



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Εφαρμογή Επαγγελματικού Προσανατολισμού Career and Personality Test
Όνοματεπώνυμο Φοιτητή	Κωνσταντίνος Αττάρτ
Πατρώνυμο	Ανδρέας
Αριθμός Μητρώου	ΜΠΠΛ/ 10009
Επιβλέπων	Μαρία Βίρβου, Καθηγήτρια

Ημερομηνία
Παράδοσης

Οκτώβριος 2014

Πανεπιστήμιο Πειραιώς

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

ΠΕΡΙΛΗΨΗ

Μια από τις σημαντικότερες αποφάσεις που καλείται κάποιος άνθρωπος να πάρει στη ζωή του είναι η επαγγελματική σταδιοδρομία που θα ακολουθήσει. Η απόφαση αυτή συνήθως λαμβάνει χώρα κατά την ολοκλήρωση των σχολικών υποχρεώσεων όπου πρέπει να επιλέξει αν θα σπουδάσει σε κάποια συγκεκριμένη σχολή ή θα βγει κατευθείαν στην αγορά εργασίας. Η στιγμή αυτή συμπίπτει περίπου με την ενηλικίωση του ατόμου και πολλές φορές δεν είναι έτοιμο να πάρει τέτοια απόφαση.

Το επάγγελμα που θα επιλέξει κάθε άτομο θα καθορίσει σε μεγάλο βαθμό το μέλλον είτε αυτό αφορά τον οικονομικό τομέα, είτε ακόμα και την ψυχική του ισορροπία και ευχαρίστηση. Η επιλογή του οπότε, εκτός από τις εντυπώσεις και τα ερεθίσματα που έχει λάβει από το περιβάλλον του και από τις εκάστοτε συνθήκες της αγοράς εργασίας, θα πρέπει να συνάδει και με την προσωπικότητα του.

Πολύ σημαντικό ρόλο στην απόφαση του ατόμου θα παίξει το οικογενειακό και σχολικό περιβάλλον του, το οποίο εκτός από τη συμμετοχή στη διαμόρφωση του χαρακτήρα του θα πρέπει να παρέχει και τις απαραίτητες πληροφορίες ώστε να έχει το άτομο την κατάλληλη εικόνα για να κάνει τις επιλογές του. Η έγκυρη ενημέρωση για τις συνθήκες που επικρατούν στην αγορά εργασίας μπορεί να βοηθήσει το άτομο να αποφύγει λανθασμένες επιλογές και μελλοντικά επαγγελματικά αδιέξοδα.

Ο ενδεδειγμένος επαγγελματικός προσανατολισμός θα πρέπει να καλύπτει τρία βασικά στάδια: α) την πληροφόρηση, β) την χρήση ψυχομετρικών εργαλείων (τεστ) και γ) την συμβουλευτική διαδικασία. Και τα τρία αυτά στάδια είναι εξίσου σημαντικά ώστε να οδηγηθεί το κάθε άτομο στην πιο κατάλληλη επιλογή σχολής ή επαγγέλματος.

ABSTRACT

One of the most important decisions a person must get in his life is what career should follow. This decision usually takes place at the conclusion of school commitments when the person must choose whether to study in a particular college or go directly into the job market. This moment coincides approximately with the aging of the individual, who is often not ready to take such a decision.

The profession chosen by each individual will determine in a large scale his future not only about the economic sphere, but his mental balance and pleasure too. This choice, apart from the impressions and stimuli received from his environment and the actual conditions of the job market, should be consistent with his personality.

Family and school environment should play a very important role in every person's decision which apart from being involved in shaping his character should provide the necessary information and that the person has the appropriate image to make choices. The accurate information about the conditions prevailing in the job market may help the individual to avoid bad choices and future business deadlocks.

The appropriate professional guidance should cover three main stages: a) information, b) the use of psychometric tools (tests) and c) the counseling process. These three stages are equally important in order to lead each person to the most appropriate choice of school or profession.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ.....	3
ABSTRACT	3
ΠΕΡΙΕΧΟΜΕΝΑ	4
1.ΕΙΣΑΓΩΓΗ – ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ.....	6
1.1 Εισαγωγή.....	6
1.2 Ανθρώπινη προσωπικότητα.....	6
1.3 Θεωρία του Holland.....	6
1.4 Οι έξι τύποι προσωπικότητας του Holland	7
1.5 Το εξάγωνο του Holland.....	9
1.6 Εφαρμογή Επαγγελματικού Προσανατολισμού	10
2. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ	11
2.1 Η θεωρία του Donald E. Super.....	11
2.2 Η θεωρία του John Krumboltz	12
2.3 Εφαρμογές Επαγγελματικού Προσανατολισμού σύμφωνα με τη θεωρία του Holland	14
3. ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΧΡΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	16
Φόρμες Εφαρμογής.....	16
ΦΟΡΜΑ 1	16
ΦΟΡΜΑ 2	19
ΦΟΡΜΑ 3	20
ΦΟΡΜΑ 4	21
ΦΟΡΜΑ 5	23
ΦΟΡΜΑ 6	25
ΦΟΡΜΑ 7	27
ΦΟΡΜΑ 8	31
ΦΟΡΜΑ 9	33
4. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ	35
4.1 Το πλαίσιο λογισμικού .NET Framework	35
4.2 Το περιβάλλον SQL SERVER.....	35
4.3 Το Visual Studio 2010	36
4.4 Η γλώσσα προγραμματισμού C#.....	36
4.5 Η εφαρμογή Rational Rose	36
4.6 Διαγράμματα UML.....	37
4.7 Τμήματα Κώδικα Εφαρμογής	50
5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	75
5.1 Συμπεράσματα	75
5.2 Μελλοντικές Επεκτάσεις της Εφαρμογής	76
ΒΙΒΛΙΟΓΡΑΦΙΑ	77

ΕΙΚΟΝΕΣ

Εικόνα 1.1 Το εξάγωνο του Holland	9
Εικόνα 2.1 Το ουράνιο τόξο Ζωής-Καριέρας του Super.....	12
Εικόνα 2.2 Τεστ ΕΟΠΠΕΠ.....	14
Εικόνα 2.3 Αποτελέσματα τεστ ΕΟΠΠΕΠ.....	14
Εικόνα 2.4 Τεστ HORIZON.....	15
Εικόνα 3.1 Φόρμα “message.show”.....	16
Εικόνα 3.2 Φόρμα “frm_firsttime”(1).....	17
Εικόνα 3.3 Φόρμα “frm_firsttime” (2)	18
Εικόνα 3.4 Φόρμα “frm_firsttimePUK”(1).....	19
Εικόνα 3.5 Φόρμα “frm_firsttimePUK” (2)	19
Εικόνα 3.6 Φόρμα “ frm_login”	20
Εικόνα 3.7 Φόρμα “frm_menou”(1).....	21
Εικόνα 3.8 Φόρμα “frm_menou”(2).....	21
Εικόνα 3.9 Φόρμα “frm_menou”(3).....	22
Εικόνα 3.10 Φόρμα “frm_new_item”(1).....	23
Εικόνα 3.11 Φόρμα “frm_new_item”(2).....	24
Εικόνα 3.12 Φόρμα “frm_exist_item”(1).....	25
Εικόνα 3.13 Φόρμα “frm_exist_item”(2).....	25
Εικόνα 3.14 Φόρμα “frm_answer”(1).....	27
Εικόνα 3.15 Φόρμα “frm_answer”(2).....	28
Εικόνα 3.16 Φόρμα “frm_answer”(3).....	29
Εικόνα 3.17 Φόρμα “frm_answer”(4).....	30
Εικόνα 3.18 Φόρμα “message.show”(1).....	31
Εικόνα 3.19 Φόρμα “message.show”(2).....	32
Εικόνα 3.20 Φόρμα “message.show”(3).....	32
Εικόνα 3.21 Φόρμα “ frm_edit_users ”(1).....	33
Εικόνα 3.22 Φόρμα “ frm_edit_users ”(2).....	33
Εικόνα 3.23 Φόρμα “ frm_edit_users ”(3).....	34
Εικόνα 4.1 Διάγραμμα περιπτώσεων χρήσης (Διαχείριση Χρηστών).....	38
Εικόνα 4.2 Διάγραμμα περιπτώσεων χρήσης (Τεστ).....	38
Εικόνα 4.3 Διάγραμμα περιπτώσεων χρήσης (Ολοκληρωμένη Εφαρμογή).....	39
Εικόνα 4.4 Διάγραμμα κλάσης (1 ^η έκδοση).....	40
Εικόνα 4.5 Διάγραμμα κλάσης (2 ^η έκδοση).....	41
Εικόνα 4.6 Διάγραμμα ακολουθίας (Master Admin-User-Database).....	42
Εικόνα 4.7 Διάγραμμα ακολουθίας (User-Database).....	43
Εικόνα 4.8 Διάγραμμα ακολουθίας (Admin-Database).....	44
Εικόνα 4.9 Διάγραμμα ακολουθίας (Ολοκληρωμένη Εφαρμογή).....	45
Εικόνα 4.10 Διάγραμμα καταστάσεων.....	46
Εικόνα 4.11 Διάγραμμα εξαρτημάτων.....	47
Εικόνα 4.12 Διάγραμμα ανάπτυξης.....	48
Εικόνα 4.13 Διάγραμμα δραστηριοτήτων.....	49

1.ΕΙΣΑΓΩΓΗ – ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

1.1 Εισαγωγή

Στην σημερινή εποχή είναι σημαντικό το κάθε άτομο να συνειδητοποιήσει τα ιδιαίτερα χαρακτηριστικά που το διακρίνουν, το χαρακτήρα και την ιδιοσυγκρασία του, την προσωπικότητά του, τις κλίσεις και τα ενδιαφέροντα του, τις ικανότητες και τις δεξιότητες που κατέχει, ώστε μέσα από αυτά να οδηγηθεί στη σωστή λήψη αποφάσεων.

Πολλά επαγγέλματα απαιτούν συνδυασμό προσωπικότητας, ικανοτήτων και δεξιοτήτων και γι' αυτό πρέπει με συνειδητή σκέψη ο καθένας να διαπιστώνει σε ποιον τομέα ανταποκρίνεται περισσότερο, δεδομένου ότι η επιλογή της επαγγελματικής σταδιοδρομίας είναι μια καθοριστική στιγμή της ζωής.

Λόγω του σημαντικού ρόλου που παίζει η απόφαση του κάθε ατόμου για την επαγγελματική του σταδιοδρομία, η Επιστήμη της Ψυχολογίας έχει αναπτύξει αρκετές θεωρίες σχετικά.

Το πρόβλημα που ενδιαφέρει τους ψυχολόγους μπορεί να αναλυθεί σε τρία κυρίως θέματα:

- Στον προσδιορισμό των κοινών στοιχείων που αφορούν όσο το δυνατόν περισσότερα επαγγέλματα.
- Στη δημιουργία μεθόδων εκτίμησης για την οριοθέτηση των διαφόρων επαγγελμάτων και τη συγκριτική μελέτη διαφόρων ατόμων εντός και εκτός καθενός από τα επαγγέλματα αυτά.
- Στο σχεδιασμό των διαδικασιών εκείνων που θα μπορούσαν να διαγνώσουν ένα σύνολο χαρακτηριστικών για κάθε άτομο, με σκοπό τον προσδιορισμό των επαγγελματικών του προτιμήσεων.

1.2 Ανθρώπινη προσωπικότητα

Κύριο χαρακτηριστικό της ανθρώπινης κοινωνίας είναι ότι αποτελείται από διαφορετικούς τύπους ατόμων που έχουν πολλές ομοιότητες αλλά και διαφορές στην εμφάνιση, το χαρακτήρα, τα ενδιαφέροντα, τις κλίσεις, τις προτιμήσεις, τις ικανότητες, τις δεξιότητες, τα ταλέντα. Κάθε άτομο διακρίνεται για ορισμένα χαρακτηριστικά της προσωπικότητάς του, που λείπουν από κάποιο άλλο και το αντίθετο.

Όπως δεν υπάρχουν δύο τύποι όμοιων δακτυλικών αποτυπωμάτων, έτσι και κάθε άνθρωπος έχει τη δική του ξεχωριστή προσωπικότητα. Δεν είναι όλοι γεννημένοι για τα ίδια πράγματα όσο και αν παρόμοιες εργασίες μπορούν να εκτελεστούν από διάφορα άτομα. Αυτή η ιδιαιτερότητα του καθενός κάνει τη ζωή των κοινωνιών πιο γόνιμη, δημιουργική και ενδιαφέρουσα. Μια κοινωνία όπου οι άνθρωποι θα ήταν σε όλα ίδιοι, δε θα είχε λόγο ύπαρξης. Και το αντίθετο, μια κοινωνία όπου οι άνθρωποι θα διέφεραν σε όλα μεταξύ τους, θα είχε προβλήματα συνοχής και δε θα επιβίωνε.

Η διεθνής επιστημονική βιβλιογραφία του κλάδου της συμβουλευτικής καριέρας, έχει εμπλουτιστεί με αρκετές θεωρίες που επιχειρούν να συνδυάσουν επιμέρους προσωπικότητες, προτιμήσεις, κλίσεις, ενδιαφέροντα, ταλέντα, ικανότητες και δεξιότητες ανθρώπων με διάφορα επαγγέλματα. Η πλέον διαδεδομένη και αποδεκτή διεθνώς θεωρία είναι αυτή του John L. Holland.

1.3 Θεωρία του Holland

Η θεωρία του Holland ανήκει στις θεωρίες της προσωπικότητας και πιο συγκεκριμένα της σύνδεσης ανθρώπου και περιβάλλοντος, και αποδίδει την επαγγελματική επιλογή στον τύπο προσωπικότητας που έχει αναπτύξει το άτομο στη διαλεκτική του σχέση με το περιβάλλον του. Παρόλο που δεν παραβλέπει την κληρονομικότητα, το περιβάλλον, καθώς και τον τρόπο

που αυτές οι παράμετροι επηρεάζουν τις επαγγελματικές επιλογές, η θεωρία του Holland επικεντρώνεται στους παράγοντες που επηρεάζουν τις επιλογές σε ένα δεδομένο χρονικό σημείο. Συγκεκριμένα, τη στιγμή της επαγγελματικής επιλογής το άτομο έχει ένα έτοιμο ρεπερτόριο συμπεριφορών και προσπαθεί να ταιριάζει τα χαρακτηριστικά αυτά με το κατάλληλο από ψυχολογική άποψη περιβάλλον, συμβάλλοντας σημαντικά στη δυνατότητα να επιτύχει ικανοποίηση, παραμονή και συνεισφορά στο περιβάλλον αυτό.

Βασικές αρχές στη θεωρία του Holland είναι οι παρακάτω: Η εκλογή επαγγέλματος είναι έκφραση της προσωπικότητας. Τα τεστ ενδιαφερόντων είναι στην ουσία τεστ προσωπικότητας. Τα επαγγελματικά στερεότυπα προσφέρονται για αξιόπιστες ψυχολογικές και κοινωνικές ερμηνείες. Τα άτομα που ασκούν ένα επάγγελμα έχουν όμοιες προσωπικότητες και παρόμοια προσωπική εξέλιξη. Δεδομένου ότι οι άνθρωποι σε ένα επαγγελματικό περιβάλλον έχουν όμοιες προσωπικότητες, είναι επόμενο να αντιδρούν σε πολλές καταστάσεις και σε πολλά προβλήματα κατά παρόμοιο τρόπο. Η επαγγελματική ικανοποίηση, σταθερότητα και επιτυχία εξαρτώνται από το βαθμό αρμονικότητας μεταξύ της προσωπικότητας του ατόμου και του επαγγελματικού του περιβάλλοντος.

Ο Holland διέκρινε έξι τύπους προσωπικότητας: α) Ρεαλιστικός (realistic), ο οποίος προτιμά τα εξειδικευμένα και τεχνικά επαγγέλματα. β) Ερευνητικός (investigative), ο οποίος προτιμά δραστηριότητες που ανήκουν στο χώρο των φυσικών, βιολογικών και κοινωνικών επιστημών και συνεπάγονται παρατήρηση και έρευνα. γ) Καλλιτεχνικός (artistic), που προτιμά ασαφείς, ελεύθερες και μη συστηματικές δραστηριότητες. δ) Κοινωνικός (social) που προτιμά τις δραστηριότητες που συνεπάγονται σχέσεις με άλλους ανθρώπους, όπως είναι η διδασκαλία, η παροχή βοήθειας, συμβουλών και κοινωνικής εργασίας και αποφεύγουν τις πρακτικές ενασχολήσεις. ε) Επιχειρηματικός (enterprising), που διαθέτει ηγετικές και επωποτικές ικανότητες, δυνατότητες πειθούς, φιλοδοξία και κοινωνικότητα και απεχθάνονται τις επιστημονικές δραστηριότητες. στ) Συμβατικός (conventional), που προτιμά δραστηριότητες που χαρακτηρίζονται από ακρίβεια, τάξη και συστηματικότητα.

Βασικές έννοιες που επίσης εισάγει ο Holland αποτελούν η συνέπεια, δηλαδή ο βαθμός συγγένειας μεταξύ τύπων και περιβαλλόντων, η διαφοροποίηση, δηλαδή ο βαθμός που το άτομο ή το περιβάλλον είναι σαφώς καθορισμένα, η ταυτότητα, δηλαδή ο βαθμός καθαρότητας και σταθερότητας των στόχων και των ενδιαφερόντων ή των καθηκόντων και των αμοιβών του περιβάλλοντος, τη σύμπτωση, αφού κάθε τύπος προσωπικότητας βρίσκεται σε σύμπτωση με ένα περιβάλλον, και τέλος τον λογισμό, ο οποίος αναφέρεται στην εξαγωγική διάταξη των τύπων και των περιβαλλόντων.

Με βάση τη θεωρία ανάπτυξης σταδιοδρομίας του Holland, πρωταρχικός στόχος του σύμβουλου είναι να καθορίσει τον τωρινό κώδικα του συμβουλευόμενου και τα χαρακτηριστικά του, δηλαδή την ταυτότητα, τη σύμπτωση, τη διαφοροποίηση και την συνέπεια που του αντιστοιχεί. Για το λόγο αυτό, για άτομα που φαίνονται να έχουν δυσκολίες αναφορικά με τη διερεύνηση και τη λήψη αποφάσεων, όπως παράλογες πεποιθήσεις, χαμηλό επίπεδο αυτοαποτελεσματικότητας, χαμηλό επίπεδο αυτοαντίληψης ή αναποτελεσματικές μεθόδους λήψης αποφάσεων, δεν ενδείκνυται η συμβουλευτική παρέμβαση που στηρίζεται στη θεωρία του Holland. Τίθεται επίσης, το ερώτημα, ποια από τα πολλαπλά ενδιαφέροντα και ίσως πολλαπλά ταλέντα που διαθέτει το άτομο, θα αξιοποιήσει στην αμειβόμενη απασχόλησή του και με ποια θα ασχοληθεί στους άλλους ρόλους της ζωής του. Για να κάνει αυτήν τη διάκριση, θα πρέπει να ενεργοποιήσει τις προσωπικές του αξίες. Ωστόσο, δε λαμβάνεται υπόψη ότι το άτομο αλλάζει όπως και το περιβάλλον του, και επιπλέον η θεωρία δεν ερμηνεύει επαρκώς πώς ακριβώς επιτελείται η διαμόρφωση των τύπων της προσωπικότητας.

1.4 Οι έξι τύποι προσωπικότητας του Holland

Οι έξι τύποι προσωπικότητας του Holland είναι:

Πρακτικός ή Ρεαλιστικός (Realistic)

Τα άτομα που κατηγοριοποιούνται σε αυτό τον τύπο είναι πρακτικοί και ρεαλιστές, απόλυτοι και δογματικοί, εσωστρεφείς, εύρωστοι και επίμονοι, στις ενέργειες τους είναι αποτελεσματικοί μεθοδικοί, προσγειωμένοι και συνήθως ευθείς και ειλικρινείς. Διαθέτουν χειρωνακτικές, τεχνικές και μηχανολογικές ικανότητες, χειρίζονται με επιδεξιότητα εργαλεία, μηχανικά συστήματα και ηλεκτρονικό εξοπλισμό και συνήθως στρέφονται σε τεχνολογικά, μηχανολογικά και κατασκευαστικά επαγγέλματα.

Το υπερκείμενο εργασιακό περιβάλλον τους προσανατολίζει σε εργασίες με πρακτικό, τεχνικό και συχνά υπαίθριο προσανατολισμό, με άμεσα ορατό αποτέλεσμα, καθώς και στην ανάπτυξη παραδοσιακών αξιών που έχουν σχέση με την εξουσία και τις οικονομικές απολαβές. Επαγγέλματα που ταιριάζουν στον πρακτικό τύπο είναι, ο πολιτικός μηχανικός, ο μηχανολόγος μηχανικός, ο χημικός μηχανικός, ο γεωλόγος, ο γεωπόνος, ο ιχθυολόγος, ο δασολόγος κλπ.

Ερευνητικός (Investigative)

Στον ερευνητικό τύπο ανήκουν άτομα με ανήσυχη, ευρηματική, δημιουργική και κριτική σκέψη, με ιδιαίτερη κλίση στις επιστήμες και την έρευνα. Είναι ανεξάρτητοι, πολυσύνθετοι και διανοούμενοι σαν χαρακτήρες, παρατηρητικοί, έως και περίεργοι. Διαθέτουν αναλυτικές, συνθετικές ικανότητες και διακατέχονται από ριζοσπαστική, κριτική και επικριτική διάθεση, είναι εσωστρεφείς, μάλλον απαισιόδοξοι, επιφυλακτικοί και μετριοφρονες.

Το υπερκείμενο εργασιακό περιβάλλον τους συνδέει με δραστηριότητες που συνεπάγονται τη συστηματική και καινοτόμο έρευνα, την κατανόηση και τον έλεγχο των φυσικών, βιολογικών, πολιτιστικών και κοινωνικών φαινομένων.

Ο ερευνητικός τύπος θα μπορούσε να διαχωριστεί σε δύο κατηγορίες: α) τον ερευνητικό-θετικό τύπο, με ιδιαίτερες προτιμήσεις και δεξιότητες προς τα μαθηματικά, τη φυσικοχημεία, την τεχνολογία και τα συναφή επαγγέλματα και β) τον ερευνητικό-γραμματικό τύπο, με ροπή προς τις γλωσσικές και γραμματικές δεξιότητες, τις κοινωνικές και ανθρωπιστικές επιστήμες και τα αντίστοιχα επαγγέλματα.

Καλλιτεχνικός (Artistic)

Οι καλλιτεχνικοί τύποι είναι πολύπλοκοι χαρακτήρες, ευαίσθητοι, ατίθασοι και αφηρημένοι, χωρίς τάξη, δυσπροσάρμοστοι και συναισθηματικοί. Είναι εκδηλωτικοί, παρορμητικοί και ευφάνταστοι, συχνά αναποτελεσματικοί αλλά και αυθεντικοί, ιδεαλιστές και ρομαντικοί. Προσανατολίζονται σε ελεύθερες, αφηρημένες και μη συστηματικές δραστηριότητες και προφανώς ενδιαφέρονται και κλίνουν προς τα καλλιτεχνικά επαγγέλματα.

Το υπερκείμενο εργασιακό περιβάλλον τους ενθαρρύνει να θεωρούν τους εαυτούς τους αντικομφορμιστές και ανεξάρτητους χαρακτήρες. Η ιδιαίτερη κλίση, το φυσικό ταλέντο του καθενός προς κάποια τέχνη, προς τη μουσική, τη ζωγραφική, το χορό, το θέατρο κλπ., αποτελεί κύριο συστατικό του τύπου αυτού και είναι πρωταρχικής σημασίας για μια αξιοπρεπή καριέρα στη δύσβατη, όσο και ενδιαφέρουσα διαδρομή των καλλιτεχνικών επαγγελμάτων.

Κοινωνικός (Social)

Όσοι ανήκουν στο κοινωνικό τύπο, είναι κατά κύριο λόγο φιλικοί, συνεργάσιμοι, ευγενικοί, εξωστρεφείς, ευχάριστοι, θερμοί και ζωντανοί χαρακτήρες, συνήθως υπομονετικοί, αλληλέγγυοι, εξυπηρετικοί και συγκαταβατικοί. Αναπτύσσουν καλές σχέσεις με τρίτους, διαθέτουν πειθώ, παρέχουν τη βοήθειά τους σε όσους τη χρειάζονται και εμπλέκονται σε δραστηριότητες που συνεπάγονται τις δημόσιες σχέσεις, την επικοινωνία, την πληροφόρηση, την εκπαίδευση, τη θεραπεία, την αλληλεγγύη και την πρόνοια.

Το υπερκείμενο εργασιακό τους περιβάλλον τους οδηγεί στο να αναπτύσσουν δεξιότητες επικοινωνίας, κατανόησης και συνεργασίας.

Οι κοινωνικοί τύποι θα μπορούσαν να διαχωριστούν σε δύο κατηγορίες: α) τους κοινωνικούς-εξωστρεφείς και τους κοινωνικούς-αλληλέγγυους. Οι πρώτοι χαρακτηρίζονται από δεξιότητες δημοσίων σχέσεων, ικανότητες πωλήσεων και προσωπικής ανέλιξης, η κοινωνικότητα τους δηλαδή συνδυάζεται με σκοπούς, στόχους και προσωπικές φιλοδοξίες. Συναφή επαγγέλματα είναι του στελέχους δημοσίων σχέσεων, του πωλητή, του δημοσιογράφου.

Οι δεύτεροι χαρακτηρίζονται περισσότερο από «αλληλέγγυα κοινωνικότητα», από ιδεαλισμό, συναισθηματισμό, γενναιοδωρία, ανθρωπισμό ή ακόμη και αυτοθυσία και ταιριάζουν περισσότερο σε επαγγέλματα εκπαίδευσης, πρόνοιας, υγείας, περίθαλψης, ανθρωπίνης συνεργασίας και αλληλεγγύης γενικότερα.

Επιχειρηματικός (Enterprising)

Στον επιχειρηματικό τύπο υπάγονται άτομα που διαθέτουν ηγετικές ικανότητες, άνθρωποι που είναι φιλόδοξοι, κυριαρχικοί, ανταγωνιστικοί, ενεργητικοί, κοινωνικοί, αισιόδοξοι, αποφασιστικοί και δημιουργικοί. Επιπλέον μπορεί να χαρακτηριστούν ως επίμονοι, απόλυτοι, ανήσυχοι, δραστήριοι, ενθουσιώδεις, εξωστρεφείς, ριψοκίνδυνοι, επινοητικοί, δυναμικοί, γενναίοι αλλά και συχνά εγωκεντρικοί, επιθετικοί και αυταρχικοί.

Το υπερκείμενο εργασιακό περιβάλλον τους οδηγεί στην ανάπτυξη επιχειρημάτων αλλά και μεθόδων πειθούς και χειραγώγησης τρίτων, ενώ θεωρούν τους εαυτούς τους δημοφιλείς και γεμάτους αυτοπεποίθηση. Συχνά κρίνουν τον κόσμο απλουστευμένα και συμβατικά, με βάση το κοινωνικό κύρος και την εξουσία.

Οι επιχειρηματικοί τύποι ταιριάζουν με διευθυντικές και ηγετικές θέσεις και επαγγέλματα, όπου ενυπάρχει οικονομική διάσταση, επιχειρηματικότητα και δυνατότητες ιεραρχικής εξέλιξης.

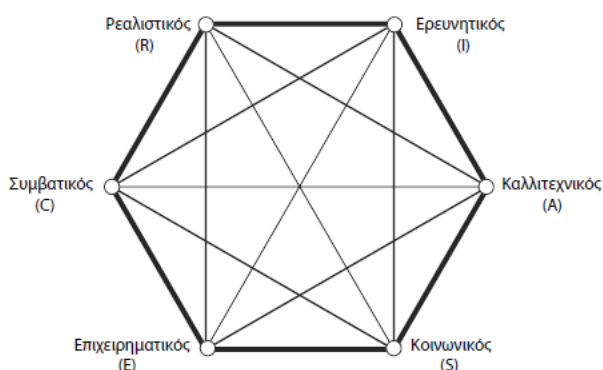
Συμβατικός-Οργανωτικός (Conventional)

Οι συμβατικοί τύποι αποδέχονται τον καθιερωμένο τρόπο ζωής, τις αποδεκτές αξίες της κοινωνίας και την υφιστάμενη ιεραρχία εξουσίας, είναι αφοσιωμένοι, ευσυνείδητοι, με ανεπτυγμένο ιδιαίτερα το αίσθημα της οικονομίας. Είναι επιμελείς, πειθαρχημένοι, ευπροσάρμοστοι, προσεκτικοί, τακτικοί, πρακτικοί, ακριβείς, ψύχραιμοι, συνεργάσιμοι, αξιόπιστοι, συχνά δογματικοί και εσωστρεφείς, μεθοδικοί και οργανωτικοί. Τα άτομα που υπάγονται στον τύπο αυτό, προσανατολίζονται σε δραστηριότητες που χαρακτηρίζονται από τάξη, ακρίβεια και μεθοδικότητα.

Το υπερκείμενο εργασιακό περιβάλλον τους οδηγεί σε καθιερωμένες επαναλαμβανόμενες εργασίες που απαιτούν πειθαρχία, υπολογιστικές, λογιστικές, υπαλληλικές και επιχειρηματικές ικανότητες. Ταιριάζουν με επαγγέλματα όπως ο στρατιωτικός, ο αστυνομικός, ο δημόσιος ή ο διπλωματικός υπάλληλος, το στέλεχος επιχειρήσεων, ο βιβλιοθηκονόμος, ο συμβολαιογράφος, ο λογιστής κλπ.

1.5 Το εξάγωνο του Holland

Για την εύρεση της συγγένειας των διαφόρων τύπων, ο Holland έχει αναπτύξει ένα σχήμα εξαγώνου, γνωστό και ως "Holland's hexagon", σύμφωνα με το οποίο οι ομοιότητες και οι διαφορές ανάμεσα στους έξι τύπους προσωπικότητας απεικονίζονται με το ακόλουθο σχήμα:



Εικόνα 1.1 Το εξάγωνο του Holland

Οι τύποι προσωπικότητας που είναι πιο κοντά μεταξύ τους παρουσιάζουν μεγαλύτερες ομοιότητες. Ο ρεαλιστικός και ο ερευνητικός τύπος π.χ., όπως εμφανίζονται στο σχήμα, συνήθως εμφανίζουν περισσότερα κοινά ενδιαφέροντα.

Αντίθετα, ο ρεαλιστικός και ο κοινωνικός έχουν μεγαλύτερες διαφορές. Ο συμβατικός τύπος έχει συνήθως περισσότερα κοινά ενδιαφέροντα με τον επιχειρηματικό και τον ρεαλιστικό και λιγότερα με τον κοινωνικό, τον ερευνητικό και τον καλλιτεχνικό.

Τα περισσότερα άτομα είναι συνήθως πολυσύνθετοι χαρακτήρες που εμπεριέχουν επιμέρους στοιχεία από τους διάφορους τύπους προσωπικότητας, με περισσότερες ή μικρότερες αναλογίες από τον καθένα.

Η διασπορά και το κατά περίπτωση εύρος του ιστογράμματος με τους τύπους προσωπικότητας, όπως εμφανίζονται στην εξατομικευμένη προσωπική έκθεση, μπορεί να οδηγήσουν σε κατάλληλα συμπεράσματα για το σύμβουλο επαγγελματικού προσανατολισμού.

Στη θεωρία του Holland, αλλά και σε διάφορες άλλες επιστημονικές αναλύσεις έχουν στηριχτεί πολυάριθμες πρακτικές εφαρμογές, με τις οποίες διερευνάται η ανθρώπινη προσωπικότητα και επιχειρείται η συσχέτιση της με διάφορα επαγγέλματα και κατ' επέκταση με συγγενείς σπουδές και καριέρα.

1.6 Εφαρμογή Επαγγελματικού Προσανατολισμού

Η εφαρμογή που παρουσιάζεται στην μεταπτυχιακή αυτή διατριβή, βασίζεται στη θεωρία του Holland για τους έξι τύπους προσωπικότητας. Αποτελείται από 200 ερωτήσεις που καλείτε να απαντήσει ο εκάστοτε χρήστης, βαθμολογώντας με 5 επιλογές από το «καθόλου» ως το «σίγουρα ναι». Στη συνέχεια αθροίζει όλες τις βαθμολογίες και εμφανίζει το ποσοστό που προκύπτει σε κάθε τύπο προσωπικότητας. Επιλέγοντας τον τύπο προσωπικότητας με το μεγαλύτερο ποσοστό εμφανίζει την εκάστοτε ανάλυση και τα πιο συμβατά επαγγέλματα.

Οι ερωτήσεις έχουν επιλεγεί με τέτοιο τρόπο ώστε οι συνολικές εφικτές βαθμολογίες τους να είναι ισοδύναμες για όλους τους τύπους προσωπικότητας.

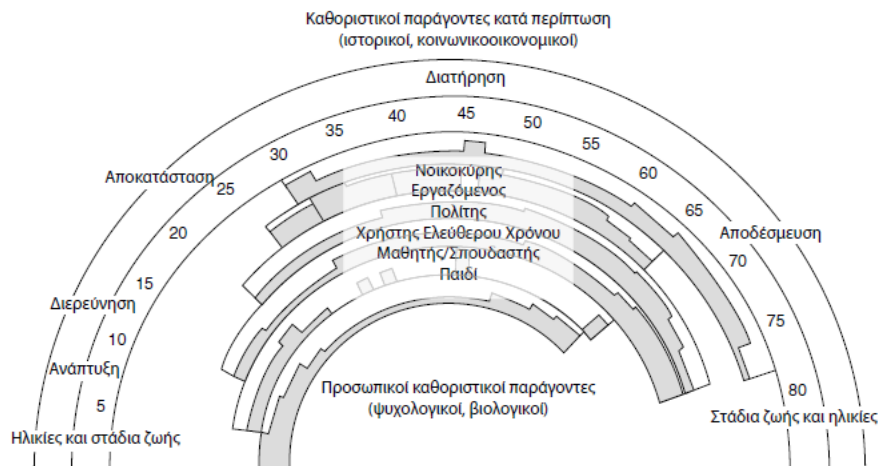
2. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ

Εκτός από τη θεωρία του Holland πολλοί ακόμα ψυχολόγοι ασχολήθηκαν με το θέμα του επαγγελματικού προσανατολισμού. Παρακάτω αναφέρονται συνοπτικά κάποιες από τις σημαντικότερες θεωρίες.

2.1 Η θεωρία του Donald E. Super

Το έργο του Super ασχολείται τόσο με τη διάρκεια όσο και με την έκταση της επαγγελματικής ανάπτυξης. Η έρευνα του εντοπίζει διαδοχικά στάδια στη ζωή και αναπτυξιακές δραστηριότητες που αφορούν την επαγγελματική ανάπτυξη και εκτείνονται από τη γέννηση ως το θάνατο του ατόμου. Η θεωρία του περιλαμβάνει το Ουράνιο Τόξο Ζωής-Καριέρας (Life Career Rainbow), το οποίο περιέχει δραστηριότητες σε όλους τους αλληλεπιδρώντες ρόλους της ζωής και εντάσσεται στα πλαίσια του ορισμού που δίνει για την σταδιοδρομία. Τα πιο καίρια σημεία της θεωρίας του Super είναι τα παρακάτω:

- Η σταδιοδρομία ενός ατόμου επηρεάζεται και ρυθμίζεται από την αυτοαντίληψή του, η οποία αρχίζει να διαμορφώνεται τη στιγμή που ένα μωρό διαχωρίζει τον εαυτό του από άλλα αντικείμενα και ανθρώπους και συνεχίζει να αναπτύσσεται σε όλη τη διάρκεια της ζωής του. Η επιλογή ενός επαγγέλματος αποτελεί την εφαρμογή της αυτοαντίληψης. Η αυτοαντίληψη ενός ατόμου μπορεί να έχει πολλά διαφορετικά χαρακτηριστικά όπως σαφήνεια, ρεαλισμό, αυτοεκτίμηση και λεπτομέρεια.
- Η ανάπτυξη σταδιοδρομίας είναι μια διά βίου διαδικασία που εκτυλίσσεται σε πέντε διαφορετικά στάδια της ζωής: ανάπτυξη, διερεύνηση, αποκατάσταση, διατήρηση και αποδέσμευση.
- Όπως ισχύει και για πολλές άλλες δεξιότητες, η δεξιότητα της πραγματοποίησης επιτυχών επαγγελματικών επιλογών εξαρτάται από την κατάκτηση κάποιων συγκεκριμένων γνώσεων και από την εκτέλεση κατάλληλων δραστηριοτήτων στις κατάλληλες χρονικές στιγμές. Αν αυτές οι προκαταρκτικές γνώσεις και οι σχετικές δεξιότητες δεν αποκτηθούν στα σωστά ηλικιακά στάδια, θα υπάρξει αργότερα επίπτωση στην επαγγελματική ωριμότητα.
- Ο ορισμός της σταδιοδρομίας περιλαμβάνει πολύ περισσότερα στοιχεία από τον ορισμό της θέσης εργασίας. Η σταδιοδρομία είναι ένας συνδυασμός όλων των δραστηριοτήτων που πραγματοποιούνται μέσα από τους ρόλους ζωής που διαδραματίζει ένα άτομο σε ένα δεδομένο χρονικό σημείο. Μερικοί από τους ρόλους αυτούς είναι: παιδί, μαθητής/σπουδαστής, εργαζόμενος, σύζυγος, γονιός, νοικοκύρης, πολίτης και χρήστης ελεύθερου χρόνου. Ο Super τους παρουσίασε με τη μορφή ενός Ουράνιου Τόξου Ζωής-Καριέρας. Στο Ουράνιο Τόξο Ζωής-Καριέρας κάθε ρόλος μπορεί να οριστεί ως ο χρόνος και η ενέργεια που δαπανάται για την εκπλήρωση του ρόλου αυτού. Οι συγκεκριμένοι ρόλοι αλληλεπιδρούν μεταξύ τους και επηρεάζουν ο ένας τον άλλο.



Εικόνα 2.1 Το ουράνιο τόξο Ζωής-Καριέρας του Super

- Ο πιο πρώιμος ρόλος είναι αυτός του παιδιού κι έπειτα του μαθητή/σπουδαστή και του χρήστη ελεύθερου χρόνου. Ο ρόλος τού εργαζομένου είναι πολύ σημαντικός καθώς επηρεάζει και επηρεάζεται από όλους τους άλλους ρόλους. Γενικά, ο Super κατέληξε στο συμπέρασμα ότι όσο περισσότερους ρόλους θα μπορούσε να παίξει ένα άτομο διατηρώντας την ισορροπία, τόσο υψηλότερη είναι η πιθανότητα να επιτύχει ικανοποίηση και ολοκλήρωση στη ζωή του.
- Αν και η αυτοαντίληψη, τα ενδιαφέροντα, οι αξίες, οι ικανότητες και οι στόχοι ενός ατόμου είναι πρωταρχικοί παράγοντες στην επιλογή και ανάπτυξη σταδιοδρομίας, οι συνθήκες του περιβάλλοντος στο οποίο πραγματοποιούνται οι επαγγελματικές επιλογές αποτελούν, επίσης, ισχυρούς παράγοντες επίδρασης.

Όπως και στον Holland, υπάρχουν πρόσθετες έννοιες σχετικές με τη θεωρία του Super που έχουν ισχυρή πρακτική εφαρμογή:

- Μερικές πολύ σημαντικές συμπεριφορές και δεξιότητες πρέπει να μαθαίνονται νωρίς στην παιδική ηλικία ώστε να διευκολύνεται η βέλτιστη επαγγελματική ανάπτυξη. Μεταξύ αυτών είναι η συνειδητοποίηση της ανάγκης για μελλοντικό προγραμματισμό, οι δεξιότητες λήψης αποφάσεων, το εσωτερικό κέντρο ελέγχου και η διερευνητική στάση. Η απουσία αυτών συμβάλλει στην επαγγελματική ανωριμότητα.
- Οι αξίες είναι πολύ σημαντικοί παράγοντες επίδρασης στην επιλογή και ανάπτυξη σταδιοδρομίας. Όταν ένα άτομο γνωρίζει καλά τις προσωπικές αξίες του είναι σε θέση να εντοπίσει τα πιθανά επαγγέλματα που του ταιριάζουν. Ωστόσο, οι αξίες μπορούν να καταμετρηθούν σε όλους τους ρόλους ζωής. Το άτομο δεν θα πρέπει να προσπαθεί να επιτύχει όλες τις σημαντικές αξίες του στο ρόλο του ως εργαζόμενος.

2.2 Η θεωρία του John Krumboltz

Προερχόμενος από το χώρο των θεωρητικών της μάθησης, ο Krumboltz εξετάζει τις επιλογές και την ανάπτυξη σταδιοδρομίας ενός ατόμου μέσα από το πρίσμα του ρεπερτορίου των συμπεριφορών που έχει καταφέρει να μάθει. Όπως ακριβώς τα άτομα έχουν καταφέρει να μάθουν τις συμπεριφορές και τις δεξιότητες που κατέχουν πριν ζητήσουν την υποστήριξη ενός συμβούλου, μπορούν να μάθουν και νέες, αν αυτό θα τους ωφελήσει στην ανάπτυξη της σταδιοδρομίας τους. Οι θεμελιώδεις έννοιες της θεωρίας είναι οι εξής:

- Υπάρχουν τέσσερις πρωταρχικοί καθοριστικοί παράγοντες της επιλογής και ανάπτυξης σταδιοδρομίας: γενετικά χαρίσματα, περιβαλλοντικές συνθήκες και

γεγονότα, καθοριστικές και συσχετιστικές μαθησιακές εμπειρίες και γνώση δεξιοτήτων προσέγγισης καθηκόντων.

- Οι σύμβουλοι δεν μπορούν να επηρεάσουν τα γενετικά χαρίσματα ή τις περιβαλλοντικές συνθήκες. Αντίθετα, ωστόσο, μπορούν να επηρεάσουν τις μελλοντικές μαθησιακές εμπειρίες ενός συμβουλευόμενου και μπορούν να διδάξουν γνωστικές δεξιότητες λήψης αποφάσεων.
- Η ύπαρξη παράλογων πεποιθήσεων σχετικά με τις επιλογές και την ανάπτυξη σταδιοδρομίας μπορεί να αποτελέσει σημαντικό εμπόδιο στον προσδιορισμό των στόχων και στην πραγματοποίηση ικανοποιητικών επιλογών.
- Κάθε άτομο φιλτράρει τα γεγονότα και τις πληροφορίες μέσα από τη δική του γενικευμένη αυτοπαρατήρηση και τις κοσμοθεωρίες του. Επομένως, η πραγματικότητα ή η κοσμοθεωρία ενός ατόμου φιλτράρεται από ένα σύστημα προσωπικών πεποιθήσεων που έχει μάθει στο περιβάλλον του.
- Η γενικευμένη μάθηση και η αυτοπαρατήρηση επηρεάζονται έντονα από τη θετική ενίσχυση και τα θετικά πρότυπα. Οι άνθρωποι μαθαίνουν όταν δέχονται θετική ενίσχυση για τη στάση ή τη συμπεριφορά τους. Μαθαίνουν, επίσης, καθώς μιμούνται τη συμπεριφορά των ανθρώπων που έχουν ως πρότυπα.
- Ειδικά στον 21ο αιώνα, η αβεβαιότητα αποτελεί μια αποδεκτή και θετική κατάσταση, η οποία μπορεί να χρησιμοποιηθεί για να ενισχύσει τη διερεύνηση και να δημιουργήσει ευκαιρίες για νέα μάθηση. Οι σύμβουλοι θα πρέπει να δίνουν προτεραιότητα στο να ωθούν τους συμβουλευόμενους στη διερεύνηση και τη μάθηση αντί να τους βοηθούν να βρουν την καλύτερη λύση ή να επιλέξουν τη «σωστή» ειδικευση ή το «σωστό» επάγγελμα.

Το πλαίσιο του Krumboltz περιλαμβάνει τους εξής στόχους συμβουλευτικής:

- Εντοπισμός και εξάλειψη των παράλογων πεποιθήσεων που επηρεάζουν την επιλογή και ανάπτυξη σταδιοδρομίας.
- Διδασκαλία δεξιοτήτων λήψης αποφάσεων και άλλων δεξιοτήτων προσέγγισης καθηκόντων, σε περίπτωση που ελλείπουν.
- Διεύρυνση του ορίζοντα εναλλακτικών επιλογών του συμβουλευόμενου, ενθαρρύνοντας τη διερεύνηση και τις πραγματικές εμπειρίες ζωής.
- Δημιουργία ευκαιριών για τον συμβουλευόμενο ώστε να λάβει θετική ενίσχυση για τις συμπεριφορές που θεωρούνται ότι είναι θετικές γι' αυτόν.
- Δημιουργία ευκαιριών για τον συμβουλευόμενο ώστε να υιοθετήσει θετικά πρότυπα.
- Ανάπτυξη κατάλληλων δεξιοτήτων επαγγελματικού σχεδιασμού (διερεύνηση, λήψη πληροφοριών, αποκρυστάλλωση επιλογής, σθάμιση εναλλακτικών επιλογών) που οδηγούν τον συμβουλευόμενο σε στόχους που θα επιλέξει ο ίδιος.
- Παροχή στήριξης στον συμβουλευόμενο για να αποδεχτεί την αβεβαιότητα ως μια φυσιολογική κατάσταση και να τη χρησιμοποιήσει για το σχεδιασμό νέων εμπειριών.

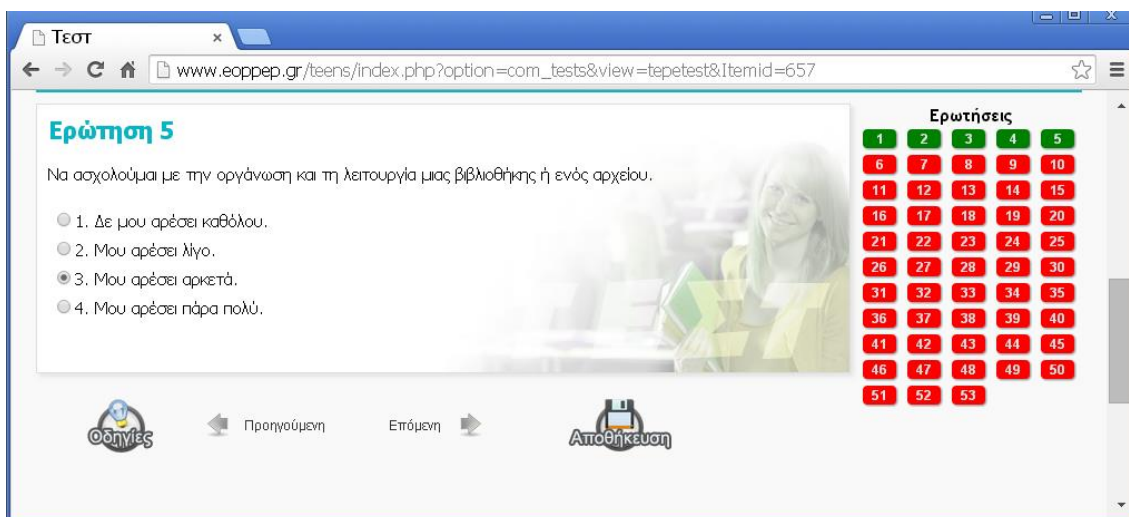
Οι Krumboltz και Baker προσδιορίζουν τα παρακάτω οχτώ στάδια στη σχέση κατά τη διαδικασία της επαγγελματικής συμβουλευτικής:

- Ορισμός του προβλήματος και των στόχων του συμβουλευόμενου.
- Επιδίωξη και εξασφάλιση αμοιβαίας συμφωνίας για την επίτευξη των στόχων.
- Δημιουργία εναλλακτικών λύσεων στο πρόβλημα.
- Συλλογή πληροφοριών σχετικά με αυτές τις εναλλακτικές.
- Εξέταση των πιθανών επιπτώσεων κάθε εναλλακτικής.
- Επαναξιολόγηση στόχων, εναλλακτικών και επιπτώσεων.
- Λήψη απόφασης ή πραγματοποίηση επιλογής.
- Γενίκευση της διαδικασίας λήψης αποφάσεων για την αντιμετώπιση νέων προβλημάτων.

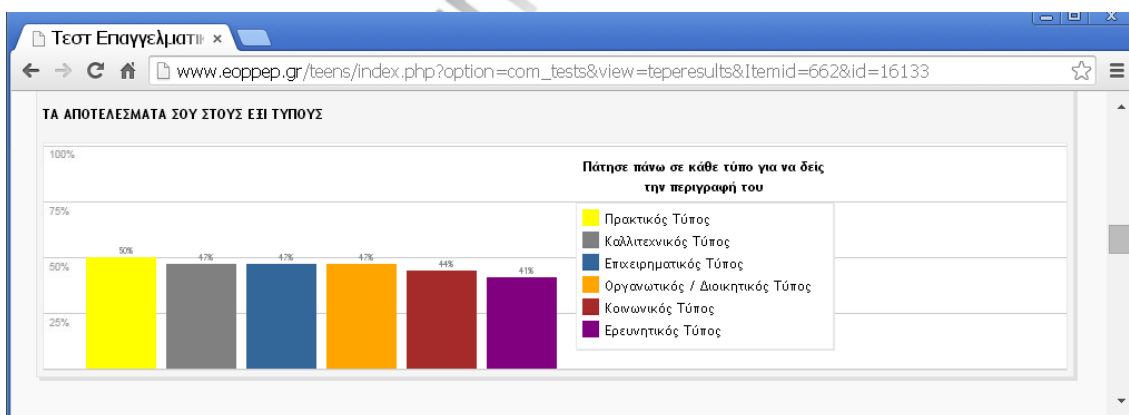
2.3 Εφαρμογές Επαγγελματικού Προσανατολισμού σύμφωνα με τη θεωρία του Holland

Στο διαδίκτυο μπορεί κάποιος να βρει αρκετά τεστ επαγγελματικού προσανατολισμού που βασίζονται στη θεωρία του Holland. Παρακάτω γίνεται μια παράθεση των πιο γνωστών από αυτά.

Στην ιστοσελίδα του Ε.Ο.Π.Π.Ε.Π. (Εθνικός Οργανισμός Πιστοποίησης Προσόντων και Επαγγελματικού Προσανατολισμού) <http://www.eoppep.gr/teens/> υπάρχει το τεστ επαγγελματικού προσανατολισμού «Γνώρισε το εαυτό σου» το οποίο αποτελείται από 53 με βαθμολογία από το 1 μέχρι το 4



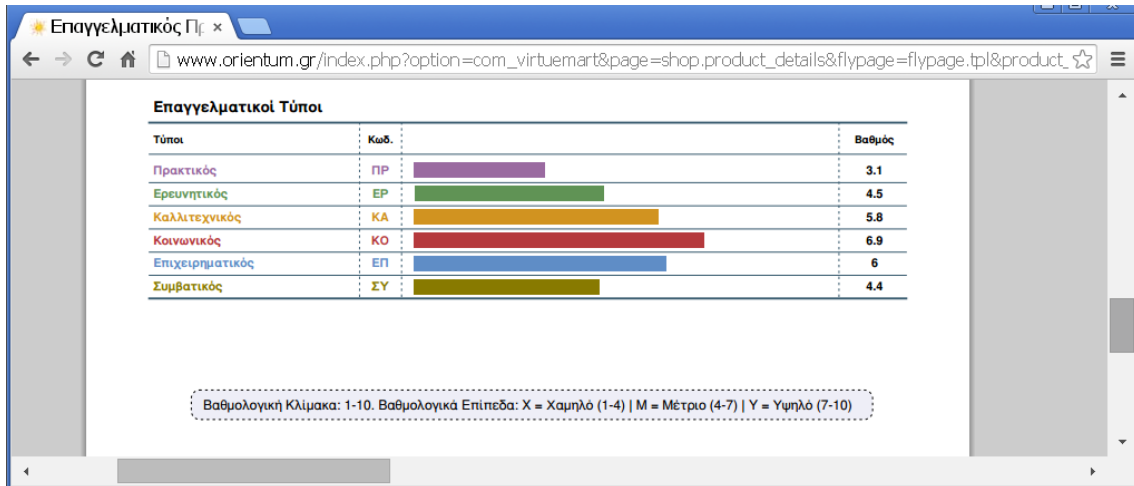
Εικόνα 2.2 Τεστ ΕΟΠΠΕΠ



Εικόνα 2.3 Αποτελέσματα τεστ ΕΟΠΠΕΠ

Από την εταιρεία Orientum – Σύμβουλοι Σταδιοδρομίας <http://www.orientum.gr/>, η οποία εξειδικεύεται σε θέματα επαγγελματικού προσανατολισμού προσφέρεται το τεστ HORIZON το οποίο δεν προσφέρεται δωρεάν. Οι κατασκευαστές, αφού μελέτησαν προσεκτικά τη σύγχρονη διεθνή και ελληνική ψυχομετρική πραγματικότητα στο επίπεδο των επαγγελματικών ενδιαφερόντων, σχεδίασαν μία εξολοκλήρου νέα μέτρηση, η οποία είχε ως στόχο, αφενός να καλύψει κάποιες ελλείψεις που εμφάνιζαν τα υπάρχοντα εργαλεία μέτρησης ενδιαφερόντων (μη συμβατότητα με το εκπαιδευτικό σύστημα, ψυχομετρικές αδυναμίες, απουσία θεωρητικού

πλασίου κ.λπ.) και αφετέρου να ικανοποιήσει στο μέγιστο δυνατό βαθμό την πάγια συμβουλευτική ανάγκη για λεπτομερειακή αποτύπωση των επαγγελματικών ενδιαφερόντων του ατόμου. Το τεστ HORIZON αποτελείται από 190 ερωτήσεις που περιγράφουν διάφορες επαγγελματικές δραστηριότητες. Το άτομο καλείται να δηλώσει σε μια θβάθμια κλίμακα το βαθμό, στον οποίο θα το ενδιέφερε να ασχοληθεί επαγγελματικά στο μέλλον με καθεμία από αυτές τις δραστηριότητες.



Εικόνα 2.4 Τεστ HORIZON

Τα πλεονεκτήματα των τεστ επαγγελματικού προσανατολισμού είναι ότι υπάρχουν στο διαδίκτυο και καθένας μπορεί εύκολα να τα χρησιμοποιήσει, δεν διαρκούν πολύ χρόνο και εμφανίζουν άμεσα τα αποτελέσματα.

Ο εκάστοτε χρήστης όμως θα πρέπει να έχει καλή γνώση του εαυτού του για να πάρει τα σωστά αποτελέσματα και γενικότερα να έχει καλή πληροφόρηση για την εικόνα της αγοράς και των σχολών για μην πάρει λανθασμένες αποφάσεις.

3. ΠΑΡΟΥΣΙΑΣΗ ΚΑΙ ΧΡΗΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Φόρμες Εφαρμογής

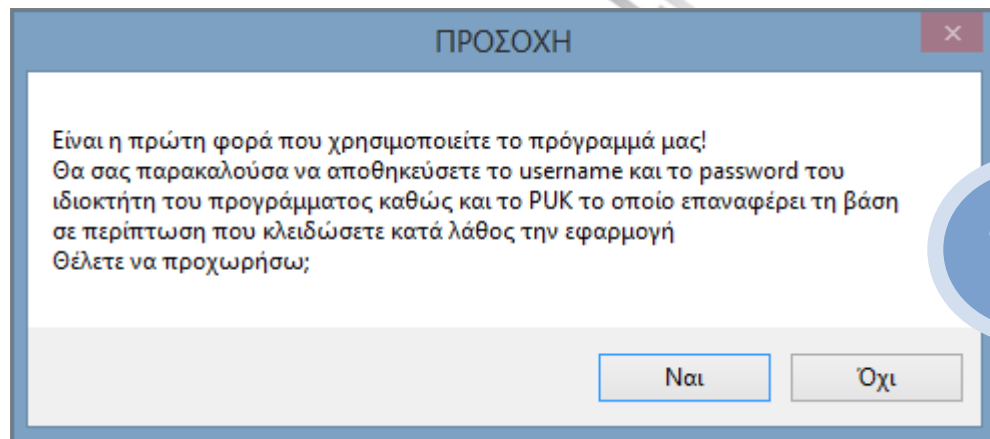
Παρακάτω παρουσιάζονται όλες οι φόρμες της εφαρμογής, η επεξήγηση τους, τα γραφικά περιβάλλοντα και η λειτουργία τους.

ΦΟΡΜΑ 1

Φόρμα [όνομα] :

message.show

Γραφικό περιβάλλον :



Λειτουργία :

Είναι η πρώτη φόρμα που βλέπει ο χρήστης όταν ανοίγει για πρώτη φορά την εφαρμογή.

Του ζητείται να αποφασίσει αν θα προχωρήσει σε αποθήκευση κωδικών διαχειριστή ή αν θα το αφήσει για μελλοντική φορά που θα συνδεθεί.

Έλεγχος ροής :

[ΝΑΙ]

Μετάβαση στην φόρμα frm_firsttime (εικόνα 3,2)

[ΟΧΙ]

Έξοδος από την εφαρμογή

Φόρμα [όνομα] :

frm_firsttime

Γραφικό περιβάλλον :


ΚΑΛΩΣΗΡΘΑΤΕ ΣΤΟ ΠΡΟΓΡΑΜΜΑ

Ορίστε το username του διαχειριστή του συστήματος

Ορίστε το password :

επιβεβαίωση password :

τέλος

 Attart A. Konstantinos

εικόνα 3.2

Λειτουργία :

Ο χρήστης πρέπει να εισάγει το username και το password εις διπλούν για να πάρει τα δικαιώματα του πρώτου χρήστη που θα είναι και τα δικαιώματα του διαχειριστή της βάσης.

Ύστερα μπορεί να επεξεργαστεί το password ή να δημιουργήσει ή να επεξεργαστεί άλλους χρήστες.

Έλεγχος ροής :

[τέλος]

Με το κουμπί γίνεται έλεγχος αν τα στοιχεία που καταχώρισε είναι αποδεκτά (εικόνα 3,3).

Σε περίπτωση που δεν είναι, του ζητείται να βάλει νέο username και password.

Σε περίπτωση που είναι, γίνεται αποθήκευση των κωδικών και γίνεται είσοδος στην φόρμα frm_firsttimePUK (εικόνα 3,4).

Φόρμα [όνομα] :

frm_firsttime

Γραφικό περιβάλλον :

ΚΑΛΩΣΗΡΘΑΤΕ ΣΤΟ ΠΡΟΓΡΑΜΜΑ


Ορίστε το username του διαχειριστή του συστήματος

Ορίστε το password :

επιβεβαίωση password :

Το όνομα σας πρέπει να αποτελείται από 7 γράμματα τουλάχιστον!

reset

 Attart A. Konstantinos

εικόνα
3.3**Λειτουργία :**

Έχει πατηθεί μη αποδεκτό username.
Εμφανίζει μήνυμα λάθους.

Έλεγχος ροής :

[reset]

Καθαρίζει τα πεδία που έχουν πληκτρολογηθεί και επαναφέρει την φόρμα στην (εικόνα 3,2).

ΦΟΡΜΑ 2

Φόρμα [όνομα] :

frm_firsttimePUK

Γραφικό περιβάλλον :

**Λειτουργία :**

Επιλέγουμε τον κωδικό επαναφοράς PUK. Ο κωδικός θα ζητηθεί αν κλειδωθεί η βάση ή θελήσουμε να επαναφέρουμε την βάση δηλαδή να διαγράψουμε όλα της τα στοιχεία. Τον κωδικό αυτόν τον ξέρει μόνο ο διαχειριστής.

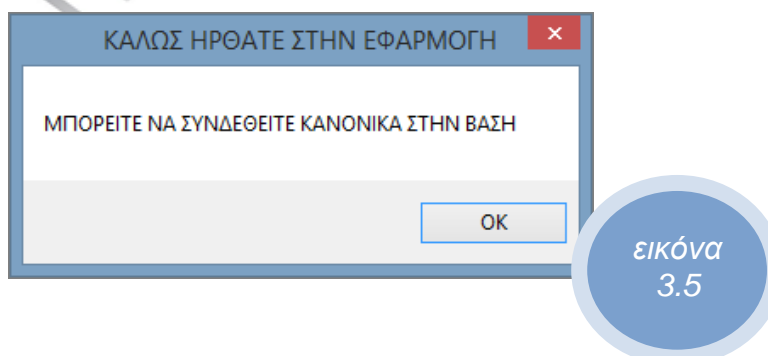
Εδώ δεν δίνεται η δυνατότητα πληκτρολόγησης.

Επιλέγουμε τον κωδικό με βάση τα βελάκια. Είναι 4ψήφιος ακέραιος.

Έλεγχος ροής :

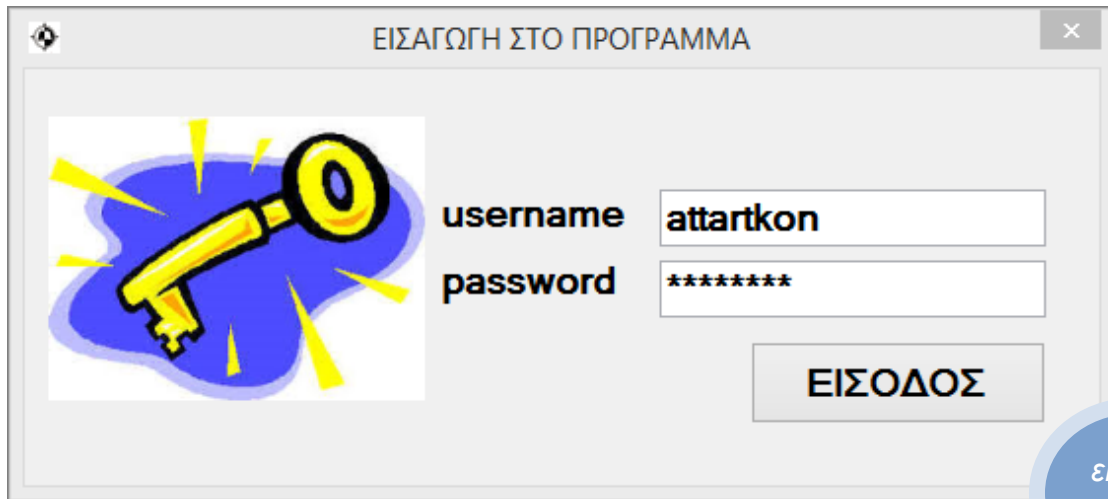
[ΑΠΟΘΗΚΕΥΣΗ]

Αποθηκεύουμε στη βάση sql τον κωδικό και εμφανίζουμε μήνυμα επιτυχίας (εικόνα 3,5). Ύστερα προχωράμε στην φόρμα frm_μενου.



ΦΟΡΜΑ 3**Φόρμα [όνομα] :**

frm_login

Γραφικό περιβάλλον :

ΕΙΣΑΓΩΓΗ ΣΤΟ ΠΡΟΓΡΑΜΜΑ

username attartkon

password *****

ΕΙΣΟΔΟΣ

εικόνα
3.6**Λειτουργία :**

Ο χρήστης πληκτρολογεί το username και το password.

Έλεγχος ροής :

[ΕΙΣΟΔΟΣ]

Σε περίπτωση που τα στοιχεία είναι αποθηκευμένα στην βάση προχωρούμε στην εισαγωγή στο κεντρικό μενού.

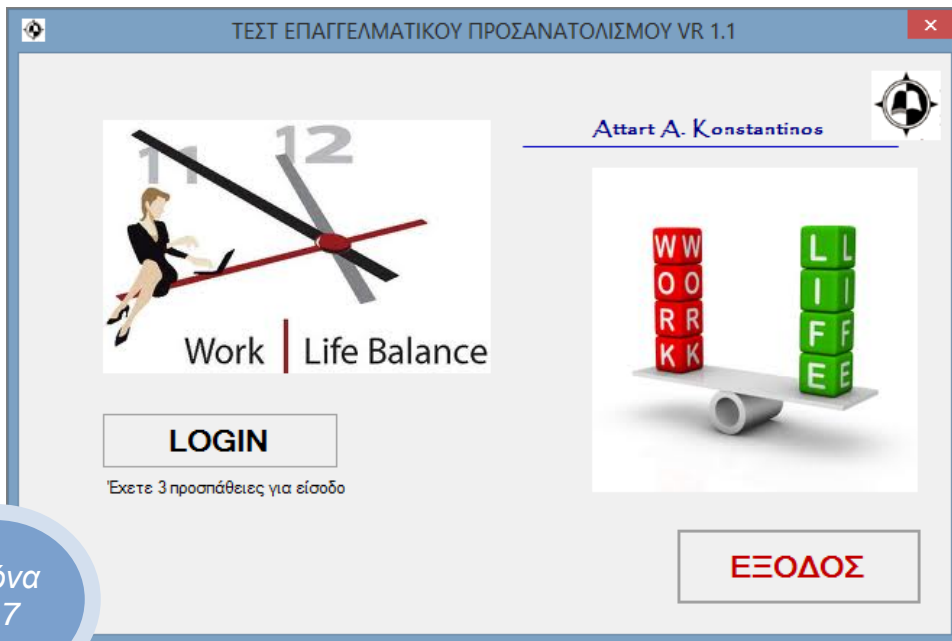
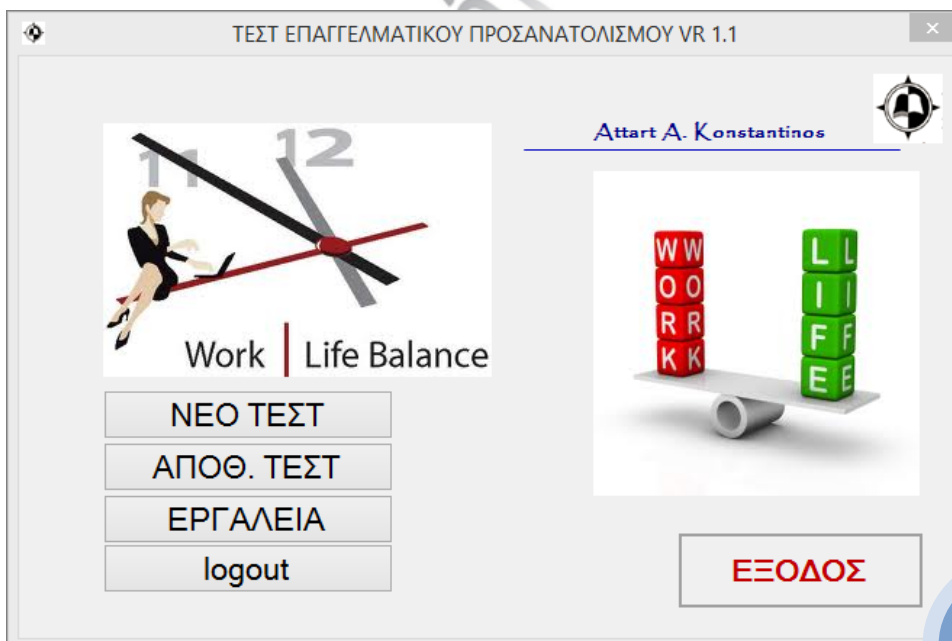
Αν τα στοιχεία δεν είναι σωστά προσθέτουμε την λάθος εισαγωγή και μας ζητείται να ξανακάνουμε login. Αν σε τρεις προσπάθειες δεν γίνει η σωστή εισαγωγή η βάση κλειδώνεται. Μπορούμε να ενεργοποιήσουμε την εφαρμογή μόνο με το username, password και PUK του διαχειριστή.

ΦΟΡΜΑ 4

Φόρμα [όνομα] :

frm_menou

Γραφικό περιβάλλον :

εικόνα
3.7εικόνα
3.8

Έλεγχος ροής:

Είναι η φόρμα εισαγωγής στην εφαρμογή. Ο χρήστης έχει επιλογή να κάνει login με τα στοιχεία του ή να βγει από την εφαρμογή.

Όταν έχει συνδεθεί με επιτυχία εμφανίζονται τα κουμπιά [ΝΕΟ ΤΕΣΤ][ΑΠΟΘ.ΤΕΣΤ][ΕΡΓΑΛΕΙΑ][logout]

[ΝΕΟ ΤΕΣΤ]

Δημιουργεί νέο τεστ επαγγελματικού προσανατολισμού.

Ανοίγει την φόρμα : [frm_new_item.cs]

[ΑΠΟΘ.ΤΕΣΤ]

Συνεχίζει αποθηκευμένο τετ που έχει γίνει κατά το παρελθόν

Ανοίγει την φόρμα : [frm_exist_item.cs]

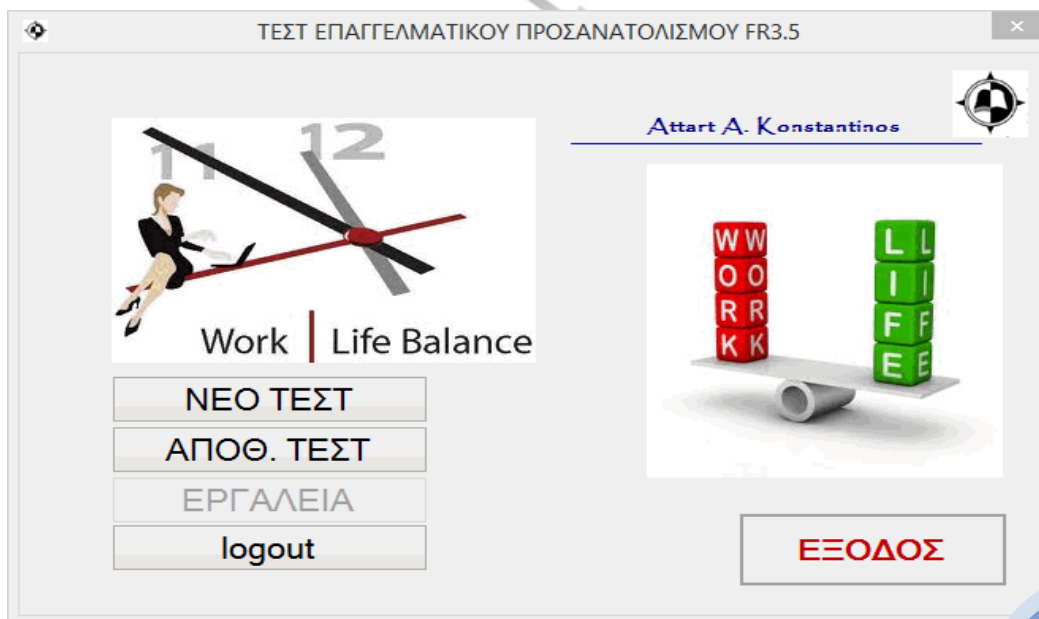
[ΕΡΓΑΛΕΙΑ]

Το πρόγραμμα δίνει την δυνατότητα να επεξεργαστούμε χρήστες, να εισάγουμε νέους χρήστες και να επαναφέρουμε την κατάσταση στην αρχική κατάσταση χωρίς καμία εισαγωγή στοιχείων.

[logout]

Γίνεται αποσύνδεση του χρήστη και δίνει την δυνατότητα να γίνει νέο login με τα στοιχεία που έχουμε αποθηκεύσει.

Εφόσον ο χρήστης δεν είναι και διαχειριστής η επιλογή εργαλεία είναι απενεργοποιημένη.



εικόνα
3.9

ΦΟΡΜΑ 5

Φόρμα [όνομα] :

frm_new_item

Γραφικό περιβάλλον :

ΤΕΣΤ ΕΠΑΓΓΕΛΜΑΤΙΚΟΥ ΠΡΟΣΑΝΑΤΟΛΙΣΜΟΥ

χρήστης : attartkon 20/08/2014
19:17:04

Κωδικός 1

Επίθετο Ατταρτ ημερομηνία 20/08/2014
 ορατό από όλους τους χρήστες;

Όνομα Κωνσταντίνος

Ξεκίνα το ΤΕΣΤ ΑΚΥΡΟ

εικόνα 3.10

Λειτουργία :

Δημιουργούμε νέο τεστ. Διαλέγουμε το επίθετο και το όνομα. Η ημερομηνία είναι η τρέχουσα. Αν επιλέξω το ορατό, το τεστ είναι ορατό από όλους τους άλλους χρήστες. Σε περίπτωση που δεν είναι ενεργοποιημένο δεν θα μπορέσει κάποιος άλλος να βλέπει το τεστ για να εξασφαλίσουμε την ιδιωτικότητα των τεστ.

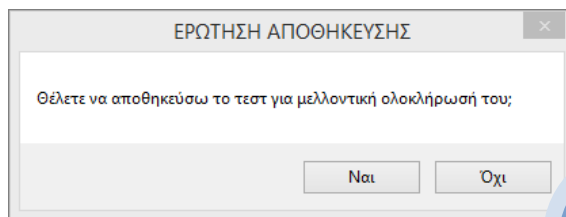
Έλεγχος ροής :

[Ξεκίνα το ΤΕΣΤ]

Αποθηκεύονται στην βάση sql τα στοιχεία του τεστ και αρχίζουμε τις ερωτήσεις του τεστ. Προχώρα στην φόρμα : frm_answer.

[ΑΚΥΡΟ]

Αν επιλέξω την ακύρωση θα εμφανιστεί μήνυμα αν θέλουμε να αποθηκευτεί το τεστ. (εικόνα 3,11). Αν επιλέξω ναι γίνεται αποθήκευση. Αν όχι δεν γίνεται. Επιστρέφω στο κεντρικό μενού, στη φόρμα: frm_menou.



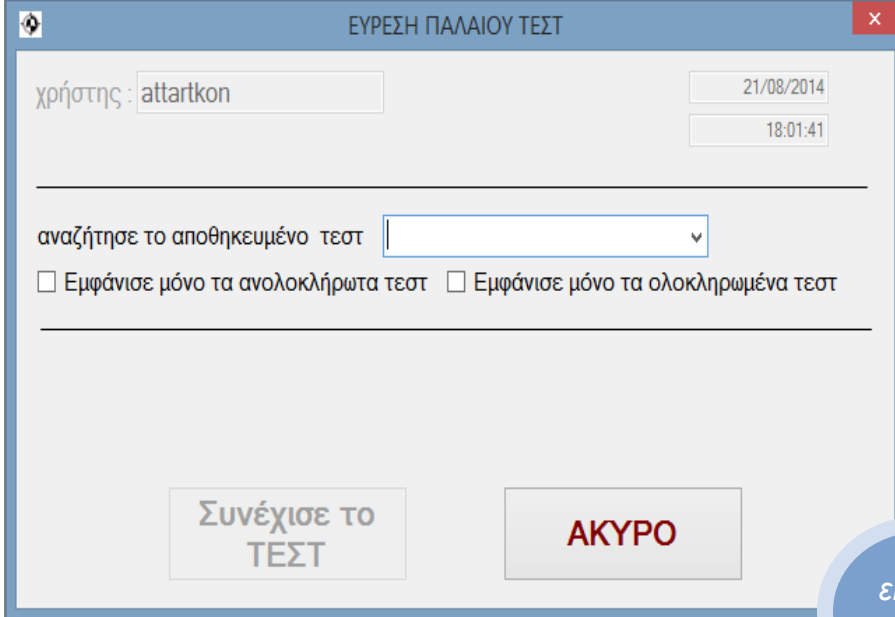
εικόνα
3.11

ΦΟΡΜΑ 6

Φόρμα [όνομα] :

frm_exist_item

Γραφικό περιβάλλον :



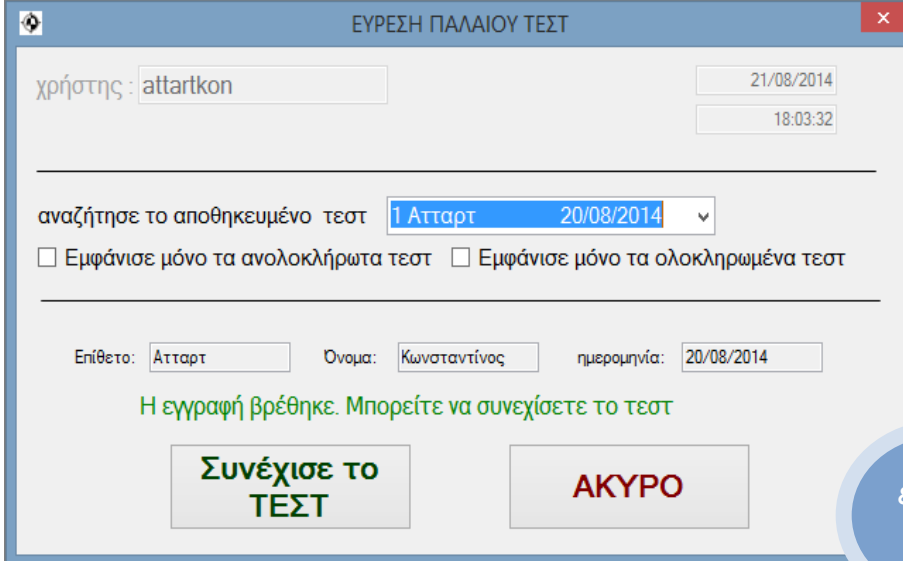
ΕΥΡΕΣΗ ΠΑΛΑΙΟΥ ΤΕΣΤ

χρήστης : attartkon 21/08/2014
18:01:41

αναζήτησε το αποθηκευμένο τεστ []

Εμφάνισε μόνο τα ανολοκλήρωτα τεστ Εμφάνισε μόνο τα ολοκληρωμένα τεστ

Συνέχισε το ΤΕΣΤ ΑΚΥΡΟ

εικόνα
3.12

ΕΥΡΕΣΗ ΠΑΛΑΙΟΥ ΤΕΣΤ

χρήστης : attartkon 21/08/2014
18:03:32

αναζήτησε το αποθηκευμένο τεστ 1 Ατάρτ 20/08/2014

Εμφάνισε μόνο τα ανολοκλήρωτα τεστ Εμφάνισε μόνο τα ολοκληρωμένα τεστ

Επίθετο: Ατάρτ Όνομα: Κωνσταντίνος ημερομηνία: 20/08/2014

Η εγγραφή βρέθηκε. Μπορείτε να συνεχίσετε το τεστ

Συνέχισε το ΤΕΣΤ ΑΚΥΡΟ

εικόνα
3.13

Λειτουργία :

Στην φόρμα αυτή ο χρήστης επιλέγει ένα τεστ το οποίο είχε ξεκινήσει στο παρελθόν και είχε αποθηκευτεί στην βάση sql.

Έχουμε τρεις δυνατότητες.

α) Να δούμε με λίστα όλα τα τεστ που έχουν αποθηκευτεί.

β) Να δούμε μόνο τα τεστ που έχουν ολοκληρωθεί. Ουσιαστικά να δούμε τα αποτελέσματά τους.

γ) Να δούμε μόνο τα ανολοκλήρωτα τεστ και να συνεχίσουμε να απαντάμε τις ερωτήσεις. Οι ερωτήσεις είναι προς το παρόν 200. Είναι μερικές φορές δύσκολο να τις απαντήσεις όλες εξαιτίας έλλειψης χρόνου. Γι' αυτό δίνεται η δυνατότητα να συνεχίσεις.

Ύστερα από την επιλογή του τεστ έχουμε δύο ενέργειες:

Έλεγχος ροής :

[ΣΥΝΕΧΙΣΕ ΤΟ ΤΕΣΤ]

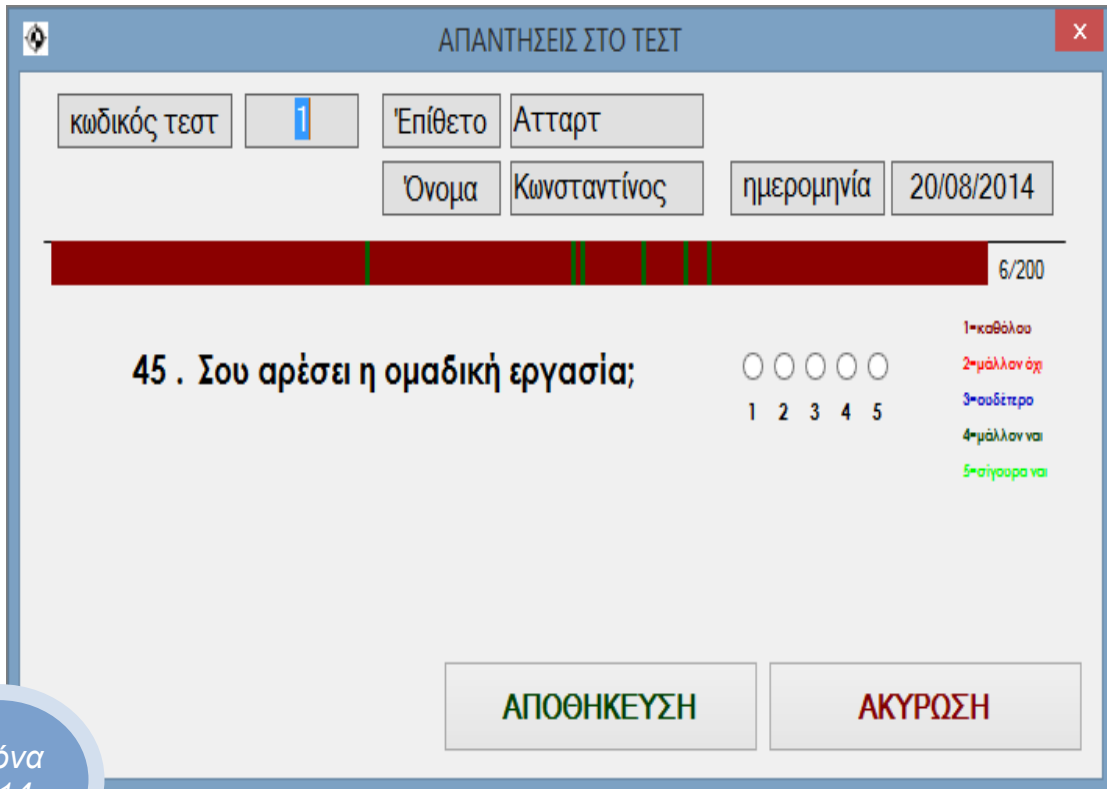
Έχοντας επιλέξει το τεστ μεταβαίνουμε στην φόρμα frm_answers.

[ΑΚΥΡΟ]

Επιστρέφουμε στο κύριο μενού. (frm_μενου)

ΦΟΡΜΑ 7**Φόρμα [όνομα] :**

frm_answer

Γραφικό περιβάλλον :

ΑΠΑΝΤΗΣΕΙΣ ΣΤΟ ΤΕΣΤ

κωδικός τεστ 1

Επίθετο Ατταρτ

Όνομα Κωνσταντίνος

ημερομηνία 20/08/2014

6/200

45 . Σου αρέσει η ομαδική εργασία;

○ ○ ○ ○ ○

1 2 3 4 5

1=καθόλου
2=μάλλον όχι
3=ουδέποτε
4=μάλλον ναι
5=σίγουρα ναι

ΑΠΟΘΗΚΕΥΣΗ ΑΚΥΡΩΣΗ

εικόνα
3.14**Λειτουργία :**

Στην φόρμα αυτή ο χρήστης δίνει την απάντηση του σε κάθε μια ερώτηση που εμφανίζεται από την εφαρμογή.

Στην έκδοση αυτή υπάρχουν 200 ερωτήσεις που είναι αποθηκευμένες στην βάση sql και δίνονται με τυχαία σειρά προς απάντηση. Η απάντηση είναι βαθμοθετημένη στην κλίμακα 1..5 που δηλώνει από την αρνητική στην θετική απάντηση.

Κάθε απάντηση αποθηκεύεται σε έναν πίνακα στην κλάση my_user.

Ο χρήστης έχει τη δυνατότητα να απαντήσει σε όλο το τεστ ή να διακόψει τις απαντήσεις και να αποθηκεύσει το τρέχον τεστ. Μπορεί να συνεχίσει το τεστ όπως περιγράφηκε στην φόρμα 6.

ΑΠΑΝΤΗΣΕΙΣ ΣΤΟ ΤΕΣΤ

κωδικός τεστ 1

Επίθετο Αττάρτ

Όνομα Κωνσταντίνος

ημερομηνία 20/08/2014

168/200

130 . Θέλεις να πας σε όλο τον κόσμο;

○ ○ ○ ○ ○

1 2 3 4 5

1=καθόλου
2=μάλλον όχι
3=ουδέτερο
4=μάλλον ναι
5=σίγουρα ναι

ΑΠΟΘΗΚΕΥΣΗ ΑΚΥΡΩΣΗ

εικόνα
3.15

Όταν οι ερωτήσεις απαντηθούν όλες η εφαρμογή εμφανίζει μήνυμα (εικόνα 3,16) και δίνει τη δυνατότητα να δούμε τα αποτελέσματα. Στην περίπτωση που το τεστ ήταν ήδη ολοκληρωμένο εμφανίζεται κατευθείαν το μήνυμα.

εικόνα
3.16

Έλεγχος ροής :

[radio 1..5]

Αποθηκεύουμε την απάντηση με κωδικό που είναι ο αριθμός της ερώτησης σε έναν πίνακα με μέγεθος όσες είναι και οι ερωτήσεις.

Δημιουργείται τυχαίος αριθμός που αντιστοιχεί σε ερώτηση που δεν έχει απαντηθεί. Αν υπάρχει τέτοιος αριθμός εμφανίζεται η ερώτηση με τον κωδικό αυτόν τον αριθμό και επιλέγουμε ξανά την απάντηση μέχρι να τελειώσουμε τις ερωτήσεις ή σταματήσουμε το τεστ προς αποθήκευση.

[ΑΠΟΘΗΚΕΥΣΗ]

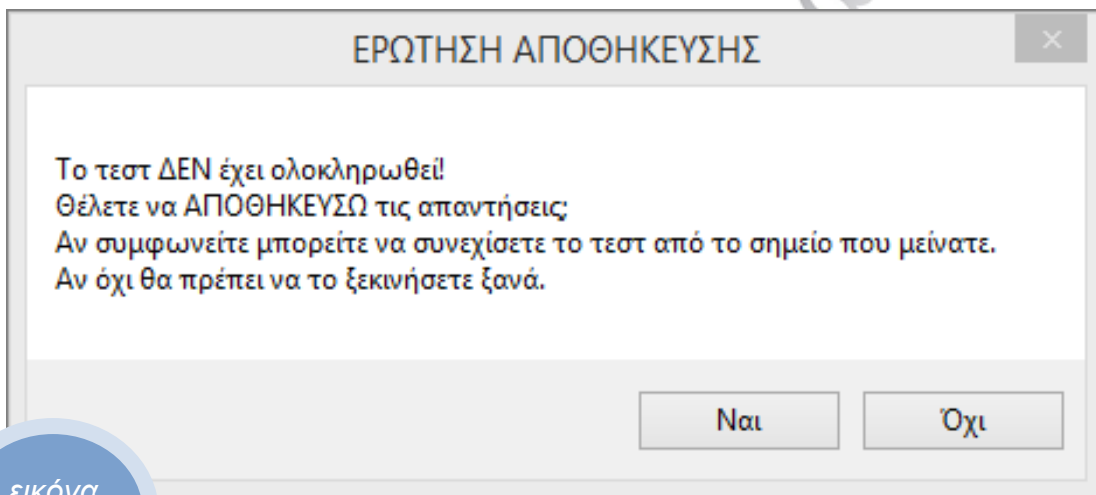
Σταματάμε το τεστ και αποθηκεύουμε τον πίνακα με τις απαντήσεις στη βάση sql. Ο πίνακας μετατρέπεται σε string και κάθε ψηφίο είναι η απάντηση. Έτσι επέλεξα να αποθηκεύσω όλες τις απαντήσεις σε ένα μόνο πεδίο χωρίς να χρειάζεται να δημιουργήσω και να κανονικοποιήσω νέο πίνακα στην σχεσιακή βάση sql.

[ΑΚΥΡΩΣΗ]

Κλείνει η φόρμα αλλά εμφανίζει μήνυμα αν θέλουμε να αποθηκεύσουμε το τεστ για να μην χαθούν οι απαντήσεις που δόθηκαν μέχρι εκείνη τη στιγμή (εικόνα 7,4). Αν απαντηθεί [ΝΑΙ] γίνεται αποθήκευση.

[ΑΠΟΤΕΛΕΣΜΑΤΑ]

Το τεστ έχει ολοκληρωθεί ή είναι αποθηκευμένο ως ολοκληρωμένο και για αυτό εμφανίζεται το κουμπί αυτό. Μεταβαίνουμε στη φόρμα 8 frm_results για να δούμε τα αποτελέσματα του τεστ.



εικόνα
3.17

ΦΟΡΜΑ 8**Φόρμα [όνομα] :**

message.show

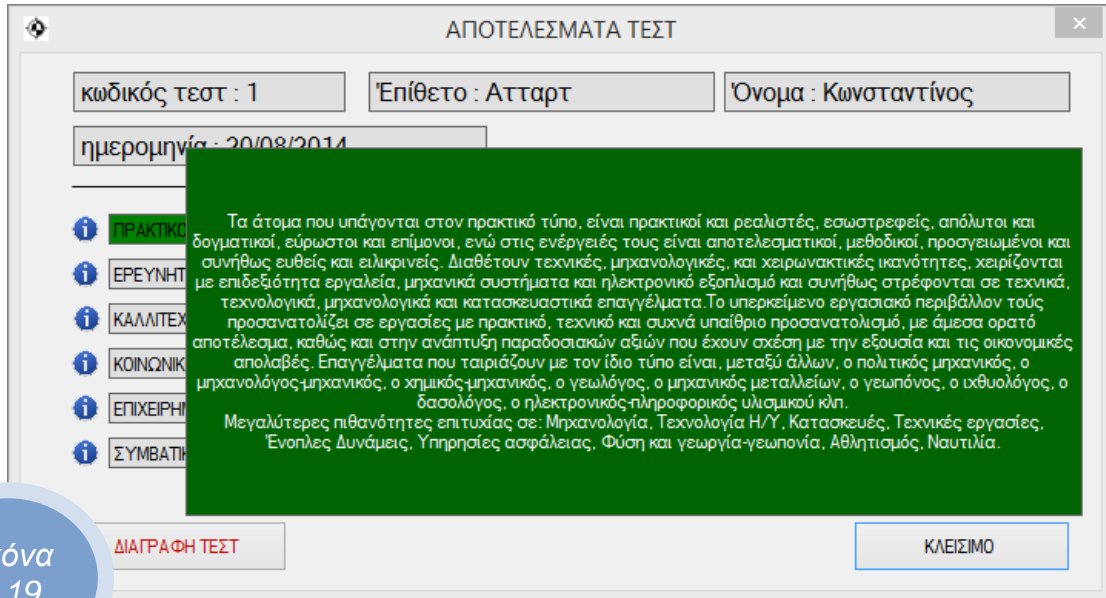
Γραφικό περιβάλλον :

Κατηγορία	Ποσοστό	Πρόοδος
ΠΡΑΚΤΙΚΟΣ-ΡΕΑΛΙΣΤΙΚΟΣ	27,09%	27,09%
ΕΡΕΥΝΗΤΙΚΟΣ	4,79%	4,79%
ΚΑΛΛΙΤΕΧΝΙΚΟΣ	17,08%	17,08%
ΚΟΙΝΩΝΙΚΟΣ	15,57%	15,57%
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΣ	17,92%	17,92%
ΣΥΜΒΑΤΙΚΟΣ-ΟΡΓΑΝΩΤΙΚΟΣ	17,55%	17,55%

εικόνα
3.18**Λειτουργία :**

Στη φόρμα αυτή ο χρήστης βλέπει τα αποτελέσματα του τεστ σε ποσοστά με βάση τις 6 κατηγορίες του Holland και έχει τη δυνατότητα να δει και σε χρωματισμένες μπάρες το αποτέλεσμα.

Περνώντας το ποντίκι πάνω από τα ερωτηματικά της κάθε κατηγορίας βλέπει τις προτάσεις των κατηγοριών και των επαγγελματών.



εικόνα
3.19

Αφού έχει δει τα αποτελέσματα μπορεί να διαγράψει το τεστ. Έτσι εξασφαλίζεται η ιδιωτικότητα του τεστ.

Έλεγχος ροής :

[ερωτηματικά]

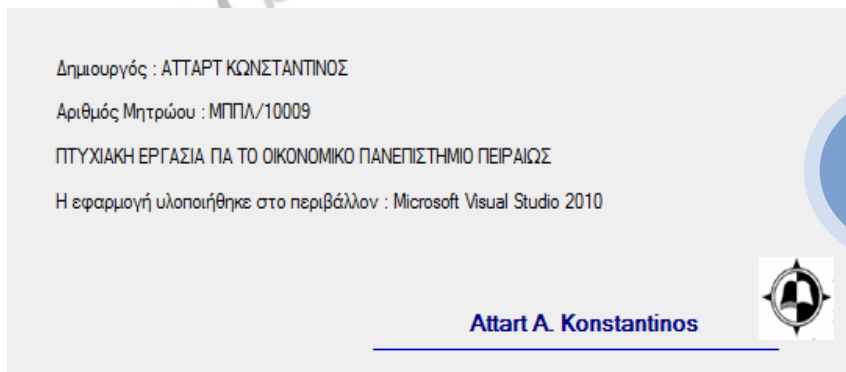
Εμφανίζει μήνυμα με τα αποτελέσματα της κάθε κατηγορίας (εικόνα 3,18).

[ΔΙΑΓΡΑΦΗ]

Διαγράφεται τελείως το τεστ από την βάση sql.

[ΚΛΕΙΣΙΜΟ]

Εμφανίζετε μήνυμα (εικόνα 3,20) και επιστρέφει στο κύριο μενού, φόρμα frm_μενου.



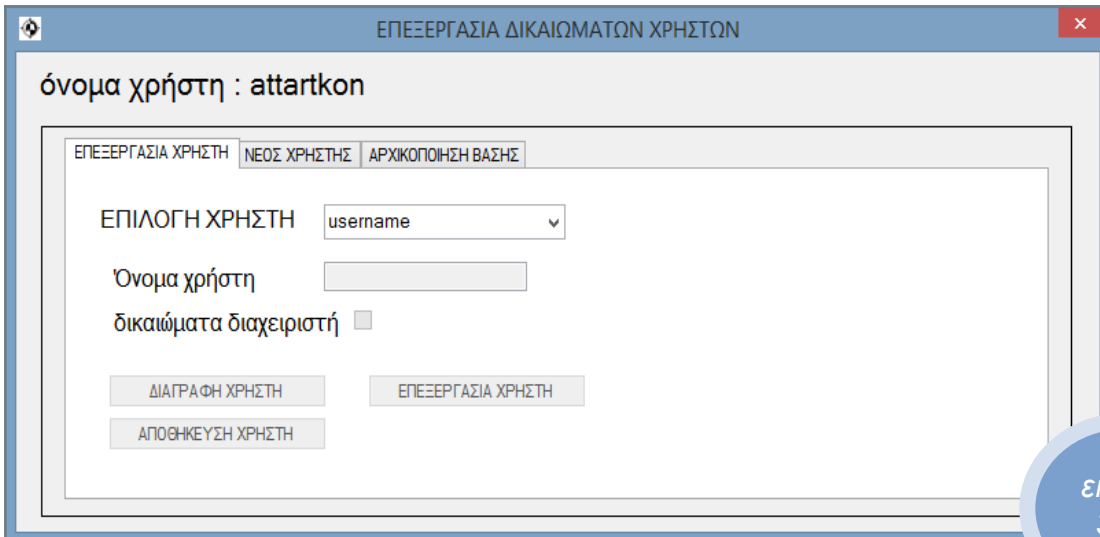
εικόνα
3.20

ΦΟΡΜΑ 9

Φόρμα [όνομα] :

frm_edit_users

Γραφικό περιβάλλον :



ΕΠΕΞΕΡΓΑΣΙΑ ΔΙΚΑΙΩΜΑΤΩΝ ΧΡΗΣΤΩΝ

όνομα χρήστη : attartkon

ΕΠΕΞΕΡΓΑΣΙΑ ΧΡΗΣΤΗ | ΝΕΟΣ ΧΡΗΣΤΗΣ | ΑΡΧΙΚΟΠΟΙΗΣΗ ΒΑΣΗΣ

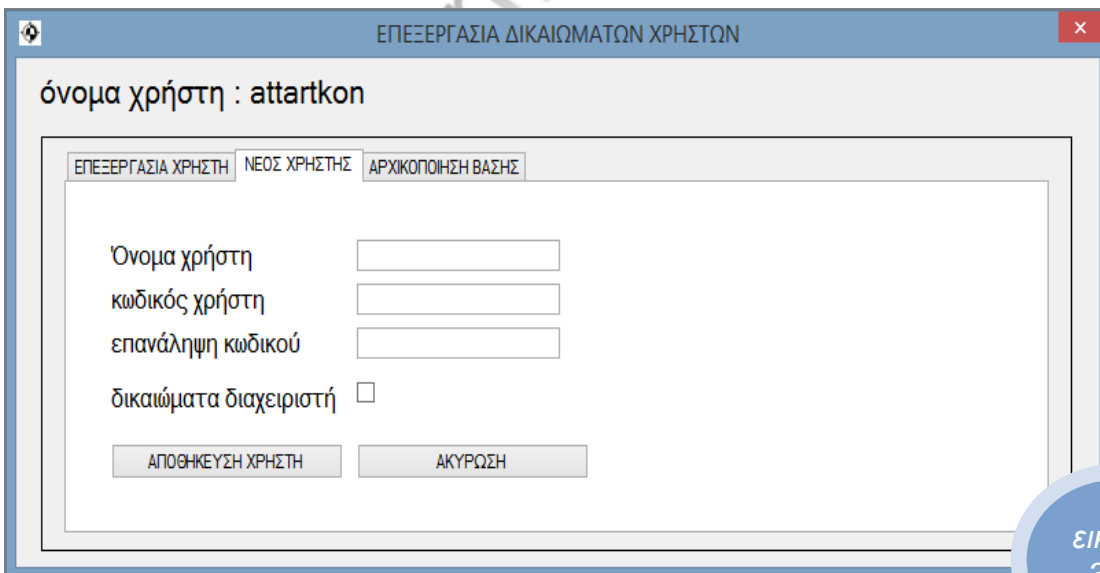
ΕΠΙΛΟΓΗ ΧΡΗΣΤΗ username

Όνομα χρήστη

δικαιώματα διαχειριστή

ΔΙΑΓΡΑΦΗ ΧΡΗΣΤΗ ΕΠΕΞΕΡΓΑΣΙΑ ΧΡΗΣΤΗ

ΑΠΟΘΗΚΕΥΣΗ ΧΡΗΣΤΗ

εικόνα
3.21

ΕΠΕΞΕΡΓΑΣΙΑ ΔΙΚΑΙΩΜΑΤΩΝ ΧΡΗΣΤΩΝ

όνομα χρήστη : attartkon

ΕΠΕΞΕΡΓΑΣΙΑ ΧΡΗΣΤΗ | ΝΕΟΣ ΧΡΗΣΤΗΣ | ΑΡΧΙΚΟΠΟΙΗΣΗ ΒΑΣΗΣ

Όνομα χρήστη

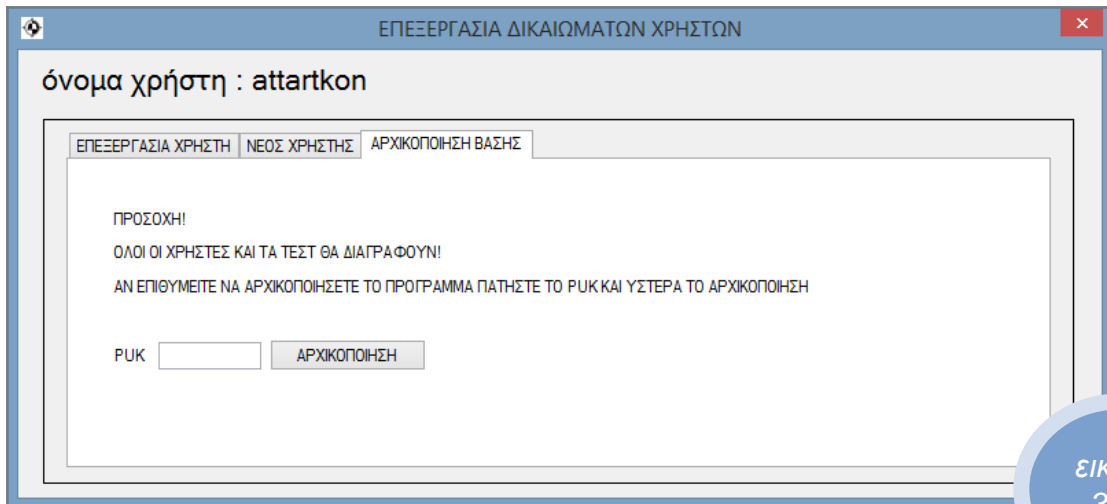
κωδικός χρήστη

επανάληψη κωδικού

δικαιώματα διαχειριστή

ΑΠΟΘΗΚΕΥΣΗ ΧΡΗΣΤΗ ΑΚΥΡΩΣΗ

εικόνα
3.22



Λειτουργία :

Είναι η φόρμα διαχείρισης χρηστών. Μπορούμε να εισάγουμε νέο χρήστη ή να επεξεργαστούμε παλαιότερο. Πρέπει να έχουμε δικαιώματα διαχειριστή.

Μπορούμε να αρχικοποιήσουμε τη βάση στην αρχική της μορφή, χωρίς δεδομένα δηλαδή και τεστ.

Έλεγχος ροής

[ΕΠΕΞΕΡΓΑΣΙΑ ΧΡΗΣΤΗ]

Επιλέγουμε τον χρήστη που θέλουμε και τροποποιούμε τα στοιχεία του. Μπορούμε να του δώσουμε ή να του αφαιρέσουμε δικαιώματα διαχειριστή, να αλλάξουμε το όνομά του ή να τον διαγράψουμε. (εικόνα 3,21)

[ΝΕΟΣ ΧΡΗΣΤΗΣ]

Προσθέτουμε χρήστη με τη δυνατότητα διαχειριστή ή όχι. (εικόνα 3,22)

Κάθε τεστ αντιστοιχεί στον χρήστη που κάνει εισαγωγή στην εφαρμογή. Οι διαχειριστές μπορούν να δουν όλα τα τεστ, ενώ οι απλοί χρήστες μπορούν να δουν μόνο τα δικά τους (έχουν δικαίωμα να τα κάνουν ορατά και σε άλλους).

[ΑΡΧΙΚΟΠΟΙΗΣΗ ΒΑΣΗΣ]

Ο αρχικός διαχειριστής και μόνο έχει τον κωδικό ΡΥΚ. Με αυτόν τον κωδικό μπορεί να ξεκλειδώσει την εφαρμογή ή να εκκαθαρίσει τα δεδομένα και να επαναφέρει την βάση στην αρχική της κατάσταση.

4. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ

Για την υλοποίηση της εφαρμογής χρειάστηκε αρχικά η καταγραφή των απαιτήσεων και των απαραίτητων προδιαγραφών. Δημιουργήθηκε μια βάση δεδομένων σε περιβάλλον SQL SERVER EXPRESS 2008 R2 όπου καταχωρούνται και αποθηκεύονται τα δεδομένα και οι πίνακες της εφαρμογής. Το γραφικό περιβάλλον υλοποιήθηκε με VISUAL STUDIO 2010. Η γλώσσα που χρησιμοποιήθηκε για την υλοποίηση είναι η C# σε .NET Framework.

4.1 Το πλαίσιο λογισμικού .NET Framework

Το .NET Framework είναι ένα πλαίσιο λογισμικού (software framework) που προορίζεται για την πλατφόρμα των Windows. Αποτελείται από μια μεγάλη βιβλιοθήκη κλάσεων και υποστηρίζει μια πλειάδα γλωσσών προγραμματισμού με τη δυνατότητα η μια να μπορεί να χρησιμοποιηθεί από την άλλη. Τα προγράμματα που γράφονται για το .NET Framework εκτελούνται σε ένα περιβάλλον εκτέλεσης γνωστό ως Common Language Runtime (CLR), ενός ειδικού λογισμικού, σχεδιασμένου να υποστηρίζει την εκτέλεση προγραμμάτων και την απρόσκοπτη συνεργασία με το λειτουργικό σύστημα. Το CLR περιέχει μια εικονική μηχανή (virtual machine) που διαχειρίζεται την εκτέλεση ενός προγράμματος και παρέχει μια σειρά σημαντικών υπηρεσιών όπως ασφάλεια, διαχείριση μνήμης και διαχείριση εξαιρέσεων.

Τα προγράμματα που γράφονται για το .NET χρησιμοποιούν τη βιβλιοθήκη κλάσεων (class library) του .NET, η οποία δίνει πρόσβαση στο περιβάλλον εκτέλεσής του (runtime environment). Βασικές λειτουργίες όπως οι γραφικές διεπαφές χρηστών (Graphical User Interfaces – GUIs), η επικοινωνία με βάσεις δεδομένων, η κρυπτογραφία, η ανάπτυξη web εφαρμογών και οι δικτυακές επικοινωνίες παρέχονται μέσω του Application Programming Interface (API) του .NET και μπορούν να συνδυαστούν με κώδικα από τους προγραμματιστές για τη δημιουργία ολοκληρωμένων εφαρμογών. Όσο ένα πρόγραμμα περιορίζεται στη χρήση της βιβλιοθήκης κλάσεων του .NET, μπορεί να τρέχει οπουδήποτε υπάρχει εγκατεστημένο το περιβάλλον εκτέλεσης του .NET.

Τα σημαντικότερα χαρακτηριστικά και πλεονεκτήματα του .NET Framework είναι:

- Υποστήριξη πολλαπλών γλωσσών
- Ανάπτυξη βασισμένη σε components
- Μεταφερσιμότητα
- Απλή εγκατάσταση εφαρμογών
- Επικοινωνία με υπάρχων κώδικα
- Αξιοπιστία
- Αυτόματη διαχείριση μνήμης
- Ασφάλεια και απόδοση
- Υποστήριξη με επαγγελματικά εργαλεία

4.2 Το περιβάλλον SQL SERVER

Ο SQL Server είναι μια σχεσιακή βάση δεδομένων, η οποία αναπτύσσεται από τη Microsoft. Οι κύριες γλώσσες που χρησιμοποιούνται είναι η T-SQL και η ANSI SQL. Ο SQL Server βγήκε για πρώτη φορά στην αγορά το 1989 σε συνεργασία με την Sybase. Η κύρια μονάδα αποθήκευσης στοιχείων είναι μια βάση δεδομένων, η οποία αποτελείται από μια συλλογή πινάκων και κώδικα.

Η κεντρική βάση δεδομένων του SQL υποστηρίζει διαφορετικούς τύπους, συμπεριλαμβανομένων των ακεραίων αριθμών, αριθμών κινητής υποδιαστολής, δεκαδικών, αλφαριθμητικών, Varchar (σειρές χαρακτήρων μεταβλητού μήκους), δυαδικών αριθμών (για τα μη δομημένα δεδομένα), κειμένων (για κείμενα). Επιτρέπει επίσης καθορισμένους από το χρήστη σύνθετους τύπους δεδομένων (UDTs), δηλαδή τύπους που βασίζονται στους βασικούς τύπους αλλά μπορούν να τροποποιηθούν. Τα στοιχεία στη βάση δεδομένων αποθηκεύονται σε ένα (ή περισσότερα) αρχεία με επέκταση .mdf. Τα δευτεροβάθμια στοιχεία αποθηκεύονται στο αρχείο με επέκταση .ndf. Το αρχείο καταγραφής το οποίο περιέχει όλες τις πρόσφατες αλλαγές στη βάση δεδομένων αποθηκεύεται σε αρχείο με επέκταση .ldf.

Η ερώτηση (query) είναι ο κύριος τρόπος για την ανάκτηση στοιχείων από μια βάση δεδομένων. Η ερώτηση εκφράζεται χρησιμοποιώντας μια παραλλαγή της αποκαλούμενου SQL T-SQL, είναι μια διάλεκτος SQL που αναπτύχθηκε από την Microsoft και Sybase. Η T-SQL είναι πολύ κοντά στα ANSI standards που έχουν καθιερωθεί διεθνώς, σε αντιδιαστολή με άλλες διαλέκτους όπως η PL-SQL της Oracle που διαφέρουν περισσότερο από το ANSI standards. Η ερώτηση διευκρινίζει επεξηγηματικά αυτό που είναι να ανακτηθεί από την βάση δεδομένων. Η ερώτηση υποβάλλεται σε επεξεργασία από τον επεξεργαστή ερώτησης, ο οποίος υπολογίζει τη σειρά των βημάτων που θα είναι απαραίτητα για να ανακτηθούν τα ζητούμενα στοιχεία. Η ακολουθία ενεργειών απαραίτητων να εκτελέσουν μια ερώτηση καλείται σχέδιο ερώτησης.

Ο Microsoft SQL Server 2005 – 2008 περιλαμβάνει ένα module που λέγεται SQL CLR μέσω του οποίου ενσωματώνει το .NET μέσα στον SQL Server. Με το SQL CLR, οι αποθηκευμένες διαδικασίες μπορούν να γραφτούν σε οποιαδήποτε γλώσσα .NET συμπεριλαμβανομένου C# και VB.NET και να δημιουργήσουν μια stored procedure. Αυτό σημαίνει ότι ο SQL Server έχει όλες τις βιβλιοθήκες και πλεονεκτήματα του .NET, αυτόχθονα μέσα στο περιβάλλον του, τα οποία μπορεί να τα καλέσει οποιαδήποτε στιγμή.

4.3 To Visual Studio 2010

Το Visual Studio 2010 είναι ένα εντοπιζόμενο περιβάλλον που απλοποιεί ολόκληρο τον κύκλο προγραμματισμού από τη φάση σχεδίασης μέχρι και την ανάπτυξη. Στο περιβάλλον του Visual Studio οι προγραμματιστές μπορούν να χρησιμοποιήσουν υπάρχουσες δεξιότητες για να δημιουργήσουν μοντέλα ενός μεγάλου αριθμού εφαρμογών, να γράψουν κώδικα για αυτές, να τις δοκιμάσουν, να βρουν τα σφάλματα τους και να τις αναπτύξουν. Το Visual Studio απλοποιεί συνήθεις εργασίες και βοηθά τους προγραμματιστές να εξερευνήσουν την πλατφόρμα σε βάθος.

Παρέχει ισχυρά εργαλεία για τη διαχείριση των έργων, τη συντήρηση του κώδικα προέλευσης και τον εντοπισμό σφαλμάτων. Οι προγραμματιστές μπορούν να χρησιμοποιήσουν μη αυτόματες και αυτοματοποιημένες δοκιμές για να διασφαλίσουν ότι δημιουργούν τη σωστή εφαρμογή με το σωστό τρόπο.

Οι επιπλέον δυνατότητες του Visual Studio 2010 από τις προηγούμενες εκδόσεις είναι:

- Η νέα σύνδεση δεδομένων με μεταφορά και απόθεση στις εφαρμογές «Υποδομή Παρουσίασης των Windows» και Microsoft Silverlight καθιστά τη δημιουργία εφαρμογών Windows και εμπλουτισμένων εφαρμογών Internet ευκολότερη και ταχύτερη από ποτέ.
- Γρήγορη ανίχνευση της ροής εκτέλεσης ενός προγράμματος χωρίς κλήση του προγράμματος εντοπισμού σφαλμάτων.
- Η νέα δυνατότητα IntelliTrace καταγράφει το ιστορικό εκτέλεσης της εφαρμογής και εγγυάται ότι θα μπορεί να αναπαράγεται πάντα το σφάλμα που έχει αναφερθεί

4.4 Η γλώσσα προγραμματισμού C#

Η C# είναι μια ολοκληρωμένη αντικειμενοστραφής γλώσσα προγραμματισμού η οποία δημιουργήθηκε από τη Microsoft και είναι σχεδιασμένη για τη δημιουργία λογισμικού σε .NET Framework. Βασικό χαρακτηριστικό της είναι ότι δεν παράγει απευθείας κώδικα μηχανής, αλλά ένα ενδιάμεσο κώδικα που στοχεύει το .NET.

Δανείζεται πολλά στοιχεία και έχει παρόμοια σύνταξη με την C++ και την Java, κάνοντας την εκμάθηση της σχετικά εύκολη. Τα πάντα στην C# είναι αντικείμενα. Επιπλέον παρέχει άμεση πρόσβαση σε τεράστιες βιβλιοθήκες κλάσεων του .NET Framework και ασφάλεια των τύπων της.

4.5 Η εφαρμογή Rational Rose

Το Rational Rose είναι ένα προηγμένο πακέτο εργαλείων ανάπτυξης λογισμικού βασισμένων στη UML. Μπορεί να μοντελοποιήσει την αρχιτεκτονική ολόκληρης της εφαρμογής και να τη

μετατρέψει σε ένα πλήθος τεχνολογιών. Υποστηρίζει αντικείμενο-σχεσιακή απεικόνιση, παραγωγή λογικών σχημάτων και συγχρονισμό κώδικα και μοντέλου για ανάπτυξη σε Java και άλλες γλώσσες προγραμματισμού.

Τα πλεονεκτήματα του Rational Rose είναι τα εξής:

- Είναι κατάλληλο για μεγάλες εφαρμογές και πολυμελείς ομάδες ανάπτυξης λογισμικού.
- Οδηγεί σε καλύτερο σχεδιασμό με τη βοήθεια των μηχανισμών ελέγχου.
- Προωθεί την χρήση προτύπων και την τεκμηρίωση.
- Αυτοματοποιεί τις περισσότερες από τις διαδικασίες ρουτίνας των κατασκευαστών του συστήματος.
- Με τη χρήση σωστών τεχνικών, περιορίζει σημαντικά τα λάθη.

4.6 Διαγράμματα UML

Η Unified Modeling Language είναι η πρότυπη γλώσσα μοντελοποίησης στη μηχανική λογισμικού. Χρησιμοποιείται για τη γραφική απεικόνιση, προσδιορισμό, κατασκευή και τεκμηρίωση των στοιχείων ενός συστήματος λογισμικού. Μπορεί να χρησιμοποιηθεί σε διάφορες φάσεις ανάπτυξης, από την ανάλυση απαιτήσεων ως τον έλεγχο ενός ολοκληρωμένου συστήματος. Αποτελείται από ένα σύνολο προσυμφωνημένων όρων, συμβόλων και διαγραμμάτων που επιτρέπουν:

- την εμφάνιση των ορίων ενός συστήματος και των βασικών λειτουργιών του, χρησιμοποιώντας «περιπτώσεις χρήσης» (use-cases) και «actors».
- την επεξήγηση της πραγματοποίησης των περιπτώσεων χρήσης με «διαγράμματα αλληλεπίδρασης».
- την αναπαράσταση μιας στατικής δομής ενός συστήματος χρησιμοποιώντας «διαγράμματα κλάσεων».
- τη μοντελοποίηση της συμπεριφοράς των αντικειμένων με «διαγράμματα καταστάσεων».
- τη μοντελοποίηση της εργασιακής ροής με «διαγράμματα δραστηριοτήτων».
- την αποκάλυψη της υλοποίησης της αρχιτεκτονικής με «διαγράμματα συστατικών» και «ανάπτυξης».
- την επέκταση της λειτουργικότητας με «στερεότυπα».

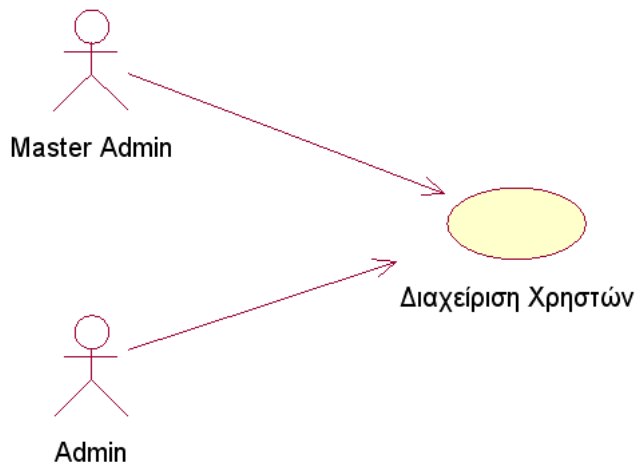
Τα διαγράμματα κλάσεων της UML ορίζουν γεωμετρικά σχήματα ως συμβολισμούς για τα αντικείμενα, τις κλάσεις και τις διασυνδέσεις, ενώ διαφόρων τύπων γραμμές χρησιμοποιούνται για να συνδέουν αυτά τα σχήματα και να υποδηλώνουν έτσι τον τρόπο που κληρονομούν, συνεργάζονται ή εξαρτώνται μεταξύ τους. Τα αντικείμενα της ίδιας κλάσης αναπαριστώνται με ένα μόνο γεωμετρικό σχήμα. Όταν ένα αντικείμενο χρησιμοποιεί κώδικα κάποιας άλλης κλάσης (π.χ. καλώντας μία μέθοδο της), σύμφωνα με το πρότυπο της UML υπάρχει μία «εξάρτηση» (dependency) μεταξύ τους η οποία αναπαρίσταται με μία διακεκομμένη γραμμή. Αυτή η εξάρτηση μπορεί να είναι «συσχέτιση» (association), ένας τύπος εξάρτησης που υπονοεί πραγματική συνύπαρξη στη μνήμη στιγμιοτύπων των συμμετεχόντων κλάσεων κατά τον χρόνο εκτέλεσης, «συνάθροιση» (aggregation), ένας τύπος συσχέτισης ο οποίος σημαίνει ότι το ένα αντικείμενο μπορεί να περιέχει στιγμιότυπα της άλλης κλάσης ως γνωρίσματα του, ή «σύνθεση» (composition), ένας πιο ισχυρός τύπος συνάθροισης που υπονοεί πως ο χρόνος ζωής των αντικειμένων είναι κοινός (δημιουργούνται και καταστρέφονται στη μνήμη ταυτόχρονα). Καθεμία από αυτές τις σχέσεις συμβολίζεται οπτικά με έναν διαφορετικό τύπο γραμμής μεταξύ των συμμετεχόντων κλάσεων, ενώ μπορεί να υπάρχουν και εξαρτήσεις οι οποίες δεν είναι καν συσχετίσεις (π.χ. όταν ένα αντικείμενο καλεί μία στατική μέθοδο κάποιας κλάσης).

Οι χρήσεις της UML είναι στα:

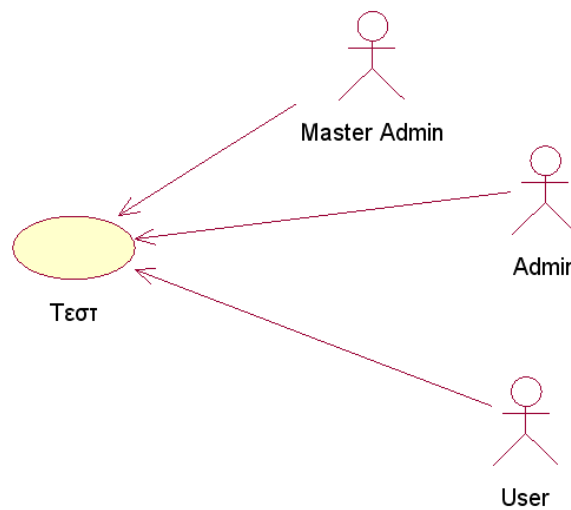
- Πληροφοριακά Συστήματα
- Τεχνολογικά Συστήματα
- Συστήματα Πραγματικού Χρόνου
- Κατανεμημένα Συστήματα
- Συστήματα Επιχειρήσεων

Διαγράμματα Περιπτώσεων Χρήσης

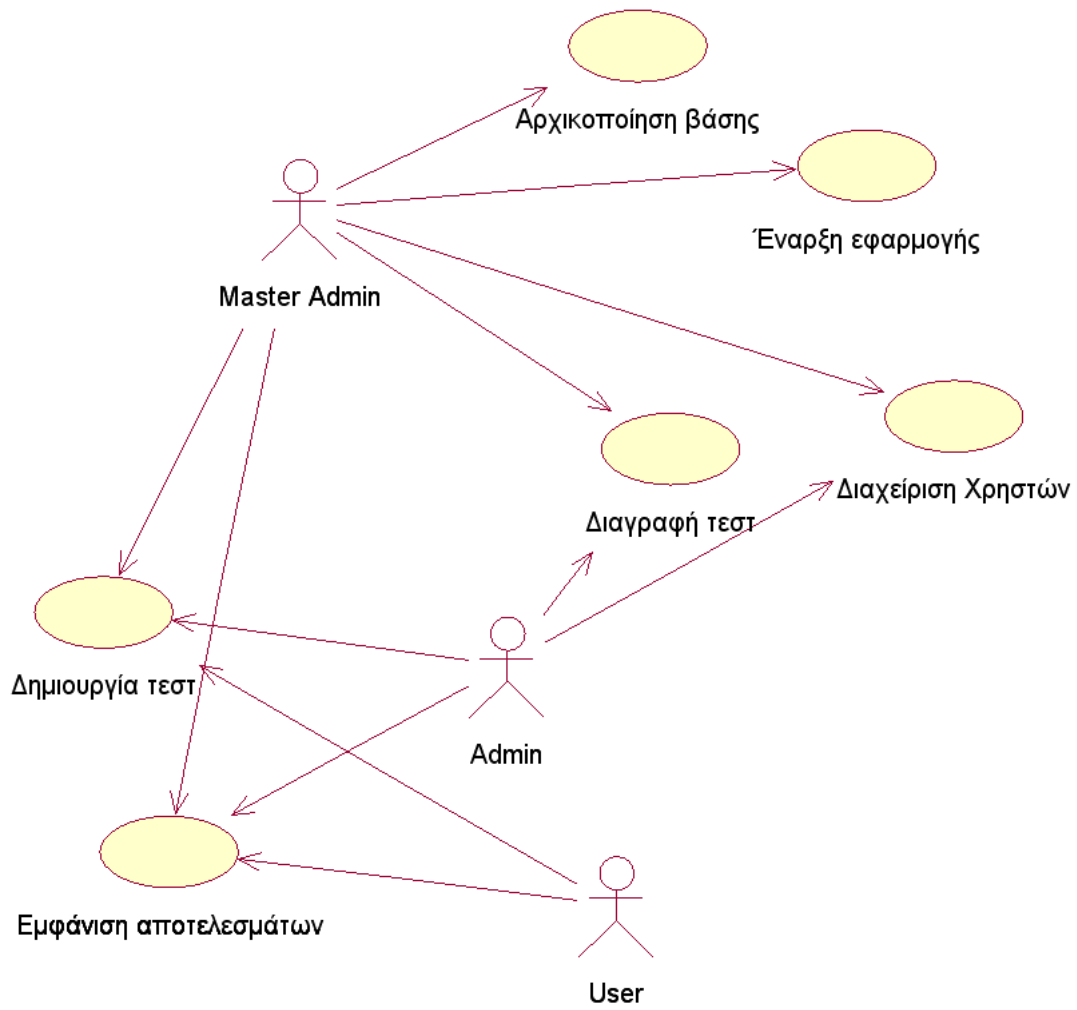
Τα διαγράμματα περιπτώσεων χρήσης (use case) χρησιμοποιούνται για τη μοντελοποίηση του πλαισίου λειτουργίας του συστήματος και των προδιαγραφών του.



Εικόνα 4.1 Διάγραμμα περιπτώσεων χρήσης (Διαχείριση Χρηστών)



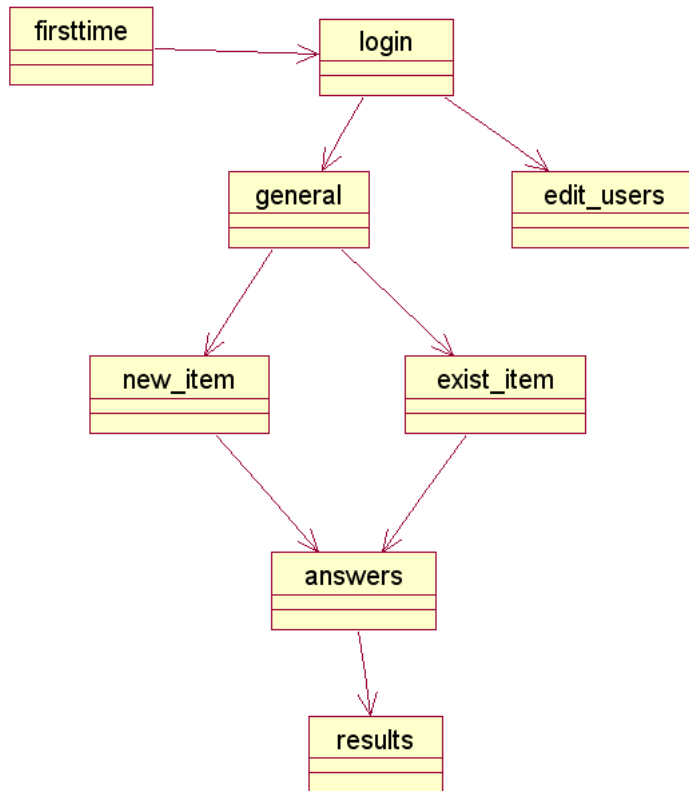
Εικόνα 4.2 Διάγραμμα περιπτώσεων χρήσης (Τεστ)



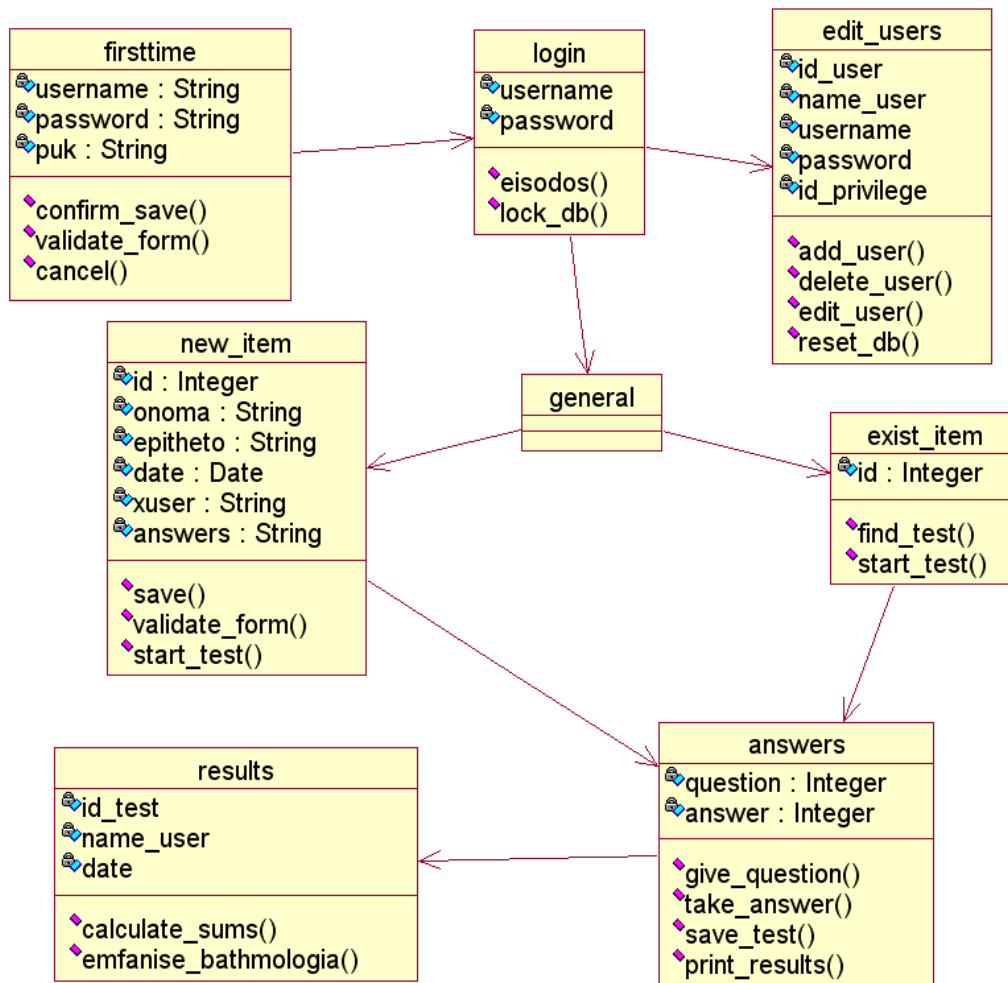
Εικόνα 4.3 Διάγραμμα περιπτώσεων χρήσης (Ολοκληρωμένη Εφαρμογή)

Διαγράμματα Κλάσεων

Το διάγραμμα των κλάσεων (class diagram) ενός συστήματος είναι ένα διάγραμμα δομής που περιέχει τις κλάσεις μαζί με τους αντίστοιχους δεσμούς εξάρτησης, γενίκευσης και σύνδεσης.

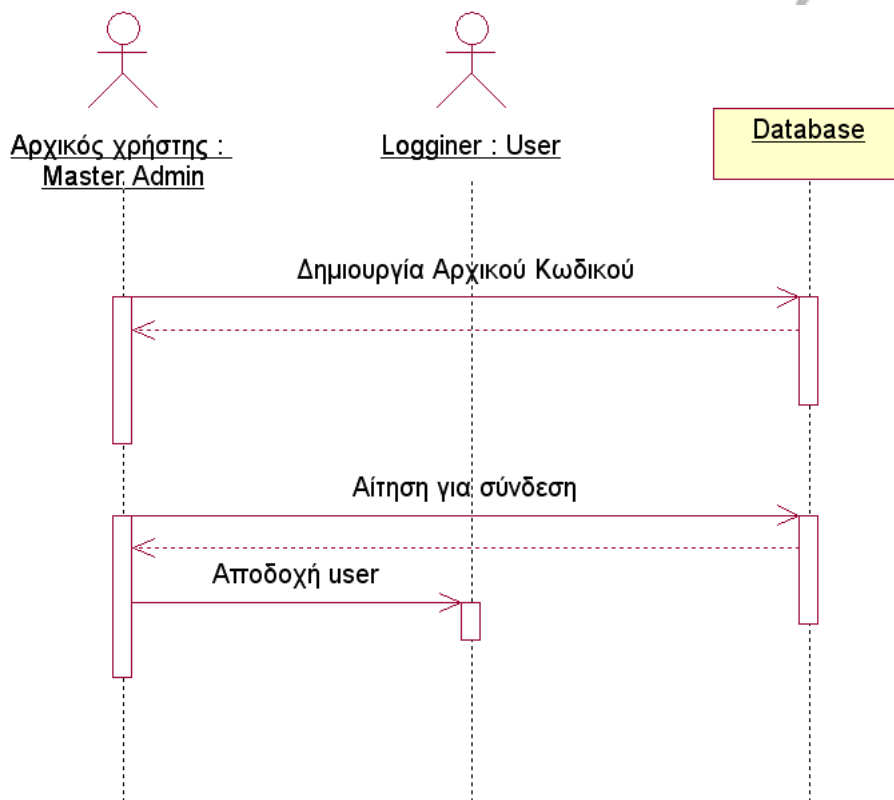


Εικόνα 4.4 Διάγραμμα κλάσης (1^η έκδοση)

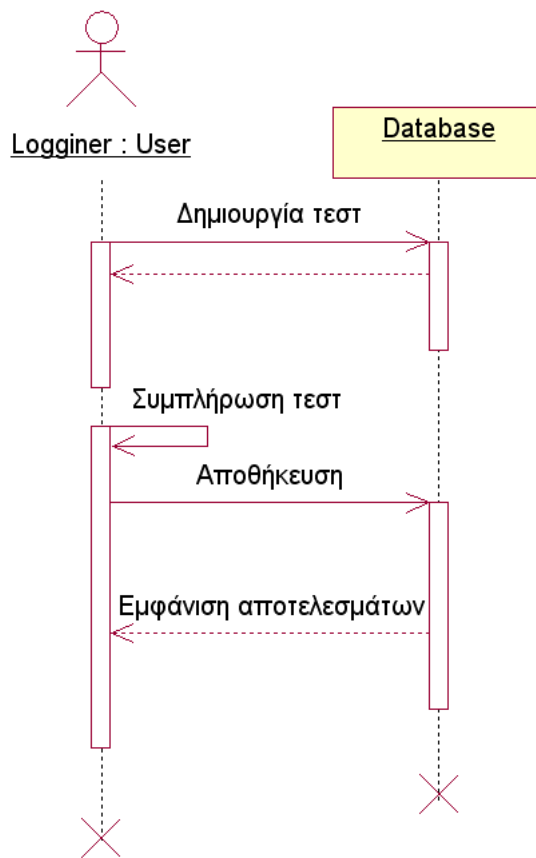
Εικόνα 4.5 Διάγραμμα κλάσης (2^η έκδοση)

Διαγράμματα Ακολουθίας

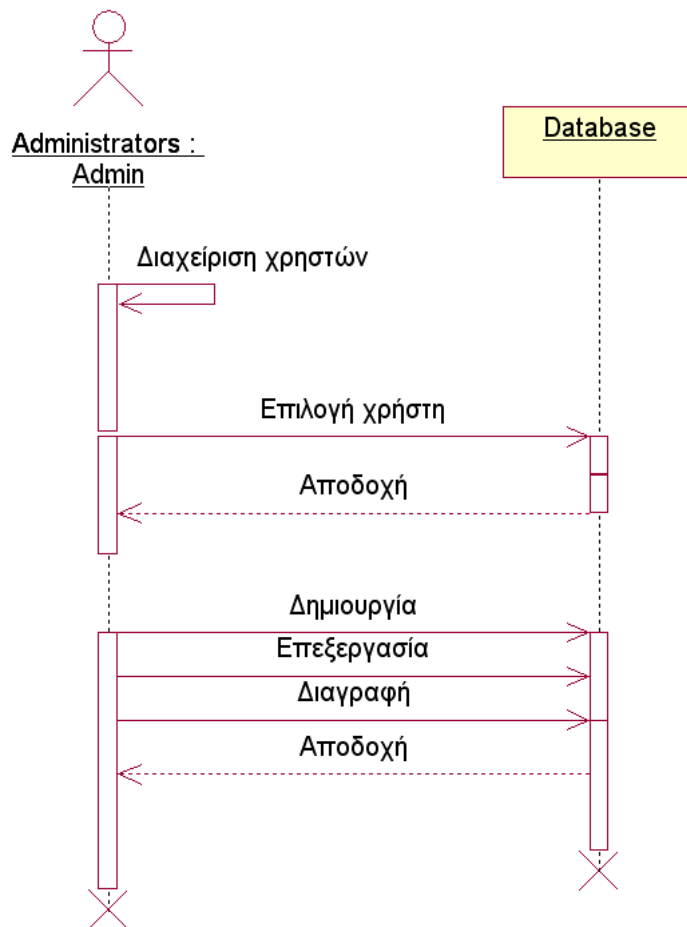
Το διάγραμμα ακολουθίας (sequence diagram) είναι ένα διάγραμμα αλληλεπίδρασης που παρουσιάζει τον τρόπο που διαφορετικά αντικείμενα συνεργάζονται μεταξύ τους σε μια χρονική ακολουθία.



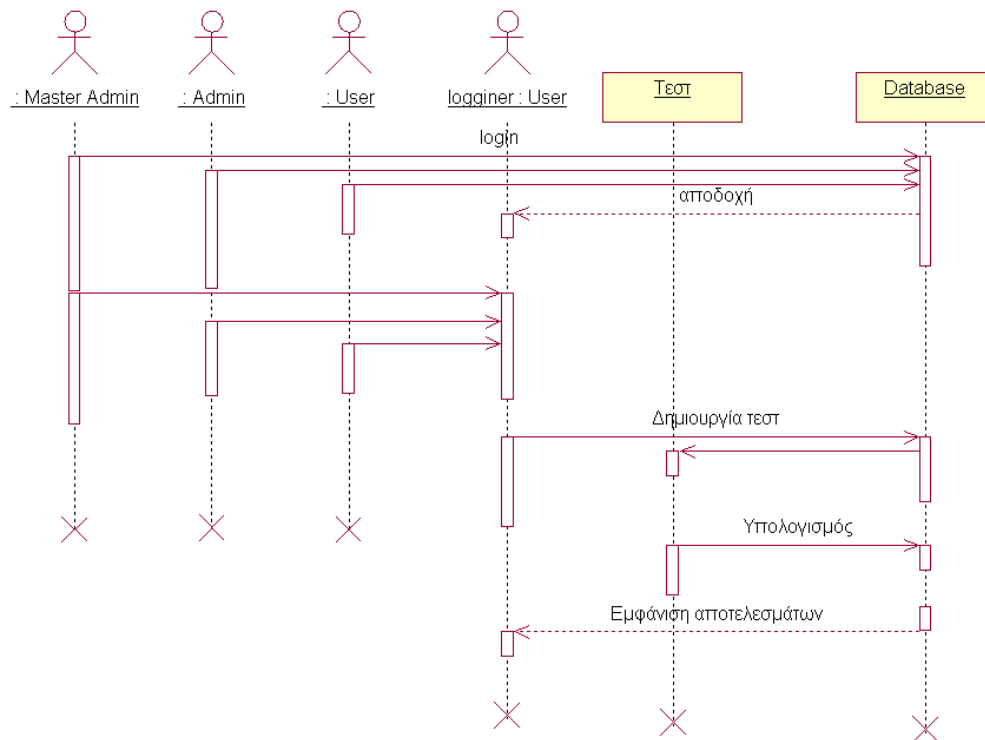
Εικόνα 4.6 Διάγραμμα ακολουθίας (Master Admin-User-Database)



Εικόνα 4.7 Διάγραμμα ακολουθίας (User-Database)



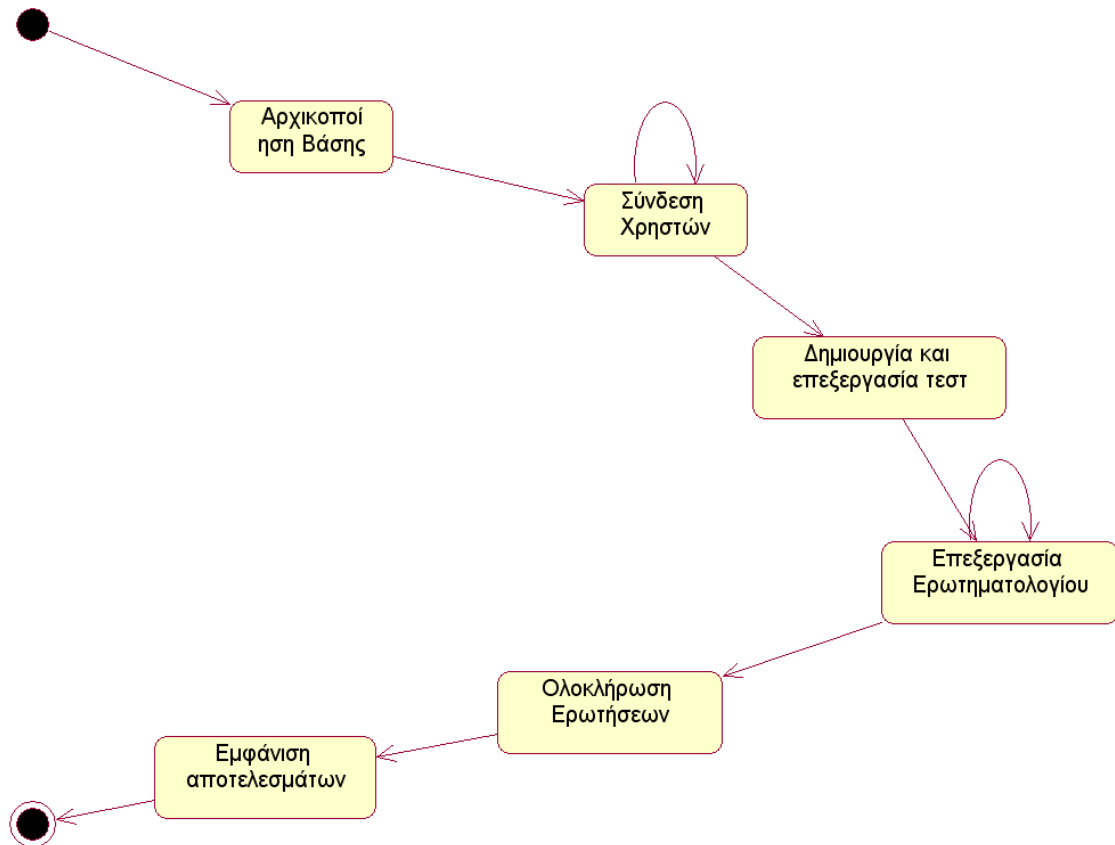
Εικόνα 4.8 Διάγραμμα ακολουθίας (Admin-Database)



Εικόνα 4.9 Διάγραμμα ακολουθίας (Ολοκληρωμένη Εφαρμογή)

Διάγραμμα Καταστάσεων

Το διάγραμμα καταστάσεων (statechart diagram) είναι ένα διάγραμμα συμπεριφοράς που εμφανίζει μια μηχανή καταστάσεων με έμφαση στις μεταπτώσεις μεταξύ καταστάσεων από διάφορα γεγονότα.

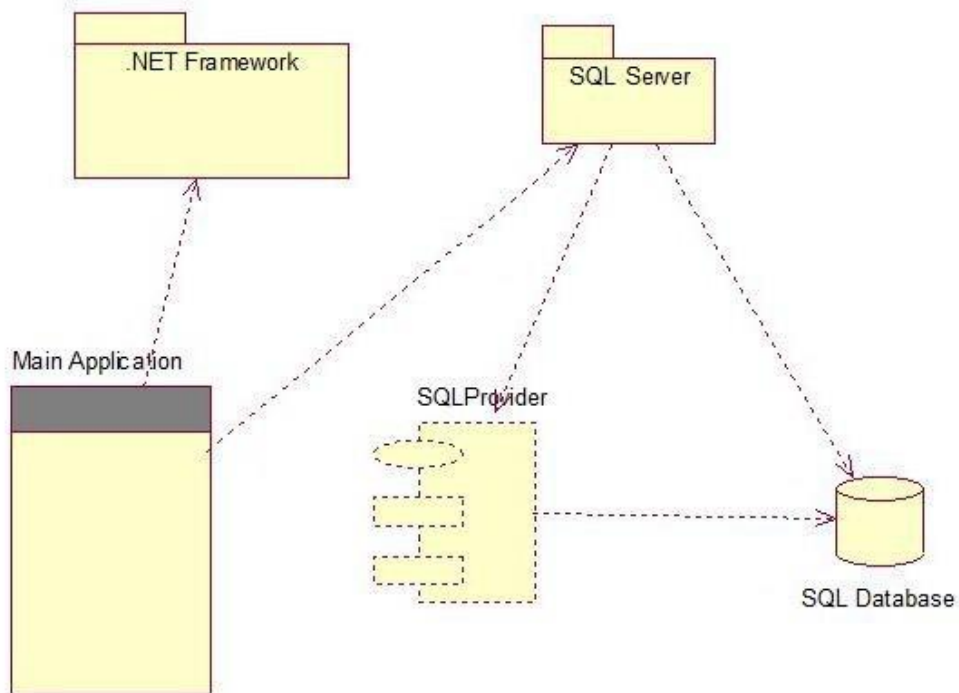


Εικόνα 4.10 Διάγραμμα καταστάσεων

Διάγραμμα Εξαρτημάτων

Το διάγραμμα εξαρτημάτων (component diagram) είναι ένα διάγραμμα δομής υλοποίησης που χρησιμοποιείται για να μοντελοποιήσει:

- Πηγαίο κώδικα
- Εκτελέσιμες εκδόσεις
- Βάσεις δεδομένων
- Δυναμικά προσαρμοζόμενα συστήματα

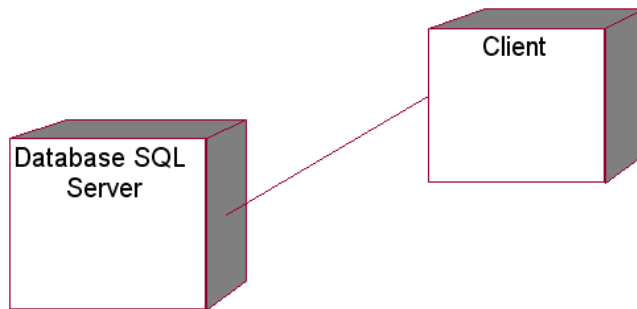


Εικόνα 4.11 Διάγραμμα εξαρτημάτων

Διάγραμμα Ανάπτυξης

Το διάγραμμα ανάπτυξης (deployment diagram) είναι ένα διάγραμμα δομής υλοποίησης που παρουσιάζει τον τρόπο διαμόρφωση των υπολογιστικών κόμβων του συστήματος κατά τη λειτουργία του. Περιλαμβάνει:

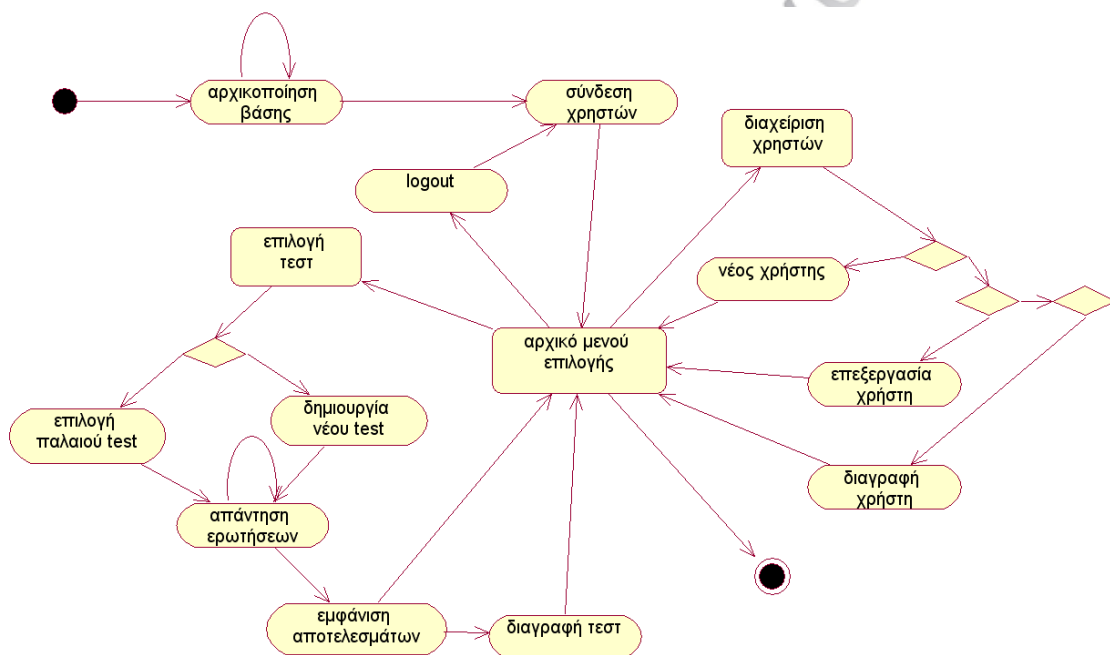
- Υπολογιστικούς κόμβους
- Σχέσης εξάρτησης και σύνδεσης
- Εξαρτήματα (μέσα σε κόμβους)
- Πακέτα (που ομαδοποιούν κόμβους)



Εικόνα 4.12 Διάγραμμα ανάπτυξης

Διάγραμμα Δραστηριοτήτων

Το διάγραμμα δραστηριοτήτων (activity diagram) είναι ένα διάγραμμα συμπεριφοράς που εμφανίζει μεταπτώσεις. Η διαφορά του από το διάγραμμα καταστάσεων είναι πως οι μεταπτώσεις είναι ανάμεσα σε διαφορετικές δραστηριότητες.



Εικόνα 4.13 Διάγραμμα δραστηριοτήτων

4.7 Τμήματα Κώδικα Εφαρμογής

Παρακάτω παρατίθενται μερικά κομμάτια από τον κώδικα υλοποίησης της εφαρμογής.

Ορισμός PUK και είσοδος στην εφαρμογή

Κώδικας :

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace WindowsFormsApplication1
{
    public partial class frm_firsttimePUK : Form
    {
        public frm_firsttimePUK()
        {
            InitializeComponent();
        }

        private void frm_firsttimePUK_Load(object sender, EventArgs e)
        {
        }

        //auksano kai meiono ta psifia tou puk
        private void pictureBox1_Click(object sender, EventArgs e)
        {
            int x=Int32.Parse(lbl1.Text);
            x++; if (x == 10) { x = 0; }
            lbl1.Text = x.ToString();
        }

        private void pictureBox2_Click(object sender, EventArgs e)
        {
            int x = Int32.Parse(lbl2.Text);
            x++; if (x == 10) { x = 0; }
            lbl2.Text = x.ToString();
        }

        private void pictureBox3_Click(object sender, EventArgs e)
        {
            int x = Int32.Parse(lbl3.Text);
            x++; if (x == 10) { x = 0; }
            lbl3.Text = x.ToString();
        }

        private void pictureBox4_Click(object sender, EventArgs e)
        {
            int x = Int32.Parse(lbl4.Text);
            x++; if (x == 10) { x = 0; }
            lbl4.Text = x.ToString();
        }
    }
}
```

```
private void pictureBox8_Click(object sender, EventArgs e)
{
    int x = Int32.Parse(lbl11.Text);
    x--; if (x == -1) { x = 9; }
    lbl11.Text = x.ToString();
}

private void pictureBox7_Click(object sender, EventArgs e)
{
    int x = Int32.Parse(lbl12.Text);
    x--; if (x == -1) { x = 9; }
    lbl12.Text = x.ToString();
}

private void pictureBox6_Click(object sender, EventArgs e)
{
    int x = Int32.Parse(lbl13.Text);
    x--; if (x == -1) { x = 9; }
    lbl13.Text = x.ToString();
}

private void pictureBox5_Click(object sender, EventArgs e)
{
    int x = Int32.Parse(lbl14.Text);
    x--; if (x == -1) { x = 9; }
    lbl14.Text = x.ToString();
}

//apothikeyse to puk kai sundesou sthn bash
private void button1_Click(object sender, EventArgs e)
{
    int intpuk=0;
    intpuk = Int32.Parse(lbl11.Text) * 1000 + Int32.Parse(lbl12.Text)
        * 100 + Int32.Parse(lbl13.Text) * 10 + Int32.Parse(lbl14.Text);

    SqlConnection db1 = new
        SqlConnection(global_variables.my_database);
    string txt1 = "update tbl_globals set value='"+intpuk+"'where
        name='PUK'";
    SqlDataAdapter dat1 = new SqlDataAdapter(txt1, db1);
    DataTable tbl1 = new DataTable();
    dat1.Fill(tbl1);
    db1.Close();
    MessageBox.Show("ΜΠΟΡΕΙΤΕ ΝΑ ΣΥΝΔΕΘΕΙΤΕ ΚΑΝΟΝΙΚΑ ΣΤΗΝ ΒΑΣΗ",
        "ΚΑΛΩΣ ΗΡΘΑΤΕ ΣΤΗΝ ΕΦΑΡΜΟΓΗ");
    this.Close();
    frm_menou f1 = new frm_menou();
    f1.Show();
}
}
```

Έλεγχος και εισαγωγή κωδικών και είσοδος στην εφαρμογή

Κώδικας:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.Data.SqlClient;

namespace WindowsFormsApplication1
{
    public partial class frm_menu : Form
    {
        private int is_login;
        public int openfromwindow = 1;

        public frm_menu()
        {
            InitializeComponent();
            set_is_login(0);
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }

        public void set_is_login(int x) {
            is_login = x;
        }

        int check_is_login() {
            return is_login;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            this.Hide();
            frm_login f1 = new frm_login();
            f1.Show();
        }

        private void set_program_lock(int x)
        {
            //anoikse thn bash ston pinaka tbl_globals kai kleidose thn bash
            //thetontas to db_lock=1
            try
            {
                SqlConnection my_database = new
                    SqlConnection(global_variables.my_database);
                SqlDataAdapter my_sqladapter = new SqlDataAdapter("update
                    tbl_globals set value="+x+" where name='db_lock'",
                    my_database);
            }
        }
    }
}
```

```

        DataTable my_datatable = new DataTable();
        my_sqladapter.Fill(my_datatable);
        my_database.Close();

        frm_unlock_application F2 = new frm_unlock_application();
        this.Close();
        F2.Show();
        F2.Focus();
    }
    catch (Exception e1) {
        MessageBox.Show("ΠΡΟΒΛΗΜΑ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
            (db1)\nΕΠΙΚΟΙΝΩΝΗΣΤΕ ΜΕ ΤΟΝ ΔΗΜΙΟΥΡΓΟ ΤΟΥ
            ΠΡΟΓΡΑΜΜΑΤΟΣ\nΜας συγχωρείτε για την αναξιοπιστία της
            εφαρμογής\n" + e1.message);
        Application.Exit();
    }
}
private int check_database_is_locked()
{
    //anoikse thn bash ston pinaka tbl_globals kai des an einai
    kleidomenh
    //tsekarontas an to db_lock==1
    try
    {

        SqlConnection my_database = new
            SqlConnection(global_variables.my_database);
        SqlDataAdapter my_sqladapter = new SqlDataAdapter("select
            value from tbl_globals where name='db_lock'",
            my_database);
        DataTable my_datatable = new DataTable();
        my_sqladapter.Fill(my_datatable);

        my_database.Close();
        //elegxo an einai kleidomenh kai epistrefo 1 an einai kai 0
        an den einai
        if (my_datatable.Rows[0][0].ToString() == "1")
        {
            return 1;
        }
        else
        {
            return 0;
        }
    }
    catch (Exception e1)
    {
        MessageBox.Show("ΠΡΟΒΛΗΜΑ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
            (db2)\nΕΠΙΚΟΙΝΩΝΗΣΤΕ ΜΕ ΤΟΝ ΔΗΜΙΟΥΡΓΟ ΤΟΥ
            ΠΡΟΓΡΑΜΜΑΤΟΣ\nΜας συγχωρείτε για την αναξιοπιστία της
            εφαρμογής\n" + e1.message);
        Application.Exit();
        return -1;
    }
}

//anoikse thn forma gia nea test
public void opennewitem(object sender, EventArgs e)
{
    this.Close();
}

```

```
        frm_new_item f1 = new frm_new_item();
        f1.Show();
    }

    //des an mpikes sthn efarmogh gia proth fora
    private bool check_first_time()
    {
        try
        {
            SqlConnection db1 = new
                SqlConnection(global_variables.my_database);
            string txt1 = "select value from tbl_globals where
                name='visit'";
            SqlDataAdapter dat = new SqlDataAdapter(txt1, db1);
            DataTable tbl1 = new DataTable();
            dat.Fill(tbl1);
            db1.Close();

            if (Int32.Parse(tbl1.Rows[0][0].ToString()) == 0)
            {
                return true;
            }
            else
            {
                return false;
            }
        }
        catch (Exception e1)
        {
            MessageBox.Show(e1.Message);
            Environment.Exit(0);
        }
        return false;
    }

    //ayksanei ton metriti ton episkepseon ths bashs
    private void visit_one_more_time()
    {
        SqlConnection db1 = new
            SqlConnection(global_variables.my_database);
        string txt1 = "select value from tbl_globals where
            name='visit'";
        SqlDataAdapter dat = new SqlDataAdapter(txt1, db1);
        DataTable tbl1 = new DataTable();
        dat.Fill(tbl1);

        int times = Int32.Parse(tbl1.Rows[0][0].ToString());
        times++;
        string txt2 = "update tbl_globals set value="+times+" where
            name='visit'";
        SqlDataAdapter dat2 = new SqlDataAdapter(txt2, db1);
        DataTable tbl2 = new DataTable();
        dat2.Fill(tbl2);

        db1.Close();
    }

    private void frm_menu_Load(object sender, EventArgs e)
    {
```

```

//elegxo an einai i proth fora pou mpaineis sthn bash
if (check_first_time())
{
    DialogResult dl= MessageBox.Show("Είναι η πρώτη φορά που
    χρησιμοποιείτε το πρόγραμμά μας!\nθα σας παρακαλούσα να
    αποθηκεύσετε το username και το password του ιδιοκτήτη
    του προγράμματος καθώς και το PUK το οποίο επαναφέρει τη
    βάση σε περίπτωση που κλειδώσετε κατά λάθος την
    εφαρμογή\nθέλετε να προχωρήσω;", "ΠΡΟΣΟΧΗ",
    MessageBoxButtons.YesNo);

    if (dl == DialogResult.Yes)
    {
        visit_one_more_time();//auksano to plithos ton
        episkepseon
        frm_firsttime f1 = new frm_firsttime();
        f1.Show();
        this.Close();
    }
    else
    {
        MessageBox.Show("Επιλέξατε να κλείσετε την
        εφαρμογή.\nΤην επόμενη φορά που θα χρησιμοποιήσετε το
        πρόγραμμα, θα σας ζητηθεί η εισαγωγή του username και
        του κωδικού...", "ΕΞΟΔΟΣ", MessageBoxButtons.OK);

        Environment.Exit(1);
    }
}
else
{
    visit_one_more_time();//auksano to plithos ton episkepseon

    //elegxo an h bash prepei na kleidothei giati eginan polles
    prospatheies apotyximenou login
    if (check_database_is_locked() == 1)
    {
        //kleidose thn bash
        set_program_lock(1);
    }
    else
    {
        //otan anoigo to parathiro na tsekarei an exo kanei
        login
        //exei ginei login sosto
        if (my_functions.check_correct_login())
        // if (1 == 1)
        {
            //epanefere ton metriti prospatheion lathous login
            my_functions.set_global_count_error(1);
            //krypse to koumpi login
            btn_login.Hide();

            //dimiourgise neo koumpi gia neo test
            Button btn_neo = new Button();
            btn_neo.Name = "btn_new_exam";
            btn_neo.Text = "NEO ΤΕΣΤ";
        }
    }
}

```

```
btn_neo.Location = new System.Drawing.Point(58,
    230);
btn_neo.Font = new System.Drawing.Font("Microsoft
    Sans Serif", 16);
btn_neo.Size = new System.Drawing.Size(200, 34);
btn_neo.Visible = true;
btn_neo.Cursor = System.Windows.Forms.Cursors.Hand;
btn_neo.Click += new EventHandler(opennewitem);
this.Controls.Add(btn_neo);

//dimiourgise neo koumpi gia yparxon test
Button btn_exist = new Button();
btn_exist.Name = "btn_exist_exam";
btn_exist.Text = "ΑΠΟΘ. ΤΕΣΤ";
btn_exist.Location = new System.Drawing.Point(58,
    230 + 34 + 2);
btn_exist.Font = new System.Drawing.Font("Microsoft
    Sans Serif", 16);
btn_exist.Size = new System.Drawing.Size(200, 34);
btn_exist.Cursor =
    System.Windows.Forms.Cursors.Hand; ;
btn_exist.Visible = true;
btn_exist.Click += new
    EventHandler(edit_exist_item);
this.Controls.Add(btn_exist);

//dimiourgise neo koumpi gia xristes
Button btn_users = new Button();
btn_users.Name = "btn_users";
btn_users.Text = "ΕΠΓΑΛΕΙΑ";
btn_users.Location = new System.Drawing.Point(58,
    230 + 34 + 34 + 4);
btn_users.Font = new System.Drawing.Font("Microsoft
    Sans Serif", 16);
btn_users.Size = new System.Drawing.Size(200, 34);
btn_users.Click += new EventHandler(fnc_click_user);
btn_users.Cursor =
    System.Windows.Forms.Cursors.Hand;
btn_users.Visible = true;
this.Controls.Add(btn_users);

//dimiourgise neo koumpi gia xristes
Button btn_logout = new Button();
btn_logout.Name = "btn_logout";
btn_logout.Text = "logout";
btn_logout.Location = new System.Drawing.Point(58,
    230 + 34 + 34 + 4 + 34);
btn_logout.Font = new System.Drawing.Font("Microsoft
    Sans Serif", 16);
btn_logout.Size = new System.Drawing.Size(200, 34);
btn_logout.Click += new EventHandler(fnc_logout);
btn_logout.Cursor =
    System.Windows.Forms.Cursors.Hand;
btn_logout.Visible = true;
this.Controls.Add(btn_logout);

}
//den exei ginei login
else
{
    try
    {
```



```
//emfanise to koumpi gia login
btn_login.Show();

//emfanise poses prospatheies sou apomenoun prin
kleidosei to programma
Label l1 = new Label();
l1.Name = "lbl_count_error";
l1.Location = new System.Drawing.Point(58, 293);
l1.AutoSize = true;
l1.Visible = true;
this.Controls.Add(l1);

//na prosthetei sto plithos ton akyron
prospatheion (global metabliti)
SqlConnection db1 = new
SqlConnection(global_variables.my_database);
string qrl = @"select value from tbl_globals
where name='count_error'";
SqlDataAdapter dat = new SqlDataAdapter(qrl,
db1);
DataTable dt1 = new DataTable();
dat.Fill(dt1);
db1.Close();
global_variables.global_count_error_login =
Int32.Parse(dt1.Rows[0][0].ToString());

if (openfromwindow == 0)
{
    global_variables.global_count_error_login +=
    1;
}

my_functions.set_global_count_error
(global_variables.global_count_error_login);

if (global_variables.global_count_error_login <=
3)
{
    l1.Text = "ΕΧΕΤΕ " + (4 -
    global_variables.
    global_count_error_login).ToString() + "
    προσπάθειες για είσοδο";
}
else if
(global_variables.global_count_error_login > 3)
{
    l1.Text = "Η εφαρμογή κλειδώθηκε.
    Επικοινωνήστε με τον διαχειριστή
    \νουσστήματος για ξεκλείδωμα";
    set_program_lock(1);
}

l1.Show();
}
catch (Exception e11)
{
    MessageBox.Show(e11.ToString());
}
```

```
        }
    }
}

//pigaine sta ergaleia
void fnc_click_user(object sender, EventArgs e)
{
    this.Close();
    frm_edit_users f1 = new frm_edit_users();
    f1.Show();
}

//anoikse thn forma gia yparxonta tests
void edit_exist_item(object sender, EventArgs e)
{
    this.Close();
    frm_exist_item f1 = new frm_exist_item();
    f1.Show();
}

//kane logout kai ksanabges sthn idia forma
protected void fnc_logout(object sender, EventArgs e)
{
    string kaneis = "";
    my_functions.set_login_name(kaneis);
    this.Close();
    frm_menou f1 = new frm_menou();
    f1.Show();
}
}
```

Εμφάνιση αποθηκευμένων τεστ και επιλογή ολοκλήρωσης τους αν είναι ανολοκλήρωτα

Κώδικας:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace WindowsFormsApplication1
{
    public partial class frm_exist_item : Form
    {
        public frm_exist_item()
        {
            InitializeComponent();
        }

        //bale xrono na pairneis thn ora apo to systima
        private void timer1_Tick(object sender, EventArgs e)
        {
            string md = DateTime.Now.ToString("d/MM/yyyy");
            string mt = DateTime.Now.ToString("HH:mm:ss");
            txt_mydate.Text = md;
            txt_mytime.Text = mt;
        }

        //bres ta test kai balta sto combobox gia epilogh
        private void Find_Exist_test(int epilogh)
        {
            try
            {
                comboboxfind.Items.Clear();
                //anoikse thn bash na pareis ta test
                SqlConnection dbl = new
                    SqlConnection(global_variables.my_database);

                string qr1 = "SELECT id,epitheto,date,answers from tbl_test
                    where xuser='" + txt_login_user.Text + "' OR xuser='*';";

                SqlDataAdapter dat1 = new SqlDataAdapter(qr1, dbl);
                DataTable tbl1 = new DataTable();
                dat1.Fill(tbl1);

                int i, j; //boithitikes gia for
                string str2, stranswers;
                char[] answchar; //pinakas char me tis erotiseis
                                //apo thn bash
                int[] answint; //pinakas int me tis erotiseis
                                //apo thn bash
                int count0 = 0; //metritis
            }
        }
    }
}
```

```
//dimiourgise ta stoixeia tou pinaka me to plithos ton
erotiseon tou test
answint = new int[global_variables.count_erotiseis];

//gia kathe eggraphh
for (i = 0; i < tbl1.Rows.Count; i++)
{
    count0 = 0;
    //ftiakse to select
    str2 = tbl1.Rows[i][0].ToString() + " " +
        tbl1.Rows[i][1].ToString() + " " +
        tbl1.Rows[i][2].ToString();

    //an einai patimenh h epilogh na emfanizontai mono
    ta anoloklirota
    if (checkboxfind1.Checked == true ||
        checkBoxfind2.Checked==true)
    {
        //pare tis erotiseis
        stranswers = tbl1.Rows[i][3].ToString();
        //spastes se pinaka char
        answchar = stranswers.ToArray();
        //gia kathe mia erotisi
        for (j = 0; j < answchar.Count(); j++)
        {
            //ftiakse ton pinaka int me tis erotiseis
            answint[j] =
                Int32.Parse(answchar[j].ToString());
            //metra se poses den edose apantisi
            if (answint[j] == 0)
            {
                count0++;
            }
        }
        //an den exei dosei se oles apantisi emfanise
        thn eggraphh sto combo
        if (count0 != 0 && checkboxfind1.Checked==true)
        {
            comboboxfind.Items.Add(str2);
        }
        //an exei dosei se oles apantisi emfanise thn
        eggraphh sto combo
        else if (count0 ==0 && checkBoxfind2.Checked ==
            true)
        {
            comboboxfind.Items.Add(str2);
        }
    }
    //an einai patimenh h epilogh na emfanizontai ola ta
    test
    else
    {
        comboboxfind.Items.Add(str2);
    }
}

    dbl.Close();
}
catch (Exception e1)
{
```

```
        MessageBox.Show("ΠΡΟΒΛΗΜΑ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ\nΕΠΙΚΟΙΝΩΝΗΣΤΕ  
        ΜΕ ΤΟΝ ΔΗΜΙΟΥΡΓΟ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ\nΜας συγχωρείτε για  
        την αναξιοπιστία της εφαρμογής\n" + e1);  
    }  
  
    }  
private void frm_exist_item_Load(object sender, EventArgs e)  
{  
    //bale sto txt_login na emfanizete to onoma aytou pou ekane  
    login sthn efarmogh  
    txt_login_user.Text = my_functions.take_login_name();  
  
    Find_Exist_test(1);  
  
    }  
  
    //kleise thn forma kai pigaine pali sto menou  
private void btn_close_Click(object sender, EventArgs e)  
{  
    frm_menou f1 = new frm_menou();  
    f1.Show();  
    this.Close();  
    }  
  
private void checkboxfind1_CheckedChanged(object sender, EventArgs  
    e)  
{  
    Find_Exist_test(1);  
    //an einaipatimeno to check na emfanisei minima  
    if (checkboxfind1.Checked == true)  
    {  
        checkBoxfind2.Checked = false;  
        pic_err.Visible = true;  
        lbl_error.Visible = true;  
        lbl_error.Text = "ΜΟΝΟ ΑΝΟΛΟΚΛΗΡΩΤΑ ΤΕΣΤ";  
    }  
    else  
    {  
        pic_err.Visible = false;  
        lbl_error.Visible = false;  
    }  
    }  
  
    //brethike h eggrafh kai efanizo ta stoixeia  
    //kai ksekleidose to koumpi gia na synexizo to test  
private void comboboxfind_SelectedIndexChanged(object sender,  
    EventArgs e)  
{  
    lbl_find_date.Visible = true;  
    lbl_find_epitheto.Visible = true;  
    lbl_find_onoma.Visible = true;  
    lbl_txt_find.Visible = true;  
    txt_find_date.Visible = true;  
    txt_find_epitheto.Visible = true;  
    txt_find_onoma.Visible = true;  
  
    string[] words_in_combo=comboboxfind.Text.Split(' ');  
    int kodikos_epiloghs = Int32.Parse(words_in_combo[0]);  
  
    try  
    {  
        //anoikse thn bash na pareis ta test
```

```

        SqlConnection db1 = new
            SqlConnection(global_variables.my_database);
        string qrl = "SELECT id,epitheto,onoma,date from tbl_test
            where id='"+kodikos_epiloghs+"'";

        SqlDataAdapter dat1 = new SqlDataAdapter(qrl, db1);
        DataTable tbl1 = new DataTable();
        dat1.Fill(tbl1);
        txt_kodikos.Text = tbl1.Rows[0][0].ToString();
        txt_find_epitheto.Text = tbl1.Rows[0][1].ToString();
        txt_find_onoma.Text = tbl1.Rows[0][2].ToString();
        txt_find_date.Text = tbl1.Rows[0][3].ToString();
        btn_begin.Enabled = true;

        db1.Close();
    }
    catch (Exception e1)
    {
        MessageBox.Show("ΠΡΟΒΛΗΜΑ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ\nΕΠΙΚΟΙΝΩΝΗΣΤΕ
            ΜΕ ΤΟΝ ΔΗΜΙΟΥΡΓΟ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ\nΜας συγχωρείτε για
            την αναξιοπιστία της εφαρμογής\n" + e1);
    }
}

//anoikse thn forma gia na apantiseis to test
private void btn_begin_Click(object sender, EventArgs e)
{
    this.Close();
    //perna parametro to test kai kleise thn forma ayth
    my_functions.open_form_answers(Int32.Parse(txt_kodikos.Text));
}

private void checkBoxfind2_CheckedChanged(object sender, EventArgs
    e)
{
    Find_Exist_test(2);
    //an einaipatimeno to check na emfanisei minima
    if (checkBoxfind2.Checked == true)
    {
        checkBoxfind1.Checked = false;
        pic_err.Visible = true;
        lbl_error.Visible = true;
        lbl_error.Text = "ΜΟΝΟ ΟΛΟΚΛΗΡΩΜΕΝΑ ΤΕΣΤ";
    }
    else
    {
        pic_err.Visible = false;
        lbl_error.Visible = false;
    }
}
}
}
}

```

Συμπλήρωση του τεστ

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing.Drawing2D;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Timers;
using System.Data.SqlClient;

namespace WindowsFormsApplication1
{
    public partial class frm_answer : Form
    {
        public int intstart_test = 0;

        //arithmos ton erotiseon pou exei akoma na apantisei
        private int countq;

        //grafika formas
        public Graphics G;
        Pen mypen = new Pen(System.Drawing.Color.Red, 1);

        public frm_answer()
        {
            InitializeComponent();
        }

        private void frm_answer_Load(object sender, EventArgs e)
        {
            txt_kodikos.Text = my_user.kodikos.ToString();
            txt_onoma.Text = my_user.onoma;
            txt_epitheto.Text = my_user.epitheto;
            txt_date.Text = my_user.mydate;
            intstart_test = 0;

            Anapantites_erotiseis();//metra poses erotiseis exei akoma na apantisei

            //an to test einal olokliromeno emfanise ta apotelesmata
            if (countq == 0)
            {
                Open_Form_Results();
            }
        }

        private void btn_close_Click(object sender, EventArgs e)
        {
            //an den ksekinise to test bgaino
            if (intstart_test == 0)
            {
                this.Close();
                frm_menus f1 = new frm_menus();
                f1.Show();
            }
            //an ksekinise to test ton roto an thelei na apothikeuso
            else
            {
                DialogResult d1 = MessageBox.Show("Το τεστ ΔΕΝ έχει ολοκληρωθεί!\n\nΘέλετε να ΑΠΟΘΗΚΕΥΣΩ τις απαντήσεις;\n\nΑν συμφωνείτε μπορείτε να συνεχίσετε το τεστ από το σημείο που
```

```

        μείναιτε.\nΑν όχι θα πρέπει να το ξεκινήσετε ξανά.", "ΕΡΩΤΗΣΗ
        ΑΠΟΘΗΚΕΥΣΗΣ", MessageBoxButtons.YesNo);
//den apothikeyo tis apantiseis kai kleino thn forma
if (d1 == DialogResult.No)
{
    MessageBox.Show("Το τεστ δεν αποθηκεύτηκε", "ΕΙΔΟΠΟΙΗΣΗ");
}
//apothikeysh tou test mexri ayth th stigmh kai kleino thn forma
else
{
    Save_Answers();
}
this.Close();
frm menou f1 = new frm_menou();
f1.Show();
}
}

//apothikeyse tis apantiseis ston sql
private void Save_Answers()
{
    string ans = "";
    int i;

    //metatroph twn erotiseon tou array sto string ans
    for (i = 0; i < global_variables.count_erotiseis; i++)
    {
        ans+=my_user.answers[i].ToString();
    }

    //apothikeysh ston sql
    try
    {
        SqlConnection db1 = new SqlConnection(global_variables.my_database);
        string txt1 = "UPDATE tbl_test SET answers='"+ans+"' WHERE
            Id='"+txt_kodikos.Text+"'";
        SqlDataAdapter dat1 = new SqlDataAdapter(txt1, db1);
        DataTable tbl1 = new DataTable();
        dat1.Fill(tbl1);

    }
    catch (Exception e1)
    {
        MessageBox.Show(e1.ToString());
        Environment.Exit(0);
    }
}

//dose ena tyxaio aritho sto diastima pou briskontai oi erotiseis
private int GiveRandom()
{
    Random rnd = new Random();
    int x = rnd.Next(1, global_variables.count_erotiseis+1);

    return x;
}

private string Give_data_Question(int x)
{
    string ques1 = "";

    //anoikse thn bash kai pare thn erotisi me arithmo x
    try
    {
        SqlConnection db1 = new SqlConnection(global_variables.my_database);
        string txt1 = "SELECT erotisi FROM tbl_questions WHERE id="+x;
        SqlDataAdapter dat1 = new SqlDataAdapter(txt1, db1);
        DataTable tbl1 = new DataTable();
        dat1.Fill(tbl1);
        ques1 = tbl1.Rows[0][0].ToString();
        db1.Close();
    }
}

```



```

        return ques1;
    }
    catch(Exception e1)
    {
        MessageBox.Show(e1.ToString());
        Environment.Exit(0);
        return "error";
    }
}

private void Give_A_Question()
{
    int qn = 0 ;
    bool flag = false;

    //an den yparxoun erotiseis
    //emfanise kleisimo tou test
    if (countq == 0)
    {
        DialogResult d1= MessageBox.Show("Το τεστ ολοκληρώθηκε\nΝα προχωρήσω
        αποθήκευσή του και την εμφάνιση των αποτελεσμάτων;",
        "ΟΛΟΚΛΗΡΩΣΗ ΤΕΣΤ",MessageBoxButtons.YesNo);

        if (d1 == DialogResult.Yes)
        {
            Save_Answers();
            flag = true;
            Open_Form_Results();
        }
    }
    else
    {
        while (!flag)
        {
            qn = GiveRandom(); //dose ena egkyro tyxaio arithmo
            if (my_user.answers[qn - 1] == 0)
            {
                flag = true;
                countq--;
            }
        }

        if (flag)//dose erotisi
        {
            radioButton1.Checked = false;
            radioButton2.Checked = false;
            radioButton3.Checked = false;
            radioButton4.Checked = false;
            radioButton5.Checked = false;
            radioButton1.Enabled = true;
            radioButton2.Enabled = true;
            radioButton3.Enabled = true;
            radioButton4.Enabled = true;
            radioButton5.Enabled = true;

            //simplirose apo thn bash ton titlo ths erotisis
            lbl_index_er.Text = qn.ToString();
            lbl_er.Text = Give_data_Question(qn);
        }
    }
}

//pare thn apantisi apo ta controls radio
private void Take_answer(int x)
{
    int nq = Int32.Parse(lbl_index_er.Text);

    //bale sthn classh thn apantisi
    my_user.answers[nq - 1] = x;
}

```

```
lbl_index_er.Text = "";
lbl_er.Text = "";

//energopoihse to timer
System.Windows.Forms.Timer timer1 = new System.Windows.Forms.Timer();

timer1.Enabled = true;
timer1.Start();
timer1.Interval = 1000;
radioButton1.Enabled=false;
radioButton2.Enabled = false;
radioButton3.Enabled = false;
radioButton4.Enabled = false;
radioButton5.Enabled = false;

timer1.Tick += new EventHandler(TimerEventProcessor);

//emfanise ta xromata gia to poses erotiseis apantithikan
DrawPercent();
}

private void TimerEventProcessor(object sender, EventArgs e)
{
    System.Windows.Forms.Timer timer1 = (System.Windows.Forms.Timer) sender;
    timer1.Stop();

    Give_A_Question();
}

//ksekina to test
//
private void StartTest()
{
    lbl_er.Visible = true;
    lbl_erotiseis.Visible = true;
    lbl_index_er.Visible = true;
    lbl_dot.Visible = true;
    panel1.Visible = true;

    DrawPercent(); //emfanise thn mpara me to poies erotiseis apantithikan

    Give_A_Question();//dose erotisi
}

private void Anapantites_erotiseis()
{
    //metra poses erotiseis exei akoma
    int i;

    countq = 0;

    for (i = 0; i < global_variables.count_erotiseis; i++)
    {
        if (my_user.answers[i] == 0)
        {
            countq++;
        }
    }
}

private void btn_start_Click(object sender, EventArgs e)
{
    //einai se katastash enarksis
    if (intstart_test == 0)
    {
        intstart_test = 1;//metabliti gia na ksero an ksekinise to test
    }
}
```

```

        btn_start.Text = "ΑΠΟΘΗΚΕΥΣΗ";
        StartTest();

    }
    //einai se katastash apothikeyshs
    else
    {
        Save Answers();
        intstart test = 0;
        btn_start.Text = "ΕΝΑΡΞΗ";
    }
}

private void radioButton1_Click(object sender, EventArgs e)
{
    Take_answer(1);
}

private void radioButton2_Click(object sender, EventArgs e)
{
    Take_answer(2);
}

private void radioButton3_Click(object sender, EventArgs e)
{
    Take_answer(3);
}

private void radioButton4_Click(object sender, EventArgs e)
{
    Take_answer(4);
}

private void radioButton5_Click(object sender, EventArgs e)
{
    Take_answer(5);
}

//oloklirosh tou test//
//zita na pas se result
private void Open_Form_Results()
{
    Label l1 = new Label();
    l1.Left = 5;
    l1.Top = 5;

    l1.AutoSize = false;
    l1.BackColor = System.Drawing.Color.BurlyWood;

    l1.Width = this.Width-25;
    l1.Height = this.Height-49;
    l1.Show();

    this.Controls.Add(l1);

    panell.Hide();
    lbl_er.Hide();
    lbl_index_er.Hide();
    lbl_dot.Hide();
    lbl_erotiseis.Hide();
    btn_start.Hide();
    lbl_er.Text = "Το τεστ ολοκληρώθηκε.\nΠατήστε ΑΠΟΤΕΛΕΣΜΑΤΑ για να δείτε
        τον τύπο προσωπικότητας που σύμφωνα με τη θεωρία του Holland
        ταιριάζετε...";
    lbl_er.Show();
    btn_results.Show();
}

//pigaine se result
private void btn_results_Click(object sender, EventArgs e)

```

```
{
    frm results f1 = new frm_results();
    this.Close();
    f1.Show();
}

//zografizei thn mpara ton erotiseon pou apantithikan me prasino kai ayton pou
den apantithikan me kokkino
private void DrawPercent()
{
    try
    {
        G = this.CreateGraphics();
        int startx = 20;
        int starty = 80;
        int countsimeia;
        int counts = 0;
        int countf = 0;
        int j = 0;

        //emfanise tis erotiseis poy exoun ginei
        for (countsimeia = 0; countsimeia < global_variables.count_erotiseis;
            countsimeia++)
        {
            if (my_user.answers[countsimeia] != 0)
            {
                mypen.Color = System.Drawing.Color.DarkGreen;
                counts++;
            }
            else
            {
                mypen.Color = System.Drawing.Color.DarkRed;
                countf++;
            }
            for (j = 0; j < 600 / global_variables.count_erotiseis; j++)
            {
                G.DrawLine(mypen, new Point(startx, starty), new Point(startx,
                    starty + 20));
                startx += 1;
            }
        }

        G.Dispose();
        lbl erotiseis.Text = counts.ToString() + "/" +
            global_variables.count_erotiseis.ToString();
    }
    catch (Exception e1)
    {
        MessageBox.Show(e1.Message);
        Environment.Exit(0);
    }
}
}
```

Εμφάνιση βαθμολογίας, υπολογισμός και εμφάνιση αποτελεσμάτων

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace WindowsFormsApplication1
{
    public partial class frm_results : Form
    {
        //athroisma katigorion
        int[] sumcategory = new int[6];
        //ta 6 keimena twon prosopikothtn
        string[] keimeno = new string[6];

        //i epifaneia pou tha graftei to keimeno
        Label balloon = new Label();

        //grafika gia tis mpares
        public Graphics G;
        public Pen mypen = new Pen(System.Drawing.Color.Red,25);

        //pososta ths kathe katigorias
        public double[] pos = new double[6];

        public frm_results()
        {
            InitializeComponent();
        }

        private void frm_results_Load(object sender, EventArgs e)
        {
            lbl_kodikos.Text += " " + my_user.kodikos.ToString();
            lbl_onoma.Text += " " + my_user.onoma;
            lbl_epitheto.Text += " " + my_user.epitheto;
            lbl_date.Text += " " + my_user.mydate;

            keimeno[0] = "Τα άτομα που υπάγονται στον πρακτικό τύπο, είναι πρακτικοί και ρεαλιστές, εσωστρεφείς, απόλυτοι και δογματικοί, εύρωστοι και επίμονοι, ενώ στις ενέργειές τους είναι αποτελεσματικοί, μεθοδικοί, προσγειωμένοι και συνήθως ευθείς και ειλικρινείς. Διαθέτουν τεχνικές, μηχανολογικές, και χειρωνακτικές ικανότητες, χειρίζονται με επιδεξιότητα εργαλεία, μηχανικά συστήματα και ηλεκτρονικό εξοπλισμό και συνήθως στρέφονται σε τεχνικά, τεχνολογικά, μηχανολογικά και κατασκευαστικά επαγγέλματα. Το υπερκείμενο εργασιακό περιβάλλον τους προσανατολίζει σε εργασίες με πρακτικό, τεχνικό και συχνά υπαίθριο προσανατολισμό, με άμεσα ορατό αποτέλεσμα, καθώς και στην ανάπτυξη παραδοσιακών αξιών που έχουν σχέση με την εξουσία και τις οικονομικές απολαβές. Επαγγέλματα που ταιριάζουν με τον ίδιο τύπο είναι, μεταξύ άλλων, ο πολιτικός μηχανικός, ο μηχανολόγος-μηχανικός, ο χημικός-μηχανικός, ο γεωλόγος, ο μηχανικός μεταλλείων, ο γεωπόνος, ο ιχθυολόγος, ο δασολόγος, ο ηλεκτρονικός-πληροφορικός υλισμικού κλπ.\n Μεγαλύτερες πιθανότητες επιτυχίας σε: Μηχανολογία, Τεχνολογία Η/Υ, Κατασκευές, Τεχνικές εργασίες, Ένοπλες Δυνάμεις, Υψηλές ασφάλειας, Φύση και γεωργία-γεωπονία, Αθλητισμός, Ναυτιλία.";
            keimeno[1] = "Στον ερευνητικό τύπο υπάγονται άτομα με ευρηματική, ανήσυχη, δημιουργική και κριτική σκέψη, με ιδιαίτερη κλίση στις επιστήμες και την έρευνα. Είναι ανεξάρτητοι, πολυσύνθετοι και διανοούμενοι ως χαρακτήρες, ευρηματικοί, παρατηρητικοί, έως και περίεργοι. Διαθέτουν αναλυτικές, συνθετικές ικανότητες και διακατέχονται από ριζοσπαστική, κριτική και επικριτική διάθεση, είναι εσωστρεφείς, μάλλον απαισιόδοξοι, επιφυλακτικοί

```

και μετριοφρονες. Το υπερκείμενο εργασιακό περιβάλλον τούς συνδέει με δραστηριότητες που συνεπάγονται τη συστηματική και καινοτόμο έρευνα, την κατανόηση και τον έλεγχο των φυσικών, βιολογικών, πολιτιστικών και κοινωνικών φαινομένων. Ο ερευνητικός τύπος θα μπορούσε να διαχωριστεί σε δύο υποτύπους: α) τον ερευνητικό-θετικό τύπο, με ιδιαίτερες προτιμήσεις και δεξιότητες προς τα μαθηματικά, τη φυσικοχημεία, την τεχνολογία και τα συναφή επαγγέλματα και β) τον ερευνητικό-γραμματικό τύπο, με ροπή προς τις γλωσσικές και γραμματικές δεξιότητες, τις κοινωνικές και ανθρωπιστικές επιστήμες και τα αντίστοιχα επαγγέλματα και θέσεις εργασίας. \n Μεγαλύτερες πιθανότητες επιτυχίας σε: Έρευνα, Φυσικές επιστήμες, Χημεία και επιστήμες υγείας, Μαθηματικά υπολογιστές.";

keimeno[2] = "Οι καλλιτεχνικοί τύποι είναι πολύπλοκοι χαρακτήρες, ευαίσθητοι, ατίθασοι και αφηρημένοι, χωρίς τάξη, δυσπροσάρμοστοι και συναισθηματικοί. Είναι εκδηλωτικοί, παρορμητικοί και ευφάνταστοι, συχνά αναποτελεσματικοί αλλά και αυθεντικοί, ιδεαλιστές και ρομαντικοί. Προσανατολίζονται σε ελεύθερες, αφηρημένες και μη συστηματικές δραστηριότητες και προφανώς ενδιαφέρονται και κλίνουν προς τα καλλιτεχνικά επαγγέλματα. Το υπερκείμενο εργασιακό περιβάλλον τούς ενθαρρύνει να θεωρούν τους εαυτούς τους αντικομφορμιστές και ανεξάρτητους χαρακτήρες. Η ιδιαίτερη κλίση, το φυσικό ταλέντο του καθενός προς κάποια τέχνη, προς τη μουσική, τη ζωγραφική, το χορό, το θέατρο κλπ., αποτελεί κύριο συστατικό του τύπου αυτού και είναι πρωταρχικής σημασίας για μια αξιοπρεπή καριέρα στη δύσβατη, όσο και ενδιαφέρουσα διαδρομή των καλλιτεχνικών επαγγελμάτων. \n Μεγαλύτερες πιθανότητες επιτυχίας σε: Τέχνες και ψυχολογία, ΜΜΕ και συγγραφή, Αισθητική, μαγειρική, ξένες γλώσσες, εικαστικές τέχνες.";

keimeno[3] = "Όσοι υπάγονται στον κοινωνικό τύπο, είναι κατά κύριο λόγο φιλικοί, συνεργάσιμοι, ευγενείς, εξωστρεφείς, ευχάριστοι, θερμοί και ζωντανοί χαρακτήρες, συνήθως υπομονετικοί, αλληλέγγυοι, εξυπηρετικοί και συγκαταβατικοί. Αναπτύσσουν καλές σχέσεις με τρίτους, διαθέτουν πειθώ, παρέχουν τη συνδρομή τους σε όσους τη χρειάζονται και εμπλέκονται σε δραστηριότητες που συνεπάγονται τις δημόσιες σχέσεις, την εικοινωνία, την πληροφόρηση, την εκπαίδευση, τη θεραπεία, την αλληλεγγύη και την πρόνοια. Το υπερκείμενο εργασιακό περιβάλλον τούς οδηγεί στο να αναπτύσσουν δεξιότητες επικοινωνίας, κατανόησης και συνεργασίας. Οι κοινωνικοί τύποι θα μπορούσαν να διαχωριστούν σε δύο υποτύπους: α) τους κοινωνικούς-εξωστρεφείς και τους κοινωνικούς-αλληλέγγυους. Οι πρώτοι χαρακτηρίζονται από δεξιότητες δημοσίων σχέσεων, ικανότητες πωλήσεων και προσωπικής ανέλιξης, η κοινωνικότητά τους δηλαδή συνδυάζεται με σκοπούς, στόχους και προσωπικές φιλοδοξίες. Συναφή επαγγέλματα είναι του στελέχους δημοσίων σχέσεων, του πωλητή, του δημοσιογράφου. Οι δεύτεροι χαρακτηρίζονται περισσότερο από «αλληλέγγυα κοινωνικότητα», από ιδεαλισμό, συναισθηματισμό, γενναιοδωρία, ανθρωπισμό ή ακόμη και αυτοθυσία και ταιριάζουν περισσότερο σε επαγγέλματα εκπαίδευσης, πρόνοιας, υγείας, περίθαλψης, ανθρωπίνης συνεργασίας και αλληλεγγύης γενικότερα. Επισημαίνεται και εδώ ότι επιμέρους χαρακτηριστικά της κάθε προσωπικότητας, όπως τα παραπάνω, μπορεί να ενυπάρχουν ταυτόχρονα σε κλιμακούμενες αναλογίες, σε διάφορα άτομα. \n Μεγαλύτερες πιθανότητες επιτυχίας σε: Ψυχολογία, Θεολογία, Φιλανθρωπία εθελοντισμός, εκπαίδευση, φροντίδα και πρόνοια, κοινωνικές επιστήμες, νομική.";

keimeno[4] = "Στον επιχειρηματικό τύπο υπάγονται άτομα που διαθέτουν ηγετικές ικανότητες, άνθρωποι που είναι φιλόδοξοι, κυριαρχικοί, ενεργητικοί, ανταγωνιστικοί, κοινωνικοί, αισιόδοξοι, αποφασιστικοί και δημιουργικοί. Επιπλέον, μπορεί να χαρακτηριστούν ως επίμονοι, απόλυτοι, ανήσυχοι, δραστήριοι, εξωστρεφείς, ενθουσιώδεις, ριψοκίνδυνοι, επινοητικοί, δυναμικοί, γενναίοι, αλλά και συχνά εγωκεντρικοί, επιθετικοί και αυταρχικοί. Το υπερκείμενο εργασιακό περιβάλλον τούς οδηγεί στην ανάπτυξη επιχειρημάτων αλλά και μεθόδων πειθούς και χειραγώγησης τρίτων, ενώ θεωρούν τους εαυτούς τους δημοφιλείς, και γεμάτους αυτοπεποίθηση. Συχνά κρίνουν τον κόσμο απλουστευμένα και συμβατικά, με βάση το κοινωνικό κύρος και την εξουσία. Είναι προφανές ότι οι επιχειρηματικοί τύποι ταιριάζουν με διευθυντικές και ηγετικές θέσεις και επαγγέλματα, όπου ενυπάρχει οικονομική διάσταση, επιχειρηματικότητα και δυνατότητες ιεραρχικής ανέλιξης. \n Μεγαλύτερες πιθανότητες επιτυχίας σε: Επιχειρηματικότητα, διοίκηση, πωλήσεις, μάρκετινγκ και διαφήμιση, πολιτική και διπλωματία.";

keimeno[5] = "Οι συμβατικοί τύποι αποδέχονται τον καθιερωμένο τρόπο ζωής, τις αποδεκτές αξίες της κοινωνίας και την υφιστάμενη ιεραρχία εξουσίας, είναι αποσιωμένοι, ευσυνείδητοι με ανεπτυγμένο ιδιαίτερα το αίσθημα της οικονομίας. Είναι επιμελείς, ευπροσάρμοστοι, πειθαρχημένοι, προσεκτικοί, τακτικοί, πρακτικοί, ακριβείς, ψύχραιμοι, συνεργάσιμοι, αξιόπιστοι, συχνά

δογματικοί και εσωστρεφείς, ευσυνείδητοι, μεθοδικοί και οργανωτικοί. Τα άτομα που υπάγονται στον τύπο αυτό, προσανατολίζονται σε δραστηριότητες που χαρακτηρίζονται από τάξη, ακρίβεια και μεθοδικότητα. Το υπερκείμενο εργασιακό περιβάλλον τούς οδηγεί σε καθιερωμένες επαναλαμβανόμενες εργασίες που απαιτούν πειθαρχία, υπολογιστικές, λογιστικές, υπαλληλικές και επιχειρηματικές ικανότητες. Ταιριάζουν με επαγγέλματα όπως ο στρατιωτικός, ο αστυνομικός, ο δημόσιος ή ο διπλωματικός υπάλληλος, το στέλεχος επιχειρήσεων, ο βιβλιοθηκονόμος, ο συμβολαιογράφος, ο λογιστής κλπ.\n Μεγαλύτερες πιθανότητες επιτυχίας σε: Οργάνωση γραφείου, λογιστική, οικονομία - χρηματοοικονομικά, τουρισμός, logistics.";

```
//bres to athroisma ths kathe katigorias
Calculate_sum_types();
//emfanise tis mpares
Emfanise_Bathmologia();

}

private void Calculate_sum_types()
{
    try
    {
        SqlConnection db1 = new SqlConnection(global_variables.my_database);

        int i,j;
        int pososto;
        int bathmospoupatise;

        //midenise ta athroismata ton katigoriion
        for (i = 0; i < 6; i++)
        {
            sumcategory[i] = 0;
        }

        DataTable tbl1 = new DataTable();

        SqlDataAdapter dat1;
        string txt1;
        dat1 = new SqlDataAdapter();
        //anoigo thn erotisi
        txt1 = "select * from tbl questions";
        dat1.SelectCommand = new SqlCommand(txt1, db1);
        dat1.Fill(tbl1);

        for (i = 0; i < global_variables.count_erotiseis; i++)
        {
            //bazo se kathe katigoria to pososto ths erotisis EPI ayto pou
            //patise'
            //-3 gia na yparxei arnitiki
            //1=-2,2=-1,3=0,4=1,5=2
            for (j = 0; j < 6; j++)
            {
                pososto = Int32.Parse(tbl1.Rows[i][2+j].ToString()); //pososto
                lhs katigorias
                bathmospoupatise = my_user.answers[i];
                sumcategory[j] += pososto * (bathmospoupatise-3);
            }
        }
        db1.Close();
    }
    catch (Exception e1)
    {
        MessageBox.Show(e1.Message);
        Environment.Exit(0);
    }
}

private void Emfanise_Bathmologia()
```

```
{
    int min;
    int i;

    //bres to mikrotero athroisma
    min = sumcategory[0];
    for (i = 0; i < 6; i++)
    {
        if (sumcategory[i] < min) {
            min = sumcategory[i];
        }
    }
    //epeidh den mporoume na emfanisoume arnitiki bathmologia
    //ayksanoume ta panta toso oste i mikroterh bathmologia na einai 0
    if (min < 0)
    {
        for (i = 0; i < 6; i++)
        {
            sumcategory[i] += -min;
        }
    }
    //bres to athroisma olon ton katigoriion
    //tha boithisei gia na broume to pososto
    int sumall = 0;
    for (i = 0; i < 6; i++)
    {
        sumall += sumcategory[i];
    }

    textBox1.Text = sumcategory[0].ToString();
    textBox2.Text = sumcategory[1].ToString();
    textBox3.Text = sumcategory[2].ToString();
    textBox4.Text = sumcategory[3].ToString();
    textBox5.Text = sumcategory[4].ToString();
    textBox6.Text = sumcategory[5].ToString();

    //bres to pososto ths kathe katigorias
    if (sumall != 0)
    {
        for (i=0;i<6;i++)
        {
            pos[i] = ((double)sumcategory[i] / (double)sumall * 100);
            pos[i]=Math.Round(pos[i],2);//stroggylopoisi sto 2o dekadiko
        }
        Proteine(pos);
        txt_pos1.Text = pos[0].ToString()+"%";
        txt_pos2.Text = pos[1].ToString()+"%";
        txt_pos3.Text = pos[2].ToString()+"%";
        txt_pos4.Text = pos[3].ToString()+"%";
        txt_pos5.Text = pos[4].ToString()+"%";
        txt_pos6.Text = pos[5].ToString()+"%";
    }
}

//an kleisei bges sto kentriko menou
private void btn_close_Click(object sender, EventArgs e)
{
    this.Close();
    frm menou f1 = new frm_menou();
    f1.Show();
}

//diegrapse to test
private void btn_delete_Click(object sender, EventArgs e)
{
```



```
SqlConnection db1 = new SqlConnection(global_variables.my_database);
string txt1 = "DELETE FROM tbl test WHERE Id='"+my user.kodikos+"'";
SqlDataAdapter dat1 = new SqlDataAdapter(txt1, db1);
DataTable tbl1 = new DataTable();
dat1.Fill(tbl1);
db1.Close();
MessageBox.Show("Το τεστ διαγράφηκε με επιτυχία!\n", "ΔΙΑΓΡΑΦΗ");
this.Close();
frm menou f1 = new frm_menou();
f1.Show();
}

//emfanise to keimeno apo thn kathe katigoria
private void balloon_help(int ep)
{
    balloon.Name = "ballool";
    balloon.Left = 20;
    balloon.Top = 50;
    balloon.Location= new Point(107,61) ;
    balloon.Size=new Size(583,239);
    balloon.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
    balloon.TextAlign = ContentAlignment.MiddleCenter;

    balloon.Text = keimeno[ep - 1];
    balloon.BackColor = System.Drawing.Color.DarkGreen;
    balloon.ForeColor = System.Drawing.Color.White;

    balloon.Show();
    this.Controls.Add(balloon);
    this.Controls[balloon.Name].BringToFront();
}

private void pictureBox1_MouseHover(object sender, EventArgs e)
{
    balloon_help(1);
}

private void pictureBox2_MouseHover(object sender, EventArgs e)
{
    balloon_help(2);
}

private void pictureBox6_MouseHover(object sender, EventArgs e)
{
    balloon_help(3);
}

private void pictureBox4_MouseHover(object sender, EventArgs e)
{
    balloon_help(4);
}

private void pictureBox5_MouseHover(object sender, EventArgs e)
{
    balloon_help(5);
}

private void pictureBox6_MouseHover_1(object sender, EventArgs e)
{
    balloon_help(6);
}

private void pictureBox6_MouseLeave(object sender, EventArgs e)
{
    balloon.Hide();
}
```

```

private void View_bars()
{
    mypen.Color = dialekse_xroma_pen(pos[0]);
    G = lbl_pos1.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[0] * 3, 5);

    mypen.Color = dialekse_xroma_pen(pos[1]);
    G = lbl_pos2.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[1] * 3, 5);

    mypen.Color = dialekse_xroma_pen(pos[2]);
    G = lbl_pos3.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[2] * 3, 5);

    mypen.Color = dialekse_xroma_pen(pos[3]);
    G = lbl_pos4.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[3] * 3, 5);

    mypen.Color = dialekse_xroma_pen(pos[4]);
    G = lbl_pos5.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[4] * 3, 5);

    mypen.Color = dialekse_xroma_pen(pos[5]);
    G = lbl_pos6.CreateGraphics();
    G.DrawLine(mypen, 0, 5, (int)pos[5] * 3, 5);
}

Color dialekse_xroma_pen(double x)
{
    if (x >= 0 && x < 20) {return System.Drawing.Color.DarkRed; }
    else if (x <= 40) { return System.Drawing.Color.Red;}
    else if (x <= 60) { return System.Drawing.Color.Blue; }
    else if (x <= 75) { return System.Drawing.Color.DarkGreen; }
    else if (x <= 90) { return System.Drawing.Color.Green; }
    else { return System.Drawing.Color.YellowGreen; }
}

//bres to megalytero pososto kai kane to labal prasino
private void Proteine(double[] pos)
{
    double max;
    int i,th;
    max = pos[0];
    th=0;
    for (i = 1; i < 6; i++)
    {
        if (pos[i] > max)
        {
            max = pos[i];
            th = i;
        }
    }

    if (th == 0) { label1.BackColor = System.Drawing.Color.Green; }
    else if (th == 1) { label2.BackColor = System.Drawing.Color.Green; }
    else if (th == 2) { label3.BackColor = System.Drawing.Color.Green; }
    else if (th == 3) { label4.BackColor = System.Drawing.Color.Green; }
    else if (th == 4) { label5.BackColor = System.Drawing.Color.Green; }
    else if (th == 5) { label6.BackColor = System.Drawing.Color.Green; }

}

//dimiourgise ta grafika gia tis mpires
private void timebar_Tick(object sender, EventArgs e)
{
    View_bars();
}
}
}

```

5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

5.1 Συμπεράσματα

Η επιλογή επαγγέλματος στην εποχή μας είναι κρίσιμη και εξαιρετικά δύσκολη απόφαση. Εξαιτίας των τεχνολογικών και οικονομικών εξελίξεων που συντελούνται με πολύ γρήγορους ρυθμούς, το μέλλον αρκετών επαγγελμάτων είναι αβέβαιο, πολλά επαγγέλματα θεωρούνται κορεσμένα, σ' άλλα ο ανταγωνισμός εντείνεται ενώ για την άσκηση νέων επαγγελμάτων απαιτούνται επιστημονικές γνώσεις, ειδικά επαγγελματικά εφόδια και ικανότητες. Κι όμως, αυτή την απόφαση πρέπει ο άνθρωπος να την πάρει από την εφηβική κιόλας ηλικία του.

Ο επαγγελματικός προσανατολισμός, είναι άρρηκτα συνδεδεμένος με τον εκπαιδευτικό προσανατολισμό, αφού η υλοποίηση των επαγγελματικών αποφάσεων του ατόμου προϋποθέτει κάποιες εκπαιδευτικές εμπειρίες και δραστηριότητες και αφού κάθε απόφαση εκπαιδευτικής κατεύθυνσης του ατόμου έχει επαγγελματικές προεκτάσεις. Η σχέση εκπαίδευσης και επαγγέλματος είναι στενή, ώστε να γίνεται λόγος για «εκπαιδευτικές – επαγγελματικές» αποφάσεις. Η σημασία και η χρησιμότητα του επαγγελματικού προσανατολισμού καταδεικνύεται μεταξύ άλλων και από την έμφαση που έχει δοθεί σ' αυτόν από τις αρχές του αιώνα μας, τόσο από μέρους μεμονωμένων ατόμων όσο και από μέρους κρατών αλλά και οικονομικών ενώσεων (π.χ. Ε.Ο.Κ.).

Σήμερα η χρησιμότητα του επαγγελματικού προσανατολισμού είναι σχεδόν απόλυτα αυτονόητη. Ακόμη και στις περιπτώσεις που εκφράζονται κάποιες επιφυλάξεις, οι πιο πολλές από αυτές σχετίζονται περισσότερο με την αποτελεσματικότητα ή τη μεθοδολογία και τα μέσα εφαρμογής του επαγγελματικού προσανατολισμού. Η διαδικασία του λεγόμενου επαγγελματικού προσανατολισμού δεν περιορίζεται πάντως μόνο στο να βοηθήσει το άτομο στην επιλογή επαγγέλματος. Αυτή σχετίζεται με την ολική ανάπτυξη του ατόμου. Έτσι με τον όρο «επαγγελματική ανάπτυξη» του ατόμου εννοείται η ανάπτυξη και ωρίμαση εκείνης της πλευράς της προσωπικότητάς του, η οποία σχετίζεται με την επαγγελματική του συμπεριφορά. Αυτή όμως, για να είναι επιτυχής, προϋποθέτει μια γενικότερη ωρίμαση της προσωπικότητας.

Η επιλογή του επαγγέλματος δεν μπορεί και δεν πρέπει να είναι ένα τυχαίο γεγονός. Προκειμένου το άτομο να κάνει μια διαχρονικά εύστοχη επαγγελματική επιλογή είναι σημαντικό να ανακαλύψει, να ιεραρχήσει και να συνυπολογίσει τις δυνατότητες και ανάγκες του εκ των έσω. Στόχος του επαγγελματικού προσανατολισμού είναι αυτή η «εκ των έσω» διερεύνηση της προσωπικότητας καθώς και η παρουσίαση των διαθέσιμων σπουδών, επαγγελμάτων και της αγοράς εργασίας ώστε το άτομο να κάνει την καλύτερη δυνατή επιλογή. Η εύστοχη επαγγελματική δραστηριοποίηση διαμορφώνει ανθρώπους με αίσθηση προσωπικής ικανοποίησης και κοινωνικής επιτυχίας. Η ποιότητα της επαγγελματικής δραστηριοποίησης ενός ατόμου έχει σημαντικό αντίκτυπο σ' όλα τα επίπεδα της ζωής του.

Στον **οικονομικό τομέα**, εξασφαλίζονται τα απαραίτητα για μια αξιοπρεπή ζωή, παρέχεται η οικονομική δυνατότητα για ψυχαγωγία αλλά και απόλαυση περισσότερων καταναλωτικών αγαθών.

Στον **πνευματικό τομέα**, οργανώνεται και καλλιεργείται πνευματικά ο άνθρωπος, αναπτύσσοντας τη δημιουργική φαντασία και καλλιεργώντας την κρίση και την αισθητική. Αποκτώνται καινούργιες εμπειρίες μέσω της συνεργασίας με άλλους ανθρώπους και εξελίσσονται οι επιστήμες, τα γράμματα και οι τέχνες.

Στον **ηθικό τομέα**, αναπτύσσεται η ευγενής άμιλλα και ο συναγωνισμός και γεννώνται η αισιοδοξία και η αυτοπεποίθηση, λόγω της ηθικής πληρότητας που νιώθει ο άνθρωπος από την ευσυνείδητη άσκηση του επαγγέλματος του.

Στον **πολιτικό τομέα**, η ενεργός συμμετοχή στα πολιτικά πράγματα είναι το επιστέγασμα μια επιτυχημένης επαγγελματικής καριέρας. Η ευνομία και η δημοκρατία προϋποθέτουν κοινωνία ανθρώπων με αγαστή συνεργασία.

Στον **ψυχολογικό τομέα**, ο άνθρωπος ηρεμεί, με την αποβολή του άγχους και της αβεβαιότητας. Νιώθει εσωτερική πληρότητα και ικανοποίηση, επειδή ανταποκρίνεται στο βασικό του ρόλο ως κοινωνικού όντος.

Από τα παραπάνω γίνεται εύκολα αντιληπτό ότι η ύπαρξη και η σωστή χρήση των εφαρμογών επαγγελματικού προσανατολισμού είναι πολύ σημαντικές.

«Διάλεξε ένα επάγγελμα που σου αρέσει και δεν θα ξαναχρειαστεί να δουλέψεις στη ζωή σου», Κομφούκιος, 551-479 π.Χ., Κινέζος φιλόσοφος.

5.2 Μελλοντικές Επεκτάσεις της Εφαρμογής

Ο βασικός στόχος για την επέκταση της εφαρμογής είναι η απαραίτητη παραμετροποίηση ώστε να φορτωθεί και να λειτουργήσει διαδικτυακά. Επιπλέον στόχος είναι η βάση να συνδεθεί με συγκεκριμένες σχολές του μηχανογραφικού ώστε να προτείνει πιο ολοκληρωμένες προτάσεις ειδικότερα στους μαθητές των Πανελληνίων εξετάσεων.

Πανεπιστήμιο Πειραιώς

ΒΙΒΛΙΟΓΡΑΦΙΑ

- ❖ Amundson, N. E., Harris-Bowlsbey, J.A. & Niles, S. G. (2009). Βασικές Αρχές Επαγγελματικής Συμβουλευτικής. Διαδικασίες και Τεχνικές. (2009). Published by EKEP/ National Centre For Vocational Orientation. (Elements of Career Counseling: processes and techniques. 2nd Edition, Pearson Education, Prentice Hall).
- ❖ Holland, J. L. (1953). Vocational preference inventory.
- ❖ Holland, J. L. (1982). The occupations finder.
- ❖ Holland, J. L. (1970, 1977, 1985 & 1994) Self-directed search.
- ❖ Holland, J. L. (1973, 1985 & 1997) Making vocational choices: A theory of vocational personalities and work Environments.
- ❖ Super, D. E. (1980). A life-span, life-space approach to career development. Journal of Vocational Behavior, 16, 282–298.
- ❖ Super, D. E. (1990). Career and life development. In D. Brown & L. Brooks (Eds.), Career choice and development: Applying contemporary theories to practice (2nd ed., pp. 197–261). San Francisco: Jossey-Bass.
- ❖ Krumboltz, J. D. (1991). The career beliefs inventory. Palo Alto, CA: Consulting Psychologists Press.
- ❖ Krumboltz, J. D. (1996). A learning theory of career counseling. In M. Savickas & B. Walsh (Eds.), Integrating career theory and practice (pp. 233–280). Palo Alto, CA: Consulting Psychologists Books.
- ❖ Krumboltz, J. D., & Baker, R. (1973). Behavioral counseling for vocational decisions.
- ❖ Terry Quatrani (2003). Visual Modeling with Rational Rose 2002 and UML. Addison-Wesley Professional.
- ❖ Wendy Boggs; Michael Boggs (2002). Mastering UML with Rational Rose 2002. Sybex.
- ❖ Virvou, M. & Kabassi, K. (2002) "Reasoning About Users' Actions in a Graphical User Interface". In Human Computer Interaction, Vol.14, No. 4, pp. 369-399, 2002.
- ❖ Βίρβου Μαρία «Τεχνολογία Λογισμικού, Σημειώσεις διδασκαλίας»
- ❖ Βίρβου Μαρία «Αντικειμενοστραφής Τεχνολογία Λογισμικού»
- ❖ Δημητρόπουλος, Ε. (2003). Αποφάσεις – Λήψη Αποφάσεων. Εισαγωγή στην Ψυχολογία των Αποφάσεων. Αθήνα: Γρηγόρης.

- ❖ Δημητρόπουλος, Ε. (2004). Η Επαγγελματική Ανάπτυξη του ατόμου και οι σχετικές Θεωρίες. Στο: Συμβουλευτική και Επαγγελματικός Προσανατολισμός - Θεωρία και Πράξη (επιμ. Μ. Κασσωτάκης), 101-166. Αθήνα: Τυπωθήτω – Γ. Δαρδανός
- ❖ Κάντας, Α. & Χαντζή, Α. (1991). Ψυχολογία της εργασίας – Θεωρίες Επαγγελματικής Ανάπτυξης – Στοιχεία Συμβουλευτικής. Αθήνα: Ελληνικά Γράμματα.
- ❖ Πατεσπή, Α. (2007). Η Συμβουλευτική και ο Επαγγελματικός Προσανατολισμός στο χώρο της Εκπαίδευσης. Στο: Οδηγός Συμβουλευτικής και Επαγγελματικού Προσανατολισμού στον Τομέα της Εκπαίδευσης. Αθήνα: ΕΚΕΠ, 55-80.

ΠΗΓΕΣ ΑΠΟ ΤΟ ΔΙΑΔΙΚΤΥΟ

- <http://el.wikipedia.org>
- <http://en.wikipedia.org>
- <http://www.eoppep.gr>
- <http://www.epaggelma.gr>
- <http://www.orientum.gr>
- <http://www.microsoft.com>
- <http://www.w3schools.com>

Πανεπιστήμιο Πειραιώς