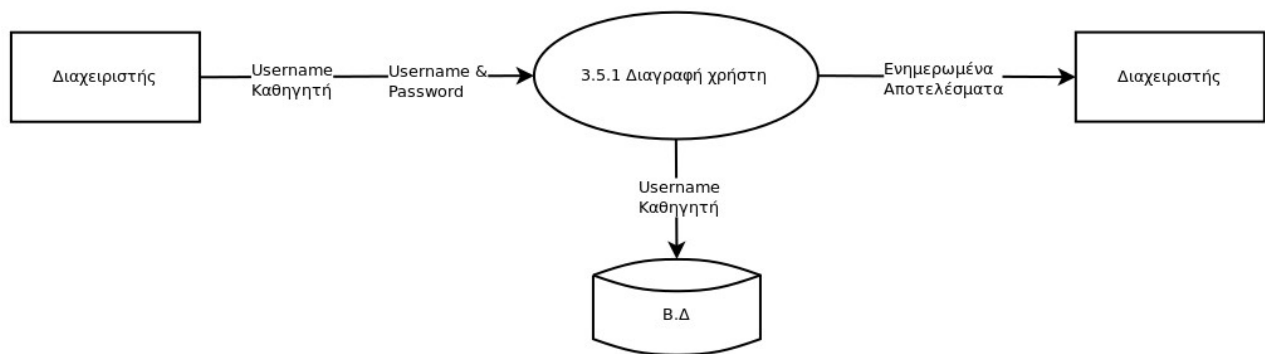












Εικόνες 6-20: Διαγράμματα ροής δεδομένων τρίτου επιπέδου.

4.3 Σχεδιασμός Υλοποίησης

4.3.1 Δεδομένα εισόδου

Σε αυτή την ενότητα καταγράφονται αναλυτικά τα δεδομένα εισόδου της εφαρμογής όπως αυτά παρουσιάστηκαν τα διαγράμματα ροής δεδομένων:

Στοιχεία μαθητή

Όνομα
Επίθετο
Αριθμός μητρώου
Ομάδα

Στοιχεία καθηγητή

Όνομα
Επίθετο
Ηλεκτρονικό ταχυδρομείο
Όνομα χρήστη
κωδικός

Κώδικας

Όνομα έργου
Γλώσσα προγραμματισμού
Πηγαίος κώδικας

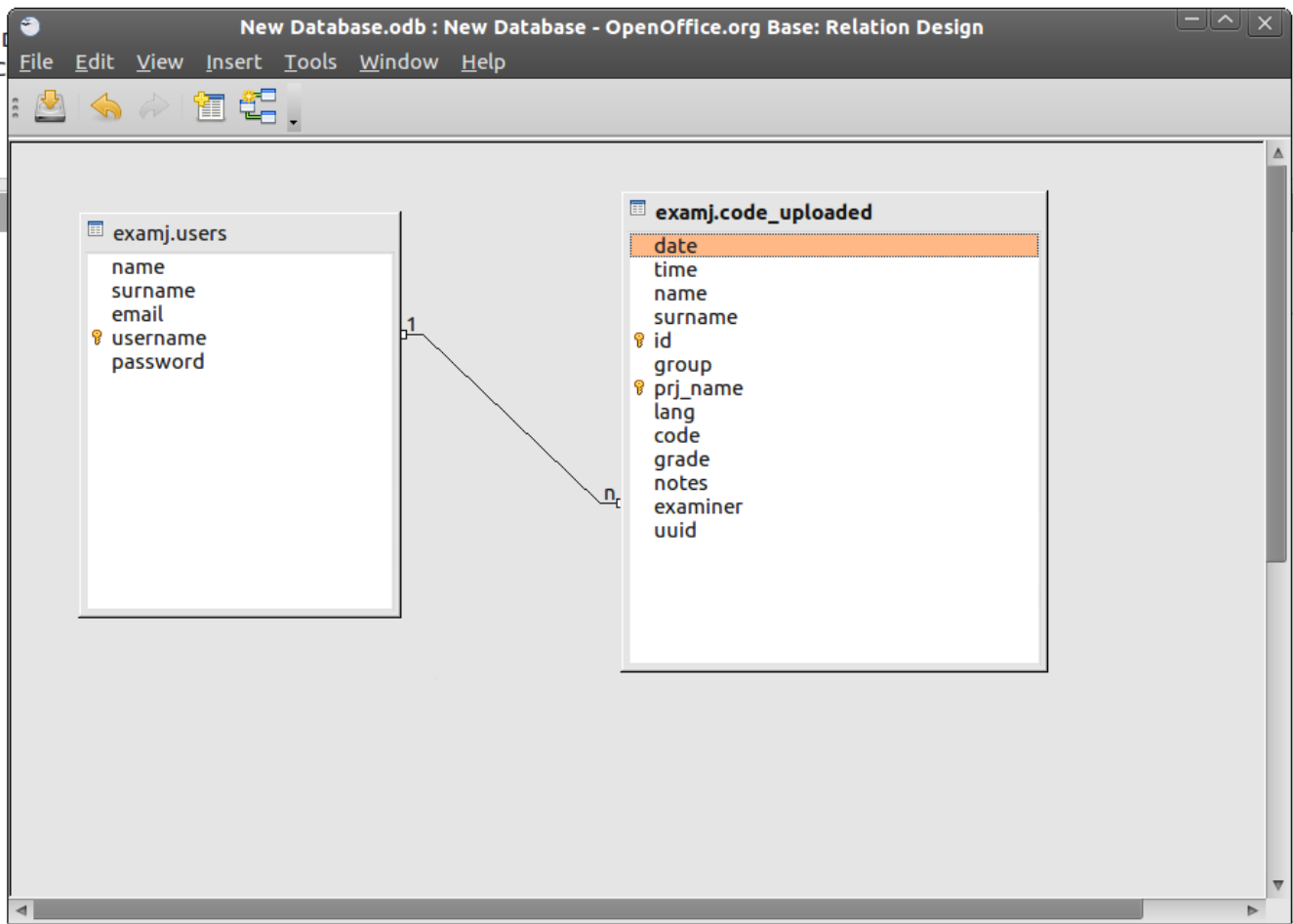
Βαθμολογία

Βαθμός
Εκθεση

4.3.2 Σχεδιασμός βάσης δεδομένων

Σύμφωνα με τα δεδομένα εισόδου της εφαρμογής σχεδιάστηκε η βάση δεδομένων, ποιο κάτω ακολουθούν το σχεδιάγραμμα συσχετίσεων μεταξύ των πινάκων της βάσης και ο κώδικας SQL που χρησιμοποιήθηκε για τη δημιουργία τους.

Για τη δημιουργία των συσχετίσεων χρησιμοποιήθηκε η εφαρμογή Database της σουίτας γραφείου OpenOffice.org.



Εικόνα 21: Διάγραμμα συσχετίσεων της βάσης δεδομένων.

Κώδικας SQL:

```

-- MySQL dump 10.13  Distrib 5.1.49, for debian-linux-gnu (i686)
--
-- Host: localhost    Database: examj
--
-----
-- Server version 5.1.49-lubuntu8
--
--
-- Current Database: `examj`
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ `examj` /*!40100 DEFAULT CHARACTER SET
latin1 */;

USE `examj`;

--
-- Table structure for table `code_uploaded`
--

DROP TABLE IF EXISTS `code_uploaded`;

CREATE TABLE `code_uploaded` (
  `date` date NOT NULL DEFAULT '0000-00-00',
  `time` time NOT NULL DEFAULT '00:00:00',
  `name` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `surname` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `id` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `group` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `prj_name` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `lang` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `code` longtext COLLATE utf8_unicode_ci NOT NULL,
  `grade` float DEFAULT NULL,
  `notes` longtext COLLATE utf8_unicode_ci NOT NULL,
  `examiner` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `uuid` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`id`,`prj_name`,`examiner`),
  KEY `examiner` (`examiner`),
  CONSTRAINT `code_uploaded_ibfk_1` FOREIGN KEY (`examiner`) REFERENCES `users`
(`username`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

--
-- Table structure for table `users`
--

DROP TABLE IF EXISTS `users`;

CREATE TABLE `users` (
  `name` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `surname` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `username` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
  `password` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`username`)

```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

--
-- Dumping data for table `users`
--

INSERT INTO `users` VALUES
('examj_user','examj_user','examj_user','examj_user','3da531d62a9c3c0b4ca09c2e3c1e2a97');

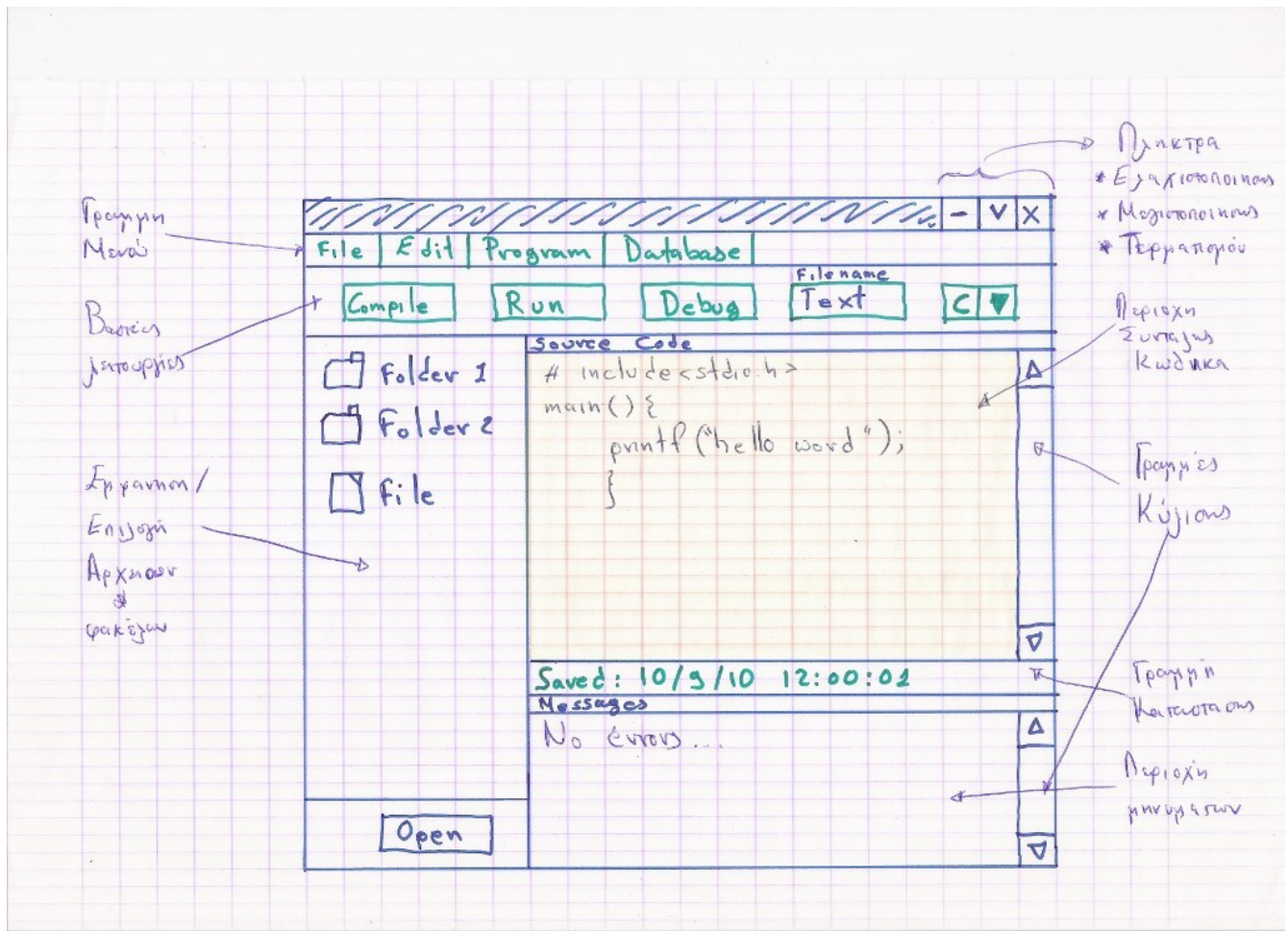
-- Dump completed on 2010-10-14 12:33:18
```

4.3.3 Σχεδιασμός γραφικής διεπαφής χρήστη

Για την γραφική διεπαφή μεταξύ του χρήστη και του υπολογιστή λήφθηκαν υπόψιν τα παρακάτω:

- Απλός σχεδιασμός.
- Εύκολη χρήση βασικών λειτουργιών.
- Εύκολη εκμάθηση εφαρμογής.

Ακολουθεί ένα πρωτότυπο σχέδιο του γραφικού περιβάλλοντος χρήστη:



Εικόνα 22: Πρωτότυπο σχέδιο του γραφικού περιβάλλοντος χρήστη

4.4 Τεχνολογίες που χρησιμοποιήθηκαν

Παρακάτω ακολουθούν σύντομες περιγραφές των τεχνολογιών που μελετήθηκαν για την υλοποίηση της εφαρμογής. Για το περιβάλλον ανάπτυξης χρησιμοποιήθηκε η γλώσσα προγραμματισμού Java, ενώ για την εφαρμογή αξιολόγησης και βαθμολόγησης των εργασιών χρησιμοποιήθηκε η γλώσσα PHP. Για βάση δεδομένων χρησιμοποιήθηκε η εφαρμογή MySQL.

4.4.1 Java

Η Java είναι μία αντικειμενοστραφής (Object Oriented) γλώσσα προγραμματισμού η οποία αναπτύχθηκε από τη Sun Microsystems. Έχει πολλά κοινά χαρακτηριστικά με τις γλώσσες C, C++, και Objective C. Η γλώσσα προγραμματισμού Java σχεδιάστηκε για να δώσει λύσεις σε μία σειρά πρακτικών προβλημάτων που εμφανίζονταν σε όλες τις ήδη υπάρχουσες γλώσσες. Η Java είναι μία απλή, αντικειμενοστραφής, έχουσα δικτυακή γνώση, μεταφράσιμη, εύρωστη, ασφαλής, ανεξάρτητη αρχιτεκτονικής, μεταφέρσιμη, υψηλής απόδοσης, πολυνηματική, δυναμική γλώσσα. Η Java αυτή τη στιγμή εκπροσωπεί την τελευταία γενιά των γλωσσών προγραμματισμού. Λόγω του ότι τα πάντα πλέον αρχίζουν να περιστρέφονται γύρω από το διαδίκτυο, με τη μορφή υπηρεσιών, η Java είναι ένα πολύ

εύχρηστο εργαλείο για την ανάπτυξη εφαρμογών που συνδυάζουν την ευκολία υλοποίησης, τη δυνατότητα να τρέχουν σε πολλά διαφορετικά περιβάλλοντα, να μην χρειάζεται ο χρήστης να κάνει καμία εγκατάσταση στον υπολογιστή του προκειμένου να τρέξει κάποια εφαρμογή (π.χ. τα applets δεν προαπαιτούν να είναι εγκατεστημένο κάτι, τρέχουν μέσα από τον ίδιο τον browser), τη δυνατότητα να είναι δικτυακές κλπ. Επιπλέον η ίδια η Sun έχει καταφέρει να κρατήσει τη Java ελεύθερη, και δίνει μάλιστα και τον κώδικά της σε όποιον το ζητήσει. Ακόμη έχουν γίνει πάρα πολλές προσπάθειες για ανάπτυξη πακέτων και APIs που κάνουν σχεδόν οτιδήποτε, από voice και βάσεις δεδομένων μέχρι κλήση και εκτέλεση απομακρυσμένου κώδικα (μέσω RMI και CORBA), και υπάρχει πάρα πολύ μεγάλη υποστήριξη πάνω σε αυτά τα θέματα.

4.4.1.1 Χρήσεις της Java

- Δικτυακές εφαρμογές οποιουδήποτε είδους, που τρέχουν σε οποιοδήποτε λειτουργικό (από e-mail και web servers και clients μέχρι Application Servers)
- Εφαρμογές για βάσεις δεδομένων
- Δυναμικές Web σελίδες (Java Server Pages)
- Εφαρμογές με γραφικά περιβάλλοντα (Graphical User Interfaces) που δεν έχουν τίποτα να ζηλέψουν από αντίστοιχες φτιαγμένες με άλλες γλώσσες προγραμματισμού.
- Υπηρεσίες κινητής τηλεφωνίας (WAP). Μπορείτε να κάνετε πολύ εύκολα μία εφαρμογή που θα προσπελαύνεται από το Internet μέσω του κινητού σας.
- Εφαρμογές για κάθε είδους μικροσυσκευή.
- Εφαρμογές ασφάλειας (JavaRing, και Smart Cards που τρέχουν Java)
- Δυνατότητα εκτέλεσης και συνεργασίας με απομακρυσμένο κώδικα. Είναι δυνατόν το πρόγραμμά σας να χρησιμοποιεί μία κλάση η οποία είναι αποθηκευμένη σε ένα άλλο απομακρυσμένο μηχάνημα (RMI), ή να χρησιμοποιεί κλάσεις που είναι γραμμένες σε άλλη γλώσσα προγραμματισμού (CORBA).
- Αναγνώριση φωνής
- Διαχείριση XML
- Δισδιάστατα και Τρισδιάστατα γραφικά
- Εφαρμογές Τεχνητής Νοημοσύνης

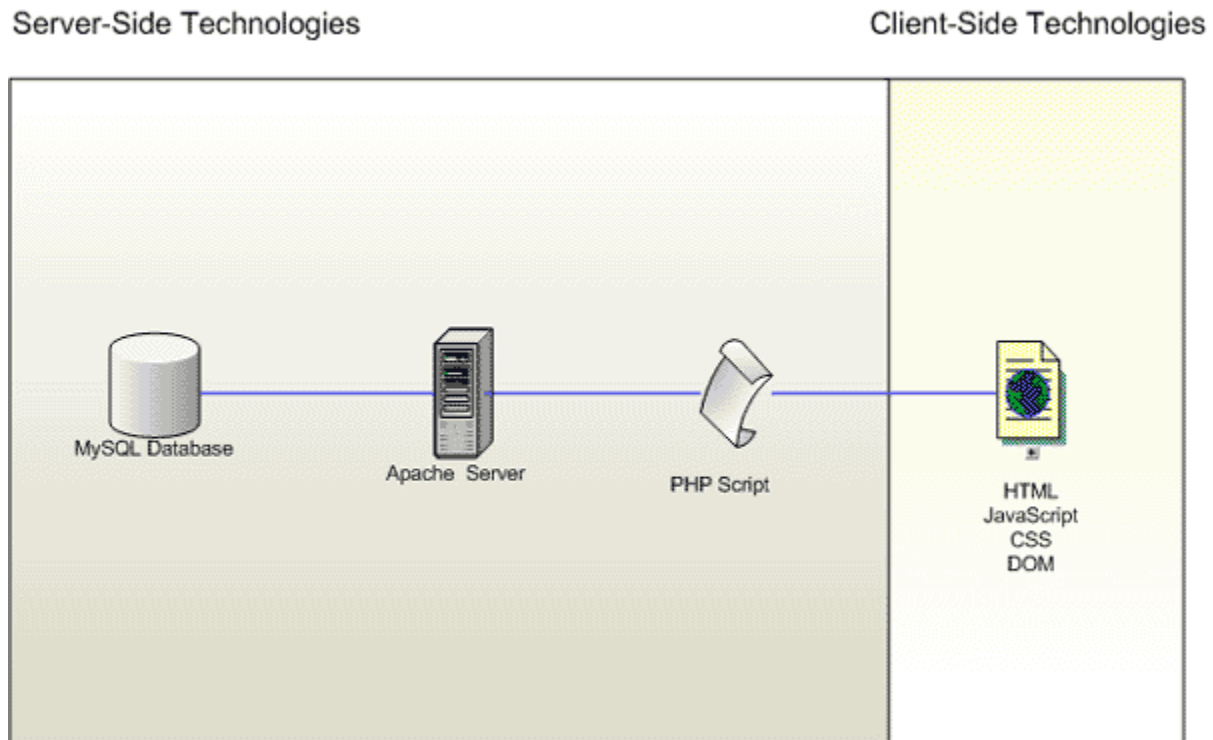
4.4.2 Apache web server

Ο APACHE αναγνωρίζεται γενικά ως ο πιο δημοφιλής Web Server. Αρχικά σχεδιάστηκε για τους κεντρικούς υπολογιστές Unix. Αργότερα χρησιμοποιήθηκε για τα Windows και άλλα λειτουργικά συστήματα δικτύων (NOS = network operating systems).

Ο APACHE Web Server παρέχει μια πλήρη σειρά χαρακτηριστικών γνωρισμάτων των Web Server, συμπεριλαμβανομένου του CGI, SSL, και των εικονικών περιοχών. Υποστηρίζει επίσης plug in ενότητες και

είναι αξιό- πιστο, ελεύθερο και σχετικά εύκολο να διαμορφωθεί. Το APACHE είναι ελεύθερο λογισμικό όπου διανέμεται από το Apache Software Foundation που προωθεί τις διάφορες ελεύθερες και προηγμένες ανοικτές πηγές τεχνολογίας Ιστού (Web technologies).

4.4.3 Αρχή λειτουργίας ενός web server



4.4.3.1 Από την πλευρά του Πελάτη.

Όταν ο χρήστης επιλέξει ένα αντικείμενο ο browser ακολουθεί το hyperlink με σκοπό να βρει και να εμφανίσει τη συγκεκριμένη σελίδα. Ο ενσωματωμένος υπερσύνδεσμος χρειάζεται ένα τρόπο για να ‘κατονομάζει’ οποιαδήποτε άλλη σελίδα στον Ιστό. Η ονομασία αυτή των σελίδων γίνεται μέσω των Ενιαίων Εντοπιστών Πόρων ή URL (Uniform Resource Locators)

Τυπική διεύθυνση η <http://www.abcd.com/products.html>

Με http - το όνομα του πρωτοκόλλου(HyperText Transfer Protocol).

Με www.abcd.com - όνομα της μηχανής στην οποία βρίσκεται η Σελίδα

Με products.html - όνομα του αρχείου που περιέχει Σελίδα

Βήματα για την επιλογή του συνδέσμου:

1. Ο φυλλομετρητής καθορίζει τη διεύθυνση URL
2. Ο φυλλομετρητής ζητά από το DNS τη διεύθυνση IP του www.int.org
3. Το DNS απαντά με τη διεύθυνση 156.106.192.32

4. Ο φυλλομετρητής εγκαθιδρύει μια σύνδεση TCP με τη θύρα 80 στη διεύθυνση 156.106.192.32
5. Ο φυλλομετρητής στέλνει μια αίτηση που ζητά το αρχείο /home/index.html
6. Ο διακομιστής www.int.org στέλνει το αρχείο /home/index.html
7. Η σύνδεση TCP αποσυνδέεται
8. Ο φυλλομετρητής εμφανίζει όλο το κείμενο του αρχείου /home/index.html
9. Ο φυλλομετρητής προσκομίζει και εμφανίζει όλες τις εικόνες σε αυτό το αρχείο

4.4.3.2 Από την πλευρά του Διακομιστή.

Έχοντας τον χρήστη να πληκτρολογεί μια URL διεύθυνση ή ένα hypertext, ο φυλλομετρητής αναλύει, ερμηνεύει ένα DNS όνομα και αναζητά. Έπειτα ο φυλλομετρητής περνώντας την IP διεύθυνση του server δημιουργεί μια σύνδεση TCP με την θύρα 80 σε αυτόν τον διακομιστή. Αργότερα στέλνοντας την υπόλοιπη διεύθυνση με το όνομα του αρχείου ο διακομιστής το βρίσκει και το επιστρέφει ώστε να το εμφανίσει .

Τα βήματα που εκτελεί ο διακομιστής στον κύριο βρόγχο είναι :

- Αποδοχή μιας σύνδεσης TCP από ένα πελάτη (φυλλομετρητή)
- Λήψη του ονόματος του ζητούμενου αρχείου
- Προσκόμιση του αρχείου
- Επιστροφή του αρχείου της σύνδεσης στον πελάτη
- Αποδέσμευση της σύνδεσης TCP

4.4.4 PHP

Η PHP είναι μια γλώσσα προγραμματισμού, που έχει σχεδιαστεί για τη δημιουργία δυναμικών ιστοθέσεων. Βρίσκεται μέσα στον διακομιστή web και επεξεργάζεται εντολές που περιέχονται μέσα σε μία ιστοσελίδα, πριν να σταλεί αυτή η ιστοσελίδα στο πρόγραμμα περιήγησης σας. Ορισμένα στοιχεία της σελίδας μπορούν να παραχθούν γρήγορα, έτσι ώστε η σελίδα να αλλάζει κάθε φορά που φορτώνεται. Για παράδειγμα, μπορείτε να χρησιμοποιήσετε PHP για να δείξετε την τρέχουσα ημερομηνία και ώρα επάνω σε κάθε σελίδα στην ιστοθέση σας.

Το όνομα PHP είναι κάτι σαν ακρωνύμιο των όρων “Hypertext Preprocessor”. Άρχισε την ζωή της με το όνομα PHP/FI, όπου το “FI” σήμαινε Forms Interpreter (διερμηνευτής φορμών). Αν και το όνομα της συντομήθηκε λίγο, ένα από τα ισχυρά χαρακτηριστικά της PHP είναι το πόσο εύκολα μπορεί να επεξεργάζεται δεδομένα, που υποβάλλονται σε φόρμες HTML. Η PHP μπορεί επίσης να “συνομιλήσει” με διάφορα συστήματα βάσεων δεδομένων, δίνοντας σας την δυνατότητα να παράγετε μια ιστοσελίδα με βάση ένα ερώτημα SQL. Για παράδειγμα, μπορείτε να εισάγετε μια λέξη-κλειδί αναζήτησης μέσα σε ένα πεδίο φόρμας σε μία ιστοσελίδα, να κάνετε ερώτημα σε μία βάση δεδομένων με αυτή την τιμή και να παράγετε μια σελίδα με αποτελέσματα που ταιριάζουν στο κλειδί αναζήτησης.

4.4.4.1 Πλεονεκτήματα της PHP

- Το βασικό πλεονέκτημα της PHP είναι ότι λειτουργεί δυναμικά
- Τα αποτελέσματα που παράγει, αλλάζουν σύμφωνα με τις ανάγκες του χρήστη
- Ο δυναμικός τρόπος λειτουργίας εφαρμόζεται ακόμα και μέσα στο εσωτερικό της PHP
- Έχει τη δυνατότητα να αλλάζει τον τύπο των μεταβλητών δυναμικά, σύμφωνα με τα δεδομένα που κάθε χρονική στιγμή είναι αποθηκευμένα σε αυτές.
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων.
- Έχει ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες διαδικασίες διαδικτύου.
- Χαμηλό κόστος - Παρέχεται δωρεάν.
- Ευκολία μάθησης και χρήσης. Η σύνταξη της PHP βασίζεται σε άλλες γλώσσες προγραμματισμού, βασικά στη C και στην Perl
- Φορητότητα - Είναι διαθέσιμη για πολλά λειτουργικά συστήματα
- Διαθεσιμότητα του κώδικα προέλευσης

4.4.5 MySQL

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (relational database management system) ή RDBMS. Σε γενικές γραμμές είναι ένα πακέτο λογισμικού (software package) που είναι πολύ καλό στην οργάνωση και τη διαχείριση μεγάλων ποσοτήτων πληροφοριών. Με τη χρήση της MySQL είναι εύκολη η πρόσβαση σ' αυτές τις πληροφορίες χρησιμοποιώντας μια γλώσσα συγγραφής σεναρίων στην πλευρά του διακομιστή (server-side scripting languages), όπως είναι η PHP.

4.4.5.1 Πλεονεκτήματα της MySQL

- Είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων
- Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα
- Μπορούν να υπάρχουν ταυτόχρονα περισσότερες από μια συνδέσεις με τη βάση χωρίς να υπάρχουν πολλαπλά αντίγραφα της, όπως συμβαίνει με άλλα συστήματα βάσεων δεδομένων
- Η απόδοσή της είναι καλύτερη σε μεγαλύτερο όγκο βάσεων δεδομένων
- Είναι πιο κατάλληλη για χρήση στο Internet

- Παρέχει ευκολίες στο backup
- Είναι ιδιαίτερα βελτιωμένη σε ταχύτητα για την ανάκτηση δεδομένων
- Είναι συμβατή και μεταφέρσιμη σε διάφορες πλατφόρμες και για διάφορα εργαλεία ανάπτυξης
- Είναι οικονομική
- Η MySQL είναι λογισμικό ανοιχτού κώδικα.

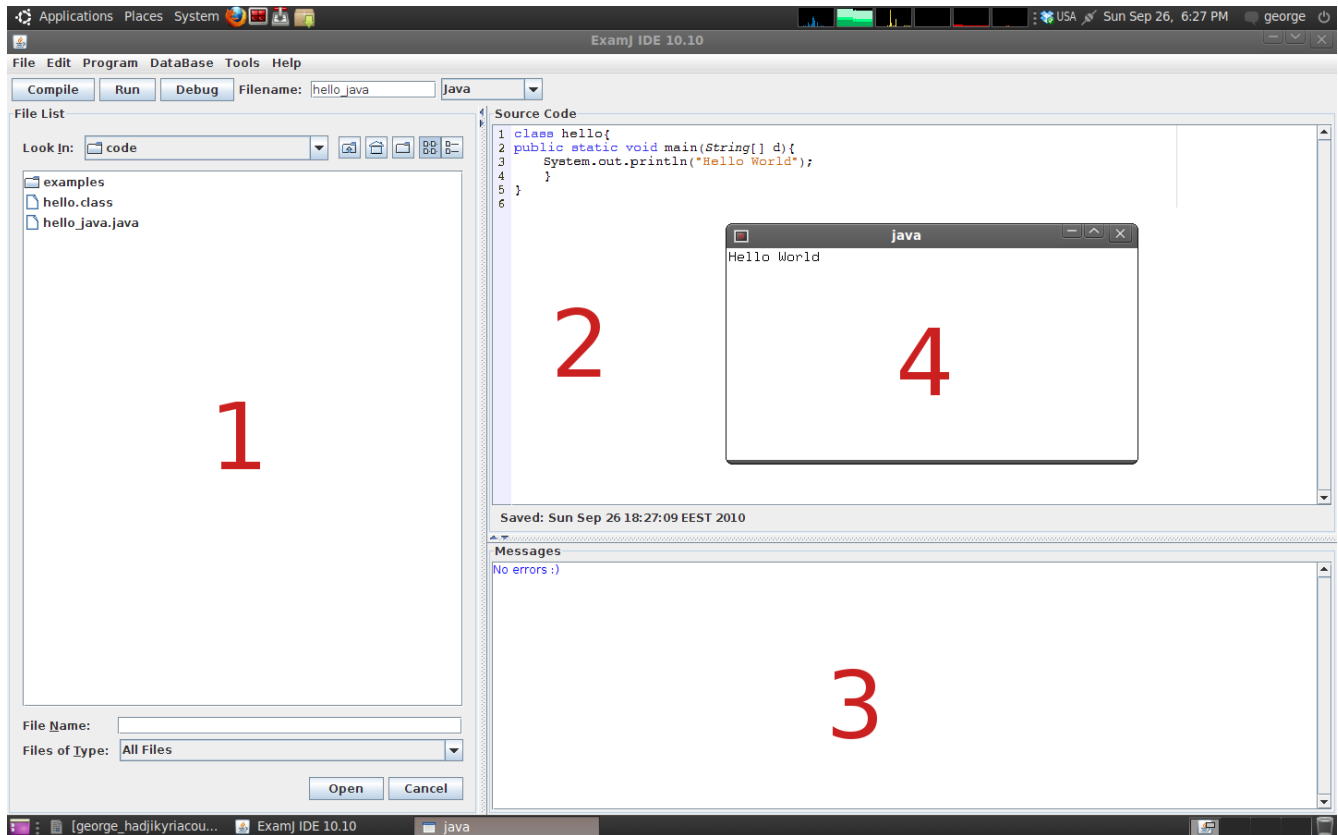
4.5 Υλοποίηση

Για την υλοποίηση της γραφικής διεπαφής χρησιμοποιήθηκε η γλώσσα προγραμματισμού Java και η εργαλειοθήκη Swing, ενώ για την δημιουργία του περιβάλλοντος των εξεταστών χρησιμοποιήθηκε η γλώσσα προγραμματισμού PHP.

Επιπλέον χρησιμοποιήθηκαν ορισμένες εξωτερικές βιβλιοθήκες και εργαλεία ανοικτού κώδικα που βοήθησαν στην υλοποίηση της εφαρμογής. Συγκεκριμένα χρησιμοποιήθηκαν:

Όνομασία	Χρήση	Άδεια χρήσης	Σύνδεσμος
jsyntaxpane	Ένα απλό στη χρήση και την επέκταση JeditorKit που υποστηρίζει αρκετές γλώσσες προγραμματισμού. Στόχος του έργου είναι να δημιουργήσει εμφανισιακά όμορφους swing editors με τονισμό σύνταξης.	Apache License 2.0	http://code.google.com/p/jsyntaxpane/
jasypt	Μια απλή στη χρήση βιβλιοθήκη κρυπτογράφησης.	Apache License 2.0	http://sourceforge.net/projects/jasypt/
JavaMail	API για την ανάπτυξη εφαρμογών που απαιτούν χρήση e-mail.	CDDL-1.0, BSD, GPL-2.0	http://www.oracle.com/technetwork/java/index-jsp-139225.html
Connector/J	Ο επίσημος JDBC driver για την MySQL βάση δεδομένων.	GPL-2.0	http://dev.mysql.com/downloads/connector/j/
GeSHi	Προσφέρει τονισμό σύνταξης σε σελίδες PHP.	GPL-2.0	http://qbnz.com/highlighter
JTidy	Προσφέρει έλεγχο σύνταξης και μορφοποίηση κώδικα HTML.	MIT-like	http://jtidy.sourceforge.net/
Pindent	Προσφέρει μορφοποίηση κώδικα Python.		http://coverage.livinglogic.de/Tools/scripts/pindent.py.html
Bash Script Beautifier	Προσφέρει μορφοποίηση κώδικα Bash.	GPL-2.0	http://www.arachnoid.com/linux/beautify_bash/index.html
Ruby Script Beautifier	Προσφέρει μορφοποίηση κώδικα Ruby.	GPL-2.0	http://www.arachnoid.com/ruby/rubyBeautifier.html
PMD	Ανάλυση κώδικα Java.	BSD	http://pmd.sourceforge.net/

4.5.1 Γραφική διεπαφή χρήστη



Εικόνα 23: Γραφική διεπαφή χρήστη

Όπως αναφέρθηκε πιο πάνω στη εισαγωγή αυτού του κειμένου, ένας από τους κύριους στόχους της εφαρμογής είναι η απλότητα στη χρήση και η αμεσότητα, ώστε να μην είναι δύσκολη η εκμάθηση της από τους χρήστες. Στην **εικόνα 23** βλέπετε τη γραφική διεπαφή του περιβάλλοντος ανάπτυξης, η οποία είναι χωρισμένη σε τέσσερα μέρη:

1. Εμφάνιση/άνοιγμα αρχείων πηγαίου κώδικα.
2. Συντάκτης πηγαίου κώδικα. Όπως αναφέρθηκε πιο πάνω, έγινε χρήση του αντικειμένου jsyntaxpane.
3. Εμφάνιση μηνυμάτων ή σφαλμάτων που προκύπτουν κατά την αποθήκευση ή μεταγλώττιση του κώδικα
4. Τερματικό για το τρέξιμο της εφαρμογής μετά από πετυχημένη μεταγλώττιση του κώδικα. Χρησιμοποιείτε η εφαρμογή xterm για λειτουργικά συστήματα GNU/Linux ή η εφαρμογή cmd για λειτουργικά συστήματα της οικογένειας windows.

Επίσης στο πάνω μέρος της διεπαφής (κάτω από το menu) υπάρχουν τρία κουμπιά που είναι υπεύθυνα για την μεταγλώττιση (Compile), τρέξιμο (Run) και αποσφαλμάτωση (Debug) της εργασίας. Επιπλέον υπάρχει μία περιοχή για την δήλωση του ονόματος του αρχείου πηγαίου κώδικα (Filename) καθώς επίσης και ένας επιλογέας

της γλώσσας προγραμματισμού που θα χρησιμοποιηθεί (π.χ Java).

4.5.2 Εκκίνηση εφαρμογής

Αυτό που χρειάζεται για την εκκίνηση της εφαρμογής είναι το JRE (Java runtime environment), στη συνέχεια τρέχουμε το εκτελέσιμο script 'examj.sh' που βρίσκεται στο φάκελο 'examj-bin'.

4.5.3 Μενού

Όλες οι λειτουργίες της εφαρμογής, βρίσκονται ομαδοποιημένες στο κεντρικό μενού της εφαρμογής. Ποιο κάτω θα γίνει αναφορά σε κάθε μια από αυτές. Συνοπτικά υπάρχουν οι επιλογές:

File	Edit	Program	DataBase	Tools	Help
Open... Save Print Exit	Find	Compile Run Debug Send code	Database setup Database delete Add user Remove user Load/Delete code from database	Code analyser Code formatter Terminal Backup 'code' directory View error.log Document information Settings	About Licence

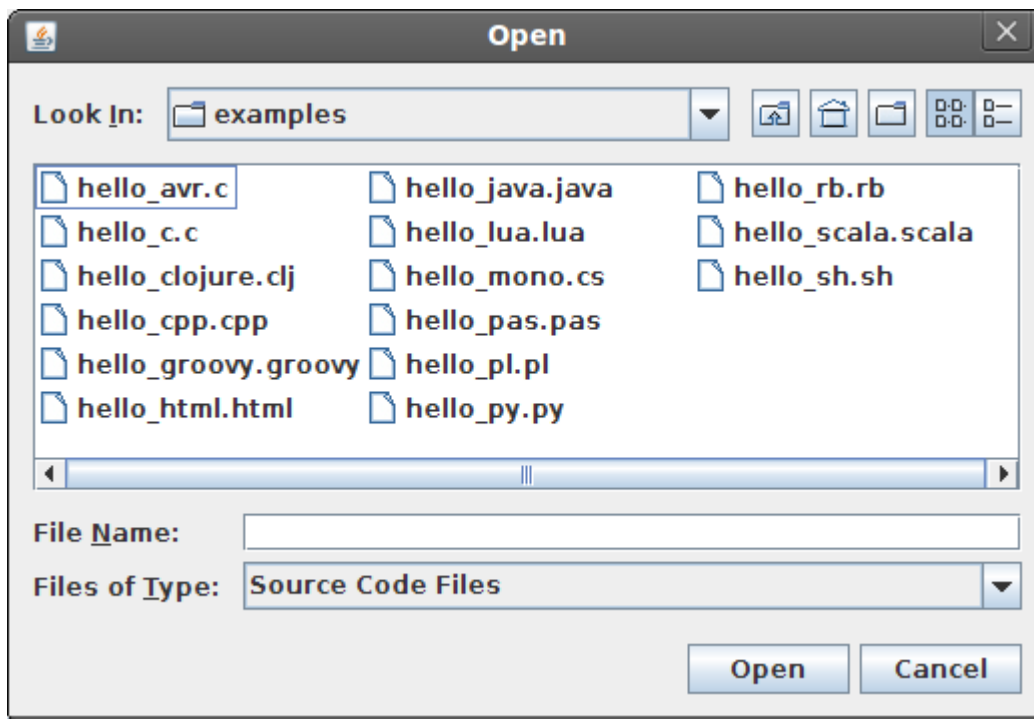
4.5.4 Άνοιγμα αρχείου πηγαίου κώδικα

(File -> Open...)

Το άνοιγμα των αρχείων πηγαίου κώδικα, μπορεί να γίνει με δύο τρόπους, είτε μέσω από την κεντρική διαπαφή χρήστη (βλ. Εικόνα 23) είτε από το μενού File -> Open...

Μετά το άνοιγμα του αρχείου και εμφάνιση του στον συντάκτη πηγαίου κώδικα ενημερώνεται αυτόματα το πεδίο filename καθώς και ο επιλογέας της γλώσσας προγραμματισμού ανάλογα με τον τύπο αρχείου που ανοίξαμε.

Για αυτό τον σκοπό χρησιμοποιήθηκε το αντικείμενο JFileChooser της εργαλειοθήκης Swing.



Εικόνα 24: Άνοιγμα αρχείου πηγαίου κώδικα

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]
void main_open_file(File file_name) {
    try {
        txt_code.setText("");
        FileInputStream fr = new FileInputStream(file_name);
        InputStreamReader isr = new InputStreamReader(fr, "UTF-8");
        BufferedReader reader = new BufferedReader(isr);
        StringBuffer buffer = new StringBuffer();

        String line = null;
        while ((line = reader.readLine()) != null) {
            buffer.append(line);
            buffer.append('\n');
        }
        reader.close();
        txt_code.setText(buffer.toString());
        txt_code.setCaretPosition(0);
    } catch (Exception exe) {
        exception_inp(exe);
    }
}
[...]
```

4.5.5 Αποθήκευση αρχείου πηγαίου κώδικα

(File -> Save)

Για την αποθήκευση του πηγαίου κώδικα χρησιμοποιήθηκαν τα αντικείμενα **BufferedWriter** και **FileWriter**. Ο κώδικας που ακολουθεί, βρίσκεται στη μέθοδο **save(int chk_exists)** και καλείται κάθε φορά που αποθηκεύεται ή μεταγλωττίζεται κάποια εργασία.

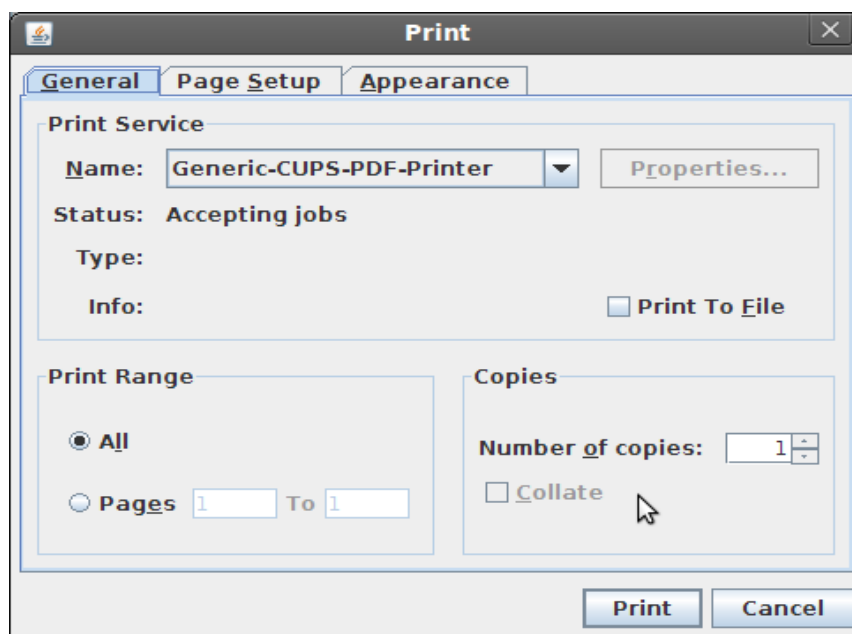
Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]  
Writer output = new BufferedWriter(new FileWriter(filename));  
output.write(txt_code.getText());  
output.close();  
[...]
```

4.5.6 Εκτύπωση πηγαίου κώδικα

(File -> Print)

Με την έκδοση 1.6 της Java ήρθαν ριζικές αλλαγές όσο αφορά την εκτύπωση από αντικείμενα εισαγωγής κειμένου, όπως για παράδειγμα τα αντικείμενα της Swing JTextArea και JeditorPane. Για να στείλουμε στον εκτυπωτή τα δεδομένα ενός JeditorPane, καλούμε την μέθοδο print() του αντικειμένου:



Εικόνα 25: Εκτύπωση πηγαίου κώδικα

Στην **εικόνα 25** βλέπουμε το διαλογικό παράθυρο που εμφανίζεται, που δίνει τη δυνατότητα να επιλέξουμε τον εκτυπωτή και τις ρυθμίσεις που θέλουμε για την σελίδα.

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]  
item_print.addActionListener(new ActionListener() {
```

```

    public void actionPerformed(ActionEvent ev) {
        try {
            txt_code.print();
        } catch (Exception exe) {
            exception_inp(exe);
        }
    }
});
[...]
```

4.5.7 Έξοδος από την εφαρμογή

(File -> Exit)

Κατά την έξοδο από την εφαρμογή καλείται η μέθοδος `on_exit()` δίνοντας τη δυνατότητα να αποθηκευτεί ή εργασία, κάνοντας κλήση της μεθόδου `save(int chk_exists)`.

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```

[...]
```

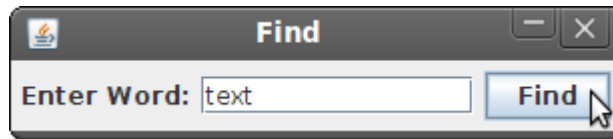
```

void on_exit() {
    autosave.stop();
    if (txt_file.getDocument().getLength() > 0 ||
txt_code.getDocument().getLength() > 0) {
        int more = JOptionPane.showConfirmDialog(null,
            "Save Before Exit ?", "Save",
JOptionPane.YES_NO_OPTION);
        if (more == JOptionPane.YES_OPTION) {
            if (txt_file.getDocument().getLength() == 0) {
                JOptionPane.showMessageDialog(win, "Set Filename
Please...");
            } else {
                save(1);
                System.exit(0);
            }
        }
        if (more == JOptionPane.NO_OPTION) {
            System.exit(0);
        }
    } else {
        System.exit(0);
    }
}
[...]
```

4.5.8 Εύρεση κειμένου

(Edit -> Find)

Είναι ένα απλό εργαλείο το οποίο ψάχνει στην εργασία για λέξεις ή προτάσεις. Τη λέξη κλειδί την δίνει ο χρήστης σε ένα πεδίο JTextField.



Εικόνα 26: Εύρεση κειμένου

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]
public void actionPerformed(ActionEvent ev) {
    txt_code.requestFocusInWindow();
    String find = txt_find.getText();
    if (find != null) {
        int index = txt_code.getText().indexOf(find, (int)
txt_code.getCaretPosition());
        txt_code.select(index, index + find.length());
        if (txt_code.getText().lastIndexOf(index) == index) {
            txt_code.setCaretPosition(0);
        }
    }
}
[...]
```

4.5.9 Μεταγλώττιση εργασίας

(Program -> Compile)

Κατά την διαδικασία μεταγλώττισης της εργασίας εκτελούνται τα παρακάτω (με τη σειρά):

1. Γίνεται η αποθήκευση του πηγαίου κώδικα καλώντας τη μέθοδο **save()**.
2. Ανάλογα με την γλώσσα προγραμματισμού που επιλέξαμε εκτελείται (**Runtime.getRuntime().exec()**) η διεργασία συστήματος (κλάση **Process**) που είναι υπεύθυνη για την μεταγλώττιση του κώδικα στη συγκεκριμένη γλώσσα προγραμματισμού (π.χ κλήση του μεταγλωττιστή javac), με όλα τα ορίσματα που είναι απαραίτητα.
3. Με την κλήση της μεθόδου **compiler_error_h()** απομονώνονται τα μηνύματα σφάλματος από την εκτέλεση της διεργασίας. Αυτό το πετυχαίνεται κάνοντας χρήση της μεθόδου **getErrorStream()** της κλάσης **Process** που καλέσαμε στο βήμα 2.

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]
else if ("Java".equals(cmp_compiler.getSelectedItem())) {
    save(1);
    String filename = txt_file.getText() + ".java";
    Process java = Runtime.getRuntime().exec( comp_java + " -g " +
code_dir + filename);
}
```



```

        compiler_error_h(java);
    }
    [...]
    void compiler_error_h(Process inp_proc) {
        int count = 0;
        String line = null;
        txt_msg.setForeground(Color.red);
        try {
            InputStream stderr = inp_proc.getErrorStream();
            InputStreamReader isr = new InputStreamReader(stderr);
            BufferedReader br = new BufferedReader(isr);
            while ((line = br.readLine()) != null) {
                txt_msg.append(line + "\n");
                count++;
            }
            if (count == 0) {
                txt_msg.setForeground(Color.blue);
                txt_msg.setText("No errors :");
            }
        } catch (Exception exe) {
            exception_inp(exe);
        }
    }
    [...]

```

4.5.10 Εκτέλεση εργασίας

(Program -> Run)

Ανάλογα με την γλώσσα προγραμματισμού που επιλέξαμε εκτελείται (**Runtime.getRuntime().exec()**) η διεργασία συστήματος (κλάση **Process**) που είναι υπεύθυνη για το τρέξιμο του εκτελέσιμου αρχείου ή της κλάσης στη συγκεκριμένη γλώσσα προγραμματισμού (π.χ κλήση του runtime java), με όλα τα ορίσματα που είναι απαραίτητα.

Όλα τα εκτελέσιμα αρχεία εκτελούνται σε τερματικό (xterm για Unix-like συστήματα ή cmd για λειτουργικά συστήματα της οικογένειας windows).

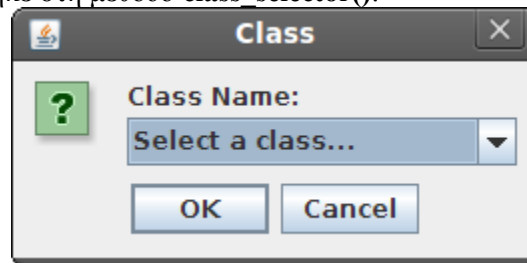


Εικόνα 27: Εκτέλεση εφαρμογής

Απόσπασμα κώδικα από το αρχείο "examj_gui.java":

```
[...]
else if ("Java".equals(cmp_compiler.getSelectedItem())) {
    class_name = class_selector();
    File class_exists = new File(code_dir + class_name + ".class");
    if (class_exists.exists() && class_name.length() > 0) {
        arguments = (String) JOptionPane.showInputDialog( win,
"Value(s):", "Arguments", JOptionPane.INFORMATION_MESSAGE);
        Runtime.getRuntime().exec( "xterm -font -*-fixed-medium-r-
*_*_"
+ font_size_x
+ " -*-*_*_*_*_*_*-iso8859-* -geometry 80x25 -hold -e "+jre+"
-classpath "
+ code_dir + " "
+ class_name + " "
+ arguments);
    } else {
        JOptionPane.showMessageDialog(win, "Class Not Found",
"ERROR", JOptionPane.ERROR_MESSAGE);
    }
[...]
```

Για μεταγλωττιστές που έχουν σαν έξοδο bytecode (π.χ Java ή Scala) πριν την εκτέλεση του εμφανίζετε ένας επιλογέας κλάσης που υλοποιήθηκε στη μέθοδο `class_selector()`.

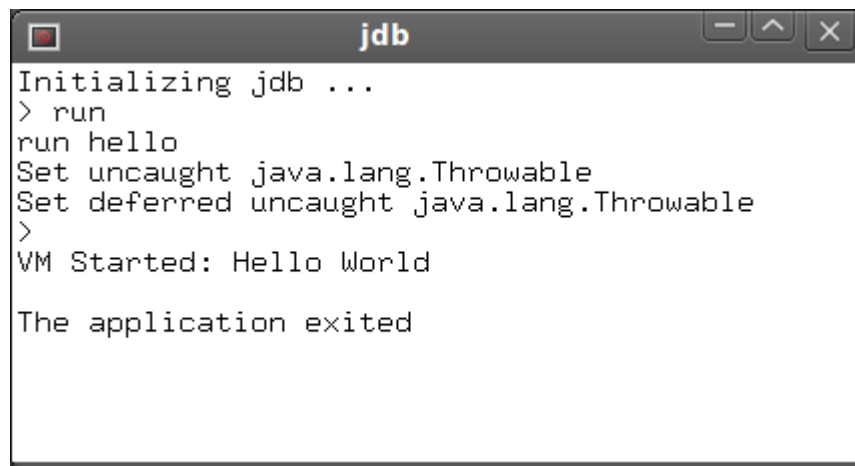


Εικόνα 28: Επιλογέας κλάσης

4.5.11 Αποσφαλμάτωση

(Program -> Debug)

Με τον ίδιο τρόπο, όπως και στη παράγραφο 5.2.7, εκτελείται ο κατάλληλος debugger για την αποσφαλμάτωση της εργασίας.



```
Initializing jdb ...
> run
run hello
Set uncaught java.lang.Throwable
Set deferred uncaught java.lang.Throwable
>
VM Started: Hello World

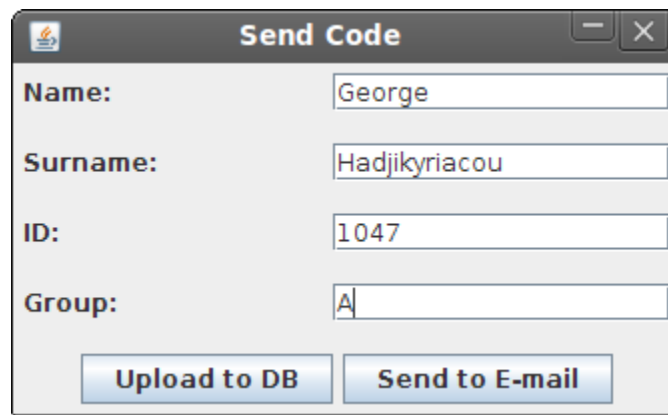
The application exited
```

Εικόνα 29: Αποσφαλμάτωση

4.5.12 Αποστολή κώδικα

(Program -> Send code)

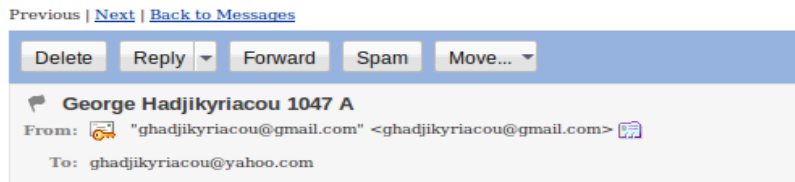
Όπως αναφέρθηκε στην Εισαγωγή αυτού του κειμένου, ο εξεταζόμενος έχει δυο τρόπους να στείλει την εργασία του για αξιολόγηση. Ο ένας είναι να στείλει την εργασία του σε μία βάση δεδομένων MySQL, ενώ ο άλλος είναι να στείλει την εργασία του κατευθείαν στο ηλεκτρονικό ταχυδρομείο του εξεταστή (πρωτόκολλο SMTP).



Name:	<input type="text" value="George"/>
Surname:	<input type="text" value="Hadjikyriacou"/>
ID:	<input type="text" value="1047"/>
Group:	<input type="text" value="A"/>

Εικόνα 30: Αποστολή κώδικα

Για την αποστολή του κώδικα στη MySQL βάση δεδομένων είναι υπεύθυνος ο driver Connector/J, ενώ για την αποστολή του κώδικα σε λογαριασμό ηλεκτρονικού ταχυδρομείου υπεύθυνο είναι το API JavaMail που αναφέρθηκε στη παράγραφο 4.5.



Εικόνα 31: Μήνυμα ηλεκτρονικού ταχυδρομείου που περιλαμβάνει μια εργασία σε Java

Απόσπασμα κώδικα από το αρχείο “code_send.java”:

```
[...]
Class.forName("org.gjt.mm.mysql.Driver");
String url = "jdbc:mysql://" + host + "/" + dbase;
connection = DriverManager.getConnection(url, username, password);
myStatement = connection.createStatement();
[...]
String myQuery = "INSERT IGNORE INTO code_uploaded
(`date`,`time`,`name`,`surname`,`id`,`group`,`prj_name`,`lang`,`code`,`exam
iner`,`uid`) values('"
+ date + "','"
+ time + "','"
+ name + "','"
+ surname + "','"
+ id + "','"
+ group + "','"
+ prj_name + "','"
+ lang + "','"
+ str_code + "','"
+ examiner + "','"
+ examj_gui.uuid_code
+ '); ";
int check_insert = myStatement.executeUpdate(myQuery);
[...]
```

Απόσπασμα κώδικα από το αρχείο “send_mail.java”:

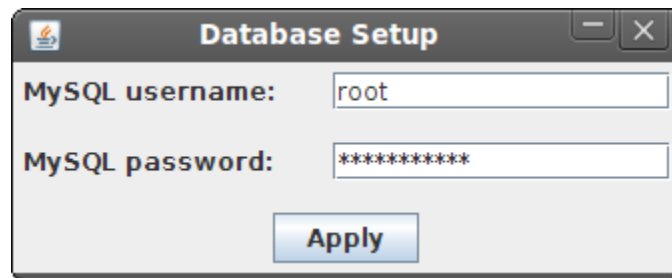
```
[...]
Authenticator auth = new SMTPAuthenticator();
Session session = Session.getInstance(props, auth);
MimeMessage msg = new MimeMessage(session);
```

```
msg.setText(m_text);
msg.setSubject(m_subject);
msg.setFrom(new InternetAddress(d_email));
msg.addRecipient(Message.RecipientType.TO, new InternetAddress(m_to));
Transport.send(msg);
JOptionPane.showMessageDialog(win_mail, "Email have been successfully
sent!");
[...]
```

4.5.13 Εγκατάσταση βάσης δεδομένων

(DataBase -> Database setup)

Με αυτή τη λειτουργία μπορούμε να εγκαταστήσουμε (εκτελώντας τις κατάλληλες εντολές SQL) τη βάση δεδομένων και τους πίνακες που είναι υπεύθυνοι για το σύστημα αξιολόγησης των χρηστών. Απαιτείτε ο κωδικός του διαχειριστή της βάσης δεδομένων.

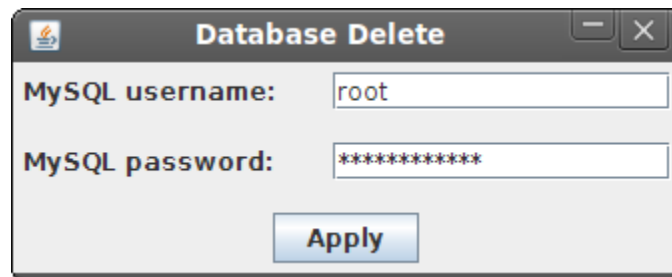


Εικόνα 32: Εγκατάσταση βάσης δεδομένων

4.5.14 Διαγραφή βάσης δεδομένων

(DataBase -> Database delete)

Με αυτή τη λειτουργία μπορούμε να διαγράψουμε (εκτελώντας τις κατάλληλες εντολές SQL) τη βάση δεδομένων και τους πίνακες που είναι υπεύθυνοι για το σύστημα αξιολόγησης των χρηστών. Απαιτείτε ο κωδικός του διαχειριστή της βάσης δεδομένων.



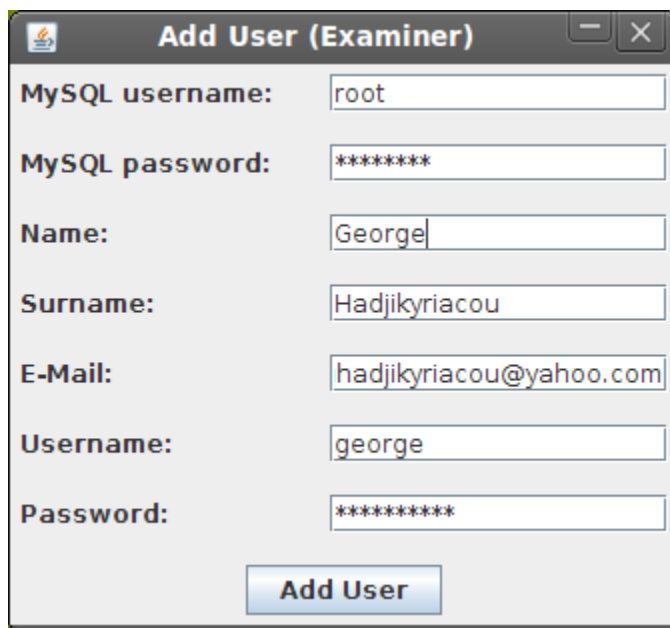
Εικόνα 33: Διαγραφή βάσης δεδομένων

4.5.15 Προσθήκη χρηστών

(DataBase -> Add user)

Με αυτή τη λειτουργία προσθέτουμε χρήστες (εξεταστές) στο σύστημα οι οποίοι έχουν το δικαίωμα να αξιολογήσουν και να βαθμολογήσουν τις εργασίες των φοιτητών. Υπεύθυνος για την προσθήκη και τη διαγραφή χρηστών (βλ. 5.2.13) στο σύστημα είναι ο διαχειριστής της βάσης δεδομένων.

Όλοι οι χρήστες εισάγονται (εκτελώντας τις κατάλληλες εντολές SQL) στον πίνακα 'users'.



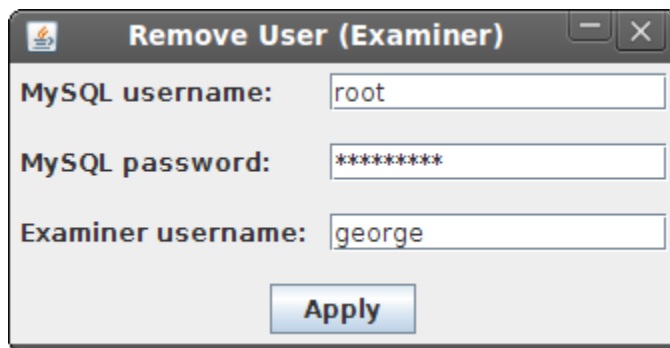
The screenshot shows a dialog box titled "Add User (Examiner)". It has a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The dialog contains several input fields: "MySQL username:" with the value "root", "MySQL password:" with masked characters "*****", "Name:" with "George", "Surname:" with "Hadjikyriacou", "E-Mail:" with "hadjikyriacou@yahoo.com", "Username:" with "george", and "Password:" with masked characters "*****". At the bottom center, there is a blue button labeled "Add User".

Εικόνα 34: Προσθήκη χρηστών

4.5.16 Διαγραφή χρηστών

(DataBase -> Remove user)

Με αυτή τη λειτουργία διαγράφουμε χρήστες (εξεταστές) στο σύστημα.



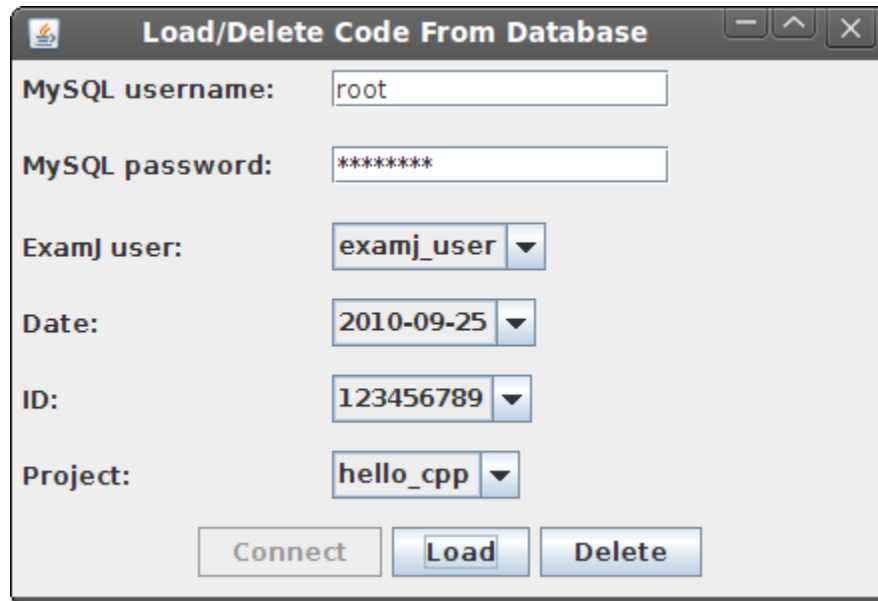
The screenshot shows a dialog box titled "Remove User (Examiner)". It has a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The dialog contains three input fields: "MySQL username:" with the value "root", "MySQL password:" with masked characters "*****", and "Examiner username:" with "george". At the bottom center, there is a blue button labeled "Apply".

Εικόνα 35: Διαγραφή χρηστών

4.5.17 Διαχείριση κώδικα από τη βάση δεδομένων

(DataBase -> Load/Delete code from database)

Πρόκειται για ένα εργαλείο που σαν σκοπό έχει να φορτώνει και να διαχειρίζεται εργασίες από την βάση δεδομένων. Ο διαχειριστής της βάσης δεδομένων μπορεί να φορτώσει ή να διαγράψει εργασίες. Επίσης μπορεί να δημιουργηθεί ένας ξεχωριστός λογαριασμός χρήστη MySQL (με δικαιώματα SELECT ή και DELETE) για τους εξεταστές ώστε να μπορούν να κάνουν αυτή τη δουλειά.



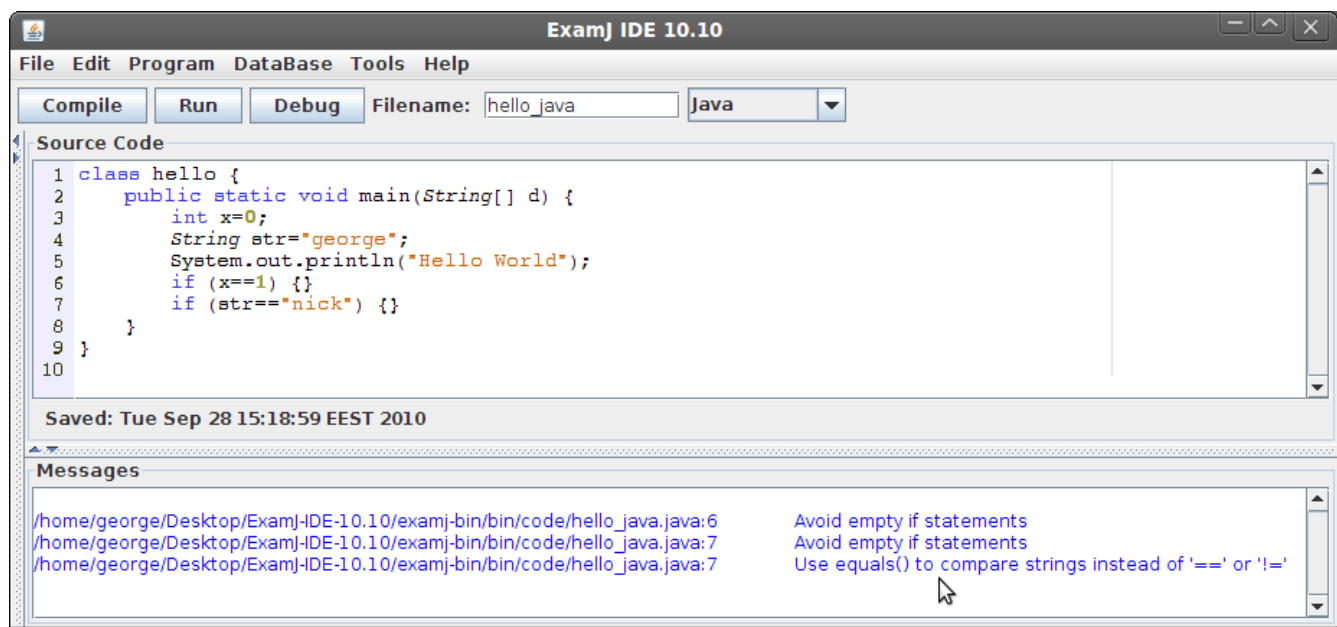
Εικόνα 36: Διαχείριση κώδικα από τη βάση δεδομένων

4.5.18 Ανάλυση κώδικα

(Tools -> Code analyzer)

Οι αναλυτές κώδικα είναι πολύ χρήσιμα και ταυτόχρονα εκπαιδευτικά εργαλεία που "σκανάρουν" τον κώδικά σας για λάθη όπως:

- Κενές δηλώσεις try/catch/finally/switch
- "Νεκρό" κώδικα, αχρησιμοποίητες μεταβλητές, παραμέτρους, imports και private μεθόδους
- Σπατάλη μεταβλητών String/StringBuffer
- Αχρησιμοποίητες δηλώσεις 'if' ή βρόγχοι 'for' που θα μπορούσαν να ήταν βρόγχοι 'while'.
- Επαναλαμβανόμενο κώδικα.
- κ.α



Εικόνα 37: Ανάλυση κώδικα Java με το εργαλείο PMD

4.5.19 Μορφοποίηση κώδικα

(Tools -> Code formatter)

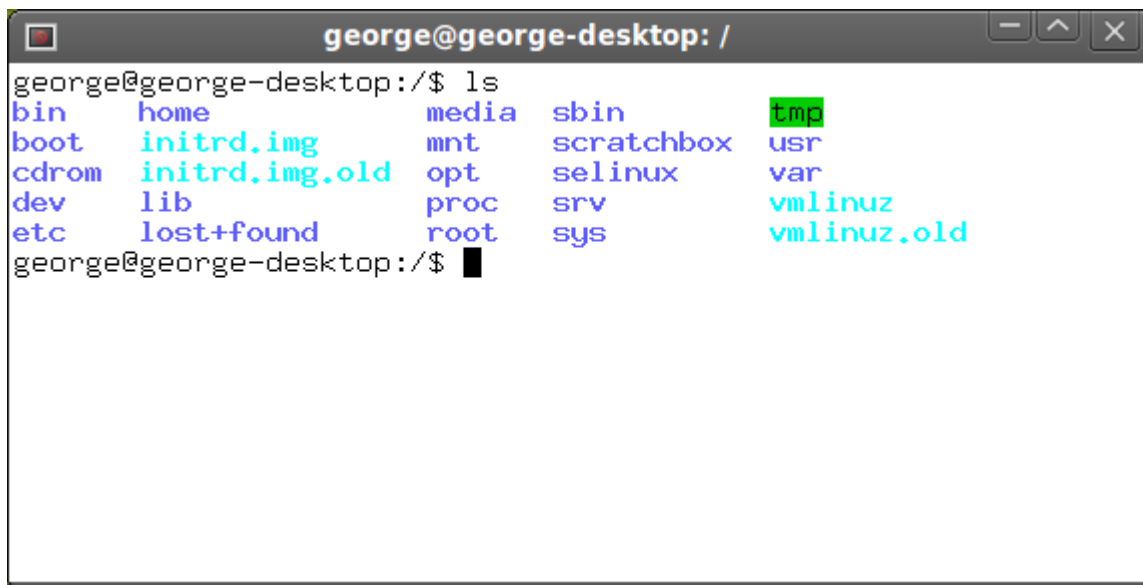
Οι Μορφοποιητές κώδικα είναι εργαλεία τα οποία “τακτοποιούν” και “ομορφαίνουν” τον κώδικα ώστε να είναι ευανάγνωστος και κατανοητός. Υπάρχουν διάφορα στυλ μορφοποίησης που κανονίζουν τη στοιχηση των εντολών και των αγκυλών.

Αρκετά έργα ανοικτού κώδικα δημιούργησαν και κανόνες στοίχισης. Οι πιο τυπικοί κανόνες είναι το στυλ GNU και το στυλ BSD.

4.5.20 Άνοιγμα τερματικού

(Tools -> Terminal)

Το τερματικό αποτελεί μια διεπαφή προς τους χρήστες του υπολογιστή (όπως και το περιβάλλον επιφάνειας εργασίας), η οποία ωστόσο δεν περιλαμβάνει γραφικά. Ο χρήστης αλληλεπιδρά με το σύστημα πληκτρολογώντας εντολές των οποίων τη σύνταξη πρέπει να γνωρίζει. Εκτός από το όνομα “Τερματικό” (terminal) το συναντάμε και με άλλες ονομασίες όπως: γραμμή εντολών (Command line interface) ή κέλυφος (shell) ή κονσόλα (console).

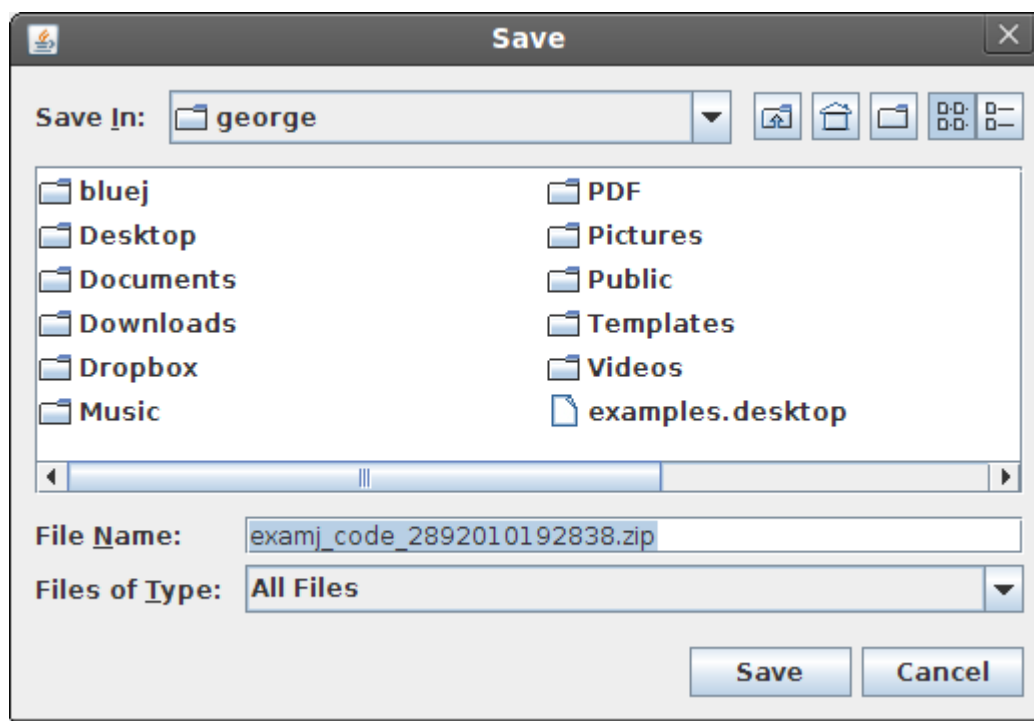


Εικόνα 38: Η εφαρμογή xterm

4.5.21 Δημιουργία αντίγραφου ασφαλείας

(Tools -> Backup 'code' directory)

Με αυτή την επιλογή δημιουργείται ένα συμπιεσμένο αρχείο (.zip) το οποίο περιλαμβάνει τα αρχεία κώδικα από το φάκελο του χρήστη. Το όνομα του συμπιεσμένου αρχείου μπορούμε να το καθορίσουμε εμείς εκτός και αν θέλουμε να κρατήσουμε το προκαθορισμένο (timestamp).

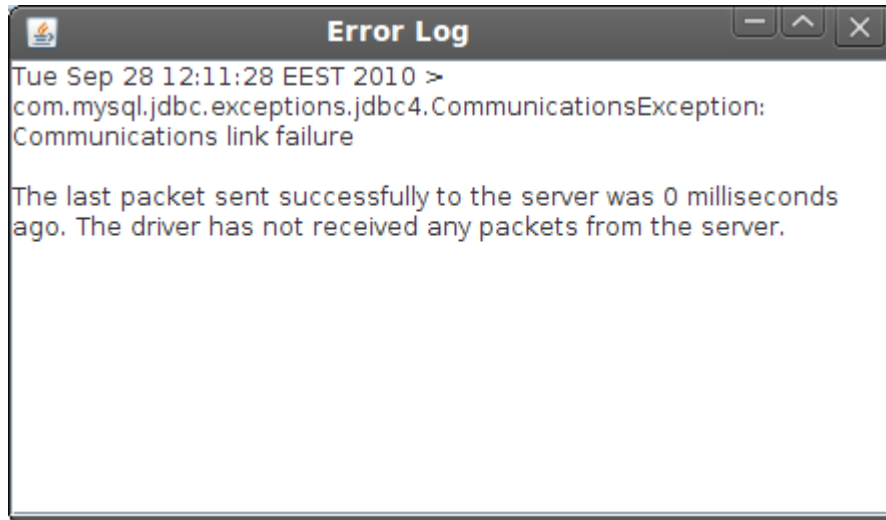


Εικόνα 39: Δημιουργία αντίγραφου ασφαλείας

4.5.22 Εμφάνιση μηνυμάτων σφαλμάτων

(Tools -> View error.log)

Επιλέγοντας από το μενού το στοιχείο “View error.log” εμφανίζονται τα μηνύματα σφάλματος της εφαρμογής. Αυτά τα μηνύματα αποθηκεύονται σε ένα αρχείο με το όνομα “error.log” στο φάκελο bin της εφαρμογής



Εικόνα 40: Εμφάνιση μηνυμάτων σφαλμάτων

4.5.23 Πληροφορίες κώδικα

(Tools -> Document information)

Αυτό το εργαλείο εμφανίζει πληροφορίες, σχετικά με τον κώδικα όπως: αριθμός γραμμών, αριθμός χαρακτήρων και αριθμό λέξεων.



Εικόνα 41: Πληροφορίες κώδικα

Απόσπασμα κώδικα από το αρχείο “examj_gui.java”:

```
[...]  
for (int i = 0; i < chars; i++) {  
    if (txt_code.getText().charAt(i) == '\n') {  
        position++;  
    }  
}
```

```

}

boolean prevWhiteSpace = true;
for (int j = 0; j < chars; j++) {
    char c = txt_code.getText().charAt(j);
    boolean currWhiteSpace = Character.isWhitespace(c);
    if (prevWhiteSpace && !currWhiteSpace) {
        words++;
    }
    prevWhiteSpace = currWhiteSpace;
}

int cur_line = (int) (position + 1);
lb_dinfo_1.setText("Lines: " + cur_line);
lb_dinfo_2.setText("Characters: " + chars);
lb_dinfo_3.setText("Words: " + words);
[...]

```

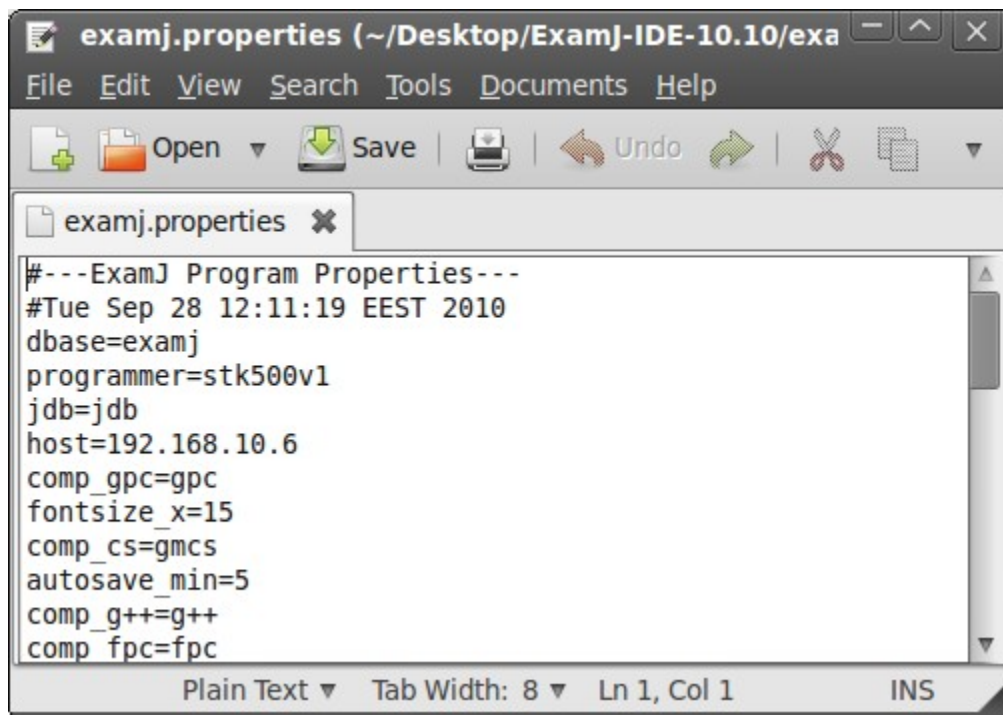
4.5.24 Ρυθμίσεις

(Tools -> Settings)

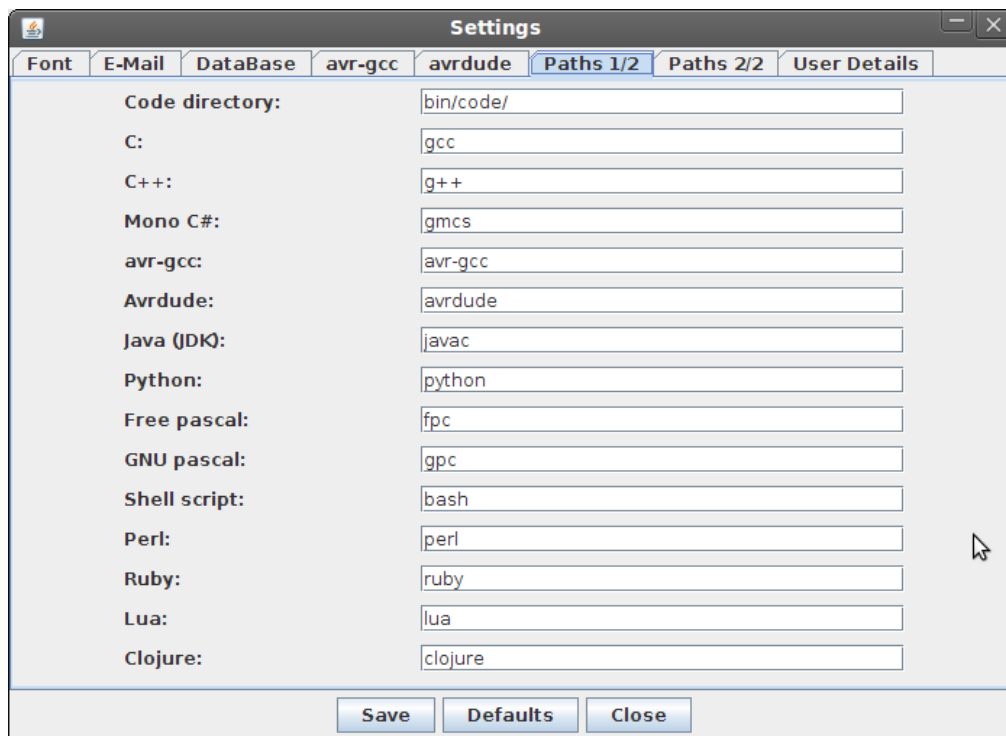
Οι ρυθμίσεις της εφαρμογής μπορούν να τροποποιηθούν είτε μέσω του αρχείου `examj.properties` είτε μέσω γραφικής διεπαφής μέσα από την εφαρμογή.

Υπάρχουν ρυθμίσεις για:

- Γραμματοσειρές του editor και του xterm.
- SMTP (για αποστολή κώδικα μέσω e-mail).
- MySQL.
- Avr-gcc.
- avrdude (για προγραμματισμό μικροελεγκτή AVR).
- Φακέλου κώδικα και εξωτερικών εφαρμογών (compilers, runtimes, debuggers, tools).
- Στοιχεία εξεταστή/εξεταζόμενου.



Εικόνα 42: Αρχείο ρυθμίσεων examj.properties



Εικόνα 43: Ρυθμίσεις

4.5.25 Η σελίδα των εξεταστών

Όπως αναφέρθηκε πιο πάνω η εφαρμογή εκτός από περιβάλλον ανάπτυξης μπορεί να χρησιμοποιηθεί και σαν εξεταστικό εργαλείο. Για αυτό τον σκοπό κατασκευαστική μια δυναμική σελίδα σε PHP όπου ο κάθε εξεταστής θα μπορεί να εισέρχεται και να αξιολογεί τις εργασίες που εστάλησαν από τους φοιτητές,

4.5.26 Εγκατάσταση

Για την εγκατάσταση της σελίδας, αντιγράφουμε τον φάκελο με τη σελίδα (examj-src-php) στο DocumentRoot φάκελο του Apache web server (/var/www) και αλλάζουμε τα δικαιώματα ώστε να μπορεί να εμφανίζεται σε τρίτους (δικαιώματα read).

Στη συνέχεια τροποποιούμε τις μεταβλητές \$localhost, \$username, \$password, \$dbase στο αρχείο 'connectdb.php' ώστε να έχουν τις ανάλογες πληροφορίες της MySQL βάσης δεδομένων του συστήματός μας.

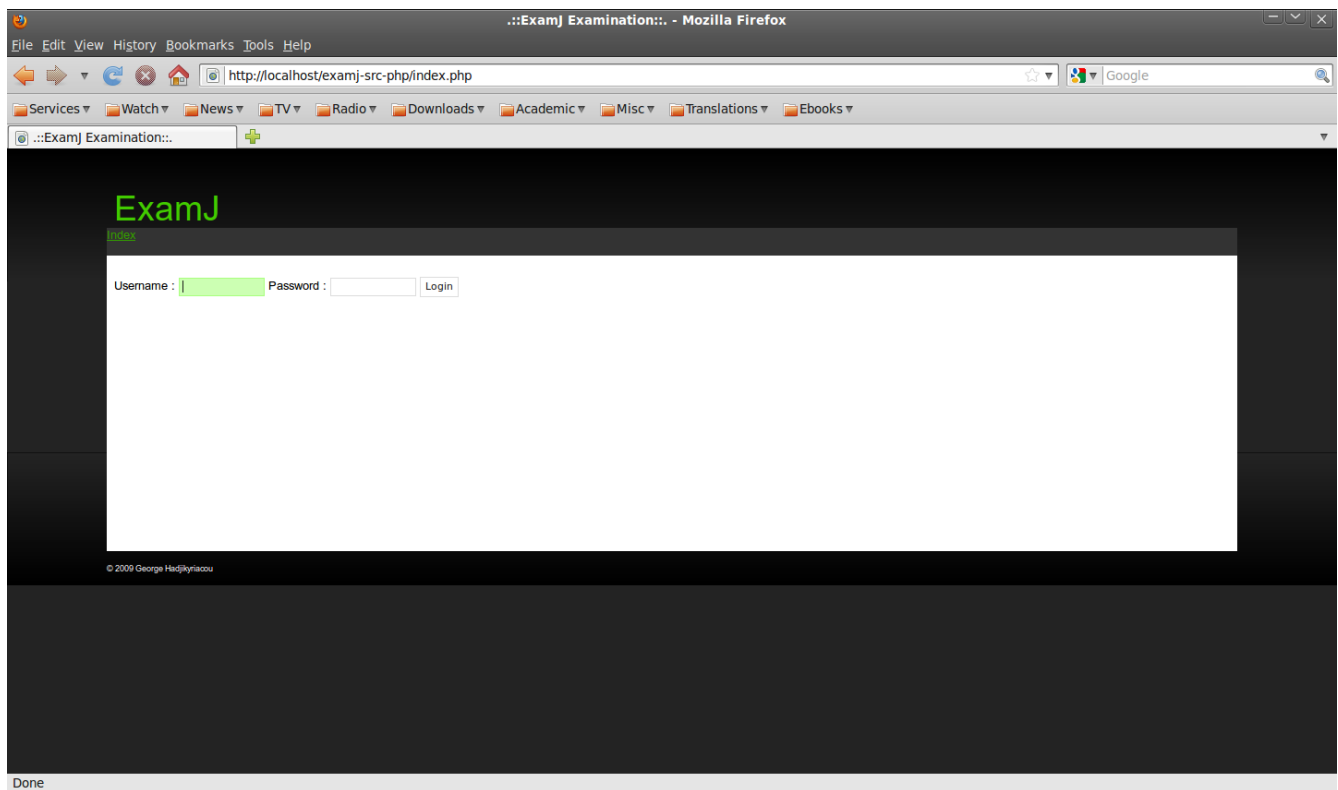
Απόσπασμα κώδικα από το αρχείο “connectdb.php”:

```
<?php
//examj-PHP By George Hadjikyriacou 2008
////////////////////////////////////
$localhost = "127.0.0.1";
$username = "root";
$password = "";
$dbase = "examj";
////////////////////////////////////
$conn = mysql_connect($localhost,$username,$password);

if(!$conn){
    die("Connection to mysql server failed: " . mysql_error($conn));
}
@mysql_select_db($dbase , $conn);
?>
```

4.5.27 Είσοδος χρήστη

Για την είσοδο του χρήστη στο σύστημα κάνουμε έλεγχο αν το όνομα και ο κωδικός του χρήστη υπάρχουν στη βάση δεδομένων και είναι έγκυρα.



Εικόνα 44: Είσοδος χρήστη

Τα δεδομένα του χρήστη λαμβάνονται μέσω μιας HTML φόρμας που βρίσκεται στο αρχείο index.php με την μέθοδο POST[]. Για την αποθήκευση και την ανάκτηση του κωδικού του χρήστη, χρησιμοποιήθηκε ο αλγόριθμος MD5.

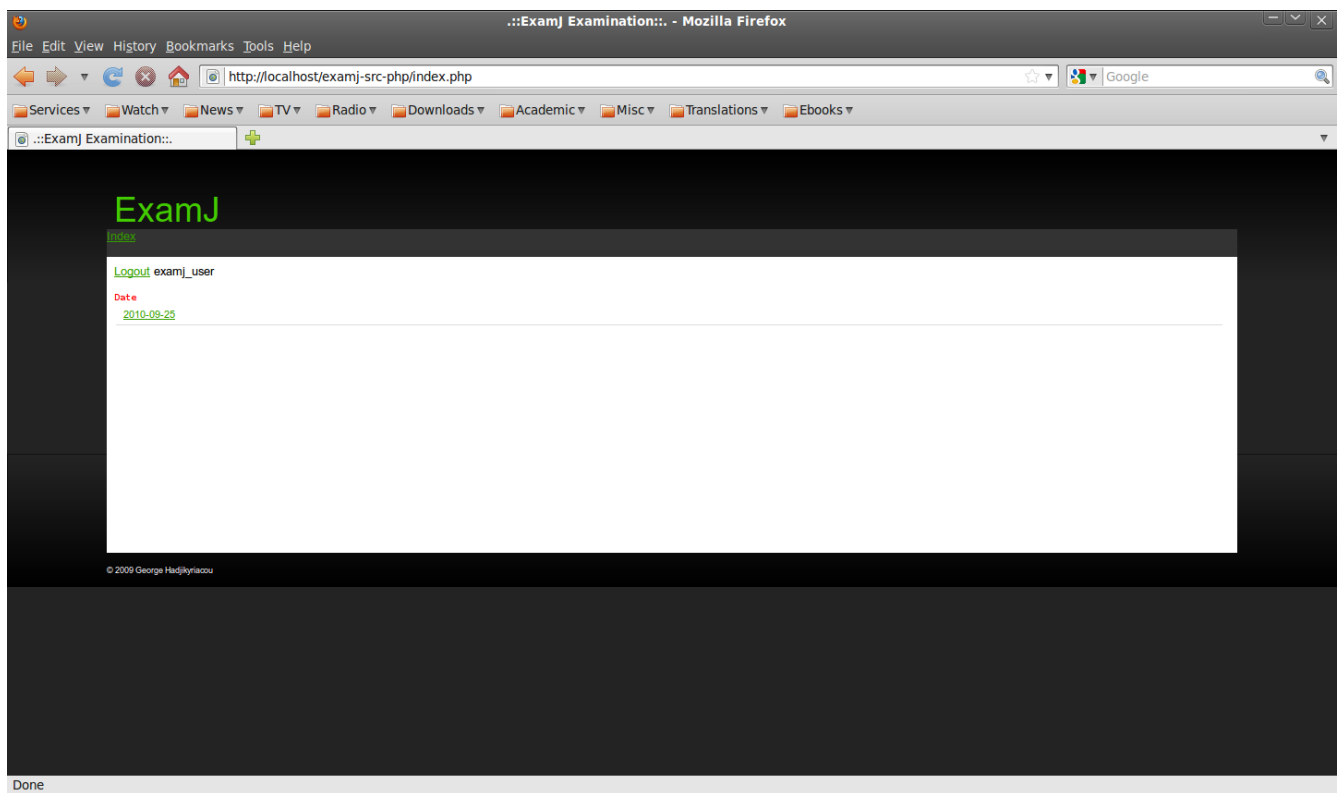
Απόσπασμα κώδικα από το αρχείο “login.php”:

```
[...]  
<?php  
    $date=$_GET[date];  
    $prj_name=$_GET[prj_name];  
    $username = $_POST[username];  
    $password = md5($_POST[password]);  
    $id=$_GET[id];  
    echo "<a href='index.php'>Index</a>";  
?>  
[...]  
<?php  
$sql = "SELECT * FROM users WHERE Username=' $username' AND  
Password=' $password'";  
$results = mysql_query($sql);  
$rows = mysql_num_rows($results);  
if ($rows==1){  
    $_SESSION[user]=$username;  
    $_SESSION[pass]=$password;  
}
```

```
    echo "<br><font color= 'green'>Welcome $_SESSION[user]</font><br>";
}
else{
    echo "<br><font color= 'red'>Login Failed</font><br>";
}
mysql_close($conn);
?>
```

4.5.28 Επιλογή ημερομηνίας

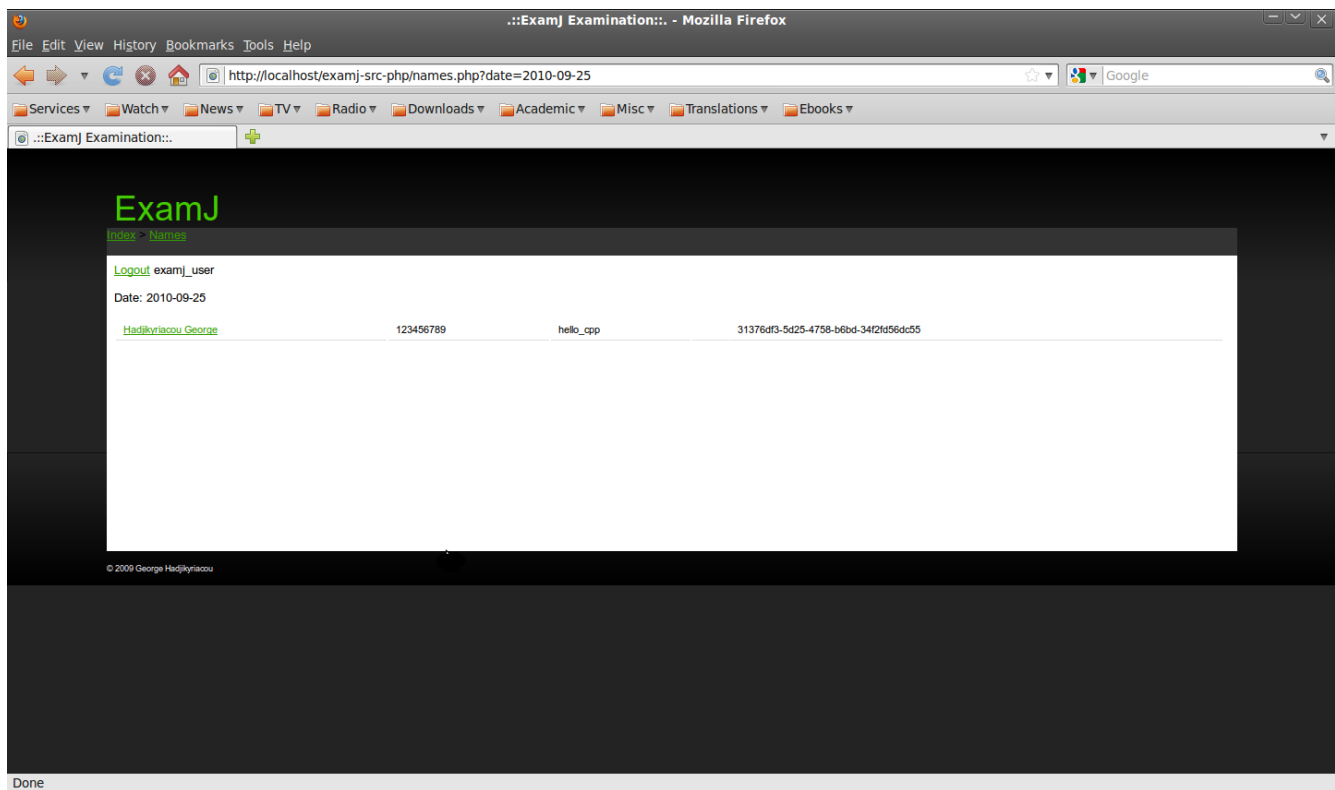
Το επόμενο βήμα, αφού εισέλθει ο χρήστης στο σύστημα, είναι να επιλέξει την ημερομηνία που διεξάχθηκε η εξέταση:



Εικόνα 45: Επιλογή ημερομηνίας

4.5.29 Επιλογή φοιτητή

Αφού επιλέξουμε την ημερομηνία εξέτασης, εμφανίζεται η λίστα των φοιτητών που εξετάστηκαν την συγκεκριμένη ημερομηνία:

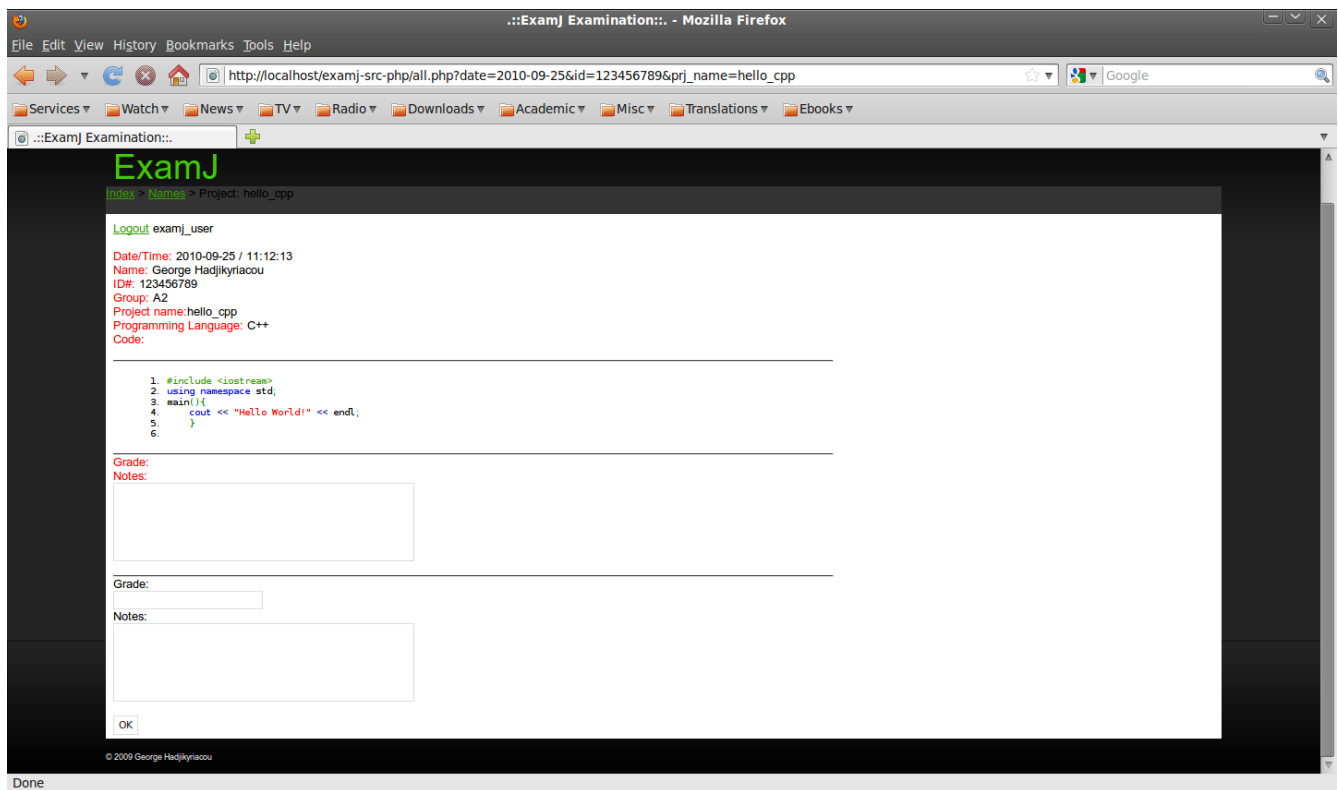


Εικόνα 46: Επιλογή φοιτητή

4.5.30 Αξιολόγηση εργασίας

Μόλις επιλέξουμε τον φοιτητή από την λίστα τότε εμφανίζονται τα στοιχεία του και η εργασία που ανέβασε στη βάση δεδομένων.

Στη συνέχεια αφού βαθμολογήσει (πεδίο Grade) και σχολιάσει (πεδίο Notes) την εργασία πρέπει να πατήσει το κουμπί 'OK' για να ανανεωθεί η βάση δεδομένων.



Εικόνα 47: Αξιολόγηση εργασίας

Απόσπασμα κώδικα από το αρχείο “insert_grade_notes.php”:

```
<?php
$check_user = "SELECT * FROM users WHERE username='$_SESSION[user]' AND
password='$_SESSION[pass]'";
$results = mysql_query($check_user);
$rows = mysql_num_rows($results);
if($rows==1){
    $query_update = "UPDATE code_uploaded SET
grade='$grade',notes='$notes' WHERE id = '$id' AND prj_name =
'$prj_name'";
    $exec_query = mysql_query($query_update,$conn);
    if($exec_query==1){
        echo "Update Complete!";
    }
    else{
        echo "Error!";
    }
}
if($rows!=1){
    echo "Access denied!";
}
?>
```

4.5.31 Απαιτήσεις συστήματος

Όπως αναφέρθηκε πιο πάνω, για να τρέξει η εφαρμογή χρειάζεται να υπάρχει εγκατεστημένος στο σύστημα μας ο JRE (Java Runtime Environment). Επιπλέον θα χρειαστεί να εγκαταστήσουμε τα παρακάτω εργαλεία ανάλογα με τις ανάγκες του μαθήματος:

- Ένας μεταγλωττιστής τουλάχιστον εγκατεστημένος στο σύστημα (π.χ gcc, python, pascal, κτλ). Η εφαρμογή είναι ρυθμισμένη να χρησιμοποιεί μεταγλωττιστές ανοικτού κώδικα.
- Ένας Web Browser (π.χ Firefox, Chrome, κτλ).
- Οι Debuggers (gdb, jdb).
- Ο Code Formatter astyle.
- Το εργαλείο avrdude για τον προγραμματισμό AVR μικροελεγκτών.
- Ο Apache web server, η γλώσσα PHP και η MySQL database server.

4.5.32 Βήματα λειτουργίας εφαρμογής

Τα βήματα που ακολουθούμε για την λειτουργία της εφαρμογής είναι:

1. Εκκίνηση εφαρμογής. Τρέχουμε το αρχείο examj.sh σε λειτουργικά συστήματα GNU/Linux ή examj.bat σε λειτουργικά συστήματα MS Windows.
2. Γράφουμε τον πηγαίο κώδικα της εργασίας μας.
3. Δηλώνουμε το όνομα αρχείου του πηγαίου κώδικα.
4. Εκτελούμε τις διεργασίες: Compile, Run, Send code, κτλ

5. Αποτελέσματα

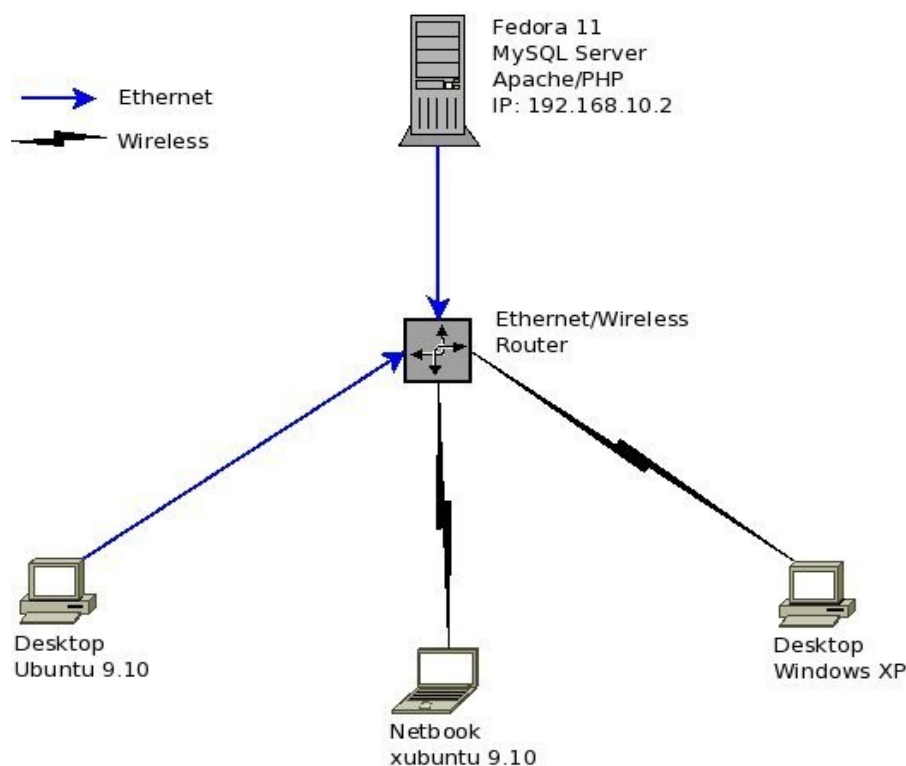
Για σκοπούς δοκιμών χρειάστηκε να δημιουργηθεί ένα τοπικό δίκτυο τεσσάρων υπολογιστών. Ένας υπολογιστής χρησιμοποιήθηκε για την εγκατάσταση της σελίδας των εξεταστών (Apache web server, PHP), ενώ στους υπόλοιπους τρεις εγκαταστάθηκε η εφαρμογή.

Δοκιμάστηκαν οι πιο κάτω λειτουργίες:

1. Εγκατάσταση της εφαρμογής σε διαφορετικά λειτουργικά συστήματα. Συγκεκριμένα χρησιμοποιήθηκαν λειτουργικά συστήματα GNU/Linux και MS Windows.
2. Συγγραφή, μεταγλώττιση, τρέξιμο κώδικα.
3. Ρύθμιση και αποστολή κώδικα σε διεύθυνση ηλεκτρονικού ταχυδρομείου.
4. Ρύθμιση και αποστολή κώδικα στη βάση δεδομένων

Ζητήθηκε από τους χρήστες να ελέγχουν τα αποτελέσματα και τη συμπεριφορά των διάφορων λειτουργιών. Τα αποτελέσματα ανταποκρίνονταν πλήρως στις προσδοκίες των χρηστών.

ExamJ over LAN



Εικόνα 48: Έλεγχος της εφαρμογής σε τοπικό δίκτυο.

5.1 Συμπεράσματα

Σε αυτή την εργασία, περιγράφεται η λειτουργία ενός περιβάλλοντος χρήστη που αποσκοπεί στη διδασκαλία μαθημάτων που απαιτούν χρήση γλώσσας προγραμματισμού. ο σύστημα απευθύνεται σε καθηγητές και μαθητές δευτεροβάθμιας και τριτοβάθμιας εκπαίδευσης. Προσφέρει εργαλεία για την ανάπτυξη και αξιολόγηση εργασιών.

Ένα από τα κύρια χαρακτηριστικά του είναι η απλότητα και η ευκολία στη χρήση. Στους μελλοντικούς στόχους είναι η διασύνδεση του συστήματος με ευφυή σύστημα διδασκαλίας.

5.2 Μελλοντική Εργασία και Επεκτάσεις

Σε αυτή την ενότητα ακολουθούν μερικές ιδέες για μελλοντική εργασία και επεκτάσεις πάνω στην εφαρμογή.

5.2.1 Επεκτάσεις

Πιθανές επεκτάσεις που θα μπορούσαν να γίνουν στην εφαρμογή είναι:

1. Υποστήριξη περισσότερων γλωσσών προγραμματισμού (π.χ η γλώσσα Prolog για το μάθημα “Τεχνητή Νοημοσύνη”).
2. Ενσωμάτωση συστήματος εκμάθησης της γλώσσας προγραμματισμού.
3. Σύστημα ελέγχου για αποφυγή αντιγραφών κατά τη διάρκεια της εξέτασης του μαθήματος.
4. Σύστημα ενσωμάτωσης έτοιμων τμημάτων κώδικα στην εργασία.
5. Εικονικός δάσκαλος. Ενσωμάτωση κάποιου είδους ευφυούς συστήματος διδασκαλίας.^[11]

5.2.2 Ευφυή συστήματα διδασκαλίας^[11]

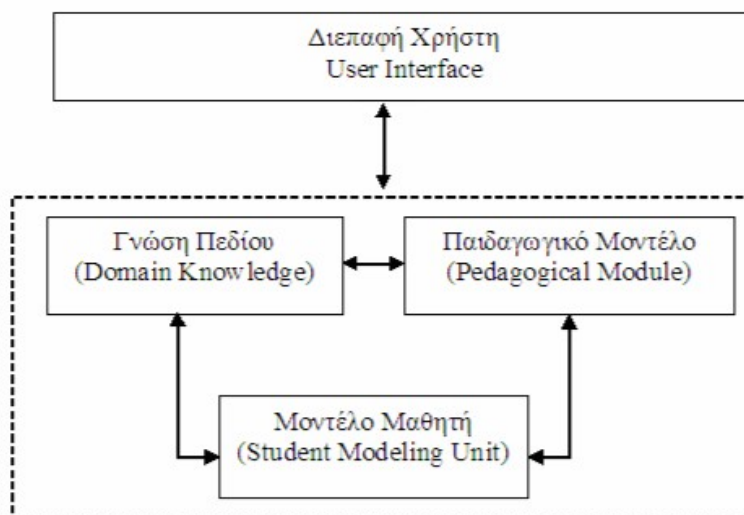
Πολλά συστήματα βασισμένα σε υπολογιστή έχουν χρησιμοποιηθεί στην εκπαίδευση τις τελευταίες δεκαετίες. Τα πρώτα αυτά συστήματα αποτελούσαν τα Computer Aided Instruction (CAI) συστήματα. Τα συστήματα αυτά πρόσφεραν αρκετή βοήθεια στους εκπαιδευόμενους. Ένα μεγάλο μειονέκτημά τους ήταν το γεγονός ότι δεν λάμβαναν υπόψη τους το γνωστικό επίπεδο και τις ικανότητες του κάθε χρήστη. Δεν μπορούσαν συνεπώς να προσαρμόσουν τη διδασκαλία στις προσωπικές ανάγκες των χρηστών. Τα μειονεκτήματα αυτά οδήγησαν στην εμφάνιση μιας νέας γενιάς εκπαιδευτικών συστημάτων που λέγονται Ευφυή Συστήματα Διδασκαλίας (Intelligent Tutoring Systems).

Ένα σημαντικό χαρακτηριστικό αυτών των συστημάτων είναι η ικανότητά τους να προσαρμόζουν την παρουσίαση του εκπαιδευτικού υλικού στις ανάγκες και ιδιαιτερότητες των χρηστών τους. Αυτό επιτυγχάνεται με τη χρήση μεθόδων της Τεχνητής Νοημοσύνης (Artificial Intelligence) για την αναπαράσταση των παιδαγωγικών αποφάσεων και των πληροφοριών που αφορούν τον κάθε εκπαιδευόμενο. Για αυτούς τους λόγους, τα Ευφυή Συστήματα Διδασκαλίας έχουν γίνει εξαιρετικά δημοφιλή τα τελευταία χρόνια και έχουν επιδείξει αποτελεσματικότητα στην αύξηση της απόδοσης των χρηστών.

Τα Ευφυή Συστήματα Διδασκαλίας συνήθως υλοποιούνταν σαν stand-alone συστήματα. Η εμφάνιση και η ραγδαία εξάπλωση του Παγκόσμιου Ιστού (World Wide Web) παρέχει τη δυνατότητα για επαναστατικές αλλαγές σε όλα τα επίπεδα της εκπαίδευσης. Ο Παγκόσμιος Ιστός δίνει σε ένα εκπαιδευτικό σύστημα όπως ένα Ευφρές Σύστημα Διδασκαλίας την ευκαιρία να προσπελαστεί από πολλούς χρήστες. Με αυτό τον τρόπο, οι λειτουργίες του μπορούν να δοκιμαστούν με πολλές και διαφορετικές περιπτώσεις χρηστών.

Ποιο κάτω παρουσιάζεται η αρχιτεκτονική ενός Ευφυούς Συστήματος Διδασκαλίας. Αποτελείται από τα ακόλουθα συστατικά μέρη:

- Τη βάση πεδίου που περιέχει το εκπαιδευτικό υλικό.
- Το μοντέλο μαθητή που καταγράφει πληροφορίες που αφορούν τον χρήστη.
- Το παιδαγωγικό μοντέλο που εμπεριέχει γνώση σχετικά με τις διάφορες παιδαγωγικές αποφάσεις.
- Τη διεπιφάνεια χρήστη.



Εικόνα 49: Η Αρχιτεκτονική του Ευφυούς Συστήματος Διδασκαλίας

5.2.2.1 Βάση Πεδίου

Η βάση πεδίου περιέχει το εκπαιδευτικό υλικό που παρουσιάζεται στους χρήστες του συστήματος. Το εκπαιδευτικό υλικό αφορά μια ποικιλία από μαθήματα που ξεκινούν από εισαγωγικά εκπαιδευτικά ζητήματα και κλιμακώνονται σε πιο προχωρημένα ζητήματα. Το εκπαιδευτικό περιεχόμενο του κάθε μαθήματος οργανώνεται σε ενότητες, υποενότητες και θέματα.

Κάθε εκπαιδευτική μονάδα συσχετίζεται με ορισμένες γνωστικές έννοιες. Οι έννοιες αυτές μπορεί να είναι είτε προαπαιτούμενες έννοιες (δηλαδή πρέπει να είναι γνωστές στον χρήστη έτσι ώστε κατανοήσει το περιεχόμενο της εκπαιδευτικής μονάδας) είτε έννοιες η γνώση των οποίων θα προκύψει μετά τη μελέτη της εκπαιδευτικής μονάδας. Οι έννοιες συσχετίζονται μεταξύ τους περιγράφοντας για κάθε έννοια τις έννοιες που είναι

προαπαιτούμενες και αποτέλεσμα.

Το κάθε θέμα αποτελείται από μια σειρά εκπαιδευτικών οθονών που περιέχουν θεωρία, παραδείγματα και ασκήσεις. Τα παραδείγματα βοηθούν το χρήστη να κατανοήσει τα βασικά σημεία της θεωρίας. Ο αριθμός των παραδειγμάτων που παρουσιάζονται εξαρτάται από το μοντέλο μαθητή. Ένας χρήστης με μεγάλη ικανότητα μάθησης και υψηλό γνωστικό επίπεδο βλέπει μικρό αριθμό παραδειγμάτων σε αντίθεση με έναν χρήστη που έχει χαμηλό γνωστικό επίπεδο ή μικρή ικανότητα μάθησης. Τα προβλήματα βασίζονται στα παραδείγματα. Κάθε πρόβλημα συσχετίζεται με μια επεξήγηση που βοηθά το χρήστη σε περίπτωση που δίνει λανθασμένη απάντηση. Οι εκπαιδευτικές μονάδες παρουσιάζονται με διάφορους τρόπους όπως με κείμενο, στατικές εικόνες και κινούμενες εικόνες (animations). Αυτό εξαρτάται από τον πολυμεσικό τύπο με τον οποίο ο χρήστης προτιμά να αλληλεπιδρά. Αυτές οι προτιμήσεις του χρήστη είναι μέρος του μοντέλου μαθητή. Για τη διευκόλυνση της επιλογής του υλικού, κάθε εκπαιδευτική μονάδα συσχετίζεται με χαρακτηριστικά που καθορίζουν τον τύπο των προτιμήσεων του χρήστη με τον οποίο αντιστοιχούν.

5.2.2.2 Το μοντέλο μαθητή

Το μοντέλο μαθητή καταγράφει πληροφορίες σχετικές με το χρήστη που αφορούν το επίπεδο γνώσης καθώς και άλλα χαρακτηριστικά του γνωρίσματα. Οι πληροφορίες αυτές είναι απαραίτητες για τη λειτουργία του συστήματος με βάση τις ανάγκες του χρήστη. Είναι δύσκολο ωστόσο να συγκεντρωθούν τέτοια δεδομένα διότι δεν είναι εύκολη η αναπαράσταση των ικανοτήτων του χρήστη. Επιπλέον, ο Παγκόσμιος Ιστός επιβάλλει ορισμένους περιορισμούς στην αντίληψη που έχει το σύστημα για το χρήστη. Προς το παρόν είναι δύσκολο και χρονοβόρο να καταγράφεται η κάθε ενέργεια του χρήστη. Επιπλέον το μοντέλο μαθητή δεν πρέπει να περιέχει μη αναγκαίες πληροφορίες έτσι ώστε το σύστημα να μην επιβαρύνεται με άχρηστες αλληλεπιδράσεις.

Ένα μοντέλο μαθητή που έχει χρησιμοποιηθεί αρκετά συχνά στο παρελθόν και χρησιμοποιείται και στο δικό μας σύστημα είναι το μοντέλο επικάλυψης (overlay model). Σε αυτό το μοντέλο, η γνώση του χρήστη θεωρείται ότι είναι υποσύνολο της γνώσης ενός εμπειρογνώμονα του γνωστικού πεδίου. Με αυτή την αναπαράσταση, το σύστημα παρουσιάζει το εκπαιδευτικό υλικό στο χρήστη έτσι ώστε στο τέλος η γνώση του να συμπίπτει με τη γνώση του ειδικού. Ένα μειονέκτημα του μοντέλου επικάλυψης είναι η ανικανότητά του να αναπαραστήσει πιθανές παρερμηνείες του χρήστη. Για το σκοπό αυτό χρησιμοποιούνται συχνά άλλα μοντέλα όπως το bug catalogue ή η βελτιωμένη εκδοχή του, το bug-parts-library.

Το μοντέλο μαθητή βασίζεται στις έννοιες που συσχετίζονται με τις εκπαιδευτικές μονάδες. Επιπλέον πληροφορίες που καταγράφει το σύστημα αφορούν την ικανότητα μάθησης και συγκέντρωσης του μαθητή. Η απόκριση του μαθητή κατά την αλληλεπίδρασή του με το σύστημα καθορίζουν πόσο υψηλές ή χαμηλές είναι αυτές οι ικανότητες. Επιπλέον καταγράφονται οι προτιμήσεις του χρήστη σχετικά με τον πολυμεσικό τύπο (π.χ. κείμενο, στατικές εικόνες, animations) των εκπαιδευτικών μονάδων. Οι προτιμήσεις αυτές καταγράφονται όταν ο χρήστης αποκτά λογαριασμό στο σύστημα ενώ μπορούν να αλλαχθούν και κατά τη διάρκεια της εκπαιδευτικής διαδικασίας.

Μόνο καταγεγραμμένοι χρήστες μπορούν να έχουν πρόσβαση στο σύστημα. Με αυτό τον τρόπο το σύστημα μπορεί να καταχωρεί το γνωστικό επίπεδο και τις άλλες ιδιαιτερότητές τους. Ένας καταγεγραμμένος χρήστης προσδιορίζει την ταυτότητά του κάθε φορά που εισάγεται στο σύστημα δίνοντας ένα όνομα χρήστη (login name) και έναν κωδικό (password). Ένας μη καταγεγραμμένος χρήστης πρέπει πρώτα να υποβάλλει στο σύστημα πληροφορίες σχετικές με αυτόν (π.χ. όνομα, ηλεκτρονική διεύθυνση, προτιμήσεις σε πολυμεσικό τύπο) έτσι ώστε να αποκτήσει λογαριασμό που θα του επιτρέψει να έχει πρόσβαση στις λειτουργίες του συστήματος.

5.2.2.3 Παιδαγωγικό Μοντέλο

Το παιδαγωγικό μοντέλο αναπαριστά την εκπαιδευτική διαδικασία. Παρέχει τη γνώση που απαιτείται έτσι ώστε να προσαρμόζεται η παρουσίαση του εκπαιδευτικού υλικού σύμφωνα με τα δεδομένα του μοντέλου μαθητή. Το παιδαγωγικό μοντέλο εμπεριέχει πληροφορίες σχετικά με τις διάφορες εκπαιδευτικές στρατηγικές.

Οι στρατηγικές αυτές καθορίζουν πως πρέπει να οργανωθεί ένα μάθημα. Επιπλέον το παιδαγωγικό μοντέλο περιέχει γνώση σχετικά με την επιλογή των διαφόρων εκπαιδευτικών μονάδων με βάση τις ιδιαιτερότητες του χρήστη.

5.2.2.4 Έμπειρο Σύστημα

Το παιδαγωγικό μοντέλο και το πεδίο γνώσης είναι μέρη ενός έμπειρου συστήματος που ελέγχει την εκπαιδευτική διαδικασία. Το έμπειρο σύστημα έχει έναν μηχανισμό εξαγωγής συμπερασμάτων έτσι ώστε να λαμβάνει αποφάσεις με βάση τα γνωστά δεδομένα.

Οι συμβολικοί κανόνες αποτελούν έναν δημοφιλή τρόπο αναπαράστασης γνώσης για την ανάπτυξη έμπειρων συστημάτων. Οι κανόνες αντιπροσωπεύουν τη γνώση ενός ειδικού στο γνωστικό πεδίο εκπαίδευσης και εκμαιεύονται κυρίως μέσω συνεντεύξεων. Οι κανόνες επιδεικνύουν ορισμένα ελκυστικά χαρακτηριστικά όπως φυσικότητα, ομοιομορφία, modularity και ευκολία παροχής επεξηγήσεων. Είναι μια τυπική προσέγγιση όσον αφορά τη συμβολική αναπαράσταση γνώσης. Ένα από τα κυριότερα μειονεκτήματά τους είναι ότι η αλληλεπίδραση με τον εμπειρογνώμονα μπορεί να αποδειχτεί προβληματική προκαλώντας καθυστερήσεις στην ανάπτυξη του συνολικού συστήματος.

Κατά τα τελευταία χρόνια, νευρωνικά δίκτυα χρησιμοποιούνται αρκετά συχνά στην υλοποίηση έμπειρων συστημάτων. Τα νευρωνικά δίκτυα αποτελούν μια εντελώς διαφορετική προσέγγιση στο πρόβλημα της αναπαράστασης γνώσης που είναι γνωστή ως συνδεσμισμός (connectionism). Ορισμένα πλεονεκτήματα των νευρωνικών δικτύων είναι η ικανότητά τους να αποκτούν γνώση από παραδείγματα εκπαίδευσης (μειώνοντας την αλληλεπίδραση με τους εμπειρογνώμονες), το υψηλό επίπεδο απόδοσης και η ικανότητα αναπαράστασης πολύπλοκης και ανακριβούς γνώσης.

Στο σύστημά μας, υιοθετούμε μια υβριδική προσέγγιση που ολοκληρώνει συμβολικούς κανόνες με νευρωνικά δίκτυα. Ο στόχος είναι να αυξάνονται τα αντίστοιχα πλεονεκτήματα των δύο αυτών φορμαλισμών αναπαράστασης γνώσης και ταυτόχρονα να μειώνονται τα μειονεκτήματά τους. άλλες γνωστές υβριδικές προσεγγίσεις έχουν αποδειχτεί αρκετά αποδοτικές.

5.2.2.5 Διεπιφάνεια Χρήστη

Η διεπιφάνεια χρήστη είναι υπεύθυνη για την αλληλεπίδραση του συστήματος με το χρήστη. Καθώς είναι το κομμάτι του συστήματος που επικοινωνεί απευθείας με το χρήστη πρέπει να σχεδιαστεί με προσοχή. Ο σχεδιασμός και η υλοποίηση της διεπιφάνειας χρήστη του συστήματος είναι μια σημαντική φάση κατά την ανάπτυξη του συστήματος καθώς η αλληλεπιδραστικότητα του συστήματος καθορίζει σε μεγάλο βαθμό την αποδοχή του από την πλευρά των χρηστών. Ο κύριος στόχος είναι ο σχεδιασμός μιας διεπιφάνειας χρήστη που θα μπορεί να χρησιμοποιηθεί από χρήστες με διαφορετικές ικανότητες, ανάγκες, απαιτήσεις και προτιμήσεις. Βασικό ζήτημα είναι η υλοποίηση ενός συστήματος που θα υποστηρίξει τις λειτουργίες που παρέχει στο χρήστη αποδοτικά και αποτελεσματικά.

Κύρια ζητήματα της διεπιφάνειας χρήστη του συστήματος είναι τα εξής:

1. Αλληλεπίδραση, ροή και πλοήγηση μεταξύ οθονών ή άλλων μερών του συστήματος.
2. Συσχετίσεις μεταξύ των μηνυμάτων του συστήματος.
3. Σχεδιασμοί οθονών.
4. Μηνύματα που θα ενημερώνουν το χρήστη και θα προσελκύουν το ενδιαφέρον του. Η ανάδραση αυτή παίρνει τις ακόλουθες μορφές:

- (a) Μηνύματα κατάστασης (status messages) που δείχνουν την πρόοδο της διεργασίας που επιτελείται.
- (b) Μηνύματα προειδοποίησης (warning messages) που γνωστοποιούν στους χρήστες τις συνέπειες των ενεργειών που εκτελούν.
- (c) Ανάδραση διόρθωσης που υποδηλώνει αν η απόκριση του χρήστη είναι σωστή ή όχι.
- (d) Ανάδραση πλοήγησης που δείχνει στους χρήστες πού βρίσκονται. Αυτό είναι ιδιαίτερα σημαντικό διότι η δομή του συστήματος μπορεί να μην είναι απαραίτητα ιεραρχική.

Μπορούν να διακριθούν δύο όψεις της διεπιφάνειας χρήστη όσον αφορά τους χρήστες:

1. Γενική άποψη: Στη γενική άποψη ο χρήστης μπορεί να προσπελάσει όλο το διαθέσιμο εκπαιδευτικό υλικό.
2. Άποψη του διαχειριστή: Στην άποψη του διαχειριστή ο χρήστης μπορεί να ενημερώσει το παιδαγωγικό μοντέλο και τη βάση πεδίου εισάγοντας νέα αντικείμενα, ή τροποποιώντας και διαγράφοντας υπάρχοντα αντικείμενα.

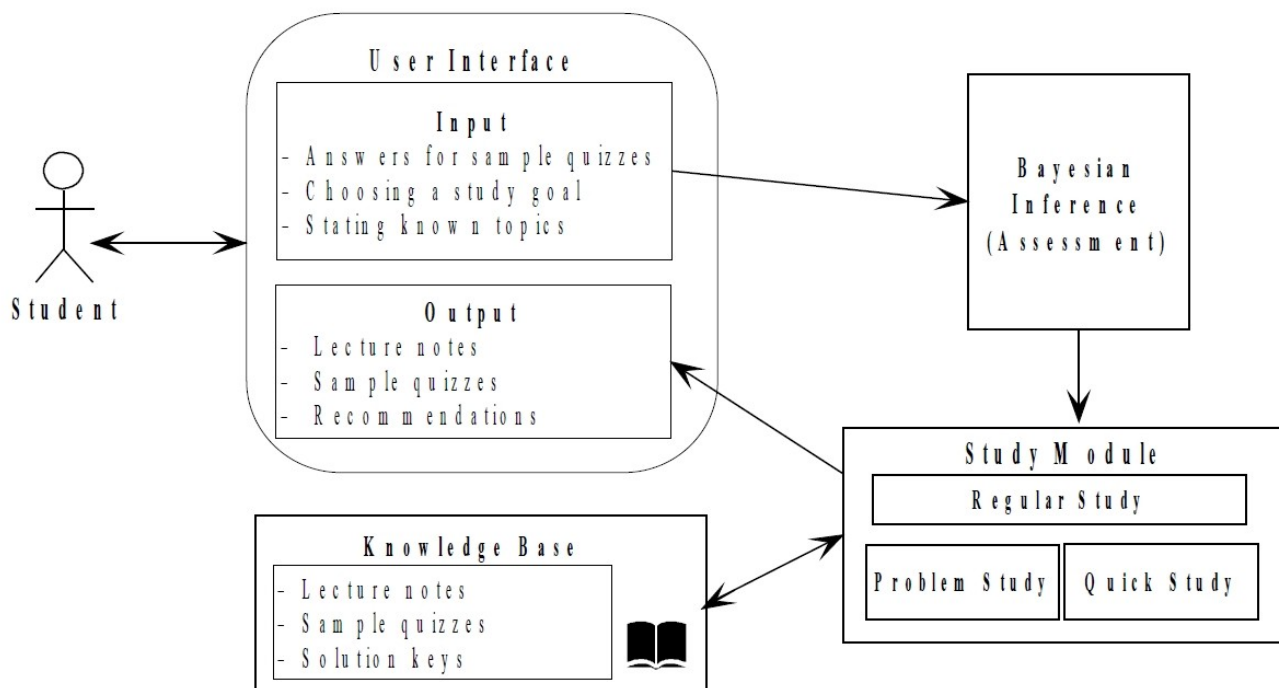
Μόνο ο διαχειριστής του συστήματος μπορεί να προσπελάσει το σύστημα μέσα από την άποψη του διαχειριστή. Η προσπέλαση μέσα από τη γενική άποψη μπορεί να γίνει τόσο από τους εκπαιδευόμενους όσο και από τον διαχειριστή.

6. Ευφυή συστήματα διδασκαλίας προγραμματισμού υπολογιστών

Σε αυτό το κεφάλαιο παρουσιάζονται ορισμένα ευφυή συστήματα διδασκαλίας και προσαρμοστικά συστήματα τα οποία έχουν να κάνουν με τη διδασκαλία γλωσσών προγραμματισμού. Στόχος αυτής της μελέτης ήταν να πάρουμε κάποιες ιδέες για το πώς μπορεί να εξελιχθεί η εφαρμογή που παρουσιάσαμε σε αυτή τη πτυχιακή.

6.1 Το σύστημα Bits

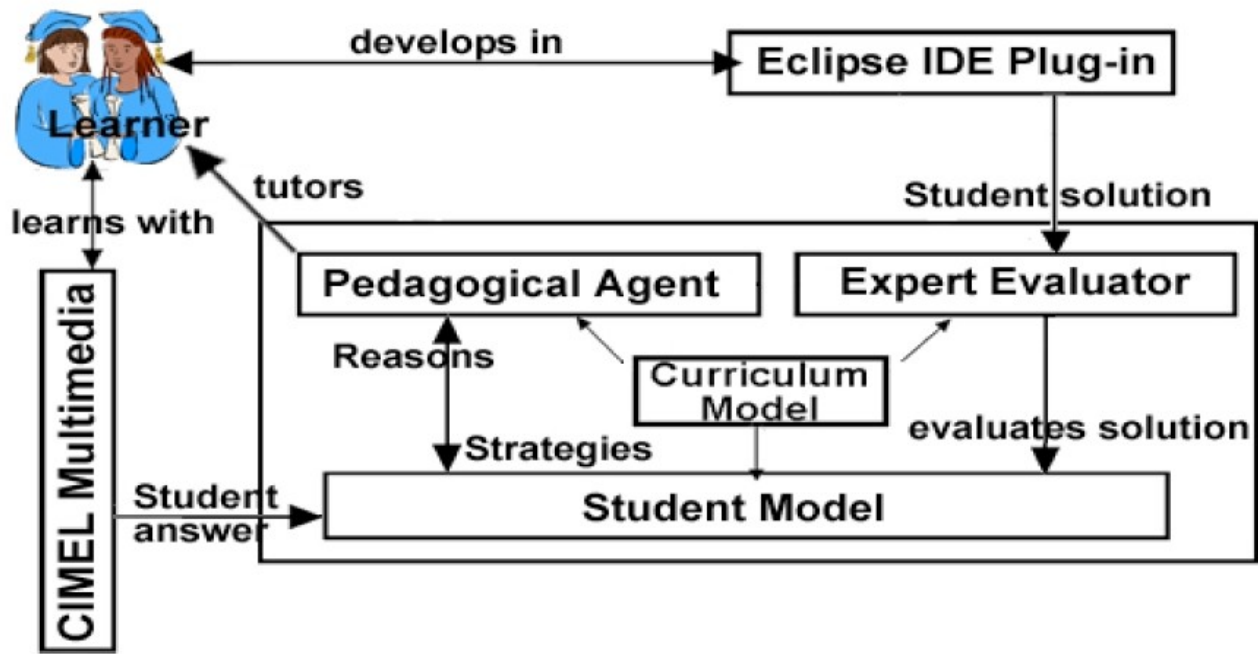
Το Bits (Bayesian intelligent tutoring system)^[12] είναι ένα ευφύες σύστημα διδασκαλίας για προγραμματισμό υπολογιστών. Στην εικόνα που ακολουθεί βλέπουμε την αρχιτεκτονική του συστήματος Bits:



Εικόνα 50: Αρχιτεκτονική του Bits

6.2 Το σύστημα CIMEL ITS

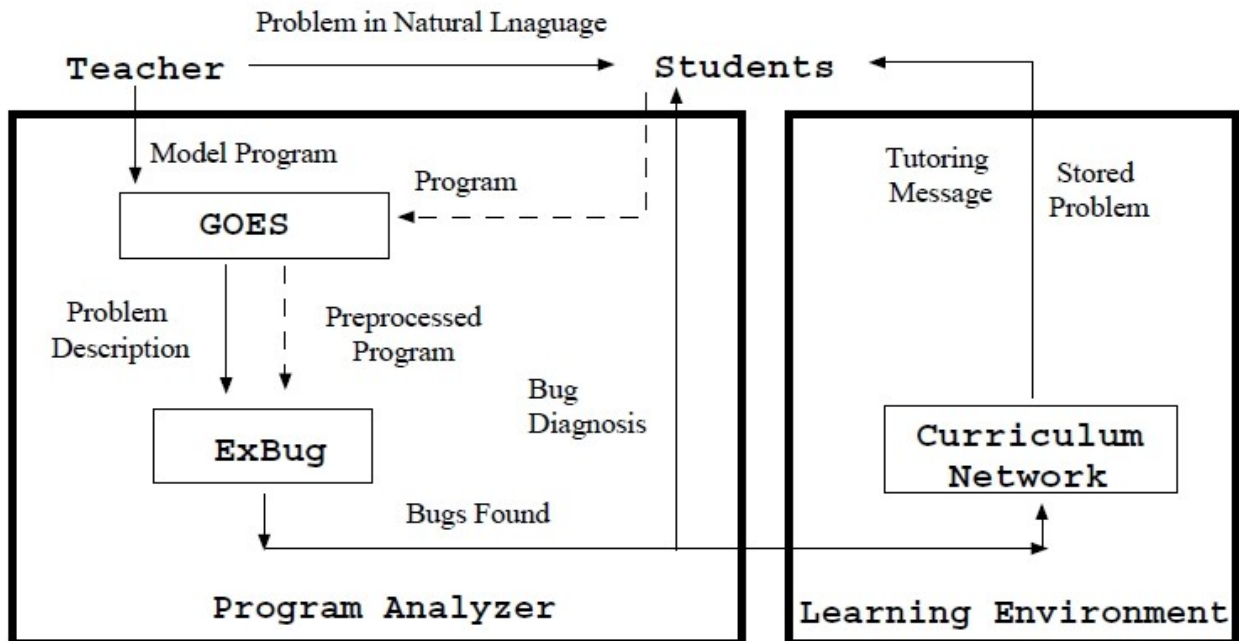
Το CIMEL ITS^[13] είναι ένα ευφύες σύστημα διδασκαλίας για αντικειμενοστραφείς γλώσσες προγραμματισμού. Μπορεί να ενσωματωθεί σε διάφορα περιβάλλοντα ανάπτυξης λογισμικού όπως για παράδειγμα το Eclipse IDE. Στην εικόνα που ακολουθεί βλέπουμε την αρχιτεκτονική του συστήματος CIMEL:



Εικόνα 51: Αρχιτεκτονική του CIMEL ITS

6.3 Το σύστημα C-Tutor

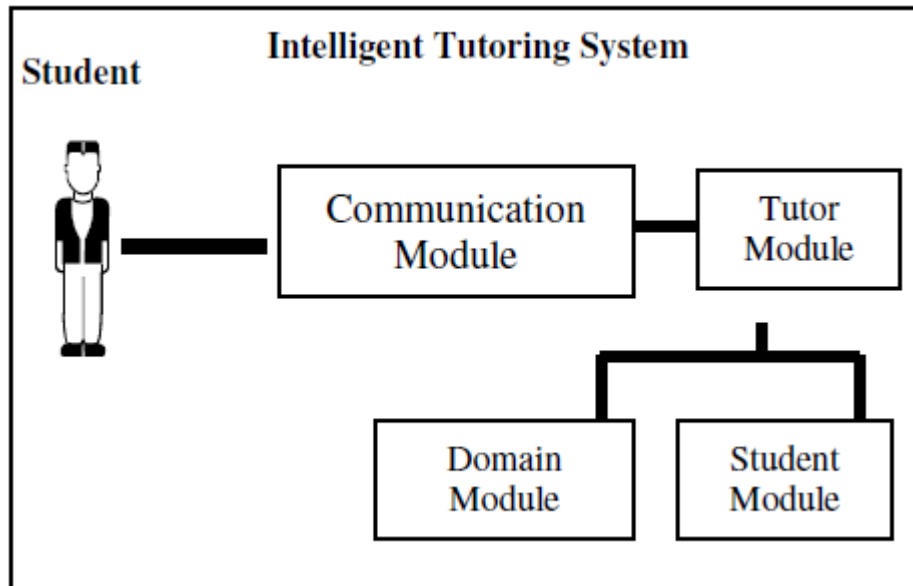
Το C-Tutor^[14] είναι ένα ευφρές σύστημα διδασκαλίας για τη γλώσσα προγραμματισμού C. Στην εικόνα που ακολουθεί βλέπουμε την αρχιτεκτονική του συστήματος C-Tutor:



Εικόνα 52: Αρχιτεκτονική του C-Tutor

6.4 Το σύστημα CITS

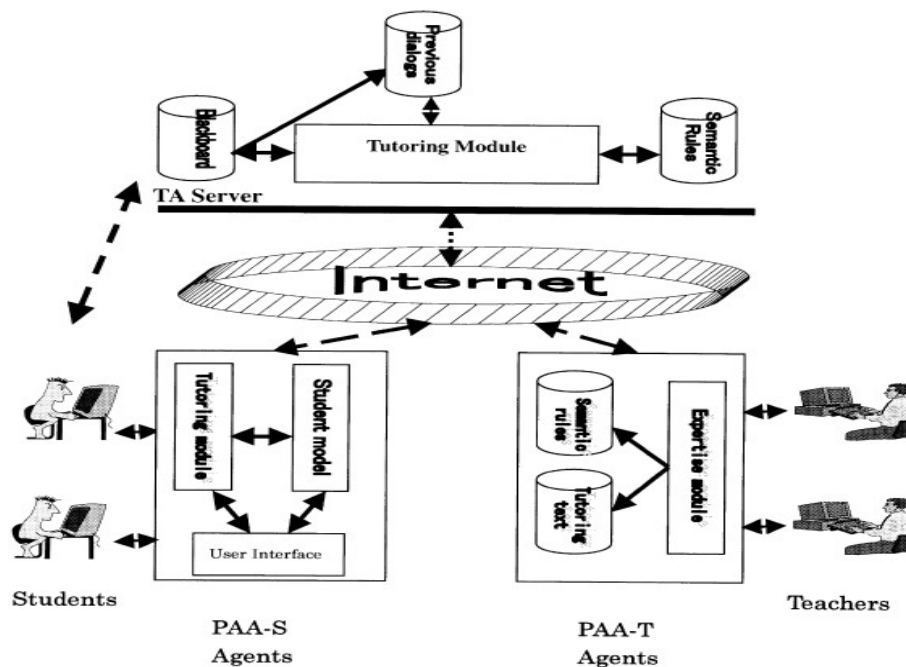
Το CITS^[15] είναι ένα ευφρές σύστημα διδασκαλίας για τη γλώσσα προγραμματισμού C++. Στην εικόνα που ακολουθεί βλέπουμε την αρχιτεκτονική του συστήματος CITS:



Εικόνα 53: Αρχιτεκτονική του CITS

6.5 Το σύστημα E-TCL

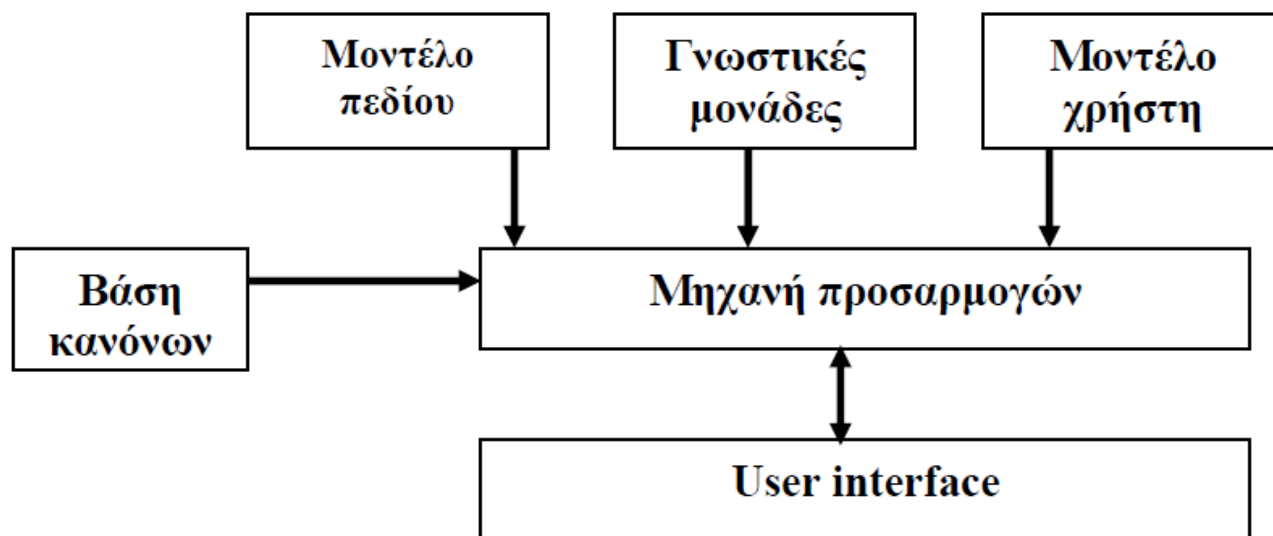
Το E-TCL (Expert tutoring system)^[16] είναι ένα ευφρές σύστημα διδασκαλίας γλώσσα προγραμματισμού μέσω διαδικτύου. Στην εικόνα που ακολουθεί βλέπουμε την αρχιτεκτονική του συστήματος E-TCL:



Εικόνα 54: Αρχιτεκτονική του E-TCS

6.6 Το σύστημα ALEA

Το ALEA^[17] (Adaptive Learning system) των (Kostelník R. και M. Bieliková. 2003) είναι ένα προσαρμοστικό σύστημα για το διαδίκτυο το οποίο έχει σκοπό να διδάξει γλώσσες προγραμματισμού χρησιμοποιώντας προσαρμοστική διαδοχή ασκήσεων. Η προσαρμογή του συστήματος βασίζεται στα διαφορετικά γνωστικά στυλ των χρηστών.



Εικόνα 55: Αρχιτεκτονική του ALEA

6.7 Το σύστημα ELM-ART

Το ELM-ART^[17] των (Brusilovsky, P., Schwarz, E., & Weber, G. 1996) βασίζεται στο ELM-PE το οποίο είναι ένα ILE για την εκμάθηση της γλώσσας LISP και υποστήριξε τις τεχνολογίες της ευφυούς ανάλυσης των λύσεων των μαθητών και του προγραμματισμού που βασίζεται σε παραδείγματα. Το ELM-PE ήταν όπως είδαμε κατάλληλο για αρχάριους χρήστες αλλά δεν έδινε την δυνατότητα σε αυτούς να εμβαθύνουν τις γνώσεις τους.

6.8 Το σύστημα ELM-ART II

Στο ELM-ART II^[17] των (Weber, G. and Brusilovsky, P. 2001): υπάρχουν περισσότερα τεστ αξιολόγησης τα οποία επιτρέπουν στο σύστημα να κάνουν καλύτερη διάγνωση για το επίπεδο του μαθητή με αποτέλεσμα το μοντέλο του μαθητή να είναι πιο πλήρης. Έτσι το σύστημα μπορεί με ακρίβεια να γνωρίζει πότε ο χρήστης έχει απλά επισκεφτεί την σελίδα, πότε έχει καταλάβει ένα μέρος μιας έννοιας και πότε την έχει κατανοήσει πλήρως.

6.9 Το σύστημα ISIS-Tutor

Το ISIS-Tutor^[17] του Brusilovsky, P. (1994) είναι ένα ευφύες περιβάλλον μάθησης (ILE) που έχει ενσωματωμένο ένα τμήμα υπερμέσων για την υποστήριξη της μαθητοκεντρικής διδασκαλίας. Χρησιμοποιείται για την εκμάθηση της print formatting γλώσσας η οποία μπορεί να θεωρηθεί ως γλώσσα προγραμματισμού. Είναι από τα πρώτα συστήματα που προσπάθησαν να συνδυάσουν τα πλεονεκτήματα των ILE και των υπερμέσων ανοίγοντας νέους δρόμους στην εκπαίδευση με τη χρήση υπολογιστή.

6.10 Το σύστημα Pl@tos

Το Pl@tos^[17] των (Liegle, Jens O. και Han-Gyun Woo. 2000) είναι ένα προσαρμοστικό ITS που μπορεί να χρησιμοποιηθεί για την διδασκαλία γλωσσών προγραμματισμού. Ο σκοπός του είναι να προσφέρει εξατομικευμένη μάθηση βασισμένη στο στυλ μάθησης του κάθε χρήστη (εξερευνητής ή παρατηρητής) και στις προηγούμενες γνώσεις του.

Πηγές

1. Phil Ballard, Michael Moncur, "Μάθετε Ajax, Javascript και PHP"
2. Julie C. Meloni, "PHP, MySQL και Apache"
3. Andrew S. Tanenbaum, "Δίκτυα Υπολογιστών", 4η Έκδοση
4. Rogers Cadenhead, "Μάθετε την Java 2 σε 24 Ωρες"
5. Γιώργος Λιακέας, "Εισαγωγή στην Java 2"
6. Matt Welsh, Mathias Kalle Dalheimer & Lar Kaufman, "Οδηγός του LINUX", 3η Αμερικάνικη έκδοσή
7. Πανεπιστήμιο Πάτρας, Εργαστήριο Γραφικών, Πολυμέσων & Γεωγραφικών Συστημάτων, <http://mmlab.ceid.upatras.gr/aigroup/undergrad/java/ergastirio/JavaFrequentlyAskedQuestions.htm> (Τελευταία πρόσβαση: 25/9/2010)
8. Πανεπιστήμιο Ιωαννίνων, PHP/MySQL, Παρουσίαση στα πλαίσια του μαθήματος Επικοινωνία ανθρώπου-μηχανής, <http://cs.uoi.gr/~fudos/chi-presentations-2008/eam.rpt> (Τελευταία πρόσβαση: 26/9/2010)
9. Πανεπιστήμιο Μακεδονίας, Αλέξανδρος Χατζηγεωργίου, Τεχνολογία Λογισμικού, http://users.uom.gr/~achat/soft_eng.html (Τελευταία πρόσβαση: 18/10/2010)
10. Πανεπιστήμιο Πειραιά, Δρ. Βασίλειος Βεσκούκης, Τεχνολογία Λογισμικού, http://users.softlab.ece.ntua.gr/~bxb/courses/unipi2001_te/unipi_te.htm (Τελευταία πρόσβαση: 18/10/2010)
11. Πανεπιστήμιο Πατρών, Δ. Πρέντζας, Ι. Χατζηλυγερούδης, Κ. Κουτσογιάννης, Μ. Ρήγκου, Η Αρχιτεκτονική ενός Ευφυούς Συστήματος Βασισμένο στο Διαδίκτυο για τη Διδασκαλία Νέων Τεχνολογιών Πληροφορικής, http://www.eap.gr/news/EXAGGELIA_SYNEDRIOU/synedrio/html/sect6/37.htm (Τελευταία πρόσβαση: 21/10/2010)
12. C.J.Butz, S.Hua, R.B.Maguire - Bits: a Bayesian Intelligent Tutoring System For Computer Programming.
13. Fang Wei, Sally H. Moritz, Shahida M. Parvez and Glenn D. Blank – A student model for object oriented design and programming
14. S.H. Hahn, J.S.Song, K.Y.Tak and J.H.Kim – An Intelligent Tutoring System For Introductory C language course
15. Hassan Zia, Qaiser S. Durrani, Rana Adnan Farrakh, Amir Riaz, Farhan Ahmed - CITS-C++ Intelligent Tutoring Tystem: A Domain Independent User Centered Curriculum Approach
16. Expert tutoring system for teaching computer programming languages - M.M. El-Khouly, B.H. Far, Z. Koono
17. Πανεπιστήμιο Πατρών - Ιωάννης Ε. Σιμιτζής - «AWBES» Προσαρμοστικά Εκπαιδευτικά συστήματα για το διαδίκτυο

Παράρτημα

• Μεταγλώττιση εφαρμογής	72
• Προγραμματισμός ηλεκτρονικής πλατφόρμας ανοικτού κώδικα ARDUINO	73
• Χρήση της MySQL βάσης δεδομένων σε λειτουργικό σύστημα Linux	74
• Άδεια χρήσης	79
• Περίληψη Πτυχιακής	81
• Παρουσίαση	86

Μεταγλώττιση εφαρμογής

Στον φάκελο examj-src/examj-src-java βρίσκεται ο πηγαίος κώδικας της εφαρμογής. Για ευκολία του προγραμματιστή, υπάρχει ένα script κελύφους Linux^[6] με το όνομα linux-compile.sh, που αυτοματοποιεί την διαδικασία. Συγκεκριμένα κάνει τα παρακάτω:

- Μεταγλώττιση κώδικα. Τα μνήματα σφάλματος που μπορεί να προκύψουν δρομολογούνται στο αρχείο compile_log.txt
- Δημιουργία εκτελέσιμου (αρχείο .jar)
- Αντιγραφή του εκτελέσιμου αρχείου στο φάκελο examj-bin/bin/
- Καθαρισμός
- Μορφοποίηση του κώδικα με το astyle

Ακολουθεί το script:

```
#!/bin/bash

#last update
echo "-----" > compile_log.txt
date >> compile_log.txt
echo "-----" >> compile_log.txt

#compile
javac -classpath \
../..examj-bin/bin/commons-codec-1.1.jar:\
../..examj-bin/bin/commons-lang-2.1.jar:\
../..examj-bin/bin/mailapi.jar:\
../..examj-bin/bin/jasypt-1.7.jar:\
../..examj-bin/bin/mysql-connector-java-5.1.14-bin.jar:\
../..examj-bin/bin/jsyntaxpane-0.9.5-b29.jar:\
../..examj-bin/bin/sntp.jar:\
../..examj-bin/bin/jtidy-r938.jar \
src/*.java 2>> compile_log.txt

#create JAR
mv src/*.class examj/
jar cfm examj-bin.jar manifest.txt examj/*.class

#copy examj-bin.jar to 'bin' directory
mv examj-bin.jar ../..examj-bin/bin/

#clean
rm examj/*.class

#code formatter (optional)
../..examj-bin/bin/tools_linux/astyle -n --style=java src/*.java
```


Προγραμματισμός ηλεκτρονικής πλατφόρμας ανοικτού κώδικα ARDUINO

Μια δυνατότητα που έχει η εφαρμογή είναι ο προγραμματισμός μικρο ελεγκτών AVR. Για την δοκιμή αυτής της λειτουργίας χρησιμοποιήθηκε η ηλεκτρονική πλατφόρμα ανοικτού κώδικα ARDUINO, η γλώσσα προγραμματισμού C (avr-gcc) και ο προγραμματιστής avrdude.

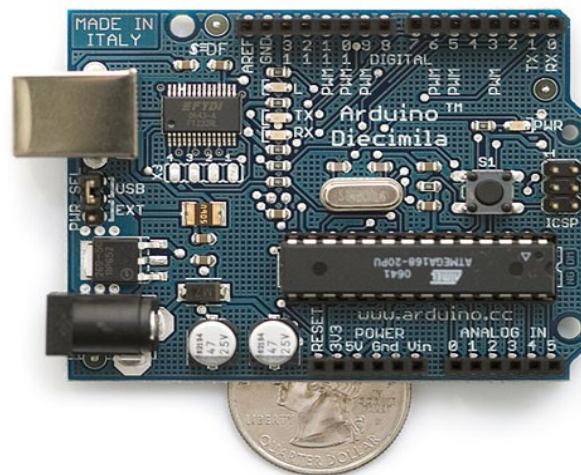
Το Arduino είναι ένας μικροελεγκτής ο οποίος περιλαμβάνει ένα chip ATmega. Διαθέτει εισόδους και εξόδους που αντιδρούν βάση του προγραμματισμού που κάναμε και που φορτώσαμε στο chip με τη βοήθεια του υπολογιστή. Η γλώσσα προγραμματισμού που χρησιμοποιεί είναι η Wiring, η οποία είναι αρκετά εύκολη στη σύνταξη και διατίθεται σε πλατφόρμες Linux, MAC και Windows με αδεια χρήσης GPL.

Αυτό όμως που κάνει το Arduino ακόμα πιο σημαντικό είναι ότι όλο το κύκλωμα της πλακέτας διατίθεται με άδεια χρήσης Creative Commons, πράγμα που σημαίνει ότι ο καθένας μπορεί να κατασκευάσει την δική του πλακέτα όπως αυτός θέλει.

Αν και μικροσκοπικό (7x5 cm) οι δυνατότητες που προσφέρει είναι πάρα πολλές. Μπορούμε να το χρησιμοποιήσουμε σε εφαρμογές ρομποτικής και γενικότερα σε αυτοματισμούς καταφέροντας έτσι πάρα πολλά όπως: την κίνηση servo, stepper και DC κινητήρων, τη λήψη πληροφοριών από διάφορους αισθητήρες (θερμοκρασίας, υγρασίας, υπέρυθρων κ.α), την αμφίδρομη σειριακή επικοινωνία μεταξύ Arduino και PC χρησιμοποιώντας γλώσσες προγραμματισμού (όπως Java και python), όπως επίσης την αναπαραγωγή και αντίληψη ήχων.

Το Arduino μέχρι αυτή τη στιγμή διατίθεται σε 12 βασικές παραλλαγές οι οποίες αναφέρονται σε διαφορετικές χρήσεις η κάθε μια, ανάλογα με τις ανάγκες της εφαρμογής μας.

Οι ρυθμίσεις του μεταγλωττιστή avr-gcc βρίσκονται στη καρτέλα Tools->Settings->avr-gcc ενώ του προγραμματιστή avrdude στη καρτέλα Tools->Settings-> avrdude



Το Arduino Diecimila

Χρήση της MySQL βάσης δεδομένων σε λειτουργικό σύστημα Linux

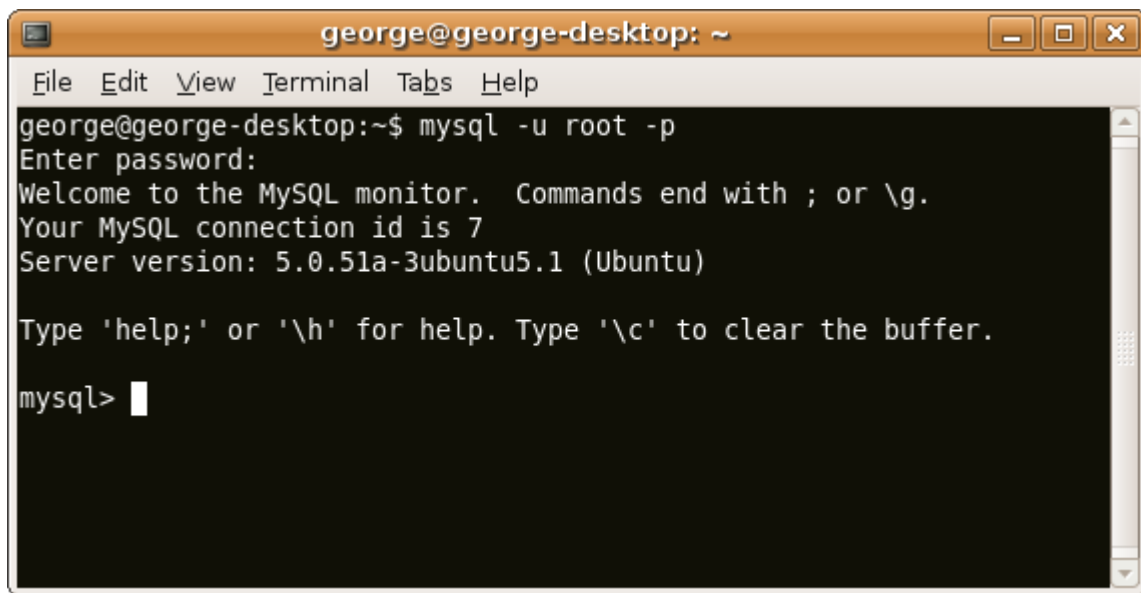
Σε αυτό το κείμενο θα δούμε συνοπτικά την διαχείριση της MySQL βάσης δεδομένων.

Εισαγωγή στην εφαρμογή

Για να γίνει η εισαγωγή ενός χρήστη στη βάση δεδομένων από την γραμμή εντολών του Linux εκτελούμε:

```
mysql -u root -p
```

Η εντολή αυτή θα δώσει το δικαίωμα του χρήστη root να διαχειριστεί τη βάση δεδομένων, αφού δώσει τον κωδικό εισόδου του. Άμεσος μετά αλλάζει η προτροπή σε “mysql>” όπως βλέπουμε στην εικόνα.



```
george@george-desktop: ~  
File Edit View Terminal Tabs Help  
george@george-desktop:~$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 7  
Server version: 5.0.51a-3ubuntu5.1 (Ubuntu)  
  
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.  
  
mysql> |
```

Δημιουργία μιας βάσης δεδομένων

Για την δημιουργία μιας βάσης δεδομένων χρησιμοποιούμε την SQL εντολή

```
CREATE DATABASE <Όνομα Βασης>;
```

Παράδειγμα:

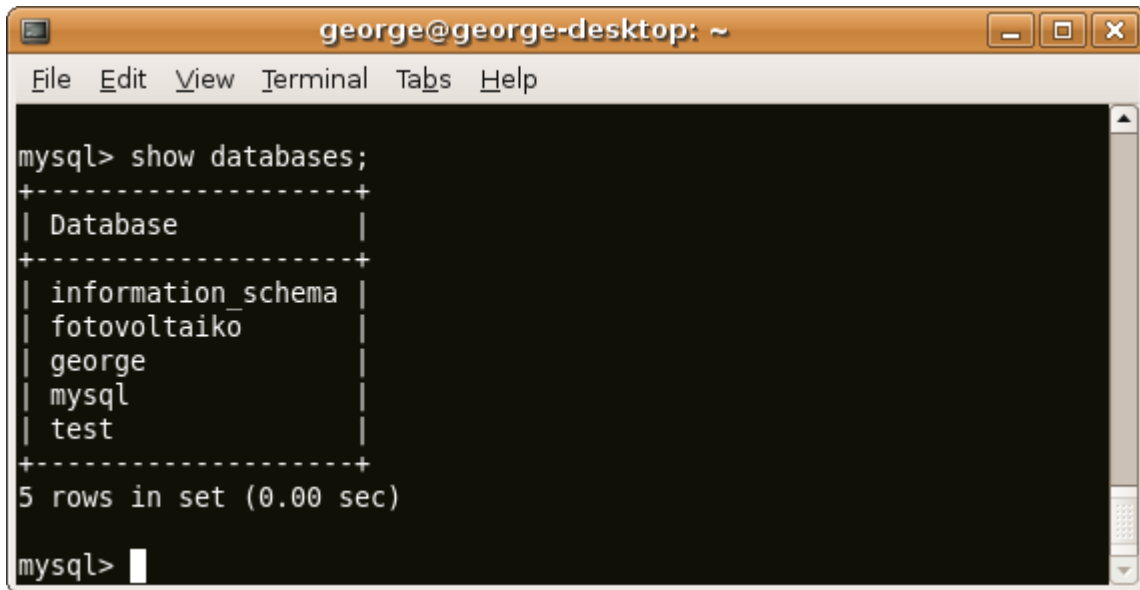
```
CREATE DATABASE test;
```

Η πιο πάνω εντολή δημιουργεί μία βάση δεδομένων με το όνομα “test”. Εδώ να σημειώσουμε ότι κάθε εντολή SQL τελειώνει με ένα ελληνικό ερωτηματικό “;”.

Για να δούμε αν όντως δημιουργήθηκε η βάση δεδομένων μπορούμε να χρησιμοποιήσουμε την εντολή:

```
SHOW DATABASES;
```

Η έξοδος που θα έχουμε θα είναι:



```
george@george-desktop: ~
File Edit View Terminal Tabs Help

mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| fotovoltaiko           |
| george                  |
| mysql                   |
| test                    |
+-----+
5 rows in set (0.00 sec)

mysql>
```

Δημιουργία ενός πίνακα στη βάση δεδομένων

Αφού δημιουργήσαμε την βάση δεδομένων τώρα πρέπει να δηλώσουμε ότι θα τη χρησιμοποιήσουμε οπότε δίνουμε την παρακάτω εντολή:

USE test;

Τότε θα έχουμε σαν έξοδο “Database Changed” ότι άλλαξε η βάση δεδομένων που χρησιμοποιούμε. Τώρα είμαστε έτοιμοι να προσθέσουμε ένα πίνακα στη βάση δεδομένων test. Η εντολή που θα χρησιμοποιήσουμε θα είναι η

CREATE TABLE `όνομα_πίνακα` (`πεδίο_1` <τύπος δεδομένων>, `πεδίο_2` <τύπος δεδομένων> ...);

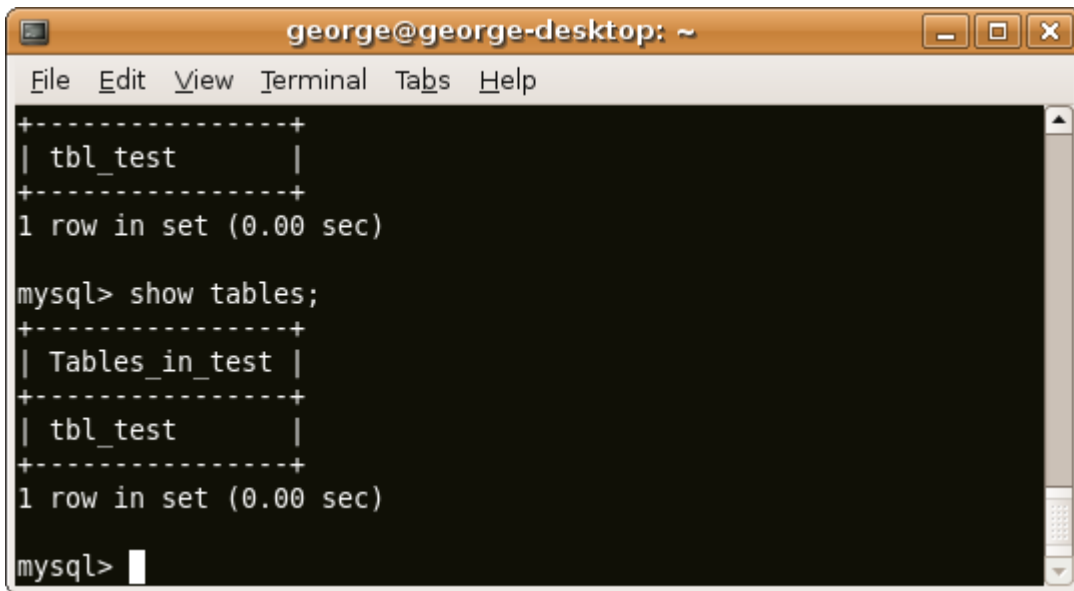
Στο πιο κάτω παράδειγμα θα δημιουργήσουμε ένα πίνακα με τρία πεδία (όνομα, επίθετο, τηλέφωνο):

```
CREATE TABLE `tbl_test`(`name` text NOT NULL,  
                        `surname` text NOT NULL,  
                        `tel` int(20) NOT NULL);
```

Αν όλα πήγαν καλά τότε θα έχουμε έξοδο: “Query OK, 0 rows affected (0.05 sec)”. Για να δούμε αν όντως δημιουργήθηκε ο πίνακας μπορούμε να χρησιμοποιήσουμε την εντολή:

SHOW TABLES;

Η έξοδος που θα έχουμε θα είναι:



```
george@george-desktop: ~
File Edit View Terminal Tabs Help
+-----+
| tbl_test |
+-----+
1 row in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| tbl_test       |
+-----+
1 row in set (0.00 sec)

mysql>
```

Εισαγωγή δεδομένων στο πίνακα

Για την εισαγωγή των δεδομένων στο πίνακα χρησιμοποιούμε την εντολή:

```
INSERT INTO `όνομα_πίνακα` VALUES ('<τιμή_στηλης_1>', '<τιμή_στηλης_2>', '<τιμή_στηλης_3>', ...);
```

Παράδειγμα:

```
INSERT INTO `tbl_test` VALUES ('Giannis', 'Kyriacou', '69566985');
```

Εμφάνιση δεδομένων πίνακα

Αφού κάναμε εισαγωγή κάποιων δεδομένων στο πίνακα `tbl_test` ας δούμε τώρα πως μπορούμε να τα εμφανίσουμε. Θα χρησιμοποιήσουμε την εντολή :

```
SELECT <πεδία> FROM <όνομα_πίνακα> WHERE <συνθήκη>;
```

Με αυτή την εντολή επιλέγουμε τα πεδία που θέλουμε από ένα πίνακα, παράδειγμα:

```
SELECT name, surname, tel FROM tbl_test WHERE 1;
```

Εδώ επιλέξαμε τα πεδία `name`, `surname` και `tel` από τον πίνακα `tbl_test`. Χρησιμοποιώντας το “where 1”, εννοούμε ότι δεν υπάρχει κάποια συνθήκη. Η έξοδος της MySQL θα είναι περίπου όπως την εικόνα (ανάλογα με τα δεδομένα που έχουμε):

```
george@george-desktop: ~
File Edit View Terminal Tabs Help
Query OK, 1 row affected (0.00 sec)

mysql> insert into `tbl_test` values('Andreas','Andreou', '99652140');
Query OK, 1 row affected (0.00 sec)

mysql> insert into `tbl_test` values('Giannis','Andreou', '99652567');
Query OK, 1 row affected (0.00 sec)

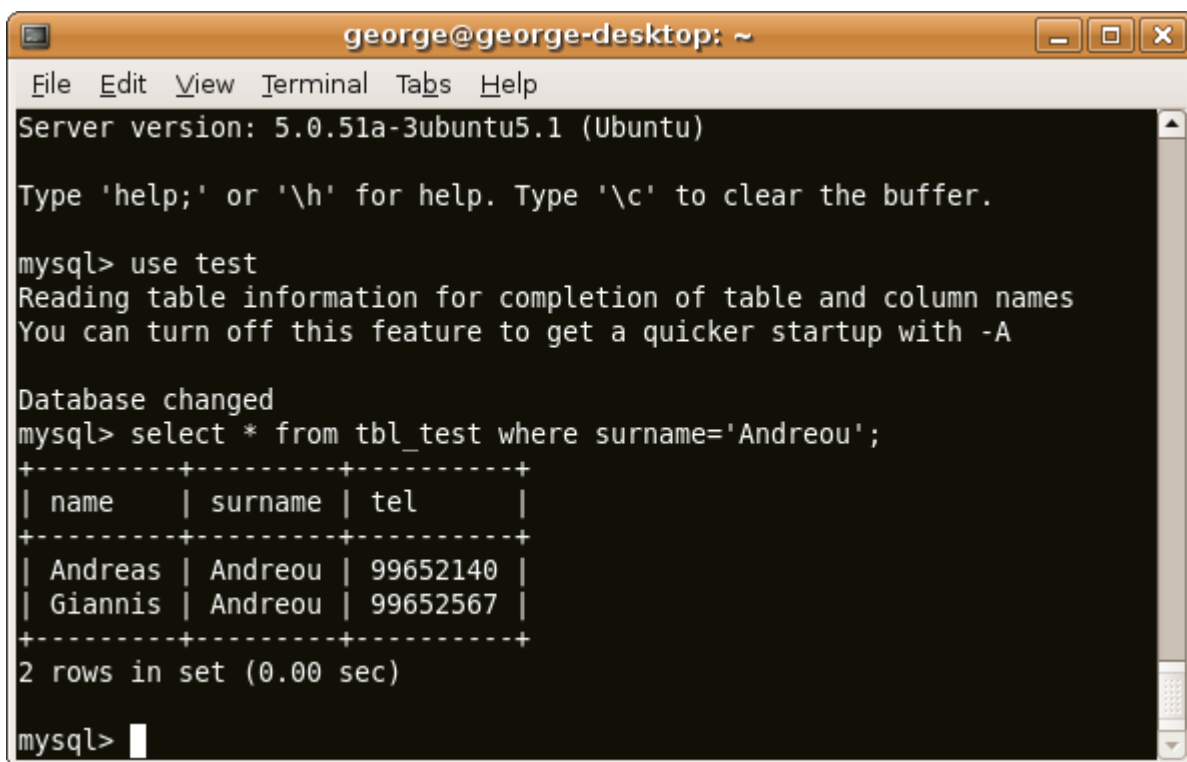
mysql> select name, surname, tel from tbl_test where 1;
+-----+-----+-----+
| name   | surname | tel   |
+-----+-----+-----+
| George | Hadjikyriacou | 99920471 |
| Andreas | Andreou | 99652140 |
| Giannis | Andreou | 99652567 |
+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> █
```

Χρησιμοποιώντας τώρα συνθήκη ας δούμε πόσα ονόματα υπάρχουν με το επίθετο “Andreou” στο πίνακα `tbl_test`. Η εντολή που θα εκτελέσουμε θα είναι:

```
SELECT * FROM tbl_test WHERE surname='Andreou';
```

Το χαρακτήρα “*” τον χρησιμοποιούμε για να επιλέξουμε ΟΛΑ τα πεδία του πίνακα. Η έξοδος της εντολής θα είναι όπως φαίνεται στην παρακάτω εικόνα:



```
george@george-desktop: ~
File Edit View Terminal Tabs Help
Server version: 5.0.51a-3ubuntu5.1 (Ubuntu)
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> use test
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from tbl_test where surname='Andreou';
+-----+-----+-----+
| name   | surname | tel   |
+-----+-----+-----+
| Andreas | Andreou | 99652140 |
| Giannis | Andreou | 99652567 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

Διαγραφή δεδομένων από τον πίνακα

Για τη διαγραφή δεδομένων χρησιμοποιούμε την εντολή :

```
DELETE FROM <όνομα_πίνακα> WHERE <συνθηκη>;
```

Αν για παράδειγμα θέλουμε να διαγράψουμε την γραμμή του πίνακα όπου το όνομα (name) είναι “Giannis” τότε εκτελούμε:

```
DELETE FROM tbl_test WHERE name='Giannis';
```

Αναβάθμιση δεδομένων πίνακα

Αν για οποιοδήποτε λόγο θέλουμε να αλλάξουμε τα δεδομένα του πίνακα τότε αυτό γίνεται με την εντολή :

```
UPDATE <όνομα_πίνακα> SET <πεδίο_αλλαγής>=<τιμή> WHERE <συνθήκη>;
```

Παράδειγμα:

```
UPDATE tbl_test SET tel='99952148' WHERE name='George' and surname='Hadjikyriacou';
```

Η παραπάνω εντολή θα αλλάξει το πεδίο tel του πίνακα όπου το όνομα είναι “George” και το επίθετο “Hadjikyriacou”.

Copyright (C) 2008-2011 George Hadjikyriacou

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.



