

**ΔΗΜΙΟΥΡΓΙΑ ΔΙΑΔΙΚΤΥΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ
ΕΚΤΕΛΕΣΗ ΕΡΓΑΣΙΩΝ ΑΠΟ ΤΟΥΣ ΦΟΙΤΗΤΕΣ ΤΟΥ
ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΚΥΠΡΟΥ**

Μιχάλης Ιωάννου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2014

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Δημιουργία Διαδικτυακού Συστήματος Για Εκτέλεση Εργασιών Από Τους Φοιτητές
Του Πανεπιστημίου Κύπρου**

Μιχάλης Ιωάννου

Επιβλέπων Καθηγητής
Αντώνης Κάκας

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων
απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου
Κύπρου

Μάιος 2014

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας κύριο Αντώνη Κάκα για την βοήθεια, τις συμβουλές του και την καθοδήγηση που μου πρόσφερε στα πλαίσια της διπλωματικής μου εργασίας.

Θα ήθελα να ευχαριστήσω την οικογένεια μου για την αμέριστη συμπαράσταση και την στήριξη τους καθ' όλη τη διάρκεια των σπουδών μου.

Τέλος, θα ήθελα να ευχαριστήσω τους φίλους μου για την στήριξη και την βοήθεια που μου παρείχαν σε αυτό το σημαντικό κομμάτι της ζωής μου.

Περίληψη

Η διπλωματική μου εργασία μελετά αρχικά τα Recommender Systems δίνοντας μια περιγραφή των συστημάτων αυτών και παρουσιάζοντας και σχολιάζοντας τα σημερινά συστήματα. Στην συνέχεια μελετά τα συστήματα τύπου Free Lance εξηγώντας την λειτουργία τους, τις παραμέτρους τους και την διαδικασία του ταιριάσματος

Το κύριο θέμα της εργασίας αυτής είναι η δημιουργία ενός συστήματος Free Lance για τους φοιτητές Πληροφορικής του Πανεπιστημίου Κύπρου και στα επόμενο κεφάλαιο καταγράφονται ο σκοπός, οι απαιτήσεις και οι είσοδοι και έξοδοι του συστήματος. Στην συνέχεια μελετώνται οι παράμετροι του συστήματος και εξηγείται με διάφορα παραδείγματα η διαδικασία του ταιριάσματος. Προς τα τελευταία κεφάλαια καταγράφονται οι αλγόριθμοι του συστήματος και εξηγούνται αναλυτικά με διάφορα παραδείγματα. Τέλος αξιολογούνται οι αλγόριθμοι και καταγράφονται συμπεράσματα για το σύστημα καθώς και τα δικαιώματα και ευθύνες που θα έχουν οι εμπλεκόμενοι.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	
1.1	Τα διαδικτυακά συστήματα σήμερα	1
1.2	Στόχος διπλωματικής εργασίας	2
Κεφάλαιο 2	Recommender Systems	
2.1	Περιγραφή	4
2.2	Ανασκόπηση	5
2.3	Τα Συστήματα στην Εποχή μας	6
2.4	Εξατομίκευση	7
Κεφάλαιο 3	Συστήματα Ελεύθερων Επαγγελματιών.....	
3.1	Περιγραφή	9
3.2	Λειτουργία Συστημάτων Ελεύθερων Επαγγελματιών	10
3.3	Παράμετροι και Ταίριασμα	10
Κεφάλαιο 4	Σύστημα Ελεύθερων Επαγγελματιών Για Τους Φοιτητές του Πανεπιστημίου Κύπρου.....	
4.1	Σκοπός Συστήματος	13
4.2	Περιγραφή Συστήματος	14
4.3	Απαιτήσεις Συστήματος	14
4.4	Είσοδοι και Έξοδοι Συστήματος	15
4.5	Πιλοτική Φάση	16
Κεφάλαιο 5	Παράμετροι Συστήματος.....	
5.1	Περιγραφή	20
5.2	Παράμετροι του Πελάτη	21

5.3.Παράμετροι του Φοιτητή	22
5.4. Ιεραρχία Πληροφορικής	24
5.5 Ταίριασμα Μεταξύ πελάτη και φοιτητή	25
Κεφάλαιο 6 Αλγόριθμοι Ταιριάσματος.....	
6.1 Ταίριασμα	29
6.2 Αλγόριθμος 1	30
6.3 Αλγόριθμος 2	32
6.4 Υπολογισμός Παραμέτρου Job Compatibility	35
Κεφάλαιο 7 Δικαιώματα και ευθύνες.....	
7.1 Περιγραφή	37
7.2 Δικαιώματα και ευθύνες πελατών	37
7.3 Δικαιώματα και ευθύνες φοιτητών	38
7.4 Δικαιώματα και ευθύνες του τμήματος Πληροφορικής	39
Κεφάλαιο 8 Αξιολογήσεις Αλγορίθμων.....	
8.1 Αξιολόγηση Προσέγγισης και Συστήματος	41
Κεφάλαιο 9 Συμπεράσματα	
9.1 Συμπεράσματα	52
9.2 Μελλοντική Εργασία	52
Βιβλιογραφία	54
Παράρτημα Α.....	A-1
Παράρτημα Β.....	B-1

Κεφάλαιο 1

Εισαγωγή

1.1 Τα διαδικτυακά συστήματα σήμερα	1
1.2 Στόχος διπλωματικής εργασίας	2

1.1 Τα διαδικτυακά συστήματα σήμερα

Στην εποχή μας, υπάρχουν αρκετά συστήματα τα οποία επιτρέπουν την παροχή υπηρεσιών μέσω του διαδικτύου. Από αγορές και πωλήσεις αντικειμένων, μέχρι και ενοικιάσεις αυτοκινήτων. Οι ανταλλαγές αυτών των υπηρεσιών γίνονται ευκολότερα μέσω του διαδικτύου, όσο αφορά την πληρωμή και την εύρεση των κατάλληλων προϊόντων για κάθε χρήστη.

Συγκεκριμένα, θα επικεντρωθούμε σε συστήματα τύπου Free Lancers (Ελεύθερων Επαγγελματιών). Τα συστήματα αυτά προσφέρουν ποικιλία υπηρεσιών, όπως βοήθεια στις δουλειές του σπιτιού, σε μαθήματα, σε θέματα ηλεκτρονικών υπολογιστών και σε άλλα πολλά. Για αρχή, θα περιγράψουμε ένα από αυτά τα συστήματα, το οποίο είναι αρκετά γνωστό και έχει μεγάλη επιτυχία, και θα προτείνουμε τρόπους για να υποβοηθήσουμε τέτοιου είδους συστήματα, με αυτοματοποίηση κάποιων λειτουργιών τους.

Το atyourservice.com.cy είναι μια κυπριακή ιστοσελίδα για πώληση υπηρεσιών. Ελεύθεροι επαγγελματίες, διάφορων ειδικοτήτων, εγγράφονται στο σύστημα αυτό, έτσι ώστε ο κόσμος που χρειάζεται τέτοιες υπηρεσίες να μπορεί να τους βρει.

Αφού γίνει η εγγραφή, ο χρήστης μπορεί να αναρτήσει εργασίες που θέλουν επίτευξη και να του κάνουν προσφορές οι ενδιαφερόμενοι, καθώς επίσης μπορεί να ψάξει για κάποιον συγκεκριμένο ελεύθερο επαγγελματία και να κάνει στον ίδιο προσφορά για μια δουλειά. Επιπρόσθετα, μπορεί να αναθέσει μια δουλειά προσωπικά σε ένα επαγγελματία, ο οποίος με την σειρά του μπορεί να την αποδεχτεί ή να την απορρίψει. Μετά από την επίτευξη μιας εργασίας, ο πελάτης μπορεί να βαθμολογήσει αυτόν που έκανε την δουλειά και να αφήσει

μερικά σχόλια όσο αφορά την ποιότητα της εργασίας του, ώστε να ενημερωθούν οι υπόλοιποι χρήστες για το πόσο έμπιστος είναι ο κάθε ελεύθερος επαγγελματίας.

Κατά την αναζήτηση εργασιών, ο χρήστης βάζει κάποιο από τα υπάρχοντα φίλτρα και επιλέγει κατηγορίες, όπως δουλειές του σπιτιού, βοήθεια σε μαθήματα κτλ. Το προφίλ του κάθε χρήστη περιλαμβάνει στοιχεία επικοινωνίας, πόσες δουλειές έκανε, βαθμολογία και τυχόν σχόλια που άφησαν οι υπόλοιποι χρήστες.

Παρόμοιες ιστοσελίδες με την atyourservice.com.cy, είναι το elance.com και peoplebythehour.com, στις οποίες γίνεται πώληση υπηρεσιών με αρκετά παρόμοιο τρόπο.

1.2 Στόχος διπλωματικής εργασίας

Στόχος της εργασίας αυτής είναι να υποβοηθήσουμε αυτού του είδους τα συστήματα με την αυτοματοποίηση της αναζήτησης των ελεύθερων επαγγελματιών και την δημιουργία ενός ταιριάσματος μεταξύ των δύο. Για να επιτευχθεί αυτό και οι ελεύθεροι επαγγελματίες και οι χρήστες θα πρέπει να έχουν κάποιες παραμέτρους, δηλαδή, αν θέλω εγώ να μου κάνουν μια εργασία, βάζω κάποιες παραμέτρους για την δουλειά και με βάση των παραμέτρων αυτών και τις παραμέτρους των ελεύθερων επαγγελματιών μου βγάζει τα κατάλληλα αποτελέσματα. Οι παράμετροι θα αποθηκεύονται με μια ιεράρχηση. Για τις δουλειές, η ιεράρχηση αυτή θα περιλαμβάνει είδος εργασίας, ποιότητα (πόσο καλή δουλειά μπορεί να γίνει), ταχύτητα (πόσο γρήγορα μπορεί να γίνει η δουλειά), χρονικό όριο (μέχρι πότε την θέλω), χρηματικό ποσό (μέχρι ποιο ποσό δίνω για την δουλειά), εμπιστοσύνη, βαθμολογία. Αντίστοιχα, για τους ελεύθερους επαγγελματίες θα υπάρχουν παράμετροι όπως, ειδικότητα (σε ποιο πεδίο ειδικεύετε), βαθμολογία, χρηματικό ποσό (αποδεχτό ποσό για την εκπλήρωση της εργασίας). Αυτές είναι και οι παράμετροι που θα συγκρίνονται για να γίνει το ταίριασμα. Το ταίριασμα φαίνεται στο ακόλουθο παράδειγμα :

Υποθέτουμε ότι ένας χρήστης ψάχνει ελεύθερο επαγγελματία για να του κάνει μια εργασία και η εργασία αυτή έχει κάποιες παραμέτρους.

$$\begin{array}{l} \Phi 1 \\ \{a, b, c\} \end{array}$$

$$\begin{array}{l} \Phi 2 \\ \{x, y, z\} \end{array}$$

Όπου Φ , οι ελεύθεροι επαγγελματίες και a, b, c, x, y, z οι παράμετροι τους. Οι παράμετροι έχουν τις ακόλουθες τιμές :

$$a=4, b=1, c=1$$

$$x=5, y=3, z=2$$

Το a και το x είναι το ίδιο είδος παραμέτρου, όπως είναι και το b με το y και το c με το z . Οι αριθμοί αντικατοπτρίζουν τον βαθμό προτεραιότητας που έχουν οι παράμετροι. Όπως βλέπουμε, οι συγκρίσεις που μπορούν να γίνουν είναι $a < x$, $b < y$ και $c < z$. Άρα για την συγκεκριμένη εργασία ο κατάλληλος είναι ο $\Phi 2$ επειδή οι βαθμοί προτεραιότητας του είναι και στις τρεις παραμέτρους μεγαλύτεροι από αυτούς του $\Phi 1$. Σημειώστε ότι κάθε χρήστης έχει τις δικές του προτεραιότητες στις παραμέτρους. Για άλλους πιο σημαντικός παράγοντας είναι η εμπιστοσύνη, για άλλους το πόσο γρήγορα θα γίνει η δουλειά κτλ. Οι πιο ασήμαντοι παράμετροι που βρίσκονται στο τέλος της ιεράρχησης, ενδέχεται να μην συγκρίνονται καν, εκτός και αν υπάρχουν συγκρίσεις για τις παραμέτρους στην αρχή της ιεράρχησης.

Αν για παράδειγμα υπάρχουν 1000 ελεύθεροι επαγγελματίες, θα ήταν πολύ δύσκολο να βρει τον κατάλληλο, ενώ με τον τρόπο που αναφέραμε θα του παρουσιάσουμε μόνο 10 από αυτούς, που σύμφωνα με τις παραμέτρους που ο ίδιος έβαλε, θεωρεί κατάλληλους για εκείνον. Για αυτόν ακριβώς τον λόγο θέλουμε να ενσωματώσουμε αυτή την λειτουργία σε αυτά τα συστήματα. Για να γίνει πιο εύκολο αυτό το ταίριασμα και για να εμπλέκετε ο χρήστης όσο το δυνατό λιγότερο σε αυτή την αναζήτηση.

Κεφάλαιο 2

Recommender Systems

2.1 Περιγραφή	4
2.2 Ανασκόπηση	5
2.3 Τα συστήματα στην εποχή μας	6
2.4 Εξατομίκευση	7

Περίληψη

Σε αυτό το κεφάλαιο μελετώνται τα Recommender Systems. Δίνεται μια περιγραφή των συστημάτων αυτών, εξηγείτε ο τρόπος λειτουργίας τους και γίνεται μια ανασκόπηση. Επίσης περιγράφονται τα συστήματα στην εποχή μας και δίνεται μια περιγραφή της εξατομίκευσης.

2.1 Περιγραφή

Τα αυτοματοποιημένα συστήματα συστάσεων (Recommender Systems) είναι συστήματα τα οποία, με διάφορους αλγόριθμους προσπαθούν να παρουσιάζουν ταιριάσματα, συστάσεις και προβλέψεις για διάφορα προϊόντα ή υπηρεσίες και εφαρμόζονται σε ποικιλία ιστοσελίδων. Πρωτοεμφανίστηκαν το 1990 στην περιοχή του μάρκετινγκ και στην διανομή εμπορίου.

Στην εποχή μας αυτά τα συστήματα αναπτυχθήκαν και χρησιμοποιούνται από το ευρύ κοινό, καθώς επιτρέπουν την παροχή διάφορων υπηρεσιών μέσω του διαδικτύου. Από αγορές και πωλήσεις αντικειμένων, μέχρι και ενοικιάσεις αυτοκινήτων[1]. Οι ανταλλαγές αυτών των υπηρεσιών γίνονται ευκολότερα μέσω του διαδικτύου όσο αφορά την πληρωμή και την εύρεση των κατάλληλων προϊόντων για κάθε χρήστη.

Στόχος των συστημάτων αυτών, είναι η πρόβλεψη και η βελτίωση των προτάσεων προς τον χρήστη ανάλογα με τις προηγούμενες του αγορές καθώς και τις προτιμήσεις του.

2.2 Ανασκόπηση

Τα Recommender Systems είναι μια υποκατηγορία των Information Filtering System, τα οποία επιδιώκουν να προβλέψουν την ικανότητα ή προτίμηση, την οποία θα έδινε ο χρήστης σε ένα αντικείμενο. Τα Recommender Systems έγιναν εξαιρετικά δημοφιλή στα πρόσφατα χρόνια και ενσωματώνονται σε πολλές εφαρμογές. Οι πιο δημοφιλείς είναι οι ταινίες, μουσική, νέα, βιβλία, άρθρα ερευνών και γενικότερα σε προϊόντα. Υπάρχουν επίσης και συστήματα για τους έμπειρους όπως εστιατόρια, χρηματοπιστωτικές υπηρεσίες, ασφάλειες ζωής και online dating.

Τα Recommender Systems παράγουν συνήθως μια λίστα συστάσεων με ένα από τους 2 ακόλουθους τρόπους.

- Collaborative Filtering

Το Collaborative filtering χτίζει ένα μοντέλο από την προηγούμενη συμπεριφορά του χρήστη, δηλαδή αντικείμενα τα οποία προμηθεύτηκε καθώς επίσης και παρόμοιες αποφάσεις που πάρθηκαν από άλλους χρήστες. Μετά χρησιμοποιώντας αυτό το μοντέλο μπορούμε να προβλέψουμε αντικείμενα για τα οποία μπορεί να ενδιαφέρεται ο χρήστης

- Content Based Filtering

Το Content Based Filtering συνδυάζει διάφορα χαρακτηριστικά κάποιου αντικειμένου έτσι ώστε να προτείνει άλλα αντικείμενα με τα οποία έχει ίδια χαρακτηριστικά.

- Συνδυασμός Collaborative filtering και Content Based Filtering

Πρόσφατες έρευνες έχουν δείξει ότι μια υβριδική προσέγγιση, που συνδυάζει το Collaborative filtering και το Content Based Filtering θα μπορούσε να είναι πιο αποτελεσματική σε ορισμένες περιπτώσεις. Οι υβριδικές προσεγγίσεις μπορούν να εφαρμοστούν με διάφορους τρόπους: κάνοντας προβλέψεις με βάση το Collaborative filtering και Content Based Filtering και στη συνέχεια να συνδυάζονται, με την προσθήκη δυνατοτήτων του Content Based Filtering σε μια Collaborative filtering προσέγγιση (και αντίστροφα), ή με την ενοποίηση των δύο προσεγγίσεων σε ένα μοντέλο.

2.3 Τα συστήματα στην εποχή μας

Μερικές από τις ιστοσελίδες στις οποίες εφαρμόζονται τα συστήματα αυτά είναι ιστοσελίδες στις οποίες γίνεται αγορά προϊόντων όπως το Ebay και το Amazon[4] , στις οποίες διάφοροι αγοραστές εισέρχονται για να προμηθευτούν διάφορα προϊόντα από άλλους πωλητές, όπως βιβλία, ηλεκτρονικές συσκευές κ.α.

Εκτός από προϊόντα , υπάρχουν ιστοσελίδες που προσφέρουν υπηρεσίες, όπως AtYourservice.com.cy , guru.com και elance.com. Σε όλες αυτές τις σελίδες γίνεται μια αγορά υπηρεσιών σε ένα μεγάλο εύρος περιοχών όπως, πληροφορική , νοικοκυριό, αρχιτεκτονική , ζωγραφική κ.α. Μετά την εγγραφή κάποιου χρήστη, μπορεί να διαλέξει την υπηρεσία της επιλογής του, ανάμεσα σε πολλές εταιρείες και ελεύθερους επαγγελματίες. Επίσης υπάρχει και η επιλογή να εγγραφεί και ο χρήστης σαν ελεύθερος επαγγελματίας.

Υπάρχουν χιλιάδες παραδείγματα τέτοιων υπηρεσιών και η ενσωμάτωση των συστημάτων συστάσεων σε όλες αυτές τις σελίδες έχει ως εξής:

Ας πάρουμε παράδειγμα υπάρχοντα συστήματα αρχίζοντας με το EBay. Κάνοντας ένα account στο EBay , ο χρήστης μπορεί να διαλέξει διάφορα προϊόντα. Για χάρη του παραδείγματος, ας ονομάσουμε τον χρήστη μας Κώστα. Ο Κώστας αγόρασε παλαιότερα κάποια μουσικά CD των AC/DC και τώρα ξαναμπαίνει στην ιστοσελίδα για να αγοράσει κάποια καινούρια. Το σύστημα θα προσπαθήσει να προβλέψει ότι ίσως θέλει να αγοράσει CD του ιδίου συγκροτήματος και ότι αρέσει στον Κώστα η ροκ μουσική. Άρα καθώς θα βλέπει διάφορα προϊόντα , θα του εμφανιστούν, εκτός άλμπουμ από τους AC/DC, και άλλα ροκ μουσικά CD. Ακόμα μια πρόβλεψη που προσπαθεί να κάνει το σύστημα είναι η ακόλουθη. Όταν ο Κώστας αγοράσει ένα προϊόν , θα του εμφανιστούν άλλα προϊόντα τα οποία προμηθεύτηκαν άλλοι χρήστες, οι οποίοι αγόρασαν και το συγκεκριμένο αυτό προϊόν. Όπως βλέπετε το σύστημα προσπαθεί να προβλέψει τα προϊόντα που τον ενδιαφέρουν , όχι μόνο από παλιές αγορές αλλά και από άλλους χρήστες που μπορεί να ενδιαφέρονται για τα ίδια προϊόντα όπως και ο Κώστας.

Ακόμα ένα παράδειγμα, είναι το carpooling.co.uk. Σε αυτή την σελίδα ,οδηγοί και επιβάτες μπορούν να βρουν , να προσφέρουν και να κλείσουν διαδρομές , και να ελέγξουν πως είναι οι ταξιδιώτες μέσω των προφίλ τους. Ο χρήστης αφού κάνει εγγραφή στο σύστημα μπορεί να

κάνει offer ή request. Αν κάποιος θέλει να πάει σε ένα συγκεκριμένο ορισμό, βρίσκει κάποιον ταξιδιώτη ο οποίος θα κάνει αυτή την διαδρομή και μπορεί να δει το προφίλ του και να τον επιλέξει. Υπάρχουν πολλοί παράμετροι τις οποίες μπορεί να επιλέξει ο χρήστης, όπως Ημερομηνία και ώρα αναχώρησης, Πόλη αναχώρησης και τελικός προορισμός, Διαθέσιμες Θέσεις, Αν είναι καπνιστής ή όχι κτλ.

2.4 Εξατομίκευση

Η εξατομίκευση επιτρέπει τη δυναμική εισαγωγή και προσαρμογή περιεχομένου σε οποιαδήποτε μορφή, που είναι σχετικές με τον συγκεκριμένο χρήστη. Δηλαδή βασίζεται στην συμπεριφορά και τις προτιμήσεις του.

Διάφορες ιστοσελίδες σήμερα προσφέρουν αυτή την τεχνολογία βασιζόμενες σε χαρακτηριστικά όπως ενδιαφέροντα και προτιμήσεις κάθε ατόμου ξεχωριστά. Για παράδειγμα θα γίνονται διάφορες αλλαγές στο σύστημα ανάλογα με ποια αντικείμενα αγόρασε ή ποιες σελίδες επισκέφτηκε.

Υπάρχουν τρεις κατηγορίες εξατομίκευσης

- Βασισμένο σε ένα προφίλ ή μια ομάδα
- Βασισμένο σε συμπεριφορά
- Βασισμένο σε συνεργασία

Τα μοντέλα εξατομίκευσης περιέχουν κανόνες με βάση το filtering και collaborative filtering το οποίο λειτουργεί με τον συνδυασμό των προσωπικών προτιμήσεων, με των προτιμήσεων άλλων παρόμοιων χρηστών. Το collaborative filtering δουλεύει καλά για βιβλία, μουσική, βίντεο κτλ.

Δεν δουλεύει όμως για ένα μεγάλο αριθμό κατηγοριών όπως ρούχα, κοσμήματα, καλλυντικά κτλ.

Υπάρχουν τρεις μέθοδοι για εξατομίκευση

- Σιωπηρή
- Σαφής
- Υβρίδιο των δύο

Με την σιωπηρή, η εξατομίκευση εφαρμόζεται από το σύστημα με βάση τις κατηγορίες που αναφέρθηκαν πιο πάνω.

Με την σαφή εξατομίκευση το σύστημα αλλάζει από τον χρήστη χρησιμοποιώντας τις δυνατότητες που παρέχονται από το σύστημα.

Η υβριδική εξατομίκευση συνδυάζει τις δύο προσεγγίσεις για να πάρει τις καλύτερες δυνατότητες των δύο.

Πολλές εταιρείες προσφέρουν υπηρεσίες για διαδικτυακές συστάσεις και συστάσεις ηλεκτρονικού ταχυδρομείου που βασίζονται σε εξατομίκευση ή συλλέγονται από συμπεριφορές ανώνυμων χρηστών.

Κεφάλαιο 3

Συστήματα Ελεύθερων Επαγγελματιών

3.1 Περιγραφή	9
3.2 Λειτουργία Συστημάτων Ελεύθερων Επαγγελματιών	10
3.3 Παράμετροι και ταίριασμα	1

Περίληψη

Σε αυτό το κεφάλαιο περιγράφονται τα συστήματα τύπου Free Lance και εξηγείται η λειτουργία τους και παρουσιάζονται παραδείγματα ταιριάσματος με παράμετρους.

3.1 Περιγραφή

Τα συστήματα Free Lance προσφέρουν ποικιλία υπηρεσιών, όπως βοήθεια στις δουλειές του σπιτιού, σε μαθήματα, σε θέματα ηλεκτρονικών υπολογιστών και σε άλλα πολλά όπως ανάφερα πιο πάνω. Συγκεκριμένα σε αυτά συστήματα γίνεται εγγραφή των χρηστών, οι οποίοι μπορεί να είναι και διάφορες εταιρείες που προσφέρουν υπηρεσίες, απλοί ελεύθεροι επαγγελματίες ή οι άνθρωποι που θέλουν να αγοράσουν υπηρεσίες. Κατά την εγγραφή πραγματοποιείται η συμπλήρωση των στοιχείων τους, στα οποία εκτός από όνομα , διεύθυνση κτλ , προσθέτουν και την ειδικότητα τους ανάλογα με την περιοχή στην οποία προσφέρουν υπηρεσίες. Στην συνέχεια όποιος κόσμος επιθυμεί να αγοράσει αυτές τις υπηρεσίες μπορεί να τους αναζητήσει και να αγοράσει τις υπηρεσίες που επιθυμεί. Μερικά από αυτά τα συστήματα είναι τα AtYourservice.com.cy , guru.com, peopleperhour.com και elance.com. Τα AtYourservice.com.cy, guru.com και peopleperhour.com επικεντρώνονται σε διάφορα πεδία όπως πληροφορική , νοικοκυριό, αρχιτεκτονική , ζωγραφική ενώ το

elance.com αφορά θέματα αποκλειστικά για την πληροφορική όπως δημιουργία προγραμμάτων, κτίσιμο δικτύων και βάσεων κτλ

3.2 Λειτουργία Free Lance Συστημάτων

Αφού γίνει η εγγραφή σε ένα από αυτά τα συστήματα , ο χρήστης μπορεί να αναρτήσει εργασίες που θέλουν επίτευξη και να του κάνουν προσφορές οι ενδιαφερόμενοι, ή μπορεί να ψάξει για κάποιον συγκεκριμένο ελεύθερο επαγγελματία και να κάνει στον ίδιο προσφορά για μια δουλειά. Επίσης μπορεί να αναθέσει μια δουλειά προσωπικά σε ένα επαγγελματία και με την σειρά του μπορεί να την αποδεχτεί ή να την απορρίψει. Μετά από την επίτευξη μιας εργασίας, ο πελάτης μπορεί να βαθμολογήσει αυτόν που έκανε την δουλειά και να αφήσει μερικά σχόλια όσο αφορά την ποιότητα της εργασίας του, ώστε να ενημερωθούν οι υπόλοιποι χρήστες για το πόσο έμπιστος είναι ο κάθε ελεύθερος επαγγελματίας.

Κατά την αναζήτηση εργασιών, ο χρήστης βάζει κάποιο από τα υπάρχοντα φίλτρα και επιλέγει κατηγορίες όπως, δουλειές του σπιτιού, βοήθεια σε μαθήματα κτλ . Το προφίλ του κάθε χρήστη περιλαμβάνει στοιχεία επικοινωνίας, πόσες δουλειές έκανε, βαθμολογία και τυχόν σχόλια που άφησαν οι υπόλοιποι χρήστες.

3.3 Παράμετροι και ταίριασμα

Στόχος μας είναι να υποβοηθήσουμε αυτά συστήματα με το να αυτοματοποιήσουμε την αναζήτηση των ελεύθερων επαγγελματιών και να γίνεται ένα ταίριασμα μεταξύ των δύο. Για να επιτευχθεί αυτό , και οι ελεύθεροι επαγγελματίες και οι χρήστες θα πρέπει να έχουν κάποιες παραμέτρους. Δηλαδή, αν θέλω εγώ να μου κάνουν μια εργασία, βάζω κάποιες παραμέτρους για την δουλειά και με βάση των παραμέτρων αυτών και τις παραμέτρους των ελεύθερων επαγγελματιών μου βγάζει τα κατάλληλα αποτελέσματα.

Πιο κάτω φαίνονται δύο παραδείγματα ταιριάσματος με παραμέτρους.

Υποθέτουμε ότι ένας χρήστης ψάχνει ελεύθερο επαγγελματία για να του κάνει μια εργασία και η εργασία αυτή έχει κάποιες παραμέτρους.

$\{a, b, c\}$

$\{x, y, z\}$

Όπου Φ , οι ελεύθεροι επαγγελματίες και a, b, c, x, y, z οι παράμετροι τους. Οι παράμετροι έχουν τις ακόλουθες τιμές :

$$a=4, b=1, c=1$$

$$x=5, y=3, z=2$$

Το a και το x είναι το ίδιο είδος παραμέτρου, όπως είναι και το b με το y και το c με το z . Οι αριθμοί αντικατοπτρίζουν τον βαθμό προτεραιότητας που έχουν οι παράμετροι. Όπως βλέπουμε, οι συγκρίσεις που μπορούν να γίνουν είναι $a < x$, $b < y$ και $c < z$. Άρα για την συγκεκριμένη εργασία ο κατάλληλος είναι ο $\Phi 2$ επειδή οι βαθμοί προτεραιότητας του είναι και στις τρεις παραμέτρους μεγαλύτεροι από αυτούς του $\Phi 1$. Σημειώστε ότι κάθε χρήστης έχει τις δικές του προτεραιότητες στις παραμέτρους. Για άλλους πιο σημαντικός παράγοντας είναι η εμπιστοσύνη, για άλλους το πόσο γρήγορα θα γίνει η δουλειά κτλ. Οι πιο ασήμαντοι παράμετροι που βρίσκονται στο τέλος της ιεράρχησης, ενδέχεται να μην συγκρίνονται καν, εκτός και αν υπάρχουν συγκρίσεις για τις παραμέτρους στην αρχή της ιεράρχησης.

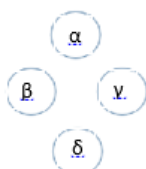
Αν για παράδειγμα υπάρχουν 1000 ελεύθεροι επαγγελματίες, θα ήταν πολύ δύσκολο να βρει τον κατάλληλο, ενώ με τον τρόπο που αναφέραμε θα του παρουσιάσουμε μόνο 10 από αυτούς, που σύμφωνα με τις παραμέτρους που ο ίδιος έβαλε, θεωρεί κατάλληλους για εκείνον. Για αυτόν ακριβώς τον λόγο θέλουμε να ενσωματώσουμε αυτή την λειτουργία σε αυτά τα συστήματα. Για να γίνει πιο εύκολο αυτό το ταίριασμα και για να εμπλέκετε ο χρήστης όσο το δυνατό λιγότερο σε αυτή την αναζήτηση.

Παράδειγμα 2

Ομάδα 1

Χρήστης που αναζητά
Υπηρεσίες.

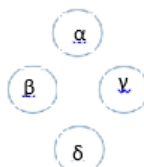
Χρήστης Α



Ομάδα 2

Χρήστης που προσφέρει
Υπηρεσίες

Χρήστης Β



Χρήστης Β1

Χρήστης Β2

Χρήστης Β3

Χρήστης Β4

Υποθέτουμε ότι έχουμε ένα χρήστη ο οποίος θέλει κάποιες υπηρεσίες (ομάδα Α) και ένα πλήθος από χρήστες που προσφέρουν κάποιες υπηρεσίες ή προϊόντα (ομάδα 2). Σε κάθε ομάδα υπάρχουν κάποιοι παράμετροι που ορίστηκαν από τους ίδιους τους χρήστες, ιεραρχικά. Δηλαδή η παράμετρος α έχει μεγαλύτερη προτεραιότητα από όλες τις υπόλοιπες παραμέτρους, δηλαδή τις β, γ και δ και η β και γ, έχουν την ίδια προτεραιότητα.

Η κύρια λειτουργία του συστήματος είναι να κάνει ένα ταίριασμα μεταξύ της ομάδας 1 και της ομάδας 2 με βάση τις παραμέτρους της κάθε ομάδας, καθώς και την ιεραρχία τους. Για παράδειγμα για τον χρήστη στην ομάδα 1, μεγαλύτερη σημασία έχει η παράμετρος α. Άρα το ταίριασμα πρέπει να γίνει με χρήστες της ομάδας 2, οι οποίοι έχουν αυτή την συγκεκριμένη παράμετρο στην αρχή της ιεραρχίας τους. Είναι πιθανόν να εμφανιστούν περισσότεροι από ένας χρήστες για το ταίριασμα, οπότε πρέπει ο χρήστης της 1ης ομάδας να επιλέξει κάποιον από μόνος του. Αν δεν υπάρχουν χρήστες για το ταίριασμα με βάση το α, τότε θα πρέπει να γίνει να γίνει το ταίριασμα με βάση την παράμετρο β ή γ, κοκ

Κεφάλαιο 4

Σύστημα Ελεύθερων Επαγγελματιών Για τους Φοιτητές Του Πανεπιστημίου Κύπρου

4.1 Σκοπός Συστήματος	13
4.2 Περιγραφή Συστήματος	14
4.3 Απαιτήσεις Συστήματος	14
4.4 Είσοδοι και Έξοδοι	15
4.5 Πιλοτική φάση	16

Περίληψη

Σε αυτό το κεφάλαιο δίνεται μια περιγραφή του συστήματος, στην οποία θα αναφερθούν οι λειτουργίες του . Μετά δίνονται οι απαιτήσεις του συστήματος, δηλαδή τι ακριβώς θα κάνει το σύστημα καθώς και ο σκοπός του συστήματος, δηλαδή το κίνητρο που ώθησε στην δημιουργία του συστήματος αυτού. Επίσης, θα παρουσιαστούν οι εισοδοι και οι έξοδοι του συστήματος, που περιλαμβάνει τα δεδομένα που παίρνει σαν είσοδο το σύστημα και οι πληροφορίες που βγάζει σαν έξοδο το σύστημα καθώς και η πιλοτική φάση του συστήματος.

4.1 Σκοπός Συστήματος

Σκοπός της εργασίας αυτής είναι να υποβοηθήσουμε τα συστήματα Ελεύθερων Επαγγελματιών με το να αυτοματοποιήσουμε το ταίριασμα των ελεύθερων επαγγελματιών με τις εργασίες. Για να επιτευχθεί αυτό , και οι ελεύθεροι επαγγελματίες και οι χρήστες θα πρέπει να έχουν κάποιες παραμέτρους. Δηλαδή, για να γίνει εισαγωγή μιας εργασίας προς εκτέλεση, ο χρήστης που το κάνει πρέπει να βάλει κάποιες παραμέτρους που θα την χαρακτηρίζουν. Με βάση αυτές τις παραμέτρους θα επιλέγονται οι πιο κατάλληλοι ελεύθεροι επαγγελματίες για την συγκεκριμένη δουλειά. Η καταλληλότητα των ελεύθερων

επαγγελματιών βρίσκεται με την σύγκριση των παραμέτρων που δόθηκαν στην εργασία και των παραμέτρων, των χαρακτηριστικών, που έχουν τα άτομα αυτά, άρα πιο κατάλληλα για αυτή την δουλεία είναι αυτά που έχουν κοινούς παραμέτρους με την συγκεκριμένη δουλεία.

Αυτή την μέθοδο θα την ενσωματώσω στην διπλωματική μου εργασία η οποία είναι η δημιουργία ενός διαδικτυακού συστήματος για τους φοιτητές του Πανεπιστημίου Κύπρου στο οποίο θα εγγράφονται όσοι φοιτητές επιθυμούν ως ελεύθεροι επαγγελματίες, με σκοπό την προσφορά των υπηρεσιών τους στον κόσμο που τις χρειάζεται σε διάφορα πεδία της Πληροφορικής. Επίσης θα εγγράφεται στο σύστημα όποιος επιθυμεί ως πελάτης και θα μπορεί να δημοσιεύει την εργασία που θέλει να εκπληρωθεί. Ο πελάτης θα δημιουργεί λογαριασμό στο σύστημα συμπληρώνοντας τα στοιχεία του.

4.2 Περιγραφή Συστήματος

Το σύστημα αυτό αποτελεί ένα σύστημα για τους φοιτητές του Πανεπιστημίου Κύπρου, στο οποίο θα εγγράφονται όσοι φοιτητές επιθυμούν. Οι φοιτητές θα επιλέγονται από το σύστημα για την εκπόνηση μιας εργασίας, η οποία θα δημοσιεύει ο κόσμος, και στην συνέχεια ο φοιτητής θα επιλέγει αν θα την εκτελέσει ή όχι. Επιπρόσθετα, στο σύστημα μπορεί να εγγραφεί οποιοσδήποτε επιθυμεί, ως πελάτης, για να δημοσιεύσει κάποια εργασία. Μετά από την δημοσίευση κάποιας εργασίας το σύστημα θα εμφανίζει στον χρήστη ένα σύνολο από προτεινόμενους φοιτητές για την συγκεκριμένη εργασία, η επιλογή των φοιτητών γίνεται με βάση την ειδικότητα κάθε φοιτητή και της εργασίας. Μετά την ολοκλήρωση κάποιας εργασίας ο πελάτης μπορεί να αξιολογήσει την δουλεία του φοιτητή, με την εισαγωγή μιας βαθμολογίας στον φοιτητή με βάση την ταχύτητα και την ποιότητα της δουλειάς του.

4.3 Απαιτήσεις Συστήματος

Οι λειτουργίες που υλοποιήθηκαν στο σύστημα είναι οι ακόλουθες:

- Εγγραφή χρηστών (Register): Οι χρήστες θα εγγράφονται στο σύστημα συμπληρώνοντας τα στοιχεία τους έτσι ώστε να έχουν τον προσωπικό τους λογαριασμό και να τους δίνεται οι δυνατότητα να δημοσιεύουν εργασίες. Οι πληροφορίες που πρέπει να δίνει ο χρήστης κατά την εγγραφή είναι το όνομα,

επίθετο, ηλεκτρονική διεύθυνση, το όνομα χρήστη και κωδικό πρόσβασης που επιθυμεί για την επιτυχή είσοδο του στο σύστημα.

- Sign In χρηστών: Οι χρήστες θα εισέρχονται στο σύστημα, με την προϋπόθεση ότι ήδη έχουν λογαριασμό, ώστε να δίνουν τον κωδικό πρόσβασης και το όνομα χρήστη.
- Δημοσίευση εργασίας: Οι χρήστες θα μπορούν να δημοσιεύουν τις εργασίες που θέλουν, γράφοντας μια περιγραφή της εργασίας και βάζοντας όποιες παραμέτρους επιθυμούν. Οι παράμετροι αυτοί θα περιγραφούν λεπτομερώς στο επόμενο υποκεφάλαιο.
- Δυνατότητα βαθμολογίας (rating) στους φοιτητές: Όταν εκτελείτε μια εργασία, δίνεται στον χρήστη η δυνατότητα να βαθμολογήσει τον συγκεκριμένο φοιτητή, ο οποίος ήταν υπεύθυνος για την εκτέλεση αυτής της εργασίας, με βάση το πόσο ευχαριστημένος ήταν από το αποτέλεσμα της δουλειάς του. Η βαθμολογία αυτή γίνεται με δύο συγκεκριμένους παραμέτρους, οι οποίοι θα περιγραφούν στο επόμενο κεφάλαιο.

Ο σκοπός της δημιουργίας αυτού του συστήματος είναι η δημιουργία κάποιων μικρών δουλειών για τους φοιτητές πληροφορικής του Πανεπιστημίου Κύπρου καθώς και για να βοηθήσουμε τον κόσμο που θέλει να του εκτελεστούν εργασίες που είναι σχετικές με την πληροφορική.

4.4 Είσοδοι και Έξοδοι Συστήματος

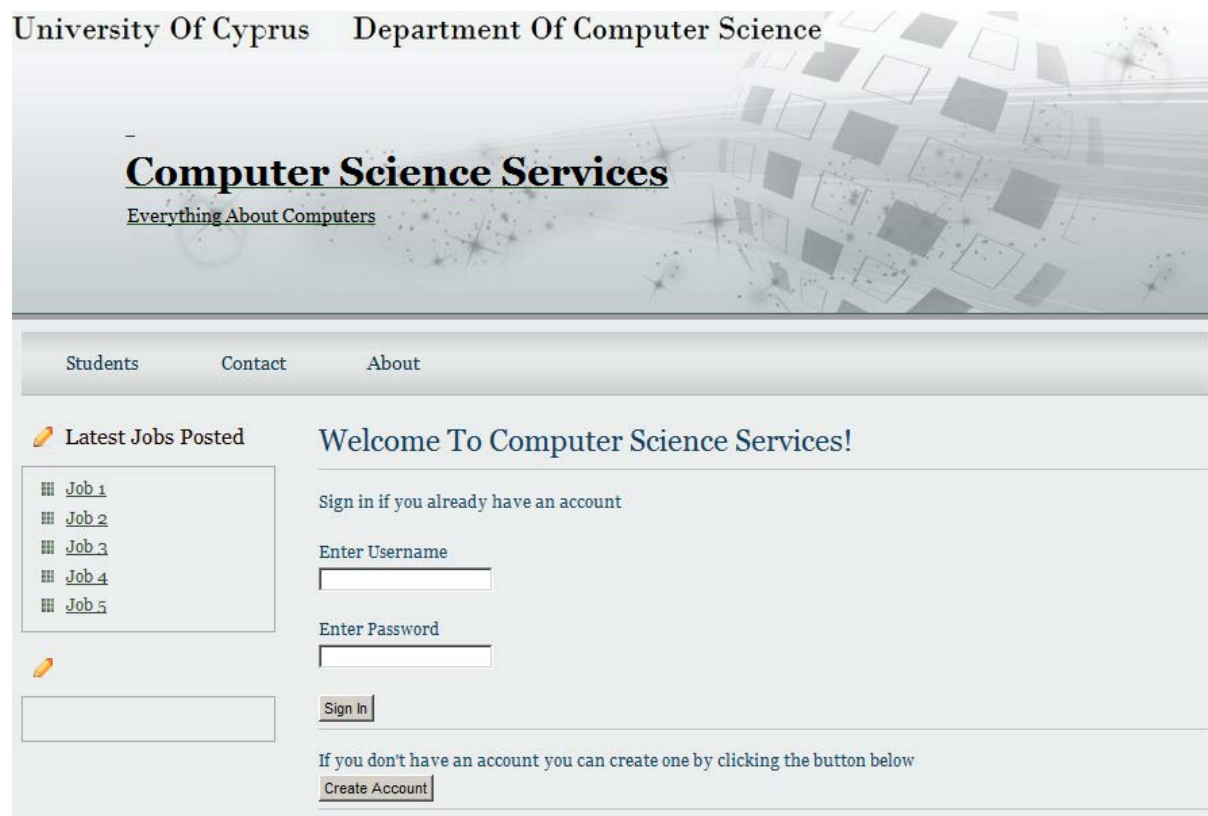
Το σύστημα αποτελείται από δύο εισόδους. Η πρώτη είσοδος είναι οι εργασίες μαζί με τις παραμέτρους τους. Ο πελάτης καλείται να εισαγάγει μια εργασία προς εκτέλεση καθώς επίσης και να προσθέσει τις απαραίτητες παραμέτρους που αντιστοιχούν στην συγκεκριμένη εργασία. Η δεύτερη είσοδος είναι το rating της εργασίας. Με το τέλος της εκτέλεσης της εργασίας από τους φοιτητές που ορίστηκαν σε αυτή, ο πελάτης και πάλι καλείται να βαθμολογήσει τους φοιτητές που εργάστηκαν σε αυτή την εργασία με κριτήριο την

ικανοποίηση του ως προς τον τρόπο διεξαγωγής της εργασίας και κυρίως του αποτελέσματος που του παρέδωσαν.

Η έξοδος του συστήματος είναι η αντιστοίχιση των εργασιών που εισαγάγουν οι πελάτες με τους καταλληλότερους φοιτητές. Η ανάθεση φοιτητών σε μια συγκεκριμένη εργασία γίνεται με βάση τους κοινούς παραμέτρους που έχουν οι φοιτητές και η εργασία.

4.5 Πιλοτική Φάση

Ο πελάτης με το μπαίνει στο σύστημα βλέπει την ακόλουθη σελίδα.



University Of Cyprus Department Of Computer Science

Computer Science Services
Everything About Computers

Students Contact About

 Latest Jobs Posted

-  [Job 1](#)
-  [Job 2](#)
-  [Job 3](#)
-  [Job 4](#)
-  [Job 5](#)



Welcome To Computer Science Services!

Sign in if you already have an account

Enter Username

Enter Password

If you don't have an account you can create one by clicking the button below

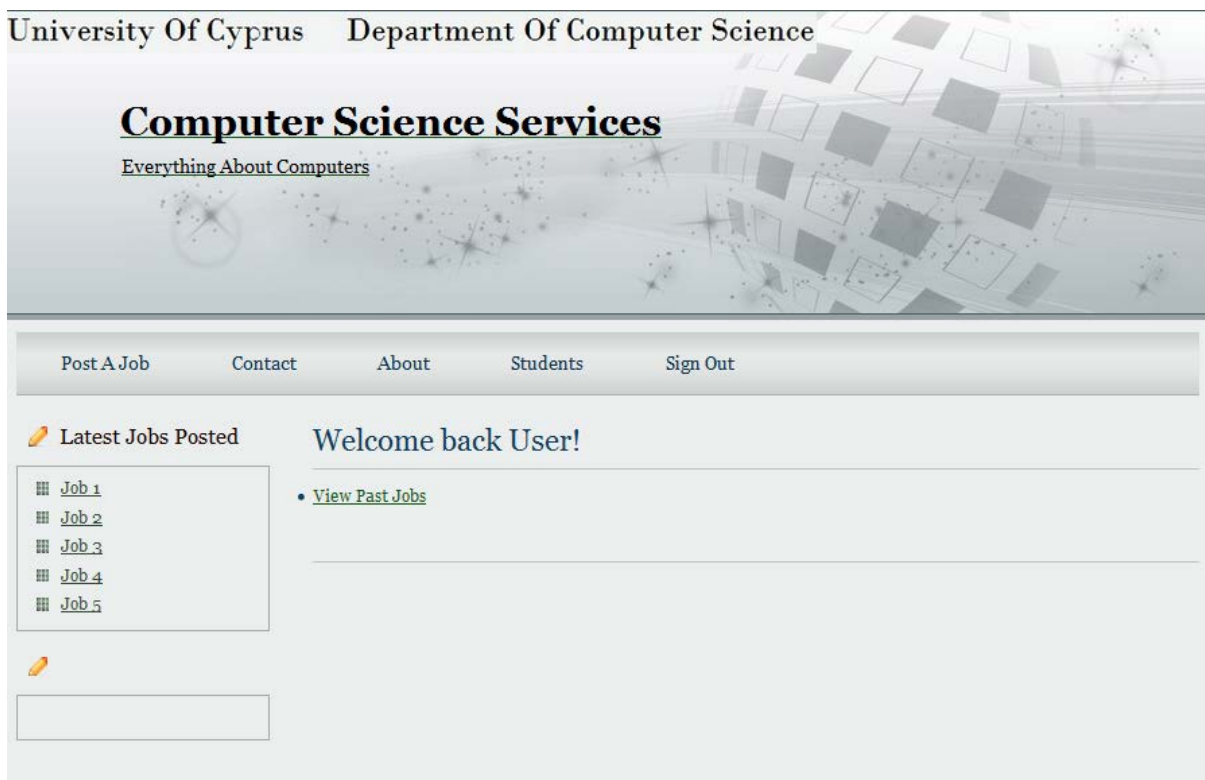
Μπορεί να κάνει ένα νέο λογαριασμό πατώντας το “Create Account” ή αν ήδη είναι εγγεγραμμένος μπορεί να κάνει Sign In στο σύστημα βάζοντας τα στοιχεία του και μετά πατώντας “Sign In”.

Στην φόρμα Create Account ο πελάτης βάζει τα στοιχεία του και μετά πατά “Submit”.



The image shows a web form titled "Create An Account". It contains six input fields stacked vertically: "First Name", "Last Name", "E-mail Address", "Username", "Password", and "Retype Password". Each field is a simple text box with a light blue border. Below the last field is a "Submit" button with a light blue background and a dark blue border. The entire form is enclosed in a light gray border.

Αφού κάνει λογαριασμό ή κάνει Sign In θα του εμφανιστεί η ακόλουθη φόρμα στην οποία μπορεί να πατήσει το “Post A Job” για να δημοσιεύσει μια εργασία.



The image shows a user dashboard after logging in. At the top, there is a header with "University Of Cyprus" and "Department Of Computer Science". Below this is a banner for "Computer Science Services" with the tagline "Everything About Computers". A navigation bar contains links: "Post A Job", "Contact", "About", "Students", and "Sign Out". The main content area is divided into two columns. The left column is titled "Latest Jobs Posted" and lists five jobs: "Job 1", "Job 2", "Job 3", "Job 4", and "Job 5". The right column is titled "Welcome back User!" and includes a link to "View Past Jobs". At the bottom of the left column, there is a small icon of a pencil and a text input field.

Αφού πατήσει το “Post A Job”, ο πελάτης θα συμπληρώσει τα διάφορα στοιχεία και θα επιλέξει τα διάφορα πεδία της εργασίας του και θα πατήσει “Submit”

Post A Job

Name Of Your Job

Describe Your Job

Select the Category Of Job

1st Field

1- Select Category - ▾

Select The Deadline Of The Job (*not required)

☐ *Urgent (Click this box if your deadline is **strict**)

Insert Your Budget For The Job

Submit

Στην συνέχεια ο πελάτης θα λάβει ένα μήνυμα στο οποίο θα λέει ότι θα πρέπει να αναμένει πριν του εμφανιστούν οι συνιστάμενοι χρήστες έτσι ώστε να ελεγχθούν από το τμήμα Πληροφορικής.

Όταν γίνει δεκτή η έγκριση ο πελάτης θα μπορεί να επιλέξει ένα από τους φοιτητές ή ομάδα από φοιτητών στην ακόλουθη φόρμα.

We recommend the following freelancers for your job

Team 1

Team Member 1:
Student 2

Team Member 2:
Student 45

Job Compatibility: 3.5 / 5
Quality Of Job: 3.2 / 5
Speed Of Job: 3.8 / 5
Team Trust: 3.5 / 5

Job Compatibility: 3.5 / 5
Quality Of Job: 2.8 / 5
Speed Of Job: 3.5 / 5
Team Trust: 3.2 / 5

Choose This Team

Team 2

Team Member 1:
Student 20

Team Member 2:
Student 28

Job Compatibility: 3.5 / 5
Quality Of Job: 2.4 / 5
Speed Of Job: 1.4 / 5
Team Trust: 1.9 / 5

Job Compatibility: 3.5 / 5
Quality Of Job: 1.6 / 5
Speed Of Job: 4.4 / 5
Team Trust: 3 / 5

Choose This Team

Αφού η εργασία εκτελεστεί από κάποιο φοιτητή, ο χρήστης μπορεί να συμπληρώσει την φόρμα ανατροφοδότησης στην οποία θα βαθμολογήσει τον φοιτητή με βάση την απόδοση και έχει την δυνατότητα αν επιθυμεί να εισάγει δικά του σχόλια, όπως φαίνονται στην ακόλουθη φόρμα.

[Post A Job](#) [Contact](#) [About](#) [Students](#)

 Latest Jobs Posted

 [Job 1](#)

 [Job 2](#)

 [Job 3](#)

 [Job 4](#)

 [Job 5](#)



Give us feedback about our students!

Rate Student 20 and Student 28 based on the quality and speed of his job

(1 = lowest, 5 = highest)

Quality

1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Timeliness

1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Comment on our freelancers (*not required)

Submit

Κεφάλαιο 5

Παράμετροι Συστήματος

5.1 Περιγραφή	20
5.2 Παράμετροι του Πελάτη	21
5.3 Παράμετροι του Φοιτητή	22
5.4 Ιεραρχία Πληροφορικής	24
5.5 Ταίριασμα Μεταξύ Πελάτη και Φοιτητή	25

Περίληψη

Σε αυτό το κεφάλαιο μελετώνται οι παράμετροι του πελάτη και του φοιτητή και εξηγούνται λεπτομερώς, δίνεται ο λόγος της δημιουργίας μιας ιεραρχίας της πληροφορικής μαζί με ένα παράδειγμα ενός κομματιού της ιεραρχίας και εξηγούνται παραδείγματα του ταιριάσματος του πελάτη και του φοιτητή.

5.1 Περιγραφή

Για να γίνει ένα επιτυχημένο ταίριασμα πρέπει και ο φοιτητής και η εργασία του πελάτη να έχουν κάποια προφίλ τα οποία θα χαρακτηρίζονται από κάποιες παραμέτρους. Με βάση τις προτιμήσεις του κάθε πελάτη θα πρέπει να του προτείνουμε τον κατάλληλο φοιτητή. Σε αυτό το κεφάλαιο καταγράφω τις παραμέτρους του πελάτη και του φοιτητή, εξήγηση και σκοπός της ιεραρχίας πληροφορικής και εξήγηση και παραδείγματα του ταιριάσματος μεταξύ του πελάτη και του φοιτητή.

5.2 Παράμετροι του Πελάτη

Όταν ένας πελάτης θέλει να λάβει υπηρεσίες από κάποιο φοιτητή, πρέπει να κάνει δημοσίευση της εργασίας του μαζί με τις ακόλουθες παραμέτρους:

-Όνομα της εργασίας προς εκτέλεση:

Κάποιο όνομα της εργασίας το οποίο θα επιλέξει ελεύθερα ο πελάτης.

-Περιγραφή της εργασίας:

Μια περιγραφή της εργασίας με όποιες λεπτομέρειες επιθυμεί ο πελάτης.

-1ο πεδίο εργασίας:

Το 1^ο πεδίο στο οποίο ανήκει η εργασία. Μια εργασία μπορεί να ανήκει σε 1 ή 2 πεδία.

-1η υποκατηγορία (Προαιρετικό):

Ο πελάτης έχει την δυνατότητα να βάλει υποκατηγορία για το 1^ο πεδίο της εργασίας του.

-2η Υποκατηγορία 1ου πεδίου:

Ο πελάτης έχει την δυνατότητα να βάλει 2^η υποκατηγορία για το 1^ο πεδίο της εργασίας του.

-2ο Πεδίο Εργασίας :

Το 2^ο πεδίο στο οποίο ανήκει η εργασία

-1η Υποκατηγορία 2ου πεδίου:

Ο πελάτης έχει την δυνατότητα να βάλει υποκατηγορία για το 2^ο πεδίο της εργασίας του.

-2η Υποκατηγορία 2ου πεδίου:

Ο πελάτης έχει την δυνατότητα να βάλει 2^η υποκατηγορία για το 2^ο πεδίο της εργασίας του.

-Χρηματικό ποσό που διατίθεται να δώσει:

Ποσό χρημάτων που διατίθεται να δώσει ο χρήστης για την εκτέλεση της εργασίας.

-Διορία:

Ημερομηνία Παράδοσης της εργασίας.

5.3 Παράμετροι του φοιτητή

-Συνολική Βαθμολογία στο Πανεπιστήμιο:

Γενική βαθμολογία στην μέχρι τώρα πορεία του φοιτητή στο Πανεπιστήμιο

-Βαθμός Βασικών Μαθημάτων:

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 131, ΕΠΛ 132, ΕΠΛ 231 και ΕΠΛ 233

-Βαθμός Θεωρητικών Μαθημάτων

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 111 και ΕΠΛ 211

-Βαθμός Software Engineering:

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 241, ΕΠΛ 361, ΕΠΛ 362, ΕΠΛ 363 και ΕΠΛ 435.

-Βαθμός Databases:

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 342 και ΕΠΛ 446

-Βαθμός Network:

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 324 και ΕΠΛ 375

-Βαθμός Computer Graphics

Γενικός βαθμός βασικών μαθημάτων ο οποίος αποτελείτε από τα μαθήματα ΕΠΛ 446 και ΕΠΛ 447

-Βαθμός Security:

Βαθμός στο μάθημα ΕΠΛ 475

-Βαθμός Gaming Technology:

Βαθμός στο μάθημα ΕΠΛ 426

-Εμπιστοσύνη χρόνου (Time Trust) :

Βαθμός εμπιστοσύνης στο κατά πόσο ο φοιτητής μπορεί να παραδώσει την εργασία στα πλαίσια της διορίας

- Εμπιστοσύνη ποιότητας (Quality Trust) :

Βαθμός εμπιστοσύνης στο κατά πόσο καλή είναι η ποιότητα της δουλειάς του

- Εμπιστοσύνη σε ομάδα (Team Trust) :

Πόσο καλά δουλεύει σε μια ομάδα ο φοιτητής

-Πεδίο Διπλωματικής Εργασίας

Πεδίο στο οποίο ανήκει η διπλωματική εργασία του φοιτητή.

-Άλλες Δραστηριότητες

Άλλες δραστηριότητες που μπορεί να έκανε ο φοιτητής, για παράδειγμα εκτέλεση άλλων εργασιών πέρα από τις εργασίες των μαθημάτων, συμμετοχή σε διαγωνισμού, συμμετοχή σε συνέδρια κτλ.

Δύο πολύ σημαντικές παράμετροι είναι η εμπιστοσύνη ταχύτητας και η εμπιστοσύνη ποιότητας. Η εμπιστοσύνη ποιότητας καθορίζει πόσο αξιόπιστος και πόσο καλά κάνει την δουλειά του ένας πελάτης. Η εμπιστοσύνη ποιότητας είναι ένας αριθμός από το 1 ως το 5, όπου το 1 ισούται με «καθόλου καλός» και το 5 «Πολύ καλός». Για να λαμβάνονται αυτές οι τιμές στο σύστημα, υπάρχει ο μηχανισμός ο οποίος επιτρέπει στους πελάτες να βάζουν βαθμολογίες στους φοιτητές ανάλογα με το πόσο ευχαριστημένοι είναι με την εργασία που ολοκληρώσανε.

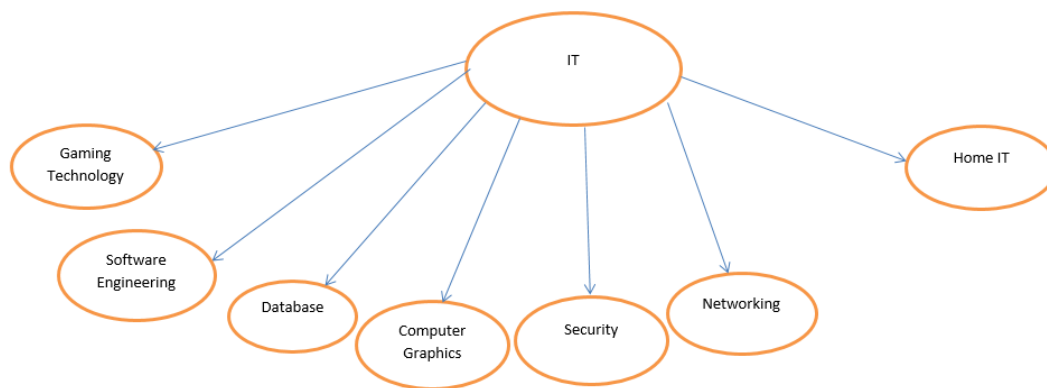
Η εμπιστοσύνη ταχύτητας είναι παρόμοια με την εμπιστοσύνη ποιότητας με την έννοια ότι θα υπάρχει στο προφίλ του Server και θα βάζει βαθμολογία ο κάθε πελάτης μετά την

ολοκλήρωση μιας εργασίας. Η εμπιστοσύνη ταχύτητας υποδηλώνει το πόσο γρήγορα εκτέλεσε μια εργασία κάποιος Server.

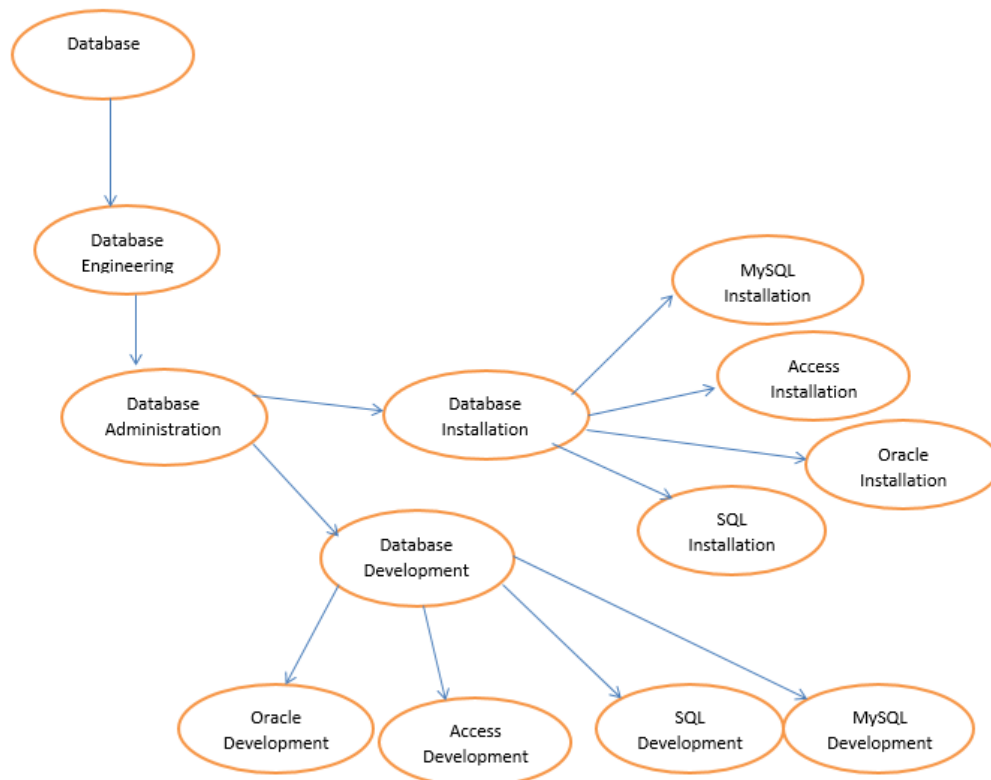
5.4 Ιεραρχία Πληροφορικής

Για να μπορούν να λειτουργήσουν αυτές οι παράμετροι χρειάστηκε να γίνει μια κατηγοριοποίηση των πεδίων της πληροφορικής με βάση τα μαθήματα που γίνονται στο Πανεπιστήμιο Κύπρου και με τις συνηθισμένες εργασίες που ζητά ο κόσμος σε παρόμοια συστήματα που μελέτησα.

Πιο κάτω φαίνεται η γενική ιεραρχία της Πληροφορικής



Εδώ φαίνεται ένα παράδειγμα της κατηγοριοποίησης του πεδίου Databases.



Σε κάθε φοιτητή υπάρχει αποθηκευμένη αυτή η ιεραρχία. Βλέπε Παράρτημα Α για ολόκληρη την ιεραρχία πληροφορικής.

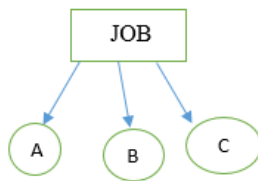
Η μελέτη και η δημιουργία της ιεραρχίας δεν έγινε αποκλειστικά με βάση της ακαδημαϊκής πλευράς αλλά έγινε και με βάση τις εργασίες που δημοσίευσαν οι πελάτες σε άλλα παρόμοια συστήματα, έχοντας υπόψιν και τα μαθήματα που προσφέρονται στο Πανεπιστήμιο Κύπρου.

5.5 Ταίριασμα μεταξύ πελάτη και φοιτητή

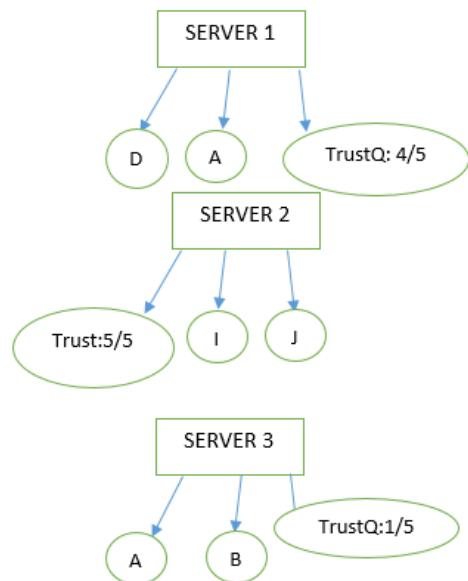
Όπως ανέφερα πιο πάνω, θέλουμε να δημιουργήσουμε μια διαδικασία ταιριάσματος ανάμεσα στις εργασίες και στους φοιτητές ανάλογα με τις παραμέτρους της εργασίας, τις οποίες βάζει ο χρήστης έτσι ώστε να επιλεγθούν οι κατάλληλοι φοιτητές. Πιο συγκεκριμένα, ο πελάτης θα δημοσιεύει μια εργασία και θα προσθέτει τις παραμέτρους που επιθυμεί. Στην συνέχεια το σύστημα θα αναλύει αυτές τις παραμέτρους και θα προσπαθεί να τις ταιριάζει όσο πιο σωστά μπορεί με τις παραμέτρους του πελάτη έτσι ώστε να δώσουμε στον Server το καταλληλότερο άτομο για την δουλειά.

Τα ακόλουθα παραδείγματα δείχνουν πως θέλουμε να δουλεύει η διαδικασία του ταιριάσματος.

Παράδειγμα 1



*Όπου Job είναι η εργασία που δημοσίευσε ο χρήστης.
Τα γράμματα του αλφαβήτου είναι οι παράμετροι της.



Όπως βλέπουμε στο πιο πάνω σχήμα έχουμε μια εργασία και τρεις διαθέσιμους φοιτητές. Πρέπει να αποφασίσουμε ποιος είναι ο πιο κατάλληλος φοιτητής για την εργασία αυτή. Ο τρόπος που θα γίνει αυτό είναι με τις παραμέτρους της συγκεκριμένης εργασίας. Η εργασία έχει τις παραμέτρους A, B και C. Για να γίνει σωστό ταίριασμα θα πρέπει να βρούμε τον φοιτητή που έχει κάποιες από αυτές τις παραμέτρους ή όλες αν είναι δυνατόν. Θα λάβουμε όμως σοβαρά υπόψη την εμπιστοσύνη ποιότητας και αυτό θέλουμε να τονίσουμε σε αυτό το παράδειγμα.

Διαδικασία:

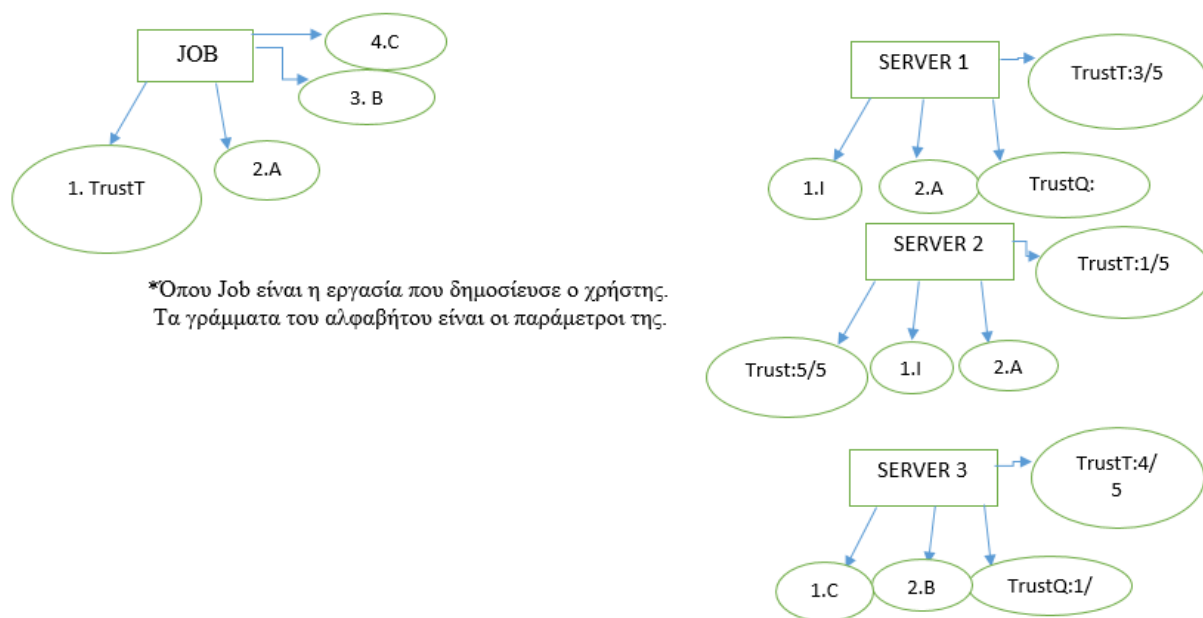
- Ο πρώτος φοιτητής έχει την κοινή παράμετρο A , με αρκετά ψηλή εμπιστοσύνη ποιότητας 4/5
- Ο δεύτερος φοιτητής έχει πολύ ψηλή εμπιστοσύνη ποιότητας αλλά δεν έχει καμία από τις παραμέτρους της εργασίας.

Ο τρίτος φοιτητής έχει δύο κοινές παραμέτρους αλλά η εμπιστοσύνη ποιότητας είναι πολύ χαμηλή με 1/5

Στην περίπτωση αυτή ο δεύτερος φοιτητής απορρίπτεται αμέσως εφόσον δεν μπορεί να γίνει ταίριασμα με τις παραμέτρους του. Ο τρίτος αν και έχει δύο κοινές παραμέτρους, τις περισσότερες και από τους τρεις, η εμπιστοσύνη ποιότητας του είναι πολύ χαμηλή. Άρα σε αυτή την περίπτωση η καλύτερη επιλογή είναι ο πρώτος φοιτητής. Ακόμα και αν μόνο μια παράμετρος είναι κοινή, η εμπιστοσύνη ποιότητας έχει μεγαλύτερο βάρος.

Αν όμως δεν λαμβάναμε υπόψη την εμπιστοσύνη ποιότητας, η καλύτερη επιλογή θα ήταν το τρίτος φοιτητής.

Παράδειγμα 2



Έχουμε και πάλι μια εργασία και τρεις διαθέσιμους φοιτητές. Σε αυτό το παράδειγμα έχουμε ιεραρχία των παραμέτρων και στις δύο πλευρές. Δηλαδή στον Client πρώτο ρόλο παίζει η εμπιστοσύνη ταχύτητας επειδή έχει τον αριθμό 1 το αμέσως μετά είναι το A με 2 και ακολουθούν το B με 3 και το C με 4. Από την μεριά των φοιτητών, για παράδειγμα στον φοιτητή 1 το πιο σημαντικό είναι το I και μετά το A. Άρα σε αυτό το παράδειγμα θα λάβουμε σοβαρά υπόψιν την εμπιστοσύνη ταχύτητα.

Ο πρώτος φοιτητής έχει την κοινή παράμετρο A στην δεύτερη θέση, με αρκετά υψηλή εμπιστοσύνη ποιότητας με 4/5 και καλή εμπιστοσύνη ταχύτητας με 3/5.

Ο δεύτερος φοιτητής έχει την μέγιστη εμπιστοσύνη ποιότητας και επίσης την κοινή παράμετρο A στην δεύτερη θέση αλλά έχει την χαμηλότερη εμπιστοσύνη ταχύτητας με 1/5

Ο τρίτος φοιτητής έχει δύο κοινές παραμέτρους C στην πρώτη θέση και B στην δεύτερη θέση, η εμπιστοσύνη ποιότητας είναι πολύ χαμηλή με $1/5$ αλλά η εμπιστοσύνη ταχύτητας του είναι πολύ ψηλή με $4/5$.

Στην περίπτωση αυτή ο δεύτερος φοιτητής απορρίπτεται αμέσως επειδή έχει την χαμηλότερη εμπιστοσύνη ταχύτητας και στις παραμέτρους του πελάτη η εμπιστοσύνη ταχύτητας είναι πρώτη. Ο τρίτος αν και έχει δύο κοινές παραμέτρους, τις περισσότερες και από τους τρείς, η εμπιστοσύνη ποιότητας του είναι πολύ χαμηλή αλλά η εμπιστοσύνη ταχύτητας του είναι πολύ ψηλή. Ο πρώτος έχει μια μία κοινή παράμετρο με καλή εμπιστοσύνη ποιότητας και καλή εμπιστοσύνη ταχύτητας. Θα μπορούσαμε να επιλέξουμε τον 3ο λόγο της πολύ καλής του εμπιστοσύνης ταχύτητας αλλά λόγω του ότι έχει την χαμηλότερη εμπιστοσύνη ποιότητας τον απορρίπτουμε.

Άρα η καλύτερη επιλογή είναι ο πρώτος φοιτητής.

Κεφάλαιο 6

Αλγόριθμοι Συστήματος

6.1 Ταίριασμα	29
6.2 Αλγόριθμος 1	30
6.3 Αλγόριθμος 2	32
6.4 Υπολογισμός Παραμέτρου Job Compatibility	35

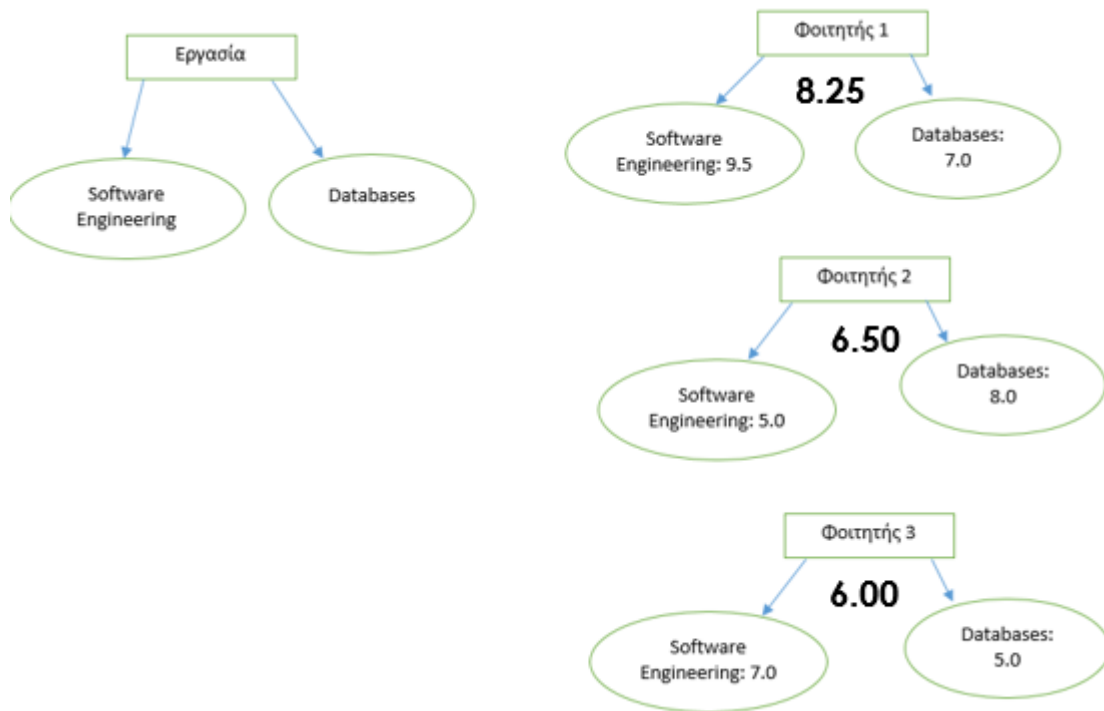
Περίληψη

Σε αυτό το κεφάλαιο εξηγείτε λεπτομερώς το ταίριασμα που προσπαθούν να επιτύχουν οι δύο αλγόριθμοι που υλοποιήθηκαν στο σύστημα και περιγράφονται με την βοήθεια παραδειγμάτων.

6.1 Ταίριασμα

Για να γίνει ένα επιτυχημένο ταίριασμα πρέπει και ο χρήστης και ο πελάτης να έχουν κάποιες παραμέτρους και για να γίνει ένα ταίριασμα μεταξύ κάποιου πελάτη και κάποιου χρήστη πρέπει αυτές οι παράμετροι να είναι κοινοί.

Αυτό φαίνεται στο πιο κάτω παράδειγμα.



Ο χρήστης δημοσίευσε μια εργασία η οποία περιλαμβάνει τα πεδία Software Engineering και Databases.

Στα δεξιά βλέπουμε του τρεις χρήστες που είναι αποθηκευμένοι στο σύστημα.

Για να βρούμε τον κατάλληλο φοιτητή σε αυτή την εργασία θα πρέπει να βρούμε τον φοιτητή με τον μεγαλύτερο μέσο όρο και στα δύο πεδία.

Ο φοιτητής 1 έχει μέσο όρο 8.25 , ο φοιτητής έχει μέσο όρο 6.50 και ο φοιτητής 3 έχει μέσο όρο 6.00. Άρα είναι προφανές πως ο καλύτερος φοιτητής για αυτή την εργασία είναι ο φοιτητής 1.

6.2 Αλγόριθμος 1

Ο πρώτος αλγόριθμος που υλοποίησα δουλεύει ως εξής:

- Αν ο χρήστης επιλέξει ένα πεδίο

Ο αλγόριθμος βρίσκει τους καλύτερους πέντε φοιτητές στο συγκεκριμένο πεδίο και τους παρουσιάζει

- Αν ο χρήστης επιλέξει δύο πεδία

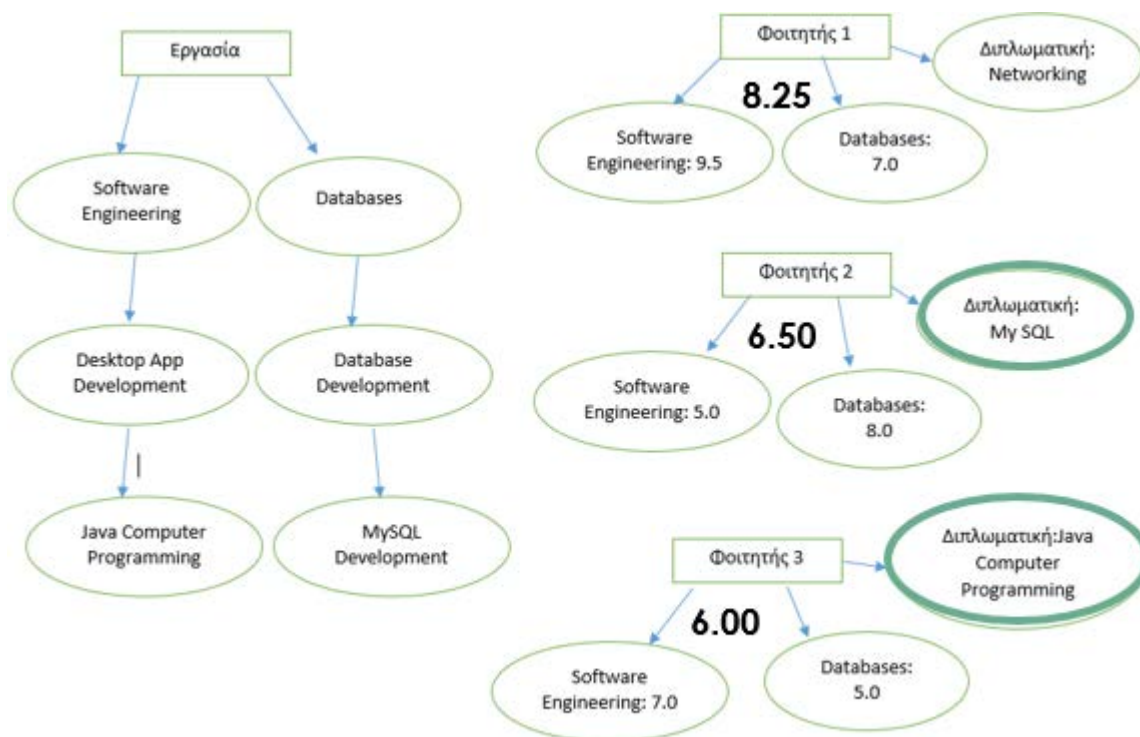
Βρίσκει τον μέσο όρο των δύο πεδίων σε κάθε φοιτητή και παρουσιάζει τους πέντε καλύτερους (με τον πιο ψηλό μέσο όρο και στα δύο πεδία)

- Αν ο χρήστης επιλέξει υποπεδίο/α προτεραιότητα θα έχουν οι φοιτητές που έχουν εργασία στην διπλωματική τους εργασία στο συγκεκριμένο θέμα άσχετα με τον μέσο όρο στα/στο πεδίο/α.

Αν υπάρχουν περισσότεροι από ένας φοιτητές που έχουν στην διπλωματική τους εργασία κάποιο από τα υποπεδία ταξινομούνται με τον βαθμό του/των πεδίου/ων και παρουσιάζονται οι πέντε καλύτεροι.

Αν δεν υπάρχουν φοιτητές κατάλληλοι για κάποιο υποπεδίο, ο αλγόριθμος θα προσπαθήσει να βρει φοιτητές κατάλληλους για το αμέσως ανώτερο επίπεδο.

Πιο κάτω φαίνεται ένα παράδειγμα του αλγόριθμου.



Ο χρήστης δημοσίευσε μια εργασία η οποία έχει δύο πεδία και το κάθε πεδίο έχει από ένα υποπεδίο. Τα πεδία μαζί με τα υποπεδία είναι :

1. Software Engineering -> Desktop App Development -> Java Computer Programming
2. Databases -> Database Development -> MySQL Development

Για να βρούμε τον κατάλληλο φοιτητή σε αυτή την εργασία θα πρέπει να βρούμε τον φοιτητή που έχει θέμα στην διπλωματική του εργασία κάποιο από τα δύο υποπεδία. Αν υπάρχουν περισσότεροι από ένα φοιτητές, τότε ο αλγόριθμος θα λειτουργήσει με τον μέσο όρο.

Ο φοιτητής 1 έχει στην διπλωματική του εργασία ένα θέμα σχετικό με τα Networking, άρα αμέσως απορρίπτεται. Ο φοιτητής 2 έχει στην διπλωματική του θέμα σχετικό με την MySQL άρα είναι ένας υποψήφιος. Ο φοιτητής 3 έχει στην διπλωματική του θέμα σχετικό με το Java Computer Programming άρα είναι και αυτός ένας υποψήφιος.

Αφού έχουμε δύο υποψήφιους πρέπει να βρούμε τον καλύτερο από τους δύο βρίσκοντας τον μέσο όρο στα δύο πεδία. Ο φοιτητής 2 έχει μέσο όρο 6.50 και ο φοιτητής 3 έχει μέσο όρο 6.00. Άρα κατάλληλος για αυτή την εργασία είναι ο φοιτητής 2.

6.3 Αλγόριθμος 2

Ο δεύτερος αλγόριθμος δουλεύει ως εξής:

- Αν ο χρήστης επιλέξει ένα πεδίο

Προτεραιότητα θα έχουν οι φοιτητές που έχουν εργασία στην διπλωματική τους εργασία στο συγκεκριμένο θέμα άσχετα με τον μέσο όρο στα/στο πεδίο/α.

Αν υπάρχουν περισσότεροι από ένας φοιτητές που έχουν στην διπλωματική τους εργασία κάποιο από τα υποπεδία ταξινομούνται με τον βαθμό του/των πεδίου/ων και παρουσιάζονται οι πέντε καλύτερες ομάδες.

- Αν ο χρήστης επιλέξει δυο πεδία

Βρίσκει τους καλύτερους πέντε φοιτητές με τον μεγαλύτερο βαθμό στο 1ο πεδίο και τους καλύτερους πέντε φοιτητές με τον μεγαλύτερο βαθμό στο δεύτερο πεδίο και τους παρουσιάζει σαν μια ομάδα.

- Αν ο χρήστης επιλέξει ένα υποπεδίο

Προτεραιότητα θα έχουν οι φοιτητές που έχουν εργασία στην διπλωματική τους εργασία στο συγκεκριμένο θέμα άσχετα με τον μέσο όρο στα/στο πεδίο/α.

Αν υπάρχουν περισσότεροι από ένας φοιτητές που έχουν στην διπλωματική τους εργασία κάποιο από τα υποπεδία ταξινομούνται με τον βαθμό του/των πεδίου/ων και παρουσιάζονται οι πέντε καλύτερες ομάδες.

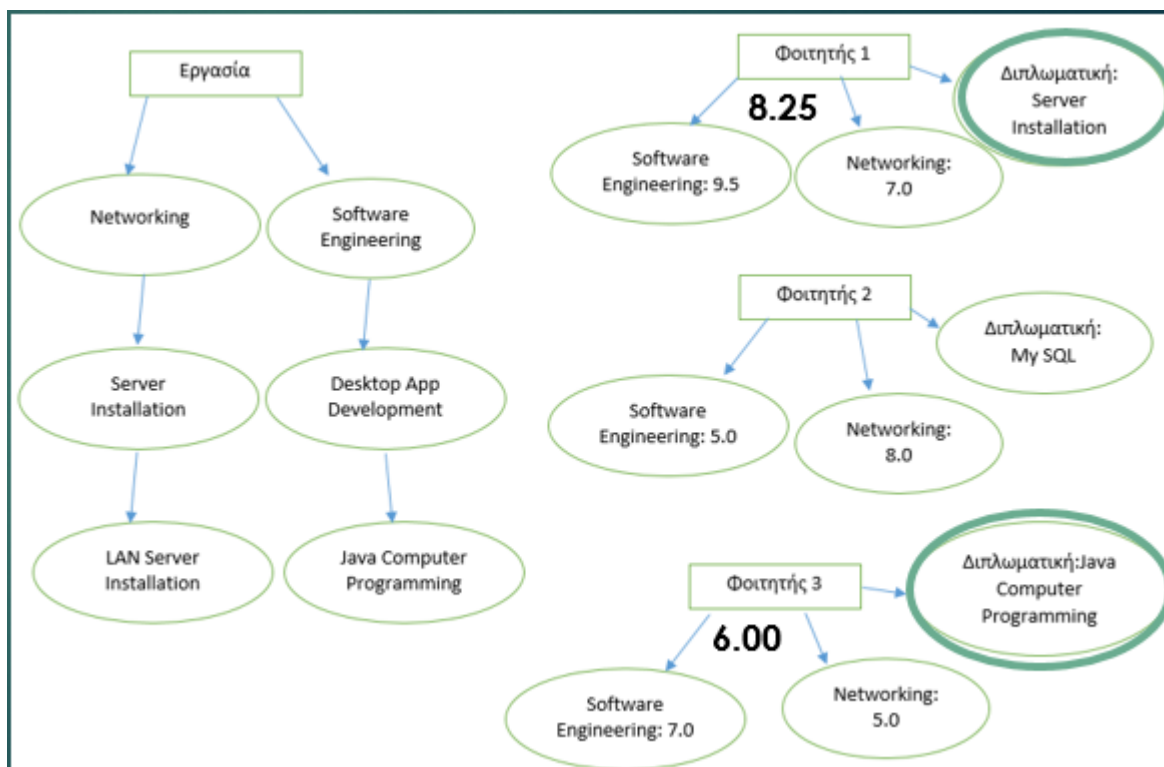
- Αν ο χρήστης επιλέξει δύο υποπεδία

Βρίσκει τους καλύτερους πέντε φοιτητές που έχουν θέμα σχετικό με το 1ο υποπεδίο και τους καλύτερους πέντε φοιτητές που έχουν θέμα σχετικό με το 2ο υποπεδίο και τους παρουσιάζει σαν μια ομάδα. Προτεραιότητα θα έχουν οι φοιτητές που έχουν εργασία στην διπλωματική τους εργασία στο συγκεκριμένο θέμα άσχετα με τον μέσο όρο στα/στο πεδίο/α.

Αν υπάρχουν περισσότεροι από ένας φοιτητές που έχουν στην διπλωματική τους εργασία κάποιο από τα υποπεδία ταξινομούνται με τον βαθμό του/των πεδίου/ων και παρουσιάζονται οι πέντε καλύτεροι.

Αν δεν υπάρχουν φοιτητές κατάλληλοι για κάποιο υποπεδίο, ο αλγόριθμος θα προσπαθήσει να βρει φοιτητές κατάλληλους για το αμέσως ανώτερο επίπεδο.

Πιο κάτω φαίνεται ένα παράδειγμα του αλγόριθμου.



Ο χρήστης δημοσίευσε μια εργασία η οποία έχει δύο πεδία και το κάθε πεδίο έχει από ένα υποπεδίο. Τα πεδία μαζί με τα υποπεδία είναι :

1. Software Engineering -> Desktop App Development -> Java Computer Programming
2. Networking -> Server Installation -> LAN Server Installation

Για να βρούμε τους δύο κατάλληλους φοιτητές σε αυτή την εργασία θα πρέπει να βρούμε τον φοιτητή που έχει θέμα στην διπλωματική του εργασία για το πρώτο υποπεδίο και ένα φοιτητή που έχει θέμα στην διπλωματική του εργασία για το δεύτερο υποπεδίο. Αν υπάρχουν περισσότεροι από ένα φοιτητές, τότε ο αλγόριθμος θα λειτουργήσει με τον μέσο όρο.

Ο φοιτητής 1 έχει στην διπλωματική του εργασία ένα θέμα σχετικό με το Server Installation, άρα είναι ένας υποψήφιος. Ο φοιτητής 2 έχει στην διπλωματική του θέμα σχετικό με την MySQL άρα απορρίπτεται. Ο φοιτητής 3 έχει στην διπλωματική του θέμα σχετικό με το Java Computer Programming άρα είναι και αυτός ένας υποψήφιος.

Αφού έχουμε μόνο δύο υποψήφιους δεν χρειάζεται να ελέγξει ο αλγόριθμος για βαθμολογίες και έτσι οι κατάλληλοι φοιτητές για αυτή την εργασία είναι οι φοιτητές 1 και 3.

Στου πιο πάνω αλγόριθμους σημαντική παράμετρος που παίζει ρόλο είναι η εμπιστοσύνη ποιότητας (Quality Trust). Ακόμα και αν επιλεγθούν οι φοιτητές με την καλύτερη βαθμολογία ή με την ύπαρξη σύνδεσης των πεδίων με την διπλωματική τους εργασία, θα απορριφθούν εάν το Quality Trust είναι μικρότερο του 2.5.

Το ίδιο συμβαίνει και με την ποιότητα χρόνου (Time Trust), εάν ο πελάτης αποφασίσει ότι η εργασία του έχει κρίσιμη διορία.

Στο τελικό σύστημα οι αλγόριθμοι συνδυάστηκαν με τέτοιο τρόπο έτσι ώστε, όταν ο πελάτης επιλέξει ένα πεδίο στην δημοσίευση της εργασίας του, να λειτουργεί ο πρώτος αλγόριθμος και όταν επιλέξει δύο πεδία να λειτουργεί ο δεύτερος αλγόριθμος.

6.4 Υπολογισμός Παραμέτρου Job Compatibility

Η παράμετρος Job Compatibility δείχνει το πόσο κατάλληλος είναι ο φοιτητής για μια συγκεκριμένη εργασία, ως προς την ειδικότητα του και τις βαθμολογίες του στα εμπλεκόμενα πεδία.

Ο υπολογισμός της παραμέτρου υπολογίζεται ως ακολούθως:

- Όταν υπάρχει ένα πεδίο στην εργασία
 - Η παράμετρος αρχικοποιείται με την τιμή «0».
 - Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 2ο υποπεδίο της εργασίας, τότε η παράμετρος παίρνει την τιμή 5 αφού αυτός ο φοιτητής είναι ο καταλληλότερος για την συγκεκριμένη εργασία.
 - Αν υπάρχουν περισσότεροι από ένας φοιτητές με το ίδιο υποπεδίο στην διπλωματική τους εργασία, τότε ο φοιτητής με τον ψηλότερο βαθμό στο κύριο πεδίο, θα εμφανιστεί πρώτος στους συνιστάμενους φοιτητές.
 - Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 1ο υποπεδίο της εργασίας, τότε η παράμετρος παίρνει την τιμή 4.7 αφού αυτός ο φοιτητής είναι ο αμέσως επόμενος κατάλληλος για την συγκεκριμένη εργασία
 - Αν υπάρχουν περισσότεροι από ένας φοιτητές με το ίδιο υποπεδίο στην διπλωματική τους εργασία, τότε ο φοιτητής με τον ψηλότερο βαθμό στο κύριο πεδίο, θα εμφανιστεί σε ψηλότερη θέση στους συνιστάμενους φοιτητές.
 - Οι υπόλοιποι φοιτητές που η διπλωματική τους εργασία δεν έχει σχέση με κανένα από τα δύο υποπεδία, η παράμετρος θα πάρει την τιμή του βαθμού τους, στο κύριο πεδίο και θα διαιρεθεί με το 2.

➤ Όταν υπάρχουν δύο πεδία στην εργασία

- Η παράμετρος αρχικοποιείται με την τιμή «0».
- Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 2ο υποπεδίο του 1^{ου} κύριου πεδίου της εργασίας, τότε η παράμετρος παίρνει την τιμή 5 αφού αυτός ο φοιτητής είναι ο καταλληλότερος για το 1^ο πεδίο της συγκεκριμένης εργασία.
- Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 2ο υποπεδίο του 2ου κύριου πεδίου της εργασίας, τότε η παράμετρος παίρνει την τιμή 5 αφού αυτός ο φοιτητής είναι ο καταλληλότερος για το 2^ο πεδίο της συγκεκριμένης εργασία.
- Αν υπάρχουν περισσότεροι από ένας φοιτητές με το ίδιο υποπεδίο στην διπλωματική τους εργασία, τότε ο φοιτητής με τον ψηλότερο βαθμό στο κύριο πεδίο, θα εμφανιστεί στην 1^η συνιστάμενη ομάδα.
- Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 1ο υποπεδίο του 1ου κύριου πεδίου της εργασίας, τότε η παράμετρος παίρνει την τιμή 4.7 αφού αυτός ο φοιτητής είναι ο αμέσως επόμενος κατάλληλος για το 1^ο πεδίο της συγκεκριμένης εργασία.
- Όταν ένας φοιτητής έχει διπλωματική, η οποία έχει συσχέτιση με το 1ο υποπεδίο του 2ου κύριου πεδίου της εργασίας, τότε η παράμετρος παίρνει την τιμή 4.7 αφού αυτός ο φοιτητής είναι ο αμέσως επόμενος κατάλληλος για το 2^ο πεδίο της συγκεκριμένης εργασία.
- Αν υπάρχουν περισσότεροι από ένας φοιτητές με το ίδιο υποπεδίο στην διπλωματική τους εργασία, τότε ο φοιτητής με τον ψηλότερο βαθμό στο κύριο πεδίο, θα εμφανιστεί σε ψηλότερη θέση στην ομάδα.
- Οι υπόλοιποι φοιτητές που η διπλωματική τους εργασία δεν έχει σχέση με κανένα από τα δύο υποπεδία, η παράμετρος θα πάρει την τιμή του βαθμού τους, στα δύο κύρια πεδία και θα διαιρεθεί με το 2.

Κεφάλαιο 7

Δικαιώματα και ευθύνες

7.1 Περιγραφή	37
7.2 Δικαιώματα και ευθύνες πελατών	37
7.3 Δικαιώματα και ευθύνες φοιτητών	38
7.4 Δικαιώματα και ευθύνες του τμήματος Πληροφορικής	39

Περίληψη

Σε αυτό το κεφάλαιο, αναφέρονται, τα διάφορα δικαιώματα και ευθύνες που πρέπει να λάβουν υπόψιν τους οι πελάτες, οι φοιτητές και το τμήμα Πληροφορικής.

7.1 Περιγραφή

Σε αυτό το κεφάλαιο δίδονται κάποιες εισηγήσεις σχετικά με τις ευθύνες και τα δικαιώματα για αυτούς που ενδέχεται να εμπλακούν με το σύστημα, δηλαδή τους πελάτες, τους φοιτητές και το τμήμα πληροφορικής. Αυτές οι εισηγήσεις είναι το αποτέλεσμα μελέτης άλλων συστημάτων τύπου «Ελεύθερου Επαγγελματία».

7.2 Δικαιώματα και ευθύνες πελατών

Οποιοσδήποτε πελάτης επιθυμεί να εγγραφεί στο σύστημα θα πρέπει να συμφωνήσει και να δεχθεί τους ακόλουθους όρους:

- Ο πελάτης θα πρέπει να είναι άνω των 18 και δεσμεύεται ότι θα δώσει αληθινές, ακριβείς και ολοκληρωμένες πληροφορίες στη φόρμα καταχώρησης (Create Account), στην φόρμα ανατροφοδότησης (Feedback Form), καθώς και στην φόρμα

καταχώρησης μιας εργασίας (Post A Job), και να ενημερώνει τις πληροφορίες του για την διατήρηση της ακρίβειας και της πληρότητας των στοιχείων του.

- Ο πελάτης είναι υπεύθυνος να διαφυλάσσει και να διατηρεί εμπιστευτικότητα για το ψευδώνυμο και τον κωδικό του, τα οποία χρησιμοποιούνται για την εισαγωγή του πελάτη στην σελίδα.
- Μετά την δημοσίευση μιας εργασίας ο πελάτης θα αναμένει έγκριση των συνιστάμενων φοιτητών , από το τμήμα Πληροφορικής εντός 48 ωρών και στην συνέχεια θα μπορεί να επιλέξει φοιτητές.
- Μετά την εκτέλεση μιας εργασίας, ο πελάτης συνιστάται θερμά να βαθμολογήσει τον/τους φοιτητή/ες με βάση την ποιότητα και την ταχύτητα της δουλείας του/τους, στην φόρμα ανατροφοδότησης (Feedback form), καθώς αυτό θα βοηθήσει στην βελτίωση των επιλογών του συστήματος για τις μελλοντικές εργασίες.
- Ο κάθε πελάτης είναι υπεύθυνος για την εισαγωγή του προϋπολογισμού της κάθε εργασίας του καθώς και για την ολοκλήρωση της συμφωνίας του με τον/τους φοιτητές.
- Το τμήμα Πληροφορικής δεν φέρει καμία ευθύνη για την ποιότητα και την ταχύτητα της εκτέλεσης μιας εργασίας από οποιονδήποτε φοιτητή. Αν κάποιος πελάτης δεν είναι ευχαριστημένος με την ολοκλήρωση μιας εργασίας μπορεί να εισάγει τις απόψεις του στην φόρμα ανατροφοδότησης (Feedback form).

7.3 Δικαιώματα και ευθύνες φοιτητών

Οποιοσδήποτε φοιτητής επιθυμεί να εγγραφεί στο σύστημα θα πρέπει να συμφωνήσει και να δεχθεί τους ακόλουθους όρους:

- Ο φοιτητής δέχεται να δώσει την άδεια για χρήση των προσωπικών του στοιχείων και επίσης τις βαθμολογίες των μαθημάτων του της πληροφορικής καθώς και την αναλυτική του βαθμολογία για να εισαχθούν στο σύστημα.
- Ο φοιτητής δεσμεύεται ότι θα είναι ενεργός χρήστης και θα δέχεται να πραγματοποιεί εργασίες εφόσον είναι διαθέσιμος. Επίσης, θα πρέπει να δέχεται οποιαδήποτε σχόλια και κριτικές από τους πελάτες για την ομαλή λειτουργία του συστήματος καθώς αυτό επηρεάζει τις επιλογές του.
- Ο φοιτητής δεσμεύεται να απαντά σε διάφορα ερωτηματολόγια όσο αφορά τις εργασίες που εκτέλεσε και τις εμπειρίες του από το σύστημα, για την συνεχή εξέλιξη και βελτίωση του συστήματος, τα οποία θα δίδονται στους εμπλεκόμενους φοιτητές κάθε τρεις μήνες.
- Οι συμφωνίες μεταξύ των πελατών και την φοιτητών για την εκτέλεση μιας εργασίας, όσο και για την συμφωνία της πληρωμής θα πραγματοποιείται μεταξύ των δύο χωρίς καμιά εμπλοκή του τμήματος Πληροφορικής.
- Όταν ένας πελάτης πληρώνει ένα φοιτητή για μια εργασία, ο φοιτητής μπορεί να δώσει εθελοντικά μέχρι και 5% της πληρωμής του στο Τμήμα Πληροφορικής για την δημιουργία, συντήρηση και παροχή υπηρεσιών του συστήματος.
- Ο φοιτητής δικαιούται να απορρίψει μια εργασία που του ανατέθηκε αφού ενημερώσει έγκαιρα τον πελάτη και το τμήμα Πληροφορικής.

7.4 Δικαιώματα και ευθύνες του τμήματος Πληροφορικής

Το τμήμα Πληροφορικής θα πρέπει να συμφωνήσει και να δεχθεί τους ακόλουθους όρους:

- Δέχεται να παρέχει στο σύστημα τα προσωπικά στοιχεία και τις βαθμολογίες των μαθημάτων της Πληροφορικής των φοιτητών που επιθυμούν να γραφτούν στο σύστημα, μετά από την έγκριση του φοιτητή.

- Μετά από την εμφάνιση των συνιστάμενων φοιτητών στον πελάτη για μια εργασία, το τμήμα θα πρέπει να ελέγχει και να εγκρίνει τις επιλογές του συστήματος καθώς και να ενημερώνει τους εμπλεκόμενους φοιτητές.
- Το τμήμα Πληροφορικής είναι υπεύθυνο για την συντήρηση, την συνεχή παροχή υπηρεσιών, εξέλιξη και βελτίωση του συστήματος.
- Το τμήμα Πληροφορικής είναι υπεύθυνο να εγκρίνει οποιουσδήποτε συνιστάμενους φοιτητές εντός 48 ωρών.

Κεφάλαιο 8

Αξιολόγηση Προσέγγισης και Συστήματος

8.1 Αξιολόγηση Προσέγγισης και Συστήματος

41

Περίληψη

Σε αυτό το κεφάλαιο εξηγούμε με πιο τρόπο θα αξιολογήσουμε το σύστημα, τις μετρικές που θα χρησιμοποιήσουμε και τα αποτελέσματα που παράγουμε.

8.1 Αξιολόγηση Προσέγγισης και Συστήματος

Για την παραγωγή αποτελεσμάτων έκανα τα πιο κάτω βήματα.

- Αποθήκευσα στην βάση 62 φοιτητές, όσο και ο αριθμός των πεδίων και των υποπεδίων, έτσι ώστε να καλυφτεί όλο το φάσμα. Οι βαθμοί των φοιτητών, καθώς και τα Time Trust και Quality Trust παράχθηκαν με τυχαίο τρόπο. Τα ονόματα τους αρχίζουν από το 1 και τελειώνουν στο 62. Τα θέματα των διπλωματικών τους εργασιών, αρχίζουν επίσης από το 1 και τελειώνουν στο 62, όσο είναι και ο αριθμός των πεδίων στο σύστημα.

Name	Surname	ID	SoftwareEngineering	GamingTechnology	Data	Networking	ComputerGraphics	Security	DiplomaProject	Basics	Theory	Total	Other	TrustQuality
#	1	1	5	8.4	7	6	6	8	1	9	8	7	0	3.2
#	2	2	5	6.9	9	5	8	6	2	8	9	7	0	3.2
#	3	3	9	5.3	8	8	6	9	3	9	6	7	0	2.7
#	4	4	8	8.3	5	5	7	7	4	5	8	7	0	1.5
#	5	5	6	5.9	6	8	7	6	5	5	7	6	0	1.5
#	6	6	9	5.7	7	5	8	8	6	6	7	7	0	1.6
#	7	7	7	6.6	6	6	7	6	7	5	7	6	0	4.3
#	8	8	8	6.7	5	8	8	6	8	8	5	7	0	3
#	9	9	7	5.7	7	8	6	5	9	5	7	6	0	1.1
#	10	10	5	8.5	6	6	6	8	10	6	6	6	0	3
#	11	11	8	7.6	6	7	6	6	11	8	6	7	0	4.7
#	12	12	9	6.6	9	5	8	5	12	8	8	7	0	1
#	13	13	7	6.7	6	7	6	5	13	7	6	6	0	1
#	14	14	5	6.2	8	6	6	8	14	8	7	7	0	2.4
#	15	15	9	7.4	6	6	8	9	15	5	6	7	0	2.7
#	16	16	8	6.9	8	8	8	9	16	8	7	8	0	2.5
#	17	17	5	7.9	8	5	5	9	17	7	7	7	0	4.1
#	18	18	8	6	5	5	6	8	18	5	7	6	0	2.7
#	19	19	6	8.8	6	8	6	8	19	7	6	7	0	1.4

- Αποθήκευσα στην βάση 50 εργασίες για να χρησιμοποιηθούν ως είσοδο του συστήματος. Τα πεδία και υποπεδία των εργασιών παράχθηκαν με τυχαίο τρόπο.

Field11	Field12	Field13	Field21	Field22	Field23	Check
3	6	45	4	8	51	0
4	8	51	7	20	0	0
7	16	0	6	14	0	0
6	14	0	2	4	34	0
6	15	0	4	10	54	0
5	11	58	3	6	42	0
5	12	60	3	6	42	0
6	13	62	5	12	60	0
5	12	59	6	15	0	0
4	8	50	2	5	40	0
6	15	0	7	16	0	0
6	13	62	1	3	32	0
5	11	57	7	17	0	0
3	6	45	2	5	40	0
3	7	48	2	5	41	0
6	14	0	4	9	0	0
6	15	0	7	20	0	0
3	7	48	4	10	55	0
5	12	59	3	6	45	0
6	13	61	1	1	21	0

Η στήλη Field11 αναφέρεται στο πρώτο πεδίο και κυμαίνεται από το 1 μέχρι το 7, όσα είναι τα κύρια πεδία.

Η στήλη Field12 αναφέρεται στο υποπεδίο του πρώτου πεδίου και κυμαίνεται από το 1 μέχρι το 20.

Η στήλη Field13 αναφέρεται στο δεύτερο υποπεδίο του πρώτου πεδίου και κυμαίνεται από το 21 μέχρι το 62.

Η στήλη Field21 αναφέρεται στο δεύτερο πεδίο αναφέρεται στο πρώτο πεδίο και κυμαίνεται από το 1 μέχρι το 7, όσα είναι τα κύρια πεδία.

Η στήλη Field22 αναφέρεται στο υποπεδίο του πρώτου πεδίου και κυμαίνεται από το 1 μέχρι το 20.

Η στήλη Field23 αναφέρεται στο δεύτερο υποπεδίο του πρώτου πεδίου και κυμαίνεται από το 21 μέχρι το 62.

Οι αριθμοί που υπάρχουν στις στήλες Field, αντιστοιχούν στα υποπεδία του συστήματος ως ακολούθως:

1	Desktop App Development	24	C++ Computer Programming	47	WAN Server Optimization
2	Mobile Application Development	25	JavaScript App Development	48	E-mail Server Optimization
3	Website Development	26	iOS App Development	49	Linux Server Optimization
4	Database Installation	27	Android App Development	50	3D Animation Development
5	Database Development	28	File Sharing	51	2D Animation Development
6	Server Installation	29	Information Website	54	Video Conversion
7	Server Optimization	30	Personal Website	55	Video Encoding
8	Animation	31	Blog	56	Windows Game Development
9	Image Editing	32	Forum	57	Linux Game Development
10	Video Editing	33	Online Shop	58	Facebook Game Development
11	PC Game Development	34	MySQL Installation	59	iOS Game Development
12	Mobile Game Development	35	Access Installation	60	Android Game Development
13	Penetration Testing	36	Oracle Installation	61	Web App Penetration Testing
14	Virus And Malware Removal	37	SQL Installation	62	Website Penetration Testing
15	File Encryption	38	MySQL Development		
16	Format PC	39	SQL Development		
17	Windows Installation	40	Access Development		
18	Linux Installation	41	Oracle Development		
19	Antivirus Installation	42	LAN Server Installation		
20	Data Recovery	43	WAN Server Installation		
21	Java Computer Programming	44	E-mail Server Installation		
22	C Computer Programming	45	Linux Server Installation		
23	.NET Development	46	LAN Server Optimization		

Το Check αναφέρεται στο αν η διορία (Deadline) της εργασίας είναι αυστηρή ή όχι.

Στο σύστημα έγιναν 7 δοκιμές με 50 τυχαίες εργασίες και με διαφορετικές παραμέτρους:

1^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου
Πεδίο 2
1 ^ο Υποπεδίο 2 ^{ου} πεδίου
2 ^ο Υποπεδίο 2 ^{ου} πεδίου

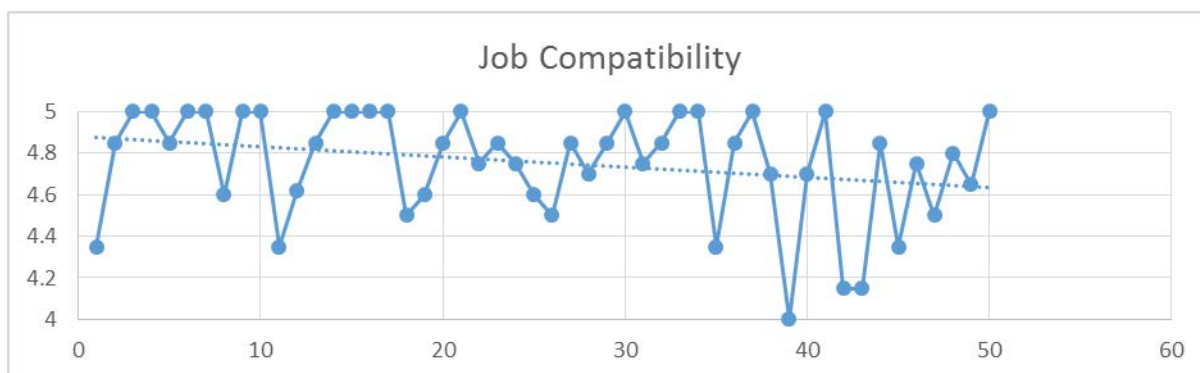
Μετρικές:

- Άξονας y: Job Compatibility
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Job Compatibility η οποία υποδηλώνει την καταλληλότητα του φοιτητή στην συγκεκριμένη εργασία.

Τα αποτελέσματα στις εργασίες αυτές εμφάνισαν 60 δυάδες φοιτητών από τους οποίους το τελικό αποτέλεσμα υπολογίστηκε με τον μέσο όρο κάθε δυάδας στο πεδίο Job Compatibility.

Η παράμετρος Job Compatibility κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι πάρα πολλές τιμές που πήραμε σαν αποτέλεσμα φτάνουν την μεγίστη τιμή της παραμέτρου. Οι τιμές αυτές κυμαίνονται μεταξύ των τιμών 4 και 5, αυτό μας δίνει την δυνατότητα να πούμε ότι ο αλγόριθμος μας βγάζει σαν έξοδο τους καλύτερους φοιτητές που έχουν την μεγαλύτερη τιμή στην συγκεκριμένη μετρική λαμβάνοντας υπόψιν και το Team Trust του κάθε φοιτητή που δεν μπορεί να είναι κάτω από 2.5.

2^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου
Πεδίο 2
1 ^ο Υποπεδίο 2 ^{ου} πεδίου
2 ^ο Υποπεδίο 2 ^{ου} πεδίου

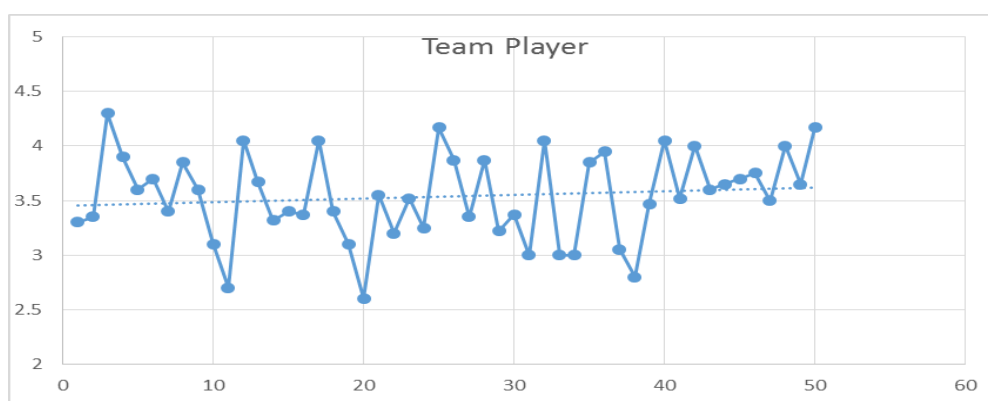
Μετρικές:

- Άξονας y: Team Player
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Team Player η οποία υποδηλώνει το πόσο καλά λειτουργεί σε μια ομάδα ο συγκεκριμένος φοιτητής.

Τα αποτελέσματα στις εργασίες αυτές εμφάνισαν 50 δυάδες φοιτητών από τους οποίους το τελικό αποτέλεσμα υπολογίστηκε με τον μέσο όρο κάθε δυάδας στο πεδίο Team Player.

Η παράμετρος Team Trust κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι τιμές που πήραμε σαν αποτέλεσμα είναι κοντά στην μέγιστη τιμή της παραμέτρου. Οι τιμές αυτές κυμαίνονται μεταξύ των τιμών 2.7 και 4.3, αυτό μας δίνει την δυνατότητα να πούμε ότι ο αλγόριθμος μας είναι αρκετά ικανοποιητικός αφού τα αποτελέσματα που βγάζει είναι αρκετά καλά και ποτέ δεν φτάνει την μικρότερη τιμή, δηλαδή το 2.5.

3^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου
Πεδίο 2
1 ^ο Υποπεδίο 2 ^{ου} πεδίου
2 ^ο Υποπεδίο 2 ^{ου} πεδίου

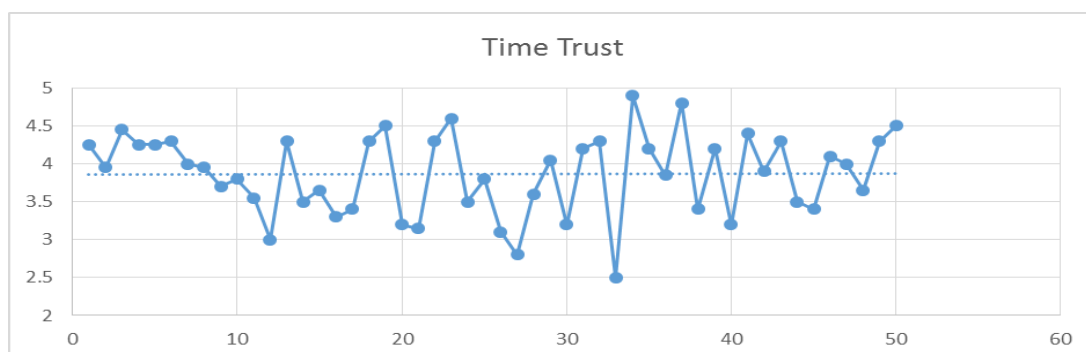
Μετρικές:

- Άξονας y: Time Trust
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Time Trust η οποία υποδηλώνει το πόσο καλά λειτουργεί σε μια ομάδα ο συγκεκριμένος φοιτητής.

Τα αποτελέσματα στις εργασίες αυτές εμφάνισαν 10 δυάδες φοιτητών από τους οποίους το τελικό αποτέλεσμα υπολογίστηκε με τον μέσο όρο κάθε δυάδας στο πεδίο Time Trust.

Η παράμετρος Time Trust κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι τιμές που πήραμε σαν αποτέλεσμα είναι πάρα πολύ κοντά στην μέγιστη τιμή της παραμέτρου. Οι τιμές αυτές κυμαίνονται μεταξύ των τιμών 2.5 και 4.9, αυτό μας δίνει την δυνατότητα να πούμε ότι ο αλγόριθμος μας βέλτιστος αφού μας βγάζει φοιτητές που έχουν ψηλή τιμή στην συγκεκριμένη μετρική που μελετάμε λαμβάνοντας υπόψιν και το Team Trust του κάθε φοιτητή που δεν μπορεί να είναι κάτω από 2.5.

.

4^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου
Πεδίο 2
1 ^ο Υποπεδίο 2 ^{ου} πεδίου
2 ^ο Υποπεδίο 2 ^{ου} πεδίου

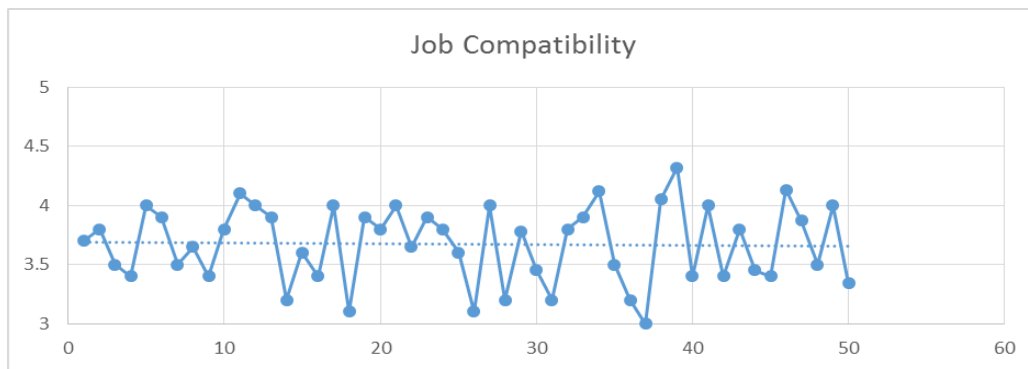
Μετρικές:

- Άξονας y: Job Compatibility Average Of 3 Teams
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Job Compatibility η οποία υποδηλώνει την καταλληλότητα του φοιτητή στην συγκεκριμένη εργασία.

Κάθε εργασία εμφάνισε 3 δυνάδες φοιτητών, από την οποία αρχικά υπολογιζόταν ο μέσος όρος για την συγκεκριμένη παράμετρο για κάθε δυνάδα και στην συνέχεια υπολογιζόταν ο μέσος όρος των αποτελεσμάτων για κάθε μια από τις τρεις δυνάδες.

Η παράμετρος Job Compatibility κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι τιμές που πήραμε σαν αποτέλεσμα δεν είναι σταθερές και δεν κυμαίνονται μεταξύ κάποιου συγκεκριμένου διαστήματος τιμών, αλλά είναι πιο σκορπισμένες στον χώρο. Από αυτό μπορούμε να συμπεράνουμε ότι οι τρεις ομάδες που εμφανίζει μια εργασία μπορούν να έχουν αρκετή διαφορά στην τιμή της συγκεκριμένης μετρικής λαμβάνοντας υπόψιν και το Team Trust του κάθε φοιτητή που δεν μπορεί να είναι κάτω από 2.5.

.

5^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου

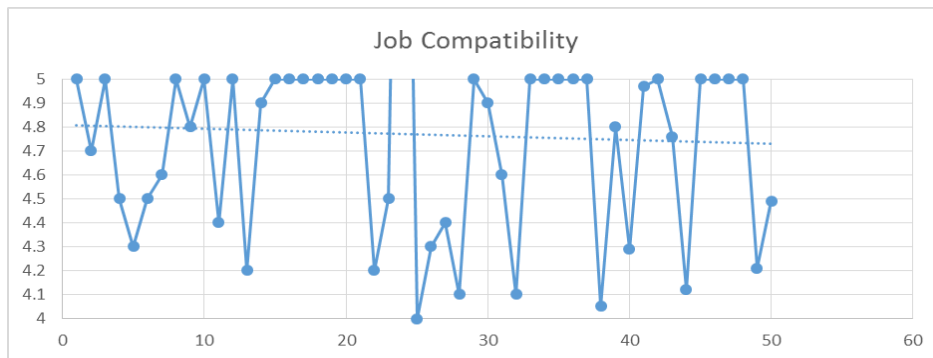
Μετρικές:

- Άξονας y: Job Compatibility
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Job Compatibility η οποία υποδηλώνει την καταλληλότητα του φοιτητή στην συγκεκριμένη εργασία.

Τα αποτελέσματα στις εργασίες αυτές εμφάνισαν 10 φοιτητές από τους οποίους το τελικό αποτέλεσμα αποτελεί το Job Compatibility που αντιστοιχεί σε κάθε φοιτητή.

Η παράμετρος Job Compatibility κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι πάρα πολλές τιμές που πήραμε σαν αποτέλεσμα φτάνουν την μεγίστη τιμή της παραμέτρου. Οι τιμές αυτές κυμαίνονται μεταξύ των τιμών 4 και 5, αυτό μας δίνει την δυνατότητα να πούμε ότι ο αλγόριθμος μας βέλτιστος αφού μας βγάζει φοιτητές που έχουν ψηλή τιμή στην συγκεκριμένη μετρική που μελετάμε λαμβάνοντας υπόψιν και το Team Trust του κάθε φοιτητή που δεν μπορεί να είναι κάτω από 2.5.

.

6^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου

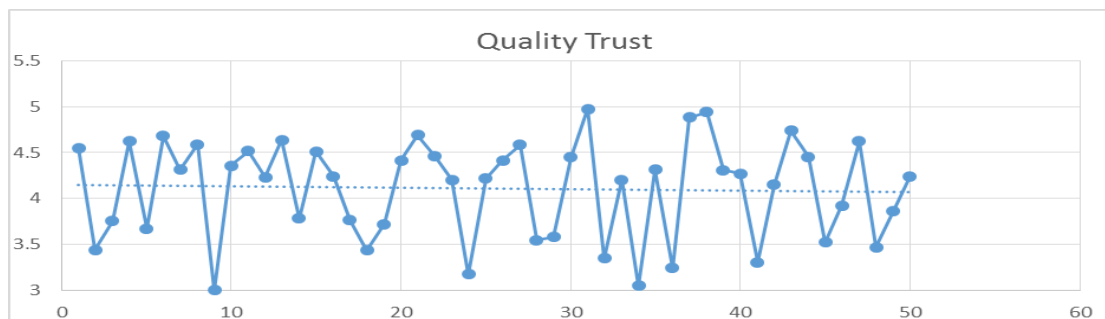
Μετρικές:

- Άξονας y: Quality Trust
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Quality Trust η οποία υποδηλώνει την καταλληλότητα του φοιτητή στην συγκεκριμένη εργασία.

Τα αποτελέσματα στις εργασίες αυτές εμφάνισαν 10 φοιτητές από τους οποίους το τελικό αποτέλεσμα αποτελεί το Quality Trust που αντιστοιχεί σε κάθε φοιτητή.

Η παράμετρος Quality Trust κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι τιμές που πήραμε σαν αποτέλεσμα κυμαίνονται μεταξύ των τιμών 3 και 4.5. Από αυτό μπορούμε να συμπεράνουμε ότι ο αλγόριθμος δίνει σημαντικό ρόλο στην ποιότητα εργασίας του φοιτητή, ανεξάρτητα αν περιέχει και αποτελέσματα λίγο χαμηλότερα, ανάλογα με τις υπόλοιπες παραμέτρους που λαμβάνονται υπόψη κατά την διεξαγωγή του αποτελέσματος.

7^η Δοκιμή:

Είσοδος:

Εργασία
Πεδίο 1
1 ^ο Υποπεδίο 1 ^{ου} πεδίου
2 ^ο Υποπεδίο 1 ^{ου} πεδίου

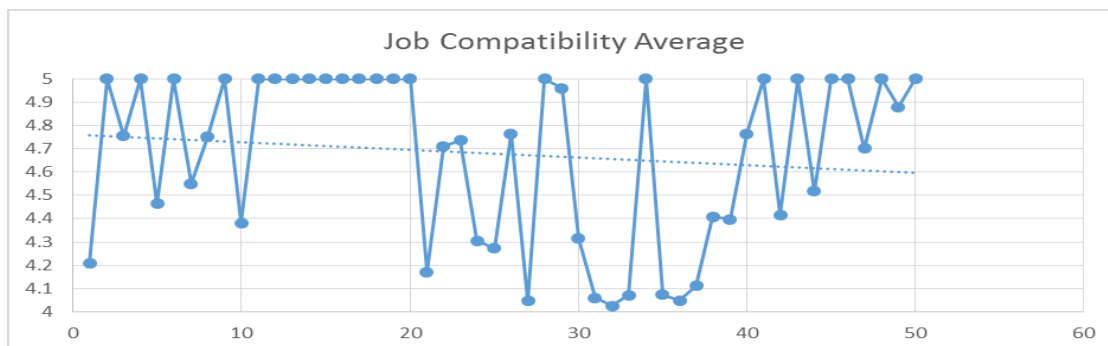
Μετρικές:

- Άξονας y: Job Compatibility Average Of 3 Students
- Άξονας x: Εργασίες

Σκοπός της δοκιμής αυτής είναι η αξιολόγηση της παραμέτρου Job Compatibility η οποία υποδηλώνει την καταλληλότητα του φοιτητή στην συγκεκριμένη εργασία.

Κάθε εργασία εμφάνισε 3 φοιτητές από τους οποίους υπολογίστηκε ο μέσος όρος τους για την παράμετρο Job Compatibility.

Η παράμετρος Job Compatibility κυμαίνεται μεταξύ των τιμών 0 και 5.



Από την πιο πάνω γραφική παράσταση μπορούμε να δούμε ότι οι τιμές που πήραμε σαν αποτέλεσμα κυμαίνονται στο διάστημα (4.2,5). Από αυτό μπορούμε να συμπεράνουμε ότι ο αλγόριθμος εξαγάγει τις περισσότερες φορές άριστα αποτελέσματα, δηλαδή οι τρεις πρώτοι φοιτητές που εμφανίζει το σύστημα για κάθε εργασία αποτελεί τους καλύτερους που μπορεί να βρει με βάση τις παραμέτρους που λαμβάνονται υπόψη, όπου ο κάθε φοιτητής στην τριάδα αυτή έχει κάποια διαφορά με τον επόμενο του και τον προηγούμενο του λαμβάνοντας υπόψιν και το Quality Trust του κάθε φοιτητή που δεν μπορεί να είναι κάτω από 2.5.

.

Κεφάλαιο 9

Συμπεράσματα

9.1 Συμπεράσματα	52
9.2 Μελλοντική Εργασία	52

9.1 Συμπεράσματα

Κύριος στόχος της εργασίας αυτής ήταν να υποβοηθήσουμε τα συστήματα τύπου «Ελεύθερων Επαγγελματιών» με το να αυτοματοποιήσουμε την αναζήτηση των ελεύθερων επαγγελματιών και την δημιουργία ενός ταιριάσματος μεταξύ των δύο, πράγμα που με την ολοκλήρωση της έχει επιτευχθεί επιτυχώς. Η επίτευξη αυτή οφείλεται στην δημιουργία ενός διαδικτυακού συστήματος τύπου «Ελεύθερων Επαγγελματιών» για τους φοιτητές της πληροφορικής του Πανεπιστημίου Κύπρου, με σκοπό την εκτέλεση εργασιών για διάφορους πελάτες.

Έχοντας τρέξει τον αλγόριθμο για διαφορετικό αριθμό εργασιών σαν είσοδο κάθε φορά, έχω συμπεράνει ότι τα αποτελέσματα του αλγορίθμου είναι βέλτιστα. Το σύνολο των φοιτητών που βγάζει σαν έξοδο το σύστημα αποτελεί τους καλύτερους φοιτητές, τους οποίους οι τιμές των παραμέτρων τους έχουν τις ψηλότερες τιμές στις ανάλογες παραμέτρους που απαιτεί η συγκεκριμένη εργασία.

9.2 Μελλοντική Εργασία

Στο μέλλον, θα μπορούσε να δίνεται η δυνατότητα στον πελάτη για επιλογή πέραν των δύο πεδίων εργασίας, κάνοντας την αναζήτηση και το εύρος του πεδίου της εργασίας πιο περίπλοκο και πιο εξειδικευμένο ταυτόχρονα. Κάτι σχετικό με το πιο πάνω, είναι η δυνατότητα πρόσθεσης από τον πελάτη επιπρόσθετου πεδίου εργασίας, το οποίο θα αναφέρεται σε ένα πεδίο το οποίο δεν περιέχεται στην λίστα που ήδη υπάρχει αλλά διευκολύνει την αναζήτηση κατάλληλων φοιτητών.

Επίσης, θα μπορούσε να γίνεται έλεγχος για το ψευδώνυμο (username), που δίνει ο πελάτης κατά την διαδικασία δημιουργίας λογαριασμού (Create Account). Με αυτό τον τρόπο θα αποφεύγεται η ύπαρξη πολλαπλών ίδιων ψευδώνυμων και σύγχυση μεταξύ αυτών των πελατών.

Μια επιπρόσθετη μελλοντική επέκταση για την εργασία αυτή είναι η επεξεργασία του προφίλ του πελάτη, από τον ίδιο, ενημερώνοντας το με τις πληροφορίες που αυτός επιθυμεί και που θέλει να γνωρίζουν οι υπόλοιποι χρήστες του συστήματος. Κάτι σχετικό με την πιο πάνω επέκταση, είναι και η δυνατότητα εμφάνισης παλαιότερων εργασιών του πελάτη, για την δική του διευκόλυνση και ενημέρωση, καθώς επίσης και η δυνατότητα αναζήτησης εργασιών που δημοσιεύτηκαν από άλλους χρήστες.

Τέλος, μια ενδιαφέρον μελλοντική επέκταση του προγράμματος είναι η δυνατότητα αναζήτησης και εμφάνισης των προφίλ των φοιτητών του συστήματος, ώστε κάθε πελάτης να ενημερώνεται πριν επιλέξει κάποιο φοιτητή για κάποια εργασία.

Βιβλιογραφία

Άρθρα:

[1] Recommender Systems In E-commerce by J.Ben Schafer University of Minnesota, Joseph Konstan University of Minnesota, John Riedl University of Minnesota

[2] Combining Collaborative Filtering with Personal Agents for Better Recommendations by Nathaniel Good, J. Ben Schafer, Joseph A. Konstan, Al Borchers, Badrul Sarwar, Jon Herlocker, and John Riedl University of Minnesota

[3] Ontological Matchmaking in Recommender Systems by Angela Bonifati CNR Italy, Giansalvatore Mecca, University Of Basilicata Italy, Domenica Sileo, University Of Basilicata Italy, Gianvito Summa University Of Basilicata Italy

[4] Recommender systems: from algorithms to user experience by Joseph A Konstan & John Riedl

[5] Recommender Systems Review: Types, Techniques and Applications by George A.Sielis of University of Cyprus, Aimilia Tzanavari of University Of Nicosia and Cyprus University of Technology and George A. Papadopoulos of University Of Cyprus

[6] Hybrid Recommender Systems: Survey and Experiments by Robin Burke of Department Of Information Systems and Decision Sciences, California State University

[7] Evaluating Collaborative Filtering Recommender Systems by Jonathan L. Herlocker of Oregon State University, Joseph A. Konstan of University of Minnesota, Loren G. Terveen of University Of Minnesota, John T. Riedl of University Of Minnesota

[8] Recommender Systems by Paul Resnick of AT&T Labs Research, Hal R. Varian Dean of the School of Information Management and Systems University Of California

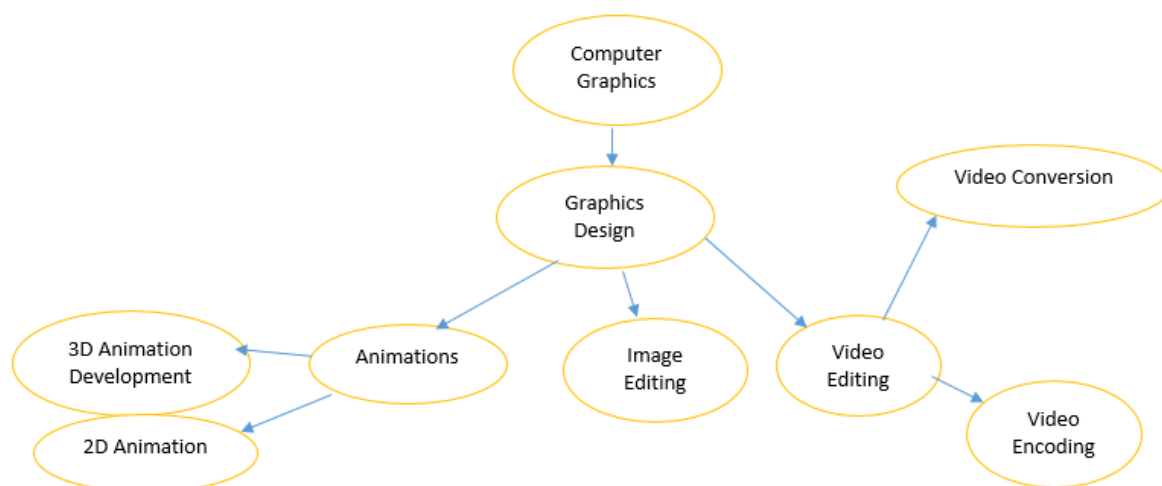
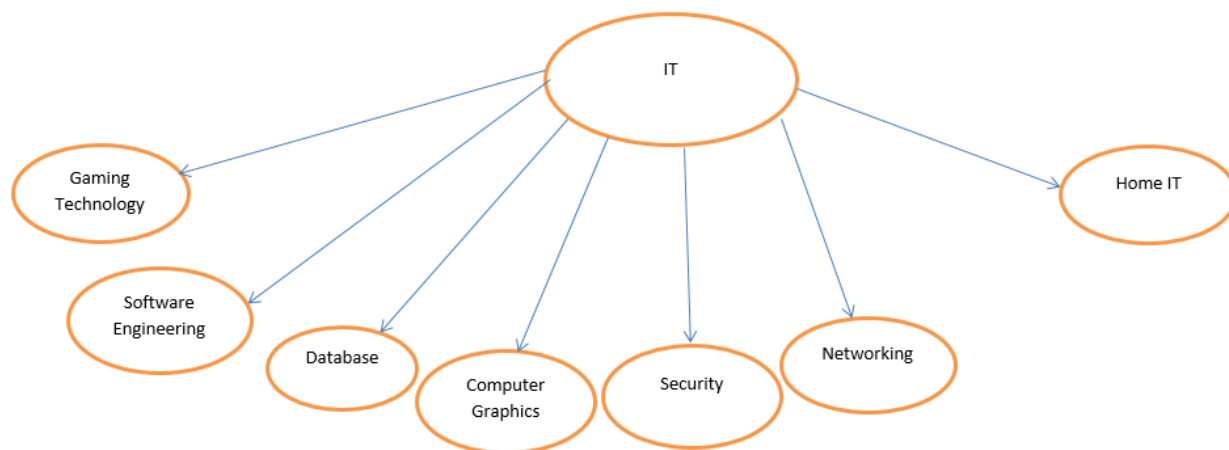
[9] Knowledge-based recommender systems by Robin Burke of Department Of Information and Computer Science of University Of California

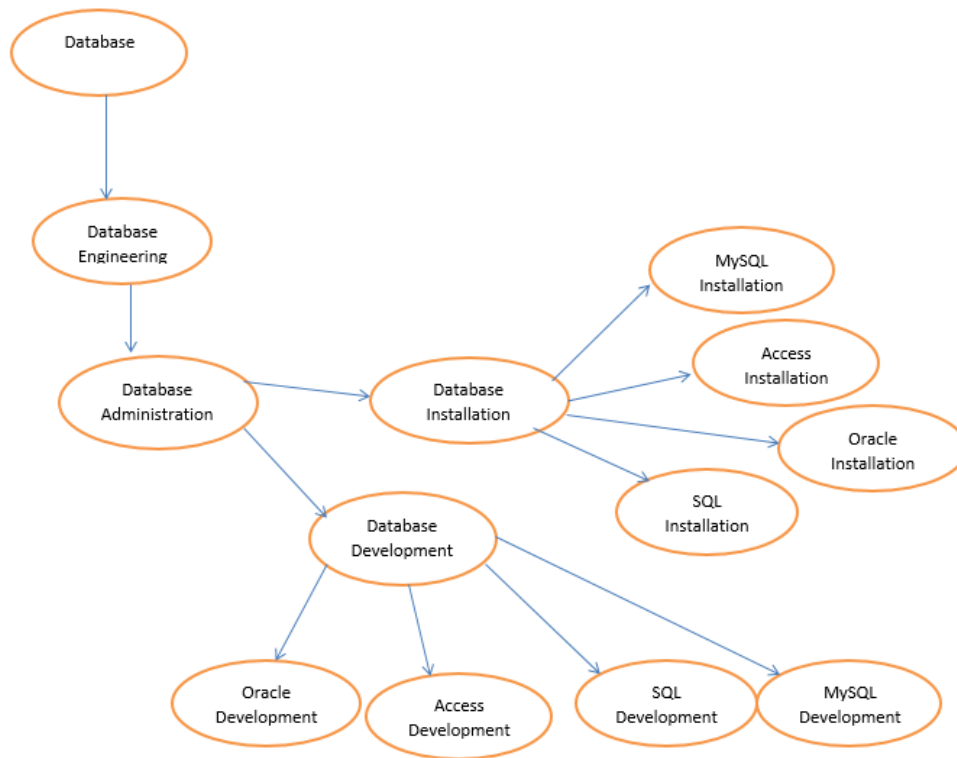
Ιστοσελίδες:

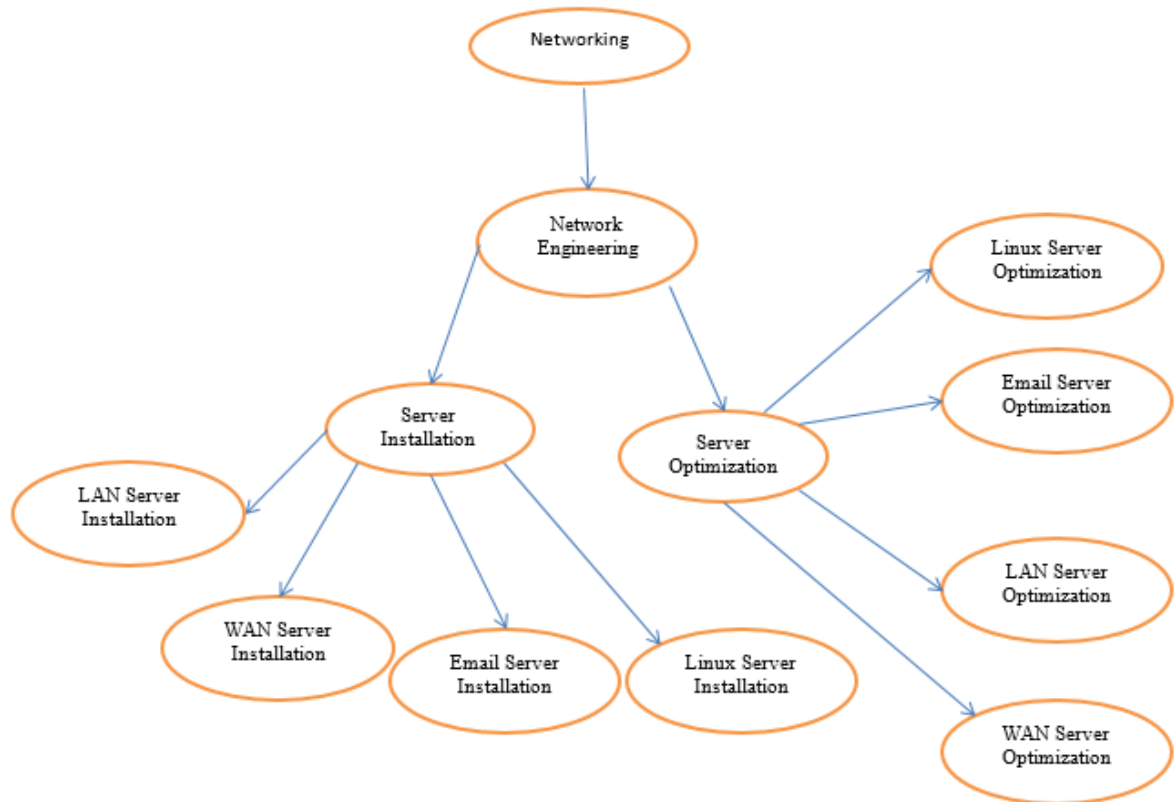
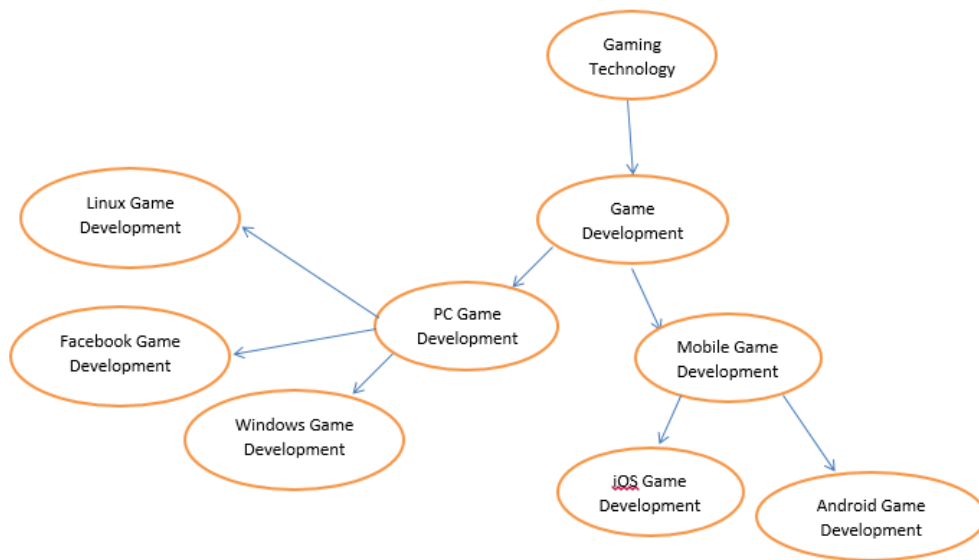
- www.atyourservice.com.cy
- www.elance.com
- www.guru.com
- www.odesk.com
- www.peopleperhour.com
- www.carpooling.com

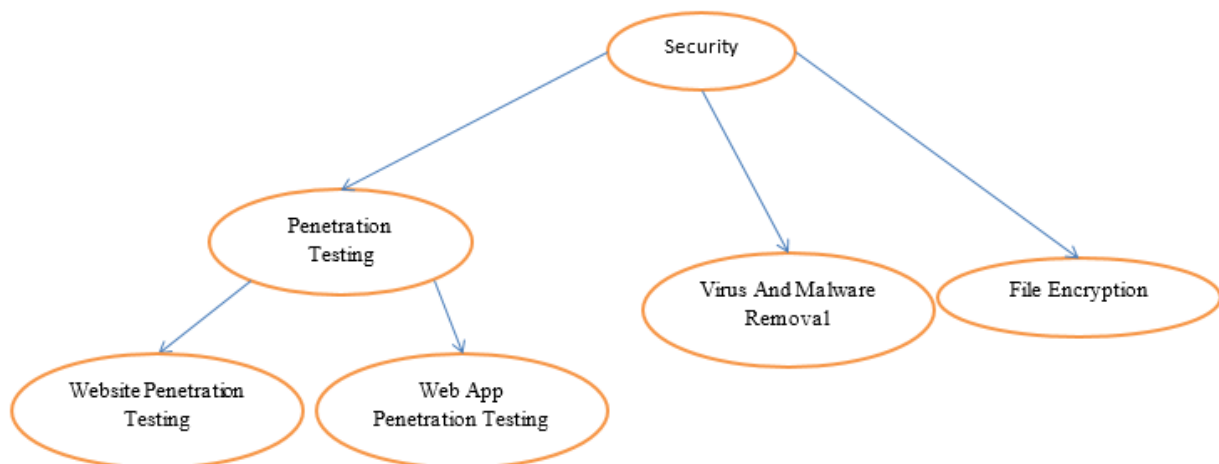
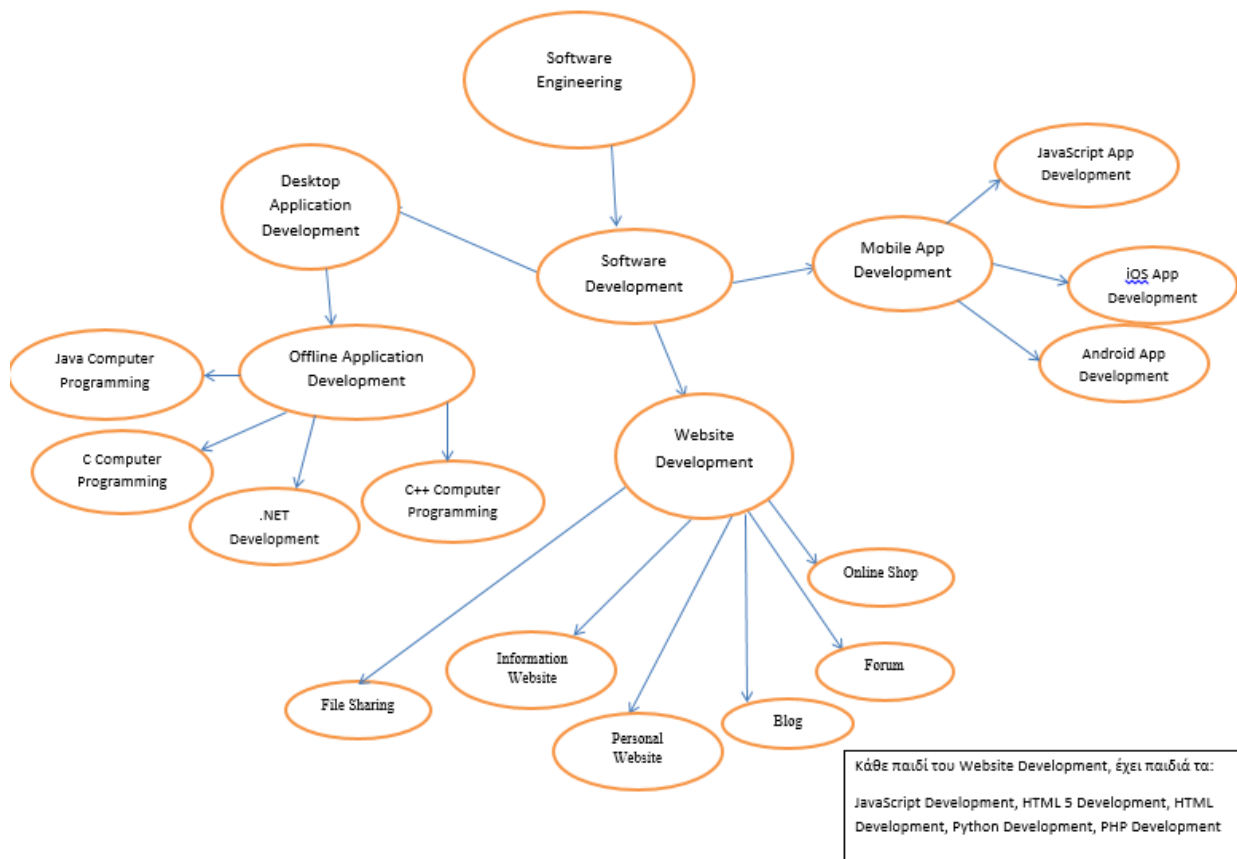
Παράρτημα Α

Εδώ φαίνεται ολόκληρη η ιεραρχία της Πληροφορικής όπως σχεδιάστηκε και μελετήθηκε.









Παράρτημα Β

1ος Αλγόριθμος ταιριάσματος

Αυτός ο αλγόριθμος εκτελείτε όταν ο πελάτης επιλέξει ένα κύριο πεδίο για την εργασία του.

B1. Αποθήκευση φοιτητών που η διπλωματική τους είναι σχετική με κάποιο από τα δύο υποπεδία.

```
Function
algorithm1(foitites, pedio, pedio2, field2, field12, field13, field22, field23, check){

    for(i=0;i<foitites.length;i++){

        if(foitites[i].DP==field13){
            foitites[i].rate+=foitites[i].rate + +2;

            f13.push(foitites[i]);
            foitites.splice(i,1);
            pinakasp3a.push(pedio[i]);
            pinakasp3b.push(pedio2[i]);
            pedio2.splice(i,1);
            pedio.splice(i,1);
            flag1=1;
            flag13=1;
        }//if

    }//for

    for(i=0;i<foitites.length;i++){

        if(foitites[i].DP==field12){

            foitites[i].rate+=foitites[i].rate + +0.5;
            f12.push(foitites[i]);
            foitites.splice(i,1);
            pinakasp2a.push(pedio[i]);
            pinakasp2b.push(pedio2[i]);
            pedio2.splice(i,1);
            pedio.splice(i,1);

            flag1=1;
            flag12=1;
        }//if

    }//for
```

B2. Αποθήκευση φοιτητών στον πίνακα υποψηφίων που η διπλωματική τους είναι σχετική με το 2^ο υποπεδίο.

```
if(flag1==1){  
  
if(flag13==1){ //iparxun foitites me diplomatiki sto pedio13  
for(i=0;i<f13.length;i++)  
temp3.push(f13[i]);  
}  
  
count=0;
```

B3. Αποθήκευση φοιτητών στον πίνακα υποψηφίων που η διπλωματική τους είναι σχετική με το 1^ο υποπεδίο.

```
if(temp.length<5){  
  
if(flag12==1){ //iparxun foitites me diplomatiki sto pedio13  
for(i=0;i<f12.length;i++)  
temp3.push(f12[i]);  
}
```

B4. Αποθήκευση φοιτητών με τους πιο ψηλούς βαθμούς στο συγκεκριμένο πεδίο, στον πίνακα υποψηφίων.

```
if(flag1==0){  
  
if(field2==0){  
while(BestCount<5){  
  
while(count<size){  
//an to best(to epomeno max) ine pio mikro apo ton vathmo  
kai to trust apo 3 kai pano  
  
if(pedio[count]==0)  
pedio[count]=foitites[count].total;  
  
if(best<pedio[count]){  
best=pedio[count];  
position=count;  
  
} //if
```

```

        ++count;
    }//foliasmeno 2owhile

        temp.push(foitites[position]);
        pedio.splice(position,1);          //ton afero apo ton pinaka me
ts vathmus
        foitites.splice(position,1);

        ++BestCount;
        count=0;
        position=0;
        best=1;

    }//while

}

```

B5. Αν το «check» είναι μαρκαρισμένο, τότε οι φοιτητές που έχουν Trust Time μικρότερο του 2.5 αφαιρούνται από τον πίνακα υποψηφίων.

```

if(check=="true"){
    count=0;
    while(count<temp.length){

        if(temp[count].trustT<2.5){
            temp.splice(count,1);
            count=0;
        }//if

        else
            ++count;

    }//while

    //TAKSINOMISI ME BASI TO TRUST TIME

```

B6. Αφαίρεση φοιτητών που το Quality Trust τους είναι μικρότερο του 2.5 και εισαγωγή άλλων φοιτητών που έχουν μεγαλύτερο του 2.5.

```
for(i=0;i<temp.length;i++){  
    if(temp[i].trustQ < 2.5)  
        temp.splice(i,1);  
}  
  
for(i=0;i<foitites.length;i++){  
    if(temp.length<5){  
        if(foitites[i].trustQ >=2.5){  
            temp.push(foitites[i]);  
        }  
    }  
}  
}
```

B7. Ταξινόμηση των υποψήφιων φοιτητών με βάση το Job Compatibility τους.

```
for (j=0; j < (temp.length)-1; j++) {  
    imax = j;  
    for ( i = j+1; i < temp.length; i++) {  
        if (temp[i].rate > temp[imax].rate) {  
            imax = i;  
        }  
    }  
}  
  
if ( imax != j ) {  
    var tmp=new Object();  
    tmp = temp[imax];  
    temp[imax] = temp[j];  
    temp[j] = tmp;  
}  
}
```

```

} //for

for(i=0;i<temp.length;i++){
    if(temp[i].rate>10)
        temp[i].rate=10;
}

for(i=0;i<temp.length;i++){
    temp[i].rate= temp[i].rate/2;
    temp[i].rate= +temp[i].rate+ +4.5;
}

    for(i=0;i<temp.length;i++){
        if(temp[i].trustQ < 2.5)
            temp.splice(i,1);
    }

return (temp);
}

```

2ος Αλγόριθμος ταιριάσματος

Αυτός ο αλγόριθμος εκτελείτε όταν ο πελάτης επιλέξει δύο κύρια πεδία για την εργασία του.

B8. Αποθήκευση φοιτητών που η διπλωματική τους είναι σχετική με κάποιο από τα τέσσερα υποπεδία.

```
function
algorithm2(foitites, foitites2, pedio, pedio2, field2, field12, field13, fi
eld22, field23, check){

    for(i=0;i<foitites.length;i++){

//vrisko tous fitites p exun diplomatiki sto pedio 13

if(foitites[i].DP==field13){
foitites[i].rate+=foitites[i].rate + +2;
f13.push(foitites[i]);
foitites.splice(i,1);
pinakasp13a.push(pedio[i]);
pinakasp13b.push(pedio2[i]);
pedio2.splice(i,1);
pedio.splice(i,1);
flag1=1;

}//if

}//for
for(i=0;i<foitites2.length;i++){

//vrisko tous fitites p exun diplomatiki sto pedio 23

if(foitites2[i].DP==field23){
    foitites2[i].rate+=foitites2[i].rate + +2;
f23.push(foitites2[i]);
pinakasp23a.push(pedio[i]);
pinakasp23b.push(pedio2[i]);
foitites2.splice(i,1);
pedio.splice(i,1);
pedio2.splice(i,1);
flag1=1;

}//if

}//for

//vrisko tous fitites p exun diplomatiki sto pedio 12

for(i=0;i<foitites.length;i++){
```

```

if(foitites[i].DP==field12){
    foitites[i].rate+=foitites[i].rate + +0.5;
    f12.push(foitites[i]);
    foitites.splice(i,1);
    pinakasp12a.push(pedio[i]);
    pinakasp12b.push(pedio2[i]);
    pedio2.splice(i,1);
    pedio.splice(i,1);
    flag1=1;

} //if

} //for

for(i=0;i<foitites2.length;i++){

//vrisko tous fitites p exun diplomatiki sto pedio 22

if(foitites2[i].DP==field22){
    foitites2[i].rate+=foitites2[i].rate + +0.5;
    f22.push(foitites2[i]);
    pinakasp22a.push(pedio[i]);
    pinakasp22b.push(pedio2[i]);
    foitites2.splice(i,1);
    pedio2.splice(i,1);
    pedio.splice(i,1);
    flag1=1;
} //if

} //for

```

B9. Υπολογισμός του μέσου όρου των δύο πεδίων για τον κάθε φοιτητή, που έχουν στην διπλωματική τους κάποιο από τα τέσσερα υποπεδία.

```

if( flag1==1 ){

var j;
var imax;
var mo;
var mot;

//taksinomisi ton ipopedion me vasi to meso oro t kathe fititi

for (j=0; j < (f13.length)-1; j++) {
    imax = j;

    for ( i = j+1; i < f13.length; i++) {

        if(pinakasp13a[i]==0)
            pinakasp13a[i]=f13[i].total;

```

```

        if(pinakasp13b[i]==0)
            pinakasp13b[i]=f13[i].total;

        mo= (+pinakasp13a[i] + +pinakasp13b[i])/2;
        mot=(+pinakasp13a[imax] + +pinakasp13b[imax])/2;

        if (mo > mot) {

            imax = i;
        }//if
    }//for

    if ( imax != j ) {
        var tmp=new Object();
        tmp = f13[imax];
        f13[imax] = f13[j];
        f13[j] = tmp;

    }//if
}//for

```

```

    for (j=0; j < (f23.length)-1; j++) {
        imax = j;

```

```

    for ( i = j+1; i < f23.length; i++) {

        if(pinakasp23a[i]==0)
            pinakasp23a[i]=f23[i].total;

        if(pinakasp23b[i]==0)
            pinakasp23b[i]=f23[i].total;

        mo= (+pinakasp23a[i] + +pinakasp23b[i])/2;
        mot=(+pinakasp23a[imax] + +pinakasp23b[imax])/2;

        if (mo > mot) {

            imax = i;
        }//if
    }//for

```

```

    if ( imax != j ) {
        var tmp=new Object();
        tmp = f23[imax];
        f23[imax] = f23[j];
        f23[j] = tmp;
    }

```

```

    }//if
}//for

for (j=0; j < (f12.length)-1; j++) {
    imax = j;

    for ( i = j+1; i < f12.length; i++) {

        if(pinakasp12a[i]==0)
            pinakasp12a[i]=f12[i].total;

        if(pinakasp12b[i]==0)
            pinakasp12b[i]=f12[i].total;

        mo= (+pinakasp12a[i] + +pinakasp12b[i])/2;
        mot=(+pinakasp12a[imax] + +pinakasp12b[imax])/2;

        if (mo > mot) {

            imax = i;
        }//if
    }//for

    if ( imax != j ) {
        var tmp=new Object();
        tmp = f12[imax];
        f12[imax] = f12[j];
        f12[j] = tmp;
    }

}

}//if
}//for

    for (j=0; j < (f22.length)-1; j++) {
        imax = j;

        for ( i = j+1; i < f22.length; i++) {

            if(pinakasp22a[i]==0)
                pinakasp22a[i]=f22[i].total;

            if(pinakasp22b[i]==0)
                pinakasp22b[i]=f22[i].total;

            mo= (+pinakasp22a[i] + +pinakasp22b[i])/2;
            mot=(+pinakasp22a[imax] + +pinakasp22b[imax])/2;

            if (mo > mot) {

                imax = i;
            }//if
        }//for
    }
}

```

```

        if ( imax != j ) {
            var tmp=new Object();
            tmp = f22[imax];
            f22[imax] = f22[j];
            f22[j] = tmp;
        }
    }
}

```

```

i=0;

if(f13.length!=0){

    for(i=0;i<f13.length;i++){
        temp.push(f13[i]);
    }

    if(f12.length!=0){

        for(i=0;i<f12.length;i++){
            temp.push(f12[i]);
        }
    }

    if(f23.length!=0){

        for(i=0;i<f23.length;i++){
            temp2.push(f23[i]);
        }

        if(f22.length!=0){

            for(i=0;i<f22.length;i++){
                temp2.push(f22[i]);
            }
        }
    }
}

```

B10. Υπολογισμός και αποθήκευση των φοιτητών με τους πιο ψηλούς βαθμούς στο 1^ο πεδίο, στον 1^ο πίνακα υποψηφίων και τους φοιτητές με τους πιο ψηλούς βαθμούς στο 2^ο πεδίο, στον 2^ο πίνακα υποψηφίων.

```
while(temp.length<5){
while(count<foitites.length){

    //an to best(to epomeno max) ine pio mikro apo ton vathmo
    kai to trust apo 3 kai pano

        if(pedio[count]==0)
            pedio[count]=foitites[count].total;

    if(best<pedio[count]){
        best=pedio[count];
        position=count;
    }//kliei to if

    ++count;

}

//kliei to 2o while

    temp.push(foitites[position]); //vallo ton pinaka to onoma tou 1ou
    upopsifiou fititi
    pedio.splice(position,1);
    foitites.splice(position,1);      //ton afero apo ton pinaka
    me ts vathmus

                                //mikraino to size kata 1
                                //megalono to size tou pinaka me ts
    ipopsisfious kata 1
        count=0;
        position=0;
        best=1;

}

count=0;

while(temp2.length<5){

while(count<foitites2.length){
```



```

        if(pedio[count]==0)
            pedio[count]=foitites[count].total;

    if(best<pedio[count]){
        best=pedio[count];
        position=count;

    }//klii to if

    ++count;

} //klii to 2o while

    temp.push(foitites[position]); //vallo ton pinaka to onoma tou 1ou
    upopsifiou fititi
    pedio.splice(position,1);
    foitites.splice(position,1);          //ton afero apo ton pinaka
    me ts vathmus

                                //mikraino to size kata 1
                                //megalono to size tou pinaka me ts
    ipopsisfious kata 1
        ++count2;
        count=0;
        position=0;
        best=1;

} //klii to 1o while

count=0;
best=1;
count2=0;
position=0;

//gia 2o pedio
//vriski tous top 5 k ts vazi stn 2o pinaka

while(count2<5){

while(count<size){
    //an to best(to epomeno max) ine pio mikro apo ton vathmo
    kai to trust apo 3 kai pano

        if(pedio2[count]==0)
            pedio2[count]=foitites2[count].total;

    if(best<pedio2[count]){
        best=pedio2[count];

```

```

        position=count;

    } //klli to if
    ++count;

} //klli to 2o while


    temp2.push(foitites2[position]); //vallo ton pinaka to onoma tou
1ou upopsifiou fititi
    pedio2.splice(position,1);
    foitites2.splice(position,1);          //ton afero apo ton pinaka
me ts vathmus


                                //mikraino to size kata 1
                                //megalono to size tou pinaka me ts
ipopsisfious kata 1
    ++count2;
    count=0;
    position=0;
    best=1;


} //klli to 1o while

}

```

B11. Αν το «check» είναι μαρκαρισμένο, τότε οι φοιτητές που έχουν Trust Time μικρότερο του 2.5 αφαιρούνται και από τους δύο πίνακες υποψηφίων.

```

if(check=="true"){
    count=0;

    while(count<temp.length){

        if(temp[count].trustT<=2){
            temp.splice(count,1);

            count=0;

        }

        else
            ++count;

    }

    count=0;

    while(count<temp2.length){

```

```

        if(temp2[count].trustT<=2){
            temp2.splice(count,1);
            count=0;

        }

        else
            ++count;
    }

    var j;
    var imax;

    for (j=0; j < (temp.length)-1; j++) {
        imax = j;

        for ( i = j+1; i < temp.length; i++) {

            if (temp[i].trustT > temp[imax].trustT) {

                imax = i;
            }//if
        }//for

        if ( imax != j ) {
            var tmp=new Object();
            tmp = temp[imax];
            temp[imax] = temp[j];
            temp[j] = tmp;

        }//if
    }//for

```

```

    for (j=0; j < (temp2.length)-1; j++) {
        imax = j;

        for ( i = j+1; i < temp2.length; i++) {

            if (temp2[i].trustT > temp2[imax].trustT) {

```

```

        imax = i;
    }
}

if ( imax != j ) {
    var tmp2=new Object();
    tmp2 = temp2[imax];
    temp2[imax] = temp2[j];
    temp2[j] = tmp2;
}
}

}

```

B12. Αφαίρεση φοιτητών που το Team Trust τους είναι μικρότερο του 2.5 και εισαγωγή άλλων φοιτητών που έχουν μεγαλύτερο του 2.5.

```

    for (j=0; j < (temp.length)-1; j++) {
        imax = j;

        for ( i = j+1; i < temp.length; i++) {

            if (temp[i].teamT > temp[imax].teamT) {

                imax = i;
            }//if
        }//for

        if ( imax != j ) {
            var tmp=new Object();
            tmp = temp[imax];
            temp[imax] = temp[j];
            temp[j] = tmp;
        }//if
    }//for

    for (j=0; j < (temp2.length)-1; j++) {
        imax = j;

        for ( i = j+1; i < temp2.length; i++) {

            if (temp2[i].teamT > temp2[imax].teamT) {

                imax = i;
            }
        }
    }
}

```

```

        }//if
    }//for

    if ( imax != j ) {
        var tmp=new Object();
        tmp = temp2[imax];
        temp2[imax] = temp2[j];
        temp2[j] = tmp;

    }//if
}//for

    i=0;
    var flagLen=0;

    for(i=0;i<temp.length;i++){
if(temp[i].teamT < 2.5){
temp.splice(i,1);

}
}

for(i=0;i<temp2.length;i++){
if(temp2[i].teamT < 2.5){
temp2.splice(i,1);

}
}

for(i=0;i<foitites.length;i++){
if(temp.length<5){
    if(foitites[i].teamT >=2.5){

        temp.push(foitites[i]);

    }

}

}

for(i=0;i<foitites2.length;i++){
if(temp2.length<5){
    if(foitites2[i].teamT >=2.5){

```

```

        temp2.push(foitites2[i]);
    }

}

}

```

B13. Έλεγχος αν υπάρχουν οι ίδιοι φοιτητές στους δύο πίνακες υποψηφίων και αφαίρεση τους.

```

    while(i<temp.length){

        if(temp[i].ID==temp2[i].ID){

            if(i==temp.length-1){
                var tmp=new Object();
                tmp = temp[i];
                temp[i] = temp[i-1];
                temp[i-1] = tmp;
            }

            else{
                var tmp=new Object();
                tmp = temp[i];
                temp[i] = temp[i+1];
                temp[i+1] = tmp;
            }
        }
        i=i+1;

    }

    i=0;
    var flagLen=0;
    while(i<temp.length){

        if(i==temp.length-1){
            break;
        } //if

        if( (temp[i].ID==temp2[i+1].ID) && (temp[i+1].ID==temp2[i].ID)
    ){

        if(i==temp.length-1){

```

```

        break;

    } //if

    else{

        temp.splice(i,1);
        temp2.splice(i,1);
        flagLen=1;
        //break;

    }//else

    }//if ==

    if(flagLen==1){
        i=0;
        flagLen=0;

    }

    else
        i++;

    for(i=0;i<temp.length;i++){

        temp[i].rate= temp[i].rate/2;
        temp2[i].rate= temp2[i].rate/2;

        temp[i].rate= +temp[i].rate+ +4.5;
        temp2[i].rate= +temp2[i].rate+ +4.5;

    }

}

```

B14. Ταξινόμηση των δύο πινάκων με τους υποψήφιους φοιτητές, με βάση το Job Compatibility τους.

```

var j;
var imax;

for (j=0; j < (temp.length)-1; j++) {
    imax = j;
    for ( i = j+1; i < temp.length; i++) {

        if (temp[i].rate > temp[imax].rate) {

            imax = i;
        }//if
    }//for
}

```

```

        if ( imax != j ) {
            var tmp=new Object();
            tmp = temp[imax];
            temp[imax] = temp[j];
            temp[j] = tmp;

        }//if
    }//for

    for (j=0; j < (temp2.length)-1; j++) {
        imax = j;
        for ( i = j+1; i < temp2.length; i++) {
            if (temp2[i].rate > temp2[imax].rate) {
                imax = i;
            }//if
        }//for

        if ( imax != j ) {
            var tmp=new Object();
            tmp = temp2[imax];
            temp2[imax] = temp2[j];
            temp2[j] = tmp;

        }//if
    }//for

    return [temp,temp2];

} //klili i sinartisi

```