



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας

Σχολή Τεχνολογικών Εφαρμογών (ΣΤΕΦ)

Τμήμα Πληροφορικής

Τίτλος:

**Ανάπτυξη Συστήματος Εργαστηριακών Εγγραφών βάσει Προτιμήσεων
(ΣΕΕΠ), με την βοήθεια Εμπείρου Συστήματος.**

Όνομα συγγραφέα:

Δημοσθένης Νικούδης

(AM 051092)

Επιβλέπουσα Καθηγήτρια:

Αικατερίνη Γεωργούλη

14/4/2011

Περίληψη

Αντικείμενο αυτής της εργασίας είναι ή ανάπτυξη ενός Συστήματος Εργαστηριακών Εγγραφών βάσει Προτιμήσεων (ΣΕΕΠ). Το σύστημα αυτό επιτρέπει σε σπουδαστές να δηλώνουν τις προτιμήσεις τους για τα εργαστηριακά τμήματα στα οποία θέλουν να εγγραφούν και στη συνέχεια τις εισάγει σε Έμπειρο Σύστημα (ΕΣ) το οποίο, αφού λάβει υπόψη διάφορες παραμέτρους, επιχειρεί να τους εγγράψει στα αντίστοιχα τμήματα. Σε περίπτωση που δεν είναι εφικτή η ικανοποίηση όλων των δηλώσεων 1ης προτίμησης, το σύστημα ικανοποιεί τις επόμενες σε σειρά προτίμησης δηλώσεις.

Εκτός από την ανάπτυξη του ΣΕΕΠ, η πτυχιακή περιλαμβάνει μια θεωρητική προσέγγιση για το πρόβλημα των εργαστηριακών εγγραφών και επίσης διερευνά εναλλακτικές λύσεις όπως την προσέγγιση του προβλήματος ως Variable-Sized Bin Packing Problem (VSBPP).

Τέλος περιλαμβάνει αξιολόγηση του ΣΕΕΠ βάσει προκαθορισμένων σεναρίων.

Περιεχόμενα

Θεωρητικό Μέρος.....	5
1 Εισαγωγή	5
1.1 Το πρόβλημα των εγγραφών	5
1.2 Υπάρχουσες Πρακτικές	6
1.3 Προτεινόμενη Προσέγγιση	7
2 Προβλήματα Bin Packing.....	8
2.1 Bin Packing Problem και Variable Sized Bin Packing Problem	8
2.2 Βασικοί Αλγόριθμοι επίλυσης του Bin Packing	9
2.3 Ο αλγόριθμος Iterative First-Fit-Decreasing	10
2.4 Generalized Bin Packing Problem (GBPP)	11
2.5 Συμπέρασμα.....	12
3 Συστήματα Γνώσης και Έμπειρα Συστήματα	13
3.1 Συστήματα Βασισμένα στη Γνώση	13
3.2 Αρχιτεκτονική Συστημάτων Γνώσης.....	15
3.3 Η γλώσσα CLIPS.....	19
3.4 Εισαγωγή στο JESS	21
Πρακτικό Μέρος	22
4 Ανάπτυξη Συστήματος Εργαστηριακών Εγγραφών Βάσει Προτιμήσεων (ΣΕΕΠ)	22
5 Ανάπτυξη Εμπείρου Συστήματος Κληρώσεων	24
5.1 Περιγραφή Λειτουργίας.....	24
5.2 Καθορισμός facts και Αντικειμένων.....	27
5.3 Καθορισμός Κανόνων.....	29
5.4 Προτεραιότητες Εγγραφής/Ειδικές Προτεραιότητες	32
6 Περιβάλλον Δήλωσης Προτιμήσεων.....	33
6.1 Εισαγωγή.....	33
6.2 Δικαιώματα χρηστών και πρόσβαση	34
6.3 Λειτουργίες για τον Σπουδαστή.....	35
6.4 Λειτουργίες για τον Καθηγητή	43
6.5 Λειτουργίες για τον Διαχειριστή	45

7	Αξιολόγηση	51
7.1	Δοκιμαστικά εργαλεία και παραδοχές	51
	A' Φάση Αξιολόγησης	52
7.2	Απλή περίπτωση εγγραφής σπουδαστή σε μη-γεμάτο τμήμα (Σενάριο 1)	52
7.3	Ολική αποτυχία εγγραφής σε μάθημα (Σενάριο 2)	53
7.4	Αποτυχία εγγραφής στην πρώτη επιλογή (Σενάριο 3)	54
7.5	Αποτυχία εγγραφής λόγω σύγκρουσης ωρών (Σενάριο 4)	55
	B' Φάση Αξιολόγησης	56
7.6	Δοκιμή με πραγματικούς σπουδαστές	56
7.7	Αποτελέσματα Πρώτης Κλήρωσης (23/03/2011)	57
7.8	Αποτελέσματα Δεύτερης Κλήρωσης (30/03/2011)	59
7.9	Συμπεράσματα B' Φάσης Αξιολόγησης	61
8	Τελικά συμπεράσματα και μελλοντικοί στόχοι	62
9	Βιβλιογραφία	63
	Παράρτημα Α	64
	Διαχείριση Δεδομένων	64
	Παράρτημα Β	66
	Οδηγίες Εγκατάστασης και Ελάχιστες Απαιτήσεις	66
	Παράρτημα Γ	67
	Επεκτασιμότητα του Περιβάλλοντος Δήλωσης Προτιμήσεων	67
	Παράρτημα Δ	68
	Γλωσσάρι	68

Θεωρητικό Μέρος

1 Εισαγωγή

1.1 Το πρόβλημα των εγγραφών

Ένα πρόβλημα που καλούνται συχνά να αντιμετωπίσουν διάφορα ακαδημαϊκά ιδρύματα είναι αυτό της εγγραφής σπουδαστών σε εργαστηριακά τμήματα. Πιο συγκεκριμένα, αφορά την όσο το δυνατό καλύτερη κατανομή των σπουδαστών μέσα στα διαθέσιμα εργαστηριακά τμήματα. Το πρόβλημα είναι περίπλοκο επειδή εμπεριέχει πολλές παραμέτρους, κάποιες από τις οποίες είναι υποκειμενικές και εξαρτώνται από ποιά όψη βλέπει κανείς το πρόβλημα.

- Από την όψη του μεμονωμένου σπουδαστή ο στόχος είναι να εγγραφεί σε όσο το δυνατό περισσότερα από τα τμήματα που επιθυμεί. Η επιλογή αυτών των τμημάτων γίνεται με βάση προσωπικά κριτήρια του σπουδαστή, όπως για παράδειγμα οι ημέρες/ώρες που διδάσκεται ένα συγκεκριμένο εργαστηριακό τμήμα, η αντίληψη που έχει ο σπουδαστής για τον διδάσκοντα του συγκεκριμένου τμήματος και άλλα.
- Από την πλευρά του ιδρύματος/υπεύθυνων καθηγητών ο στόχος είναι η όσο το δυνατό καλύτερη κατανομή στα διαθέσιμα εργαστηριακά τμήματα για την αποφυγή φαινομένων υπερπλήρωσης που θα μπορούσαν να δημιουργήσουν προβλήματα στην οργάνωση και την εκπαιδευτική διαδικασία.

Επειδή πολλές φορές οι επιλογές των σπουδαστών συμπίπτουν, κάθε λύση καταλήγει στο να υπάρχουν κάποιοι σπουδαστές που δεν καταφέρνουν να εγγραφούν στα τμήματα της επιλογής τους. Ακόμα χειρότερα, μπορούν να υπάρξουν περιπτώσεις όπου κάποιοι σπουδαστές δεν καταφέρνουν να γραφτούν σε κανένα εργαστηριακό τμήμα, λόγω του ότι δεν έχουν τη δυνατότητα να παρακολουθήσουν σε τμήματα πέραν των πρώτων τους επιλογών.

Στόχος αυτής της εργασίας είναι να σχεδιαστεί ένα σύστημα εγγραφής σπουδαστών σε εργαστηριακά τμήματα της προτίμησής τους που να παρέχει έναν ικανοποιητικό βαθμό ελέγχου στους σπουδαστές ενώ ταυτόχρονα να εξασφαλίζει την όσο το δυνατό καλύτερη κατανομή στα διαθέσιμα τμήματα.

1.2 Υπάρχουσες Πρακτικές

Μια παραδοσιακή μέθοδος εγγραφών είναι αυτή της φυσικής εγγραφής, δηλαδή οι ενδιαφερόμενοι σπουδαστές συγκεντρώνονται σε κάποιο χώρο (πχ. έξω από το γραφείο του υπεύθυνου καθηγητή) και εγγράφονται από τον καθηγητή στα αντίστοιχα εργαστηριακά τμήματα. Η σειρά με την οποία γίνονται οι εγγραφές εξαρτάται από τον καθηγητή, αλλά συνήθως είναι με μορφή First-In First-Out (FIFO). Η μέθοδος αυτή, αν και αποτελεσματική, είναι χρονοβόρα και για τους σπουδαστές που πρέπει να προσέλθουν και να περιμένουν στην ουρά, αλλά και για τον καθηγητή που πρέπει να τους εγγράψει χειροκίνητα και να διατηρήσει μια τάξη.

Με την πρόοδο της τεχνολογίας και τη δυνατότητα αξιόπιστης σύνδεσης στο Internet για τους περισσότερους σπουδαστές η παραπάνω μέθοδος έχει αντικατασταθεί σε μεγάλο βαθμό από τη μέθοδο της ηλεκτρονικής εγγραφής. Αντί να προσέλθουν οι ενδιαφερόμενοι σπουδαστές σε κάποιο φυσικό χώρο, συνδέονται σε κατάλληλο διαδικτυακό χώρο και αφού η ταυτότητα τους πιστοποιηθεί (μετά από login) μπορούν να κάνουν την εγγραφή τους ηλεκτρονικά.

Για το πρόβλημα του ανταγωνισμού στις εγγραφές χρησιμοποιείται πολλές φορές μια μέθοδος χειροκίνητης εγγραφής όπου οι εγγραφές "ανοίγουν" σε μια προκαθορισμένη ημερομηνία και ώρα και οι σπουδαστές πρέπει να ακολουθήσουν τα κατάλληλα links για να εγγραφούν. Όταν κάποιο εργαστηριακό τμήμα γεμίσει, οι εγγραφές σε αυτό "κλείνουν" και οι σπουδαστές που δεν πρόλαβαν να εγγραφούν πρέπει να επιλέξουν διαφορετικό τμήμα. Αυτό μειώνει σημαντικά τον χρόνο που απαιτείται από τους σπουδαστές, αφού δεν υπάρχει πλέον ουρά, αλλά και από τον καθηγητή αφού δεν χρειάζεται να εγγράφει ο ίδιος τους σπουδαστές.

Αν και η μέθοδος αυτή είναι απλή και αποτελεσματική, έχει και μειονεκτήματα όπως την αδυναμία ενημέρωσης του σπουδαστή με πληροφορίες όπως ο αριθμός των σπουδαστών που θα έχει να ανταγωνιστεί για την εγγραφή. Επιπλέον, σε περίπτωση που αποτύχει να εγγραφεί στην πρώτη του επιλογή, ο σπουδαστής έχει μειωμένες πιθανότητες να εγγραφεί στις επόμενες λόγω του χρόνου που έχασε προσπαθώντας για την πρώτη.

1.3 Προτεινόμενη Προσέγγιση

Στη συγκεκριμένη πτυχιακή προτείνεται μια νέα προσέγγιση στην οποία οι σπουδαστές αντί να εγγράφονται οι ίδιοι στα τμήματα δηλώνουν της προτιμήσεις τους με μορφή προτεραιοτήτων και το σύστημα αναλαμβάνει να τους τοποθετήσει στα εργαστηριακά τμήματα με τον καλύτερο δυνατό τρόπο. Οι παράμετροι που λαμβάνονται υπόψη είναι οι προτιμήσεις των σπουδαστών, τα χαρακτηριστικά των εργαστηριακών τμημάτων (χωρητικότητα, ημέρα/ώρα που διδάσκονται), τυχόν συγκρούσεις που προκύπτουν και τέλος η στρατηγική τοποθέτησης (ποιοι τοποθετούνται πρώτοι) που εξαρτάται από την πολιτική που εισάγεται από τον διαχειριστή αφού οριστεί από το Τμήμα.

Βασικό χαρακτηριστικό αυτής της προσέγγισης είναι ότι οι δηλώσεις προτιμήσεων δεν είναι ανταγωνιστικές και η ταχύτητα υποβολής μιας δήλωσης δεν επηρεάζει τις πιθανότητες τοποθέτησης στα εργαστηριακά τμήματα (εκτός αν οριστεί πολιτική που να την λαμβάνει υπόψη). Στη συνέχεια, σε μεταγενέστερη ώρα και/ή ημερομηνία, γίνεται η λεγόμενη «Κλήρωση» στην οποία το σύστημα εκτελεί τη διαδικασία τοποθέτησης που αναφέρθηκε στην παραπάνω παράγραφο. Μετά την ολοκλήρωση της διαδικασίας Κλήρωσης, προκειμένου η τοποθέτηση να ικανοποιεί όσο το δυνατό πληρέστερα τους σπουδαστές, δίνεται η δυνατότητα «Αποχώρησης» από τα εργαστηριακά τμήματα στα οποία τοποθετήθηκαν. Η Αποχώρηση τους δίνει τη δυνατότητα να δηλώσουν νέες προτιμήσεις για τα συγκεκριμένα εργαστηριακά μαθήματα και η διαδικασία Κλήρωσης πραγματοποιείται εκ νέου. Αυτός ο κύκλος εγγραφών επαναλαμβάνεται για έναν προκαθορισμένο αριθμό φορών μετά τον οποίο η διαδικασία εγγραφών ολοκληρώνεται.

Αυτή η προσέγγιση έχει σημαντικά πλεονεκτήματα για τους σπουδαστές. Ένα πλεονέκτημα είναι η δυνατότητα πρόσβασης σε περισσότερες πληροφορίες. Για παράδειγμα οι σπουδαστές μπορούν κατά τη δήλωση προτιμήσεων να δουν πόσο μεγάλος ανταγωνισμός υπάρχει σε κάθε τμήμα (δηλαδή πόσοι σπουδαστές συνολικά το έχουν δηλώσει ως πρώτη τους προτίμηση). Επίσης έχοντας τη δυνατότητα να σχηματίζουν μια εικόνα με το πώς διαμορφώνεται το πρόγραμμα τους ανάμεσα στις Κληρώσεις, οι σπουδαστές μπορούν να αποφασίζουν καλύτερα για τις επόμενες Προτιμήσεις/Αποχωρήσεις τους. Αυτό τους επιτρέπει να κάνουν πιο συνειδητές επιλογές και να έχουν μεγαλύτερο έλεγχο ως προς το τελικό αποτέλεσμα.

Παρακάτω θα μελετηθούν κάποιες προσεγγίσεις για την ανάπτυξη αυτού του συστήματος, αρχικά προσεγγίζοντας το πρόβλημα ως ένα πρόβλημα Bin Packing και στη συνέχεια ως ένα πρόβλημα Τεχνητής Νοημοσύνης.

2 Προβλήματα Bin Packing

2.1 Bin Packing Problem και Variable Sized Bin Packing Problem

Το Bin Packing Problem είναι ένα πρόβλημα βελτιστοποίησης που ανήκει σε μια ευρύτερη κατηγορία προβλημάτων που ονομάζονται προβλήματα packing. Το Bin Packing μπορεί να οριστεί ως εξής (Friesen and Langston 1986):

Έχουμε ένα σύνολο αντικειμένων με μεγέθη $\alpha_1, \alpha_2, \alpha_3$ κτλ. και έναν πεπερασμένο αριθμό (έστω B) από κάδους/δοχεία ίσης χωρητικότητας V . Ποιό είναι το ελάχιστο B που απαιτείται για μπορέσουν να τοποθετηθούν όλα τα αντικείμενα σε κάποιο κάδο;

Για τις ανάγκες της εργασίας θα μπορούσαμε να θεωρήσουμε τα εργαστηριακά τμήματα σαν κάδους και τους σπουδαστές σαν τα αντικείμενα που θα τοποθετηθούν. Αυτή η προσέγγιση όμως δεν καλύπτει πλήρως το πρόβλημα γιατί ενδέχεται τα τμήματα να μην έχουν κοινό μέγεθος ή προτεραιότητα. Έτσι χρειάζεται μια γενίκευση του προβλήματος, η οποία ονομάζεται Variable Sized Bin Packing Problem (VSBPP) (Friesen and Langston 1986).

Το VSBPP δίνει τη δυνατότητα οι κάδοι να έχουν διαφορετικά μεγέθη και διαφορετικά κόστη ο καθένας. Για κάθε κάδο πολλαπλασιάζεται το κόστος του με το συνολικό όγκο των αντικειμένων που περιέχει. Στόχος είναι να τοποθετηθούν όλα τα αντικείμενα στους κάδους με το ελάχιστο δυνατό συνολικό κόστος.

Αν και το VSBPP έχει αποτελέσει το αντικείμενο πρόσφατων μελετών και έχει προταθεί ένα εύρος μεθόδων για την επίλυση του, ακόμα δεν έχει επιτευχθεί η δημιουργία ενός γενικού πλαισίου μοντελοποίησης που να μπορεί να περιλάβει σε ικανοποιητικό βαθμό όλες τις παραλλαγές του προβλήματος. Ένας από τους λόγους είναι ότι για κάθε πρόβλημα πρέπει να εξετάζονται διάφοροι κανόνες και περιορισμοί όσον αφορά την τοποθέτηση των αντικειμένων στους κάδους, κάτι το οποίο διαφοροποιεί σε μεγάλο βαθμό τα προβλήματα και καθιστά την εύρεση γενικών λύσεων δύσκολη.

Παρακάτω θα εξεταστούν κάποιες από τις μεθόδους που έχουν προταθεί ή αναπτυχθεί μέσα στη βιβλιογραφία για την επίλυση αυτού, ή παρόμοιων προβλημάτων.

2.2 Βασικοί Αλγόριθμοι επίλυσης του Bin Packing

Εφόσον το VSBPP αποτελεί γενίκευση του Bin Packing, θα εξεταστούν αρχικά κάποιοι βασικοί αλγόριθμοι για την επίλυση αυτού του προβλήματος (Coffman, Garey and Johnson 1984). Οι διαφορές τους βρίσκονται στη σειρά τοποθέτησης των αντικειμένων στους κάδους:

- First Fit: Τοποθετεί το αντικείμενο στον πρώτο κάδο στον οποίο μπορεί να χωρέσει.
- Last Fit: Τοποθετεί το αντικείμενο στον τελευταίο κάδο στον οποίο μπορεί να χωρέσει.
- Best Fit: Τοποθετεί το αντικείμενο στον πιο γεμάτο κάδο στον οποίο μπορεί να χωρέσει.
- Worst Fit: Τοποθετεί το αντικείμενο στον πιο άδειο κάδο στον οποίο μπορεί να χωρέσει.
- Almost Worst Fit: Τοποθετεί το αντικείμενο στον δεύτερο πιο άδειο κάδο που μπορεί να χωρέσει. Ο συγκεκριμένος αλγόριθμος έχει αποδειχτεί ότι είναι καλύτερος από τον Worst Fit.

Όλοι οι παραπάνω αλγόριθμοι είναι δουλεύουν σε πραγματικό χρόνο (online), δηλαδή επεξεργάζονται τα αντικείμενα όπως τους έρχονται, χωρίς να χρειάζεται να είναι διαθέσιμη όλη η είσοδος από την αρχή. Οι offline αλγόριθμοι είναι το αντίθετο, δηλαδή απαιτούν να είναι διαθέσιμη ολόκληρη η είσοδος πριν αρχίσουν την επεξεργασία. Αν λοιπόν ταξινομήσουμε τα αντικείμενα σε φθίνουσα ή αύξουσα σειρά και στη συνέχεια εφαρμόσουμε τους παραπάνω αλγορίθμους, τότε καταλήγουμε σε μια νέα σειρά αλγορίθμων με γνωστότερους τους First Fit Decreasing και Best Fit Decreasing (Coffman, Garey and Johnson 1984).

Αυτοί είναι οι πιο απλοί αλγόριθμοι επίλυσης του Bin Packing, αλλά σίγουρα δεν είναι οι μόνοι. Πάνω στο συγκεκριμένο πρόβλημα έχουν δημοσιευτεί δεκάδες, ίσως εκατοντάδες μελέτες που στοχεύουν είτε στο γενικό πρόβλημα είτε σε διάφορες υποπεριπτώσεις του. Αυτό, όμως, που μπορεί να παρατηρηθεί στους παραπάνω απλούς αλγορίθμους σε σχέση με το VSBPP είναι ότι κανένας τους δεν μπορεί να εγγυηθεί τη βέλτιστη λύση όταν το μέγεθος των κάδων δεν είναι κοινό για όλους.

Παρακάτω θα μελετηθούν κάποιες πιο σύνθετες προσεγγίσεις, που αναπτύχθηκαν συγκεκριμένα για το VSBPP.

2.3 Ο αλγόριθμος Iterative First-Fit-Decreasing

Βασισμένοι σε γνωστούς αλγορίθμους όπως ο First-Fit Decreasing και Best-Fit Decreasing, οι Jangha Kang and Sungsoo Park εισήγαγαν δύο νέους αλγορίθμους: τον Iterative First-Fit Decreasing (IFFD) και τον Iterative Best-Fit Decreasing (IBFD) (Kang and Park 2003).

Σε μετάφραση από την παραπάνω ανακοίνωση, ο τρόπος με τον οποίο λειτουργούν οι αλγόριθμοι είναι ο εξής:

"Στον IFFD (ή αντίστοιχα, στον IBFD), τοποθετούμε πρώτα όλα τα αντικείμενα στους μεγαλύτερους κάδους χρησιμοποιώντας τον FFD (ή αντίστοιχα, BFD), και παίρνουμε μια εφικτή λύση. Στη συνέχεια παίρνουμε μια άλλη λύση πακετάροντας ξανά τα αντικείμενα που βρίσκονται στον τελευταίο κάδο της προηγούμενης λύσης στον επόμενο μεγαλύτερο χρησιμοποιώντας πάλι FFD (ή αντίστοιχα, BFD). Συνεχίζοντας αυτή τη διαδικασία, μέχρι να μην μπορούν να πακεταριστούν ξανά, έχουμε στα χέρια μας μερικές λύσεις. Με αυτό το τρόπο, μπορούμε γενικά να πάρουμε μια εφικτή λύση για κάθε τύπο κάδου. Τότε, επιλέγεται η καλύτερη λύση από αυτές σαν τελική, μετά από οτιδήποτε τροποποιήσεις χρειάζονται."

Το συμπέρασμα στο οποίο καταλήγουν, μετά από δοκιμές, είναι ότι οι αλγόριθμοι αυτοί δίνουν καλές λύσεις όταν υπάρχουν κάποιες προϋποθέσεις διαιρετότητας:

- Αν τα μεγέθη των αντικειμένων και τα μεγέθη των κάδων μπορούν να διαιρεθούν, τότε οι αλγόριθμοι δίνουν βέλτιστες λύσεις.
- Αν μόνο τα μεγέθη των κάδων μπορούν να διαιρεθούν, οι αλγόριθμοι έχουν παρόμοια απόδοση με τους FFD και BFD.

2.4 Generalized Bin Packing Problem (GBPP)

Μια άλλη προσέγγιση για την επίλυση των διαφόρων προβλημάτων Bin Packing είναι η δημιουργία ενός νέου, γενικευμένου προβλήματος. Αυτό κάνουν οι Mauro Maria Baldi κ.α. στην εισήγηση με τίτλο "The Generalized Bin Packing Problem" (Baldi, et al. 2010). Το νέο αυτό πρόβλημα γενικεύει έναν μεγάλο αριθμό από προβλήματα όπως το Knapsack Problem¹, το Bin Packing Problem, το Variable Size Bin Packing Problem και άλλα. Το συγκεκριμένο πρόβλημα το ορίζουν ως εξής:

"Το πρόβλημα περιλαμβάνει ένα σύνολο από αντικείμενα, τα οποία χαρακτηρίζονται από το μέγεθος και την αξία τους, και από ένα σύνολο κάδων διαφόρων τύπων που χαρακτηρίζονται από το μέγεθος τους και το κόστος τους. Κάποια από τα αντικείμενα πρέπει υποχρεωτικά να φορτωθούν στους κάδους, ενώ κάποια άλλα όχι. Στόχος είναι να βρεθεί ποιά από τα μη-υποχρεωτικά αντικείμενα πρέπει να επιλεγθούν και σε ποιούς κάδους πρέπει να φορτωθούν έτσι ώστε να ελαχιστοποιηθεί το συνολικό κόστος, το οποίο είναι η διαφορά ανάμεσα στο συνολικό κόστος των χρησιμοποιούμενων κάδων μείον τη συνολική αξία των αντικειμένων που φορτώθηκαν."

Για την επίλυση του προτείνουν ευριστικές μεθόδους τοποθέτησης που προέρχονται από γνώστες μεθόδους όπως οι First Fit Decreasing (FFD) και Best Fit Decreasing (BFD). Πιο συγκεκριμένα προτείνουν 3 αλγόριθμους οι οποίοι θα αναφερθούν εδώ μόνο ονομαστικά διότι δεν σχετίζονται με το πρόβλημα που έχουμε να επιλύσουμε:

1. The Profitable Heuristic for New Bin Selection
2. The BFD and FFD Upper Bound Heuristics
3. The Diving Heuristic

¹ Το Knapsack Problem έχει ως εξής: Υπάρχει ένας σάκος με περιορισμένη χωρητικότητα στον οποίο πρέπει να μπουν διάφορα αντικείμενα με διαφορετικό μέγεθος και αξία το καθένα. Πώς πρέπει να τοποθετηθούν τα αντικείμενα έτσι ώστε όταν ο σάκος γεμίσει να περιέχει τη μέγιστη δυνατή αξία;

2.5 Συμπέρασμα

Αν και το Σύστημα Κληρώσεων μπορεί κάτω από προϋποθέσεις να προσεγγιστεί ως πρόβλημα Bin Packing, στην πραγματικότητα δεν αποτελεί κάτι τέτοιο. Στο Σύστημα Κληρώσεων τα εργαστηριακά τμήματα στα οποία θα εγγραφούν οι σπουδαστές εξαρτώνται από τις προτιμήσεις τους και έτσι είναι έως ένα βαθμό προκαθορισμένα. Αντίθετα, στα προβλήματα Bin Packing υπάρχει δυνατότητα επιλογής ανάμεσα στους κάδους στους οποίους μπορεί να τοποθετηθεί ένα αντικείμενο, οπότε στόχος είναι να βρεθεί ο βέλτιστος.

Όπως έχει αναφερθεί κάποιοι από τους αλγορίθμους επίλυσης του VSBPP και του GBPP θα μπορούσαν να εφαρμοστούν στο Σύστημα Κληρώσεων, χρησιμοποιώντας μεταβλητά κόστη για τους κάδους ανάλογα με το αντικείμενο που θα τοποθετηθεί κάθε φορά (σπουδαστής), ώστε να συγκλίνουν με τις προτιμήσεις. Μια τέτοια υλοποίηση όμως θα αναιρούσε όλα τα πλεονεκτήματα που θα παρείχαν αυτοί οι αλγόριθμοι αφού θα τους χρησιμοποιούσε ουσιαστικά για στατική τοποθέτηση αφαιρώντας κάθε δυνατότητα βελτιστοποίησης².

Αυτό μας οδηγεί να ερευνήσουμε μια διαφορετική προσέγγιση με τη βοήθεια της Τεχνητής Νοημοσύνης και πιο συγκεκριμένα με τα Συστήματα Γνώσης.

² Θα μπορούσαν ωστόσο να χρησιμοποιηθούν συμπληρωματικά για τις μη τελικώς ικανοποιημένες δηλώσεις, δηλαδή σπουδαστές που βάση των προτιμήσεων τους δεν μπόρεσαν να τοποθετηθούν σε κανένα εργαστηριακό τμήμα.

3 Συστήματα Γνώσης και Έμπειρα Συστήματα

3.1 Συστήματα Βασισμένα στη Γνώση

Ο όρος Σύστημα Γνώσης ή Σύστημα Βασισμένο στη Γνώση (Knowledge-Based System) δηλώνει ένα σύστημα λογισμικού που επιδεικνύει ευφυή συμπεριφορά σε μια συγκεκριμένη λειτουργία ή πρόβλημα, αναπαριστώντας και χρησιμοποιώντας με τυπικό τρόπο τη γνώση.

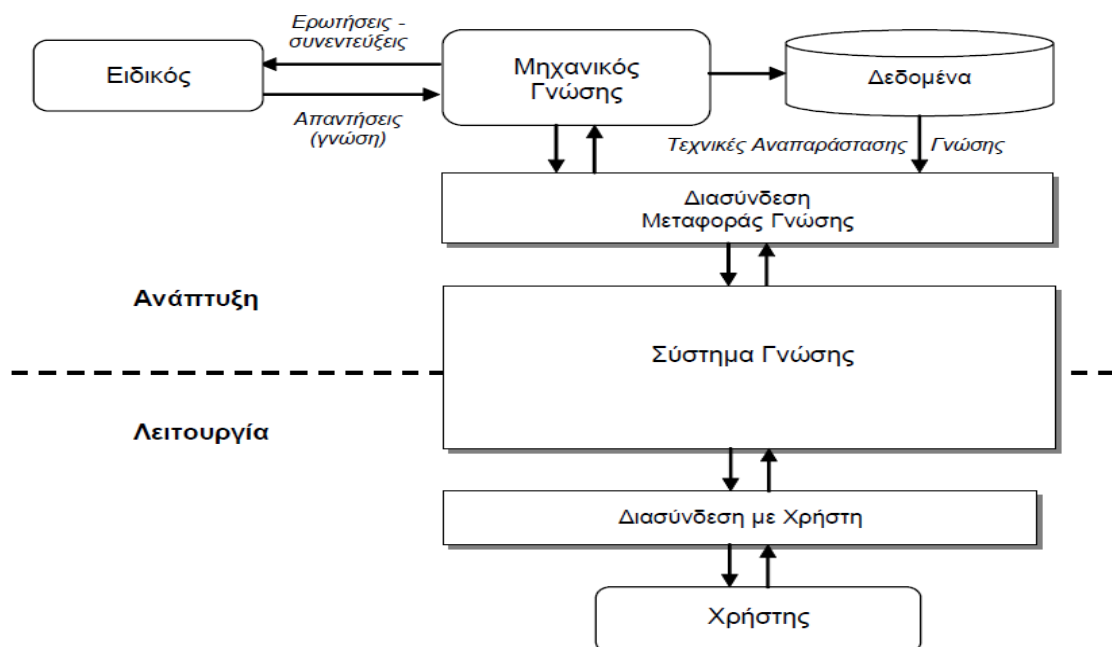
Η γνώση αυτή μπορεί να αποκτάται μέσω εμπειρίας ή μελέτης και συμπεριλαμβάνει όλες τις πληροφορίες, τις εμπειρίες, τις ικανότητες και τις δεξιότητες που κατέχει ένας άνθρωπος. Συνεπώς ένας πιο συγκεκριμένος ορισμός για τα Συστήματα Γνώσης είναι: *"Ένα πρόγραμμα υπολογιστή, το οποίο κωδικοποιεί και χειρίζεται τη γνώση και τη συλλογιστική που υπάρχει για έναν εξειδικευμένο τομέα με σκοπό την επίλυση προβλημάτων ή/και την παροχή συμβουλών στον εν λόγω τομέα"* (Βλαχάβας, et al. 2006).

Διαφοροποιούνται από τις συμβατικές εφαρμογές τόσο στη δομή όσο και στη λειτουργία τους. Το βασικότερο σημείο διαφοροποίησης είναι ότι τα συστήματα γνώσης κάνουν συλλογισμούς χρησιμοποιώντας συμβολικές αναπαράστασης της ανθρώπινης γνώσης, επιπλέον της εκτέλεσης αριθμητικών υπολογισμών και της ανάκτησης δεδομένων. Η γνώση σε ένα πρόγραμμα περιγράφεται με μια μορφή αναπαράστασης και αποθηκεύεται ξεχωριστά από τον κώδικα που εκτελεί το συλλογισμό. Αυτά τα δύο χωριστά τμήματα του συστήματος γνώσης ονομάζονται βάση γνώσης και μηχανισμός εξαγωγής συμπερασμάτων (Βλαχάβας, et al. 2006).

Τα Συστήματα Γνώσης θεωρούνται από τις πιο επιτυχημένες εφαρμογές της Τεχνητής Νοημοσύνης και χρησιμοποιούνται με ποικίλους τρόπους όπως παρακολούθηση καταστάσεων (monitoring), διάγνωση βλαβών, πρόβλεψη μελλοντικών επιπτώσεων και άλλους. Παρέχουν διάφορα πλεονεκτήματα σε σχέση με τις συμβατικές εφαρμογές όπως γρηγορότερη ανάπτυξη και ευκολότερη συντήρηση λόγω του ότι τα Συστήματα Γνώσης βρίσκονται πιο κοντά στην ανθρώπινη σκέψη και επιπλέον λόγω του ότι η αναθεώρηση της υπάρχουσας γνώσης συνήθως μπορεί να πραγματοποιηθεί χωρίς μεταβολές ευρείας κλίμακας. (Stefik 1995)

Ένα σύστημα γνώσης μπορεί να χρησιμοποιηθεί με δύο τρόπους. Ο πρώτος είναι να χρησιμοποιείται από κάποιον άνθρωπο μη-ειδικό, οπότε θα παρέχει λύσεις σε συγκεκριμένα προβλήματα που τίθενται, αντικαθιστώντας τον άνθρωπο-ειδικό και εξομοιώνοντας την ικανότητα λήψης αποφάσεων εκ μέρους του. Ο δεύτερος τρόπος χρήσης του συστήματος γνώσης είναι συμβουλευτικός: ένας άνθρωπος-ειδικός, ο οποίος καλείται να πάρει κάποια απόφαση, συμβουλευείται το σύστημα γνώσης χρησιμοποιώντας το ως βοηθό του και με τον τρόπο αυτό παίρνει καλύτερες αποφάσεις, βελτιώνοντας την παραγωγικότητα του (Βλαχάβας, et al. 2006).

Για την ανάπτυξη ενός συστήματος γνώσης πρέπει να συνεργαστούν ένας άνθρωπος-ειδικός του τομέα (domain expert) και ένας μηχανικός γνώσης (knowledge engineer). Ο ειδικός του τομέα είναι κάποιος άνθρωπος εξειδικευμένος σε έναν τομέα της ανθρώπινης δραστηριότητας, ο οποίος θα βοηθήσει στη μεταφορά της γνώσης στο σύστημα. Η γνώση που θα μεταφερθεί μπορεί να αποτελεί είτε δική του εμπειρία, είτε κοινή επιστημονική ή τεχνολογική γνώση, είτε γνώση καταγεγραμμένη σε βάσεις δεδομένων ή σε έγγραφα. Ο μηχανικός γνώσης είναι ένας επιστήμονας της πληροφορικής ειδικευμένος, ειδικευμένος σε θέματα Τεχνητής Νοημοσύνης και Συστημάτων Γνώσης. Ο μηχανικός γνώσης συνεργάζεται με τον ειδικό του τομέα με σκοπό είτε τη λήψη της γνώσης του, είτε την παροχή βοήθειας και καθοδήγησης σχετικά με τη λήψη γνώσης που προέρχεται από άλλες πηγές, όπως προαναφέρθηκε. Με βάση τα αποτελέσματα της συνεργασίας αυτής σχεδιάζει το σύστημα και τη δομή της γνώσης στη συνέχεια το αναπτύσσει (Βλαχάβας, et al. 2006).



Σχήμα 4.1 Ανάπτυξη και Λειτουργία Συστήματος Γνώσης (Βλαχάβας, et al. 2006)

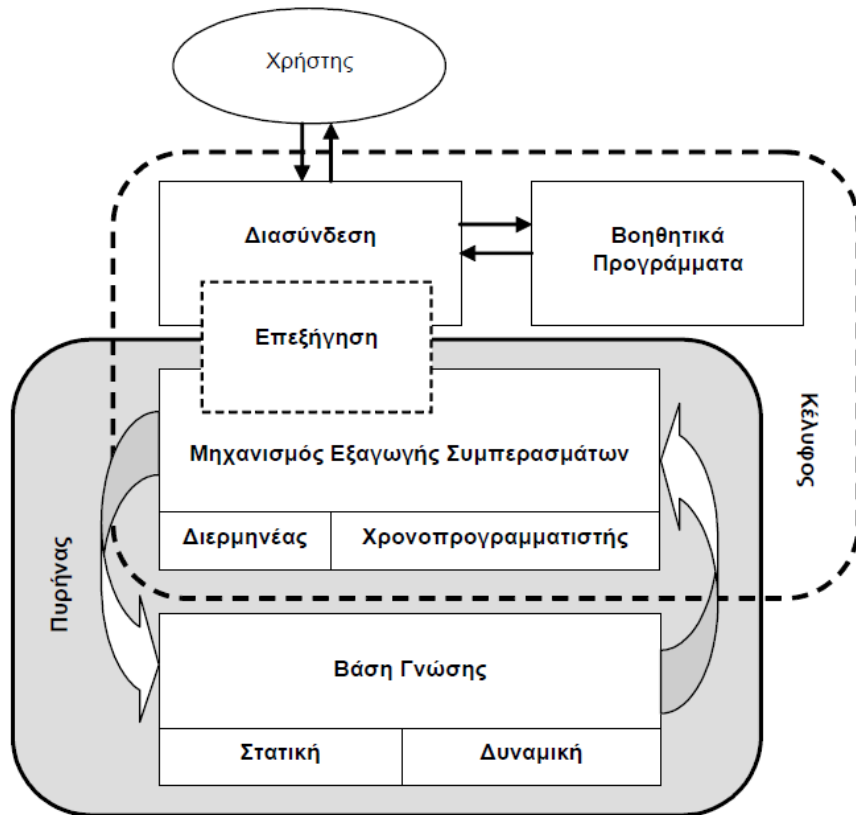
3.2 Αρχιτεκτονική Συστημάτων Γνώσης

Σύμφωνα με τους (Βλαχάβας, et al. 2006), ένα σύστημα γνώσης συνήθως αποτελείται από μία ομάδα προγραμμάτων που μπορεί να χωρισθούν σε δύο κατηγορίες: τον πυρήνα του συστήματος και ένα σύνολο βοηθητικών προγραμμάτων.

Στο βιβλίο αναφέρεται ότι ο πυρήνας ενός συστήματος γνώσης αποτελείται από δύο μέρη: τη βάση γνώσης και μηχανισμό εξαγωγής συμπερασμάτων. Αυτό γίνεται για να επιτευχθεί διαχωρισμός ανάμεσα στη γνώση του συστήματος και στον μηχανισμό χειρισμού της. Αυτός ο διαχωρισμός είναι χρήσιμος, γιατί δίνει τη δυνατότητα εύκολης προσθαφίρεσης και τροποποίησης της γνώσης, χωρίς να απαιτείται πρόσβαση στο μηχανισμό εξαγωγής συμπερασμάτων. Η ανάπτυξη του πυρήνα ενός συστήματος γνώσης συνήθως γίνεται σε κάποιο προγραμματιστικό περιβάλλον υψηλού επιπέδου, το οποίο μπορεί να είναι είτε μια γλώσσα προγραμματισμού, είτε κάποιο εξειδικευμένο εργαλείο για την ανάπτυξη συστημάτων γνώσης (Βλαχάβας, et al. 2006).

Όπως αναφέρθηκε παραπάνω, εκτός από τον πυρήνα ένα σύστημα γνώσης συνοδεύεται και από ένα σύνολο βοηθητικών προγραμμάτων. Τα προγράμματα αυτά χρησιμοποιούνται συνήθως για τη διασύνδεση και επικοινωνία του συστήματος με τον χρήστη, για το χειρισμό εξωτερικών βάσεων δεδομένων, για τη συνεργασία με περιφερειακά μηχανήματα κτλ. Αυτά τα βοηθητικά προγράμματα σε συνδυασμό με το μηχανισμό εξαγωγής συμπερασμάτων αποτελούν ένα εργαλείο ανάπτυξης συστημάτων γνώσης, το οποίο πολλές αναφέρεται ως κέλυφος γιατί προέρχεται από την αφαίρεση της βάσης γνώσης από ένα σύστημα γνώσης. Η συνηθέστερη έκφραση είναι κέλυφος εμπείρων συστημάτων (expert system shell), γιατί τα πρώτα κελύφη προήλθαν από τα έμπειρα συστήματα (Βλαχάβας, et al. 2006).

Στην επόμενη σελίδα ακολουθεί μια σχηματική απεικόνιση της δομής ενός συστήματος γνώσης από το ίδιο βιβλίο και στη συνέχεια περιγράφονται πιο αναλυτικά κάποια από τα στοιχεία του πυρήνα.



Σχήμα 4.2 Δομή Συστήματος Γνώσης (Βλαχάβας, et al. 2006)

Βάση Γνώσης

Η βάση γνώσης αποτελεί το μέρος εκείνο του συστήματος όπου περιέχεται όλη η γνώση, όπως την κατέγραψε ο μηχανικός γνώσης με τη βοήθεια του αντίστοιχου ανθρώπου-ειδικού, κατά τη διαδικασία ανάπτυξης του συστήματος γνώσης. Η βάση γνώσης αποτελείται από δύο μέρη (Βλαχάβας, et al. 2006):

- Στατική βάση γνώσης: Αυτό το μέρος περιλαμβάνει την περιγραφή του προβλήματος υπό τη μορφή διαδικασιών, κανόνων και πλαισίων, καθώς και τις γνωσιολογικές διαδικασίες επίλυσης του. Αυτό το τμήμα της βάσης γνώσης δεν μεταβάλλεται κατά τη διάρκεια εκτέλεσης του προγράμματος και από εκεί προέρχεται το όνομα "στατική" (Βλαχάβας, et al. 2006).
- Δυναμική βάση γνώσης: Αυτό το μέρος περιέχει τα δεδομένα και τα πλαίσια που περιγράφουν τη λύση του προβλήματος, δηλαδή τα μερικά συμπεράσματα που δημιουργούνται κατά την εκτέλεση του συστήματος, καθώς και την τελική λύση. Αυτό το τμήμα ονομάζεται και μνήμη/χώρος εργασίας (working memory) (Βλαχάβας, et al. 2006).

Μηχανισμός εξαγωγής συμπερασμάτων

Σύμφωνα με το βιβλίο των (Βλαχάβας, et al. 2006) " Ο μηχανισμός εξαγωγής συμπερασμάτων (*inference engine*) είναι το τμήμα του πυρήνα που είναι υπεύθυνο για το χειρισμό της βάσης γνώσης και την εξαγωγή συμπερασμάτων από αυτήν. Η δομή του μηχανισμού εξαγωγής συμπερασμάτων εξαρτάται από την οργάνωση και τον τρόπο αναπαράστασης της γνώσης. Χωρίζεται σε δύο μέρη, το διερμηνέα (*interpreter*) και το χρονοπρογραμματιστή (*scheduler*)."

Ο ρόλος του διερμηνέα είναι ο χειρισμός της υπάρχουσας γνώσης και η παραγωγή νέας. Η εξαγωγή συμπερασμάτων μπορεί να βασιστεί σε διάφορες συλλογιστικές, καθώς και τεχνικές χειρισμού ασάφειας ή αβεβαιότητας. Ο διερμηνέας συνήθως αποτελείται από αλγορίθμους που υλοποιούν την αντίστοιχη συλλογιστική. Για παράδειγμα όταν η συλλογιστική που χρησιμοποιείται είναι συνεπαγωγική και η αναπαράσταση γνώσης γίνεται με τη μορφή κανόνων, τότε αυτή μπορεί να υλοποιηθεί είτε με ορθή, είτε με ανάστροφη ακολουθία εκτέλεσης (*forward* και *backward chaining* αντίστοιχα) (Βλαχάβας, et al. 2006).

Ιδιαίτερο ενδιαφέρον έχει ο χρονοπρογραμματιστής, ο οποίος αποφασίζει πότε και με ποιά σειρά θα χρησιμοποιηθούν τα διάφορα στοιχεία της βάσης γνώσης, επιλύοντας τυχόν προβλήματα συγκρούσεων (*conflict*) μεταξύ των κανόνων. Σύγκρουση έχουμε όταν υπάρχουν δύο ή περισσότεροι κανόνες των οποίων οι συνθήκες ικανοποιούνται ταυτόχρονα. Στην περίπτωση αυτή επιλέγεται, με βάση κάποια κριτήρια³, ένας από τους υποψήφιους κανόνες και να εκτελεσθεί (Βλαχάβας, και συν. 2006).

Το τελευταίο έχει ιδιαίτερο ενδιαφέρον, γιατί παρέχει έναν μεγάλο βαθμό ελέγχου όσον αφορά την τον τρόπο και τη σειρά εκτέλεσης του συστήματος. Αυτό αποτελεί σημαντικό πλεονέκτημα στα πλαίσια της συγκεκριμένης πτυχιακής, γιατί μεταφράζεται σε ευκολία ορισμού και παραμετροποίησης της σειράς (δηλαδή των προτεραιοτήτων) εγγραφής, κάτι ιδιαίτερα σημαντικό σε ένα σύστημα αυτόματης τοποθέτησης.

Τέλος να σημειωθεί ότι στις επόμενες ενότητες τα Συστήματα Γνώσης μπορεί να αναφερθούν και ως Έμπειρα Συστήματα. Τα Έμπειρα Συστήματα αποτελούν ένα υποσύνολο των συστημάτων γνώσης, αν και πολλές φορές οι λέξεις χρησιμοποιούνται ως συνώνυμες. Η διαφοροποίηση βρίσκεται στην πηγή από όπου προέρχεται η γνώση του κάθε συστήματος. Πιο συγκεκριμένα, στα Έμπειρα Συστήματα η γνώση πρέπει να προέρχεται αποκλειστικά από έναν ή περισσότερους ανθρώπους (εμπειρογνώμονες), ενώ στα Συστήματα Γνώσης μπορεί να προέρχεται και από άλλες πηγές όπως βιβλία,

³ Σε πολλά κελύφη ανάπτυξης εμπειρών συστημάτων αυτά τα κριτήρια ονομάζονται στρατηγική.

εξόρυξη πληροφορίας από το Διαδίκτυο κτλ. (Feigenbaum, et al. 1993). Έτσι, επειδή στη συγκεκριμένη πτυχιακή η γνώση προέρχεται αποκλειστικά από ανθρώπους-εμπειρογνώμονες, ο όρος Έμπειρο Σύστημα είναι πιο ακριβής.

Στις επόμενες σελίδες θα γίνει μια εισαγωγή σε κάποια δημοφιλή εργαλεία ανάπτυξης Εμπείρων Συστημάτων. Αυτά τα εργαλεία, σε συνδυασμό με κάποιες παραδοσιακές γλώσσες προγραμματισμού, μπορούν να χρησιμοποιηθούν για την ανάπτυξη ενός Εμπείρου Συστήματος που να αναλάβει τη διαδικασία της τοποθέτησης των σπουδαστών στα εργαστηριακά τμήματα.

3.3 Η γλώσσα CLIPS

Το CLIPS (C Language Integrated Production System) είναι ένα δημοφιλές εργαλείο ανάπτυξης Εμπείρων Συστημάτων που αρχικά αναπτύχθηκε στο διαστημικό κέντρο NASA-Johnson το 1985. Ιστορικά, η σύνταξη του CLIPS είναι παρόμοια με την γλώσσα OPS5⁴, από την οποία είναι εμπνευσμένη, αλλά πιο ισχυρή, ενώ η λειτουργία του βασίζεται στον αλγόριθμο Rete⁵ κάτι που το κάνει ιδιαίτερα αποτελεσματικό σε ένα μεγάλο εύρος εφαρμογών. Το CLIPS έχει γραφτεί στη γλώσσα προγραμματισμού C και διατίθεται δωρεάν από την ιστοσελίδα <http://clipsrules.sourceforge.net/> τόσο το πρόγραμμα όσο και ο πηγαίος του κώδικας.

Για την ανάπτυξη Εμπείρων Συστημάτων το CLIPS χρησιμοποιεί μια γλώσσα η οποία είναι συντακτικά παρόμοια με την LISP, ενώ επιτρέπει και αντικειμενοστραφή ανάπτυξη μέσω της δικής του γλώσσας COOL (CLIPS Object Oriented Language).

Αντίστοιχα με άλλα Συστήματα Γνώσης, ένα σύστημα που αναπτύσσεται σε CLIPS αποτελείται από τρία βασικά συστατικά:

Γεγονότα (facts) - Είναι τα δεδομένα πάνω στα οποία θα λειτουργήσει ο μηχανισμός εξαγωγής συμπερασμάτων.

Βάση Γνώσης - Περιλαμβάνει όλους τους κανόνες που συνθέτουν τη γνώση/"εμπειρία" του συστήματος.

Μηχανισμός Εξαγωγής Συμπερασμάτων - Ελέγχει τη ροή της εκτέλεσης των κανόνων.

Παρακάτω αναφέρονται κάποιες από τις βασικές δομές της CLIPS, σύμφωνα με την επίσημη τεκμηρίωση (Riley n.d.). Πρέπει να αναφερθεί ότι η σύνταξη των παρακάτω δομών αποτελεί την πιο απλή έκδοσή τους και δεν δίνει πλήρη εικόνα όλων των δυνατοτήτων της κάθε μίας.

4 Η γλώσσα OPS5 (Official Production Language) είναι μια γλώσσα ανάπτυξης συστημάτων παραγωγής βασισμένη σε κανόνες που δημιουργήθηκε από τον Charles L. Forgy. Είναι γνωστή ως η πρώτη τέτοια γλώσσα που χρησιμοποιήθηκε σε ένα επιτυχημένο Έμπειρο Σύστημα. (Forgy 1982)

5 Ο αλγόριθμος Rete είναι ένας αλγόριθμος που αναπτύχθηκε από τον Charles L. Forgy για την υλοποίηση συστημάτων κανόνων. Σύμφωνα με τον δημιουργό του (Forgy 1982), ο Rete είναι μια αποτελεσματική μέθοδος ταιριάσματος (matching) ενός μεγάλου συνόλου από patterns με έναν μεγάλο σύνολο από αντικείμενα (objects). Ο αλγόριθμος βρίσκει όλα τα αντικείμενα που ταιριάζουν σε κάθε pattern. Ο αλγόριθμος αναπτύχθηκε για χρήση σε συστήματα παραγωγής, και έχει χρησιμοποιηθεί σε συστήματα που περιέχουν από λίγες εκατοντάδες μέχρι και πάνω από χίλια patterns και αντικείμενα.

(def facts όνομα (fact1) ... (factN)) - Αυτή η δομή δίνει τη δυνατότητα στον χρήστη να ορίζει στατικά γεγονότα. Τα γεγονότα αυτά μπορούν να ομαδοποιηθούν κάτω από ένα όνομα, αλλά αυτό χρησιμεύει μόνο για τη διευκόλυνση του μηχανικού γνώσης και δεν επιδρά στον τρόπο εκτέλεσης.

(defrule όνομα (pattern1) ... (patternN) => (action1) ... (actionN)) - Με αυτή τη δομή μπορούμε να εισάγουμε κανόνες στη βάση γνώσης. Οι κανόνες αποτελούνται από δύο μέρη, που χωρίζονται με το σύμβολο => . Στο αριστερό μέρος (Left-Hand Side - LHS, αυτό που βρίσκεται πριν από το =>) βρίσκονται οι συνθήκες που πρέπει να ισχύουν, δηλαδή τα γεγονότα που πρέπει να υπάρχουν (patterns), για να πυροδοτηθεί ο κανόνας. Στο δεξί μέρος (Right-Hand Side - RHS) βρίσκονται οι δράσεις που θα εκτελεστούν όταν πυροδοτηθεί ο κανόνας. Αυτές μπορεί να περιλαμβάνουν μεταβολές γεγονότων, κλήση συναρτήσεων, τερματισμό της εκτέλεσης του συστήματος και άλλα. Όταν ενεργοποιούνται ταυτόχρονα περισσότεροι από ένας κανόνες τότε ο κανόνας που θα πυροδοτηθεί εξαρτάται από την στρατηγική που έχει οριστεί. Για τον καλύτερο έλεγχο της ροής εκτέλεσης υπάρχει και ο μηχανισμός του salience, που δίνει τη δυνατότητα να ανατεθεί μια αριθμητική "προτεραιότητα" σε κάθε κανόνα. Σε περίπτωση σύγκρουσης οι κανόνες με μεγαλύτερο salience πυροδοτούνται πρώτοι ενώ αυτοί με χαμηλότερο τελευταίοι.

(def template όνομα (slot1 όνομα-slot1) ... (slotN όνομα-slotN)) - Τα templates επιτρέπουν την καλύτερη δόμηση των facts με τρόπο αντίστοιχο των συμβατικών γλωσσών προγραμματισμού (πχ. struct στην C).

(def module όνομα) - Τα modules είναι ένας επιπλέον μηχανισμός ελέγχου της ροής εκτέλεσης. Ένα module είναι μια δομή που μπορεί να περιέχει γεγονότα ή κανόνες, τα οποία παραμένουν αδρανή και δεν εκτελούνται μέχρι η ροή εκτέλεσης να μεταβεί στο συγκεκριμένο module μέσω κάποιου άλλου (αυτό γίνεται με τη συνάρτηση focus, που θα αναλυθεί με μεγαλύτερη λεπτομέρεια σε μεταγενέστερο κεφάλαιο). Όταν αυτό συμβεί οι κανόνες που περιλαμβάνει το module ενεργοποιούνται και πυροδοτούνται. Αυτό συνεχίζεται έως ότου σταματήσουν να υπάρχουν κανόνες που να μπορούν να πυροδοτηθούν. Τότε η ροή εκτέλεσης (που ονομάζεται και έλεγχος) επιστρέφει στο module από το οποίο μετέβη. Τα modules δηλαδή λειτουργούν σαν στοίβα όπου ενεργοποιημένο είναι εκείνο που βρίσκεται στην κορυφή της. Η συνάρτηση focus προσθέτει ένα νέο module στην κορυφή της στοίβας (push) ενώ όταν δεν υπάρχουν κανόνες που μπορούν να πυροδοτηθούν αυτό φεύγει (pop).

3.4 Εισαγωγή στο JESS

Το JESS (Java Expert System Shell) είναι μια μηχανή κανόνων βασισμένη στο CLIPS που δημιουργήθηκε το 1995 από τον Ernest Friedman-Hill των Sandia National Laboratories. Είναι διαθέσιμο εμπορικά στην ιστοσελίδα <http://www.jessrules.com/> και επιπλέον διατίθεται δωρεάν για ακαδημαϊκή χρήση.

Το JESS είναι εξολοκλήρου γραμμένο σε Java, γεγονός που του επιτρέπει να τρέχει σε μια τεράστια ποικιλία από πλατφόρμες και λειτουργικά συστήματα. Επιπλέον, το JESS παρέχει ένα απλό και καλά τεκμηριωμένο API (Application Programming Interface) κάτι το οποίο διευκολύνει σε μεγάλο βαθμό την ενσωμάτωση και χρήση του μέσα από άλλες εφαρμογές.

Η γλώσσα του JESS αποτελεί ένα υπερσύνολο του CLIPS και η σύνταξη τους είναι τόσο παρόμοια που αρκετές φορές προγράμματα που γράφτηκαν για μια γλώσσα μπορούν να μεταφερθούν και να τρέξουν αυτούσια στην άλλη. Οι διαφορές εστιάζονται κυρίως στην ύπαρξη ή μη προχωρημένων συναρτήσεων και στρατηγικών.

Ένα σημαντικό χαρακτηριστικό του JESS είναι η δυνατότητα δημιουργίας και επεξεργασίας αντικειμένων Java μέσα από το Έμπειρο Σύστημα. Οι κλάσεις αυτών των αντικειμένων φαίνονται στο JESS σαν templates και τα αντικείμενα σαν facts. Τα αντικείμενα αυτού του είδους ονομάζονται shadow facts και μπορούν να εξαχθούν από το Έμπειρο Σύστημα και να χρησιμοποιηθούν από την Java σαν κανονικά αντικείμενα. Το JESS δίνει τη δυνατότητα κλήσης μεθόδων αυτών των αντικειμένων, με παραμέτρους που μπορεί να ορίζονται μέσα από το Έμπειρο Σύστημα, κάτι που τα κάνει ιδανικά για εξαγωγή αποτελεσμάτων, συλλογή στατιστικών στοιχείων και πολλές άλλες χρήσεις.

Το Jess έχει χρησιμοποιηθεί σε διάφορες εφαρμογές, μερικές από τις οποίες παρατίθενται παρακάτω:

<http://www.hdfrazer.com/DOW> - Έμπειρο Σύστημα που βοηθά στην αντιμετώπιση του διαβήτη. Υλοποιείται ως Java applet.

<http://scialert.net/fulltext/?doi=jai.2010.239.251&org=11> - Έμπειρο Σύστημα που βοηθάει στη διάγνωση και θεραπεία ενδοκρινικών ασθενειών. Υλοποιείται ως εφαρμογή desktop.

<http://grove.cs.jmu.edu/parih/index.jsp> - Ένα Έμπειρο Σύστημα που βοηθά στην αναγνώριση ερπετών. Υλοποιείται ως servlet που ενσωματώνει το Jess.

Πρακτικό Μέρος

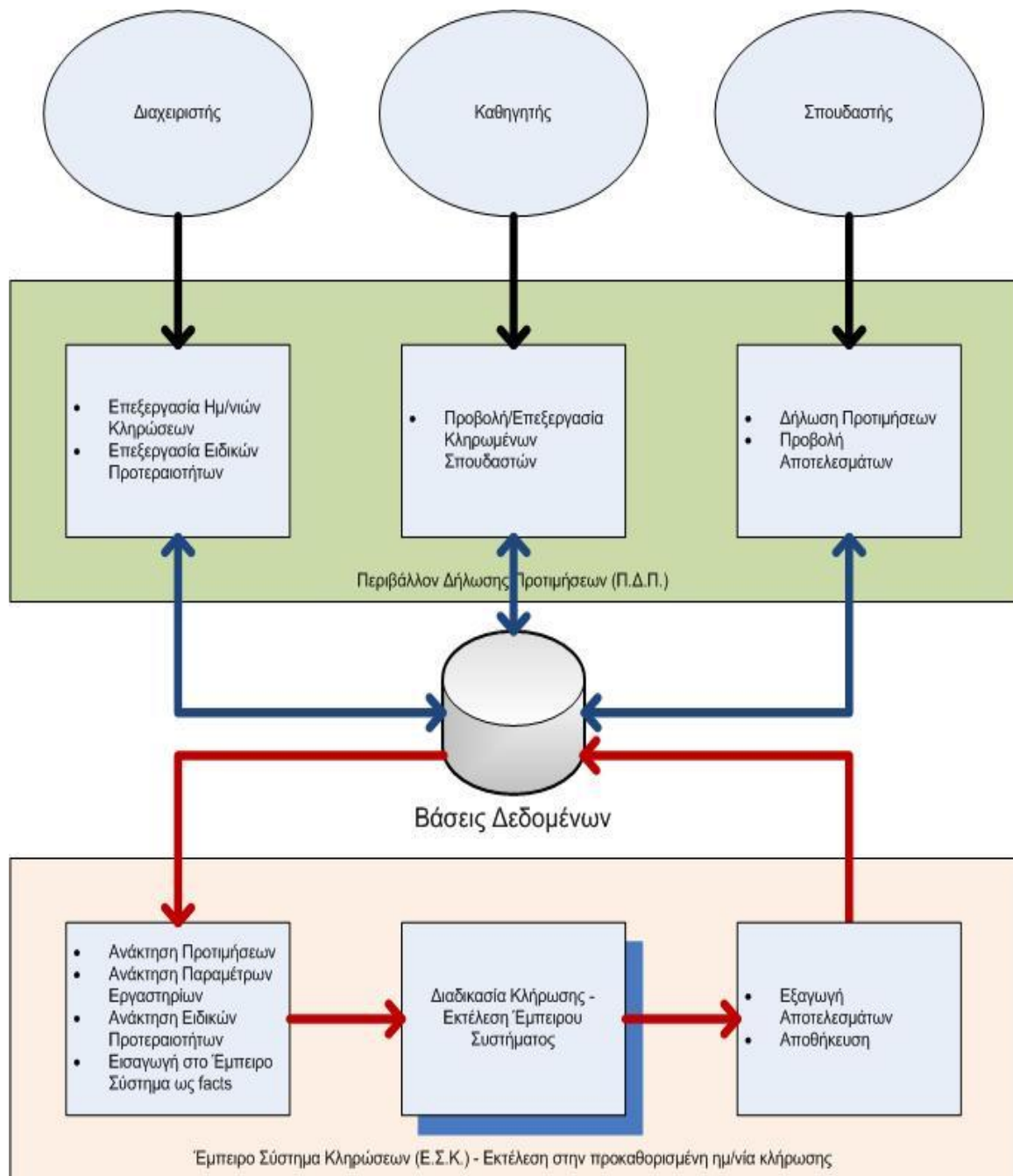
4 Ανάπτυξη Συστήματος Εργαστηριακών Εγγραφών Βάσει Προτιμήσεων (ΣΕΕΠ)

Το σύστημα που θα αναπτυχθεί αποτελείται ουσιαστικά από δύο ξεχωριστά μέρη: το Έμπειρο Σύστημα Κληρώσεων (Ε.Σ.Κ.) και το Περιβάλλον Δήλωσης Προτιμήσεων (Π.Δ.Π.). Τα δύο αυτά μέρη αποτελούν ανεξάρτητες οντότητες και θα αναπτυχθούν ξεχωριστά, χρησιμοποιώντας δημοφιλείς τεχνολογίες, αλλά ξεχωριστές για κάθε ένα.

Πιο συγκεκριμένα, το Ε.Σ.Κ. θα αναπτυχθεί σε Java με τη βοήθεια της βιβλιοθήκης του JESS, η οποία αναφέρθηκε παραπάνω. Το Έμπειρο Σύστημα θα ενσωματωθεί σε μια ευρύτερη εφαρμογή, η οποία θα αναλάβει τις βοηθητικές λειτουργίες όπως ανάκτηση δεδομένων από τις κατάλληλες βάσεις και μετατροπή τους στην αναπαράσταση που απαιτείται από το Έμπειρο Σύστημα. Επιπλέον, θα είναι υπεύθυνη για την εξαγωγή των αποτελεσμάτων και την αποθήκευσή τους.

Το Περιβάλλον Δήλωσης Προτιμήσεων (Π.Δ.Π.) θα υλοποιηθεί με την μορφή μιας web εφαρμογής χρησιμοποιώντας τεχνολογίες όπως PHP και MySQL. Στόχος του Π.Δ.Π. να παρέχει μια εύχρηστη διεπαφή προς τους σπουδαστές, εξασφαλίζοντας ταυτόχρονα την εγκυρότητα των δηλώσεων (αυθεντικοποίηση, εξασφάλιση ότι ο σπουδαστής είναι εγγεγραμμένος στο συγκεκριμένο εργαστηριακό μάθημα κτλ.) που θα εισαχθούν στο Ε.Σ.Κ. Επιπλέον, το Π.Δ.Π. χρησιμεύει και ως διεπαφή για διάφορες διαχειριστικές λειτουργίες, όπως Επεξεργασία Ημερομηνιών Κληρώσεων, Επεξεργασία Ειδικών Προτεραιοτήτων (θα αναφερθούν εκτενέστερα παρακάτω) και άλλες.

Στην επόμενη σελίδα ακολουθεί ένα διάγραμμα με τα δομικά στοιχεία του ΣΕΕΠ καθώς και τις σημαντικότερες λειτουργίες που περιλαμβάνει:



Σχήμα 2.1 Η δομή του ΣΕΕΠ

5 Ανάπτυξη Εμπείρου Συστήματος Κληρώσεων

5.1 Περιγραφή Λειτουργίας

Σύμφωνα με αυτά που είδαμε στην Εισαγωγή, σαν κλήρωση ορίζεται η διαδικασία της τοποθέτησης των σπουδαστών στα εργαστηριακά τμήματα λαμβάνοντας υπόψη τις προτιμήσεις τους και διάφορες άλλες πιθανώς ορισμένες παραμέτρους.

Έτσι, μπορούμε να συμπεράνουμε ότι στη διάρκεια μιας κλήρωσης μπορούν να υπάρξουν δύο πιθανά αποτελέσματα για κάθε προτίμηση που εισάγεται προς επεξεργασία: Επιτυχής εγγραφή, Ανεπιτυχής εγγραφή λόγω έλλειψης χώρου. Επιπλέον όταν μια εγγραφή σε ένα τμήμα είναι επιτυχής, πρέπει να διαγράφονται οποιεσδήποτε άλλες προτιμήσεις αυτού του σπουδαστή που αφορούν Εργαστηριακά Τμήματα άλλων εργαστηριακών μαθημάτων, τα οποία διδάσκονται την ίδια ημέρα και ώρα με εκείνο στο οποίο γράφτηκε. Έτσι εξασφαλίζεται ότι ένας σπουδαστής θα εγγραφεί σε ένα μόνο εργαστηριακό τμήμα για κάθε εργαστηριακό μάθημα, βελτιώνοντας επιπλέον και την ταχύτητα εκτέλεσης του συστήματος. Η σειρά εγγραφής ενός σπουδαστή στα διαφορετικά εργαστηριακά μαθήματα εξαρτάται από τη σειρά με την οποία αυτά ανακτήθηκαν, από την αντίστοιχη βάση δεδομένων.

Για την υλοποίηση των παραπάνω το Έμπειρο Σύστημα Κληρώσεων (Ε.Σ.Κ.) που θα δημιουργηθεί θα πρέπει να υποστηρίζει καταρχάς δύο βασικές διαδικασίες: τη διαδικασία εγγραφής και τη διαδικασία ολίσθησης.

- Η διαδικασία εγγραφής εντοπίζει για κάθε εργαστηριακό μάθημα την 1η προτίμηση που έχει θέσει κάθε σπουδαστής και επιχειρεί να τον εγγράψει σε εκείνο το Εργαστηριακό Τμήμα. Αν η εγγραφή γίνει επιτυχώς τότε διαγράφονται οι υπόλοιπες προτιμήσεις του σπουδαστή για το συγκεκριμένο Εργαστηριακό Μάθημα και γίνεται έλεγχος για τυχόν συγκρούσεις ωρών με δηλώσεις του σπουδαστή σε τμήματα άλλων μαθημάτων. Σε περίπτωση τέτοιων συγκρούσεων, οι δηλώσεις εκείνες διαγράφονται.
- Η διαδικασία της ολίσθησης ενεργοποιείται όταν μια εγγραφή δεν μπορεί να ολοκληρωθεί, ή όταν μια προτίμηση διαγράφεται. Στη διάρκεια της φάσης ολίσθησης όλες οι προτιμήσεις του σπουδαστή για το συγκεκριμένο εργαστηριακό μάθημα μειώνονται κατά ένα (δηλαδή η 2^η προτίμηση γίνεται 1^η, η 3^η γίνεται 2^η κτλ.).

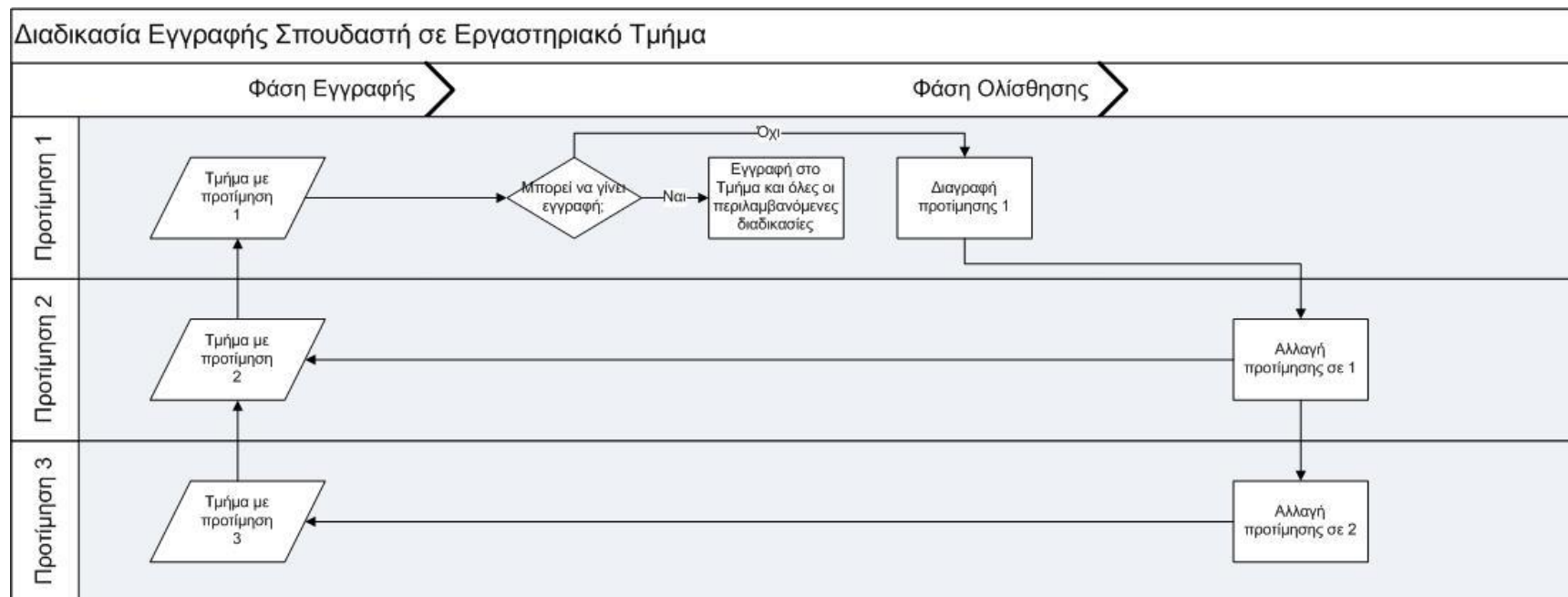
Έτσι το Ε.Σ.Κ. τελικά λειτουργεί σε δύο φάσεις, οι οποίες υιοθετούν το όνομα των διαδικασιών που υλοποιούν. Στην συνέχεια ακολουθεί μια συνοπτική αποτύπωση αυτών των φάσεων και των περιορισμών τους, καθώς και ένα διάγραμμα με την τυπική λειτουργία του συστήματος.

Φάση	Εγγραφή
Ενέργεια	Εγγράφει έναν σπουδαστή x σε ένα Εργαστηριακό Τμήμα y , το οποίο ανήκει στο Μάθημα z .
Περιορισμοί	<ul style="list-style-type: none"> - Ο σπουδαστής πρέπει να είναι εγγεγραμμένος στο Μάθημα z. - Ο σπουδαστής δεν πρέπει να είναι εγγεγραμμένος σε άλλο Εργαστηριακό Τμήμα που να διδάσκεται την ίδια ώρα με το Εργαστηριακό Τμήμα y. - Το Εργαστηριακό Τμήμα y πρέπει να είναι 1ή προτίμηση του σπουδαστή. - Το Εργαστηριακό Τμήμα y δεν πρέπει να έχει γεμίσει.

Φάση	Ολίσθησης
Ενέργεια	Ολισθαίνει τις προτιμήσεις του σπουδαστή x κατά 1 σε όλα τα τμήματα ενός μαθήματος z . Η προτίμηση που ήταν 1ή πριν την ολίσθηση διαγράφεται.
Περιορισμοί	- Πρέπει να μην μπορεί να γίνει εγγραφή στην 1ή προτίμηση του σπουδαστή λόγω περιορισμών.

Η σειρά με την οποία εγγράφονται οι σπουδαστές κατά τη Φάση Εγγραφής εξαρτάται από τη στρατηγική που έχει επιλεγεί για το Έμπειρο Σύστημα. Η προεπιλεγμένη στρατηγική είναι η *random*, δηλαδή σε περίπτωση που δεν χωρούν όλοι οι σπουδαστές, η επιλογή γίνεται τυχαία. Δίνεται ωστόσο η δυνατότητα στον διαχειριστή να διαμορφώσει τη στρατηγική, μέσω του μηχανισμού των Προτεραιοτήτων Εγγραφής ή αλλιώς Ειδικών Προτεραιοτήτων. Ο συγκεκριμένος μηχανισμός θα εξηγηθεί με μεγαλύτερη λεπτομέρεια σε μεταγενέστερη ενότητα.

Πρέπει να σημειωθεί ότι η διαδικασία κληρώσεων δεν αφορά την κατηγορία των σπουδαστών που ονομάζονται "Παλαιοί". Αυτή είναι μια κατηγορία σπουδαστών που δεν θα πρέπει να δηλώνουν προτιμήσεις για όσα μαθήματα θεωρούνται "Παλαιοί", αλλά να επιλέγουν να εγγραφούν ως τέτοιοι μέσω του Περιβάλλοντος Δήλωσης Προτιμήσεων (αυτή η κατηγορία σπουδαστών θα αναλυθεί λεπτομερέστερα σε υπο-ενότητα του Π.Δ.Π.). Είναι όμως σημαντικό να σημειωθεί, ότι στο Ε.Σ.Κ. δεν υπάρχει σχετικός έλεγχος για επιβολή αυτού του περιορισμού.



Σχήμα 5.1 Διαδικασία Εγγραφής Σπουδαστή σε Εργαστηριακό Τμήμα

Για λόγους οικονομίας χώρου η λέξη Εργαστηριακό Τμήμα έχει γραφεί ως Τμήμα. Στις «περιλαμβανόμενες διαδικασίες» που ακολουθούν την εγγραφή περιλαμβάνεται και η διαγραφή των συγκρουόμενων προτιμήσεων. Η άλλη διαδικασία που περιλαμβάνεται είναι η διαγραφή των υπόλοιπων προτιμήσεων του σπουδαστή για το συγκεκριμένο Εργαστηριακό Μάθημα, κάτι το οποίο βελτιώνει την ταχύτητα εκτέλεσης αφού μειώνει τον συνολικό αριθμό Προτιμήσεων που πρέπει να περάσουν από επεξεργασία.

5.2 Καθορισμός facts και Αντικειμένων

Σε αυτήν την ενότητα θα καθοριστεί λεπτομερώς η δομή των facts που θα χρησιμοποιηθούν από το Έμπειρο Σύστημα. Εφόσον σαν πλατφόρμα έχει επιλεγεί το JESS, τα facts θα έχουν τη μορφή λιστών και για την καλύτερη δόμηση θα χρησιμοποιηθούν templates. Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, τα templates ορίζουν τη δομή και τις πληροφορίες που θα περιέχουν τα facts αυτού του τύπου, είναι δηλαδή κάτι παρόμοιο με τις δομές των συμβατικών γλωσσών προγραμματισμού. Τα templates που θα δημιουργηθούν εδώ είναι τα εξής:

```
(deftemplate studentPreference
  "Δομή που αποθηκεύει την προτίμηση ενός σπουδαστή για ένα εργαστηριακό τμήμα."
  (slot AM) ; Ο αριθμός μητρώου του σπουδαστή.
  (slot labID) ; Ο κωδικός του εργαστηριακού τμήματος.
  (slot PREFERENCE (type INTEGER)) ; Ο αριθμός προτίμησης του σπουδαστή.
)
```

```
(deftemplate MAIN::labHours
  "Δομή που αποθηκεύει πληροφορίες για τις ώρες/ημέρες που διδάσκεται ένα εργαστηριακό τμήμα."
  (slot labID (type STRING)) ; Ο κωδικός του εργαστηριακού τμήματος.
  (slot labDay (type STRING)) ; ID της ημέρας (0 Δευτέρα, 1 Τρίτη κτλ.)
  (slot labStartTime (type INTEGER)) ; Ώρα έναρξης εργαστηρίου (πχ 16)
  (slot labEndTime (type INTEGER)) ; Ώρα λήξης εργαστηρίου (πχ 18)
)
```

```
(deftemplate MAIN::labInfo
  "Δομή που αποθηκεύει πληροφορίες για ένα εργαστηριακό τμήμα."
  (slot labID (type STRING)) ; Ο κωδικός του εργαστηριακού τμήματος.
  (slot courseID (type STRING)) ; Ο κωδικός του μαθήματος.
  (slot curSize (type INTEGER)) ; Τρέχων αριθμός εγγεγραμμένων σπουδαστών
  (slot maxSize (type INTEGER)) ; Μέγιστος αριθμός εγγεγραμμένων σπουδαστών
)
```

Επίσης, όπως αναφέρθηκε, ένα χρήσιμο χαρακτηριστικό που παρέχει η μηχανή κανόνων JESS είναι η δυνατότητα άμεσης αλληλεπίδρασης του Εμπείρου Συστήματος με αντικείμενα της Java. Πιο συγκεκριμένα, δίνεται η δυνατότητα καταχώρησης τέτοιων αντικειμένων στη βάση γνώσης του συστήματος και η πρόσβαση τους σε μορφή fact. Έτσι για την εύκολη καταχώρηση και ανάκτηση των αποτελεσμάτων των κληρώσεων (επιτυχής και ανεπιτυχής εγγραφές) θα χρησιμοποιηθεί μια κλάση Java αντί για ένα τυπικό template:

```
/**
 * Δομή που περιέχει την προσπάθεια εγγραφής ενός σπουδαστή σε ένα τμήμα.
 */
public class Registration {
    /**
     * Ο αριθμός μητρώου του σπουδαστή.
     */
    public String AM;
    /**
     * Ο κωδικός του εργαστηριακού τμήματος στο οποίο κληρώθηκε ή απέτυχε να
     * κληρωθεί ο σπουδαστής.
     */
    public String labID;
    /**
     * Ο κωδικός του μαθήματος στο οποίο ανήκει το τμήμα.
     */
    public String courseID;
    /**
     * Επιτυχής ή ανεπιτυχής εγγραφή.
     */
    public Boolean successful;
    /**
     * Επιπρόσθετες λεπτομέρειες για τη συγκεκριμένη εγγραφή (πχ. λόγος αποτυχίας).
     */
    public String details;
```

Ο λόγος που χρησιμοποιήθηκαν templates αντί για κλάσεις Java στην προηγούμενη σελίδα, είναι η καλύτερη δόμηση του κώδικα. Πιο συγκεκριμένα, τα αρχεία το Εμπείρου Συστήματος δεν μεταγλωττίζονται, αντίθετα από τις κλάσεις Java, αλλά διερμηνεύονται από το JESS. Έτσι, συγκεντρώνοντας όσο περισσότερα από τα δομικά στοιχεία του Εμπείρου Συστήματος είναι δυνατό στα συγκεκριμένα αρχεία, καθίσταται εύκολη η κατανόηση της δομής και της λειτουργίας του Εμπείρου Συστήματος ακόμα και ελλείψει του πηγαίου κώδικα του υπόλοιπου προγράμματος.

5.3 Καθορισμός Κανόνων

Η διαδικασία κληρώσεων χωρίζεται σε φάσεις. Οι φάσεις αυτές θα διαχωριστούν μεταξύ τους με τη βοήθεια του μηχανισμού των modules, όπου κάθε module θα περιλαμβάνει τους κανόνες που απαιτούνται για την διεκπεραίωση των λειτουργιών.

Έτσι έχουμε αρχικά την **φάση εγγραφής**, η οποία υλοποιείται στο module `LAB_REGISTRATION_MAIN`. Επίσης πρέπει να σημειωθεί ότι η στρατηγική εγγραφών δεν επιλέγεται σε αυτή τη φάση, αλλά επιλέγεται πριν από την έναρξη του συστήματος.

Ο πρώτος και ίσως ο σημαντικότερος κανόνας είναι αυτός που εγγράφει έναν σπουδαστή σε κάποιο εργαστηριακό τμήμα. Το σύστημα επιλέγει κάθε φορά την πρώτη προτεραιότητα ενός σπουδαστή και ελέγχει ότι το συγκεκριμένο εργαστηριακό τμήμα δεν έχει γεμίσει. Αν αυτή η προϋπόθεση πληρείται τότε ο σπουδαστής εγγράφεται στο συγκεκριμένο τμήμα και ο έλεγχος του προγράμματος μεταφέρεται στο module με όνομα `POST_REGISTRATION_TASKS`.

```
(defrule registerStudentToLab
?r1 <- (MAIN::studentPreference (AM ?AM) (labID ?labID) (PREFERENCE 1) (INITIAL-
PREFERENCE ?initialPref)) ; Επιλέγουμε την πρώτη προτεραιότητα
; Συσχετίζουμε το labID με τις υπόλοιπες πληροφορίες για το εργαστήριο
?rlab_1 <- (MAIN::labInfo (labID ?labID) (courseID ?courseID) (curSize ?curSize) (maxSize
?maxSize))
(test (< ?curSize ?maxSize)) ; Ελέγχουμε ότι το εργαστήριο δεν έχει γεμίσει
=>
(retract ?r1) ; Αφαίρεση της συγκεκριμένης προτίμησης από τον σπουδαστή
(modify ?rlab_1 (curSize (+ ?curSize 1))) ; Αύξηση του αριθμού σπουδαστών στο εργαστήριο
(addRegistration ?AM ?labID ?courseID ?initialPref "true" "") ; Εγγραφή του σπουδαστή
; Δίνουμε το focus στο module των εργασιών που έπονται της επιτυχημένης εγγραφής
(focus POST REGISTRATION TASKS)
(run)
)
```

Ο επόμενος κανόνας είναι παρόμοιος με τον πρώτο, αλλά αφορά την περίπτωση που το εργαστηριακό τμήμα έχει γεμίσει και δεν μπορεί να γίνει εγγραφή του σπουδαστή σε αυτό. Σε αυτή την περίπτωση ο έλεγχος μεταφέρεται στη φάση ολίσθησης (module με όνομα `SHIFT_PHASE`) αφού πρώτα δημιουργηθεί σχετικό fact που να δηλώνει για ποιον σπουδαστή και για πιο μάθημα θα πρέπει να πραγματοποιηθεί η ολίσθηση.

```
(defrule failedCozFull
?r1 <- (MAIN::studentPreference (AM ?AM) (labID ?labID) (PREFERENCE 1) (INITIAL-PREFERENCE
?initialPref)) ; Επιλέγουμε την πρώτη προτεραιότητα
; Συσχετίζουμε το labID με τις υπόλοιπες πληροφορίες για το εργαστήριο
(MAIN::labInfo (labID ?labID) (courseID ?courseID) (curSize ?curSize) (maxSize ?maxSize))
(test (>= ?curSize ?maxSize)) ; Ελέγχουμε ότι το εργαστήριο έχει γεμίσει
=>
(retract ?r1) ; Αφαίρεση της συγκεκριμένης προτίμησης από τον σπουδαστή
(addRegistration ?AM ?labID ?courseID ?initialPref "false" "labFull")
(assert (SHIFT_PHASE::bumpPreferences ?AM ?courseID))
(focus SHIFT_PHASE) ; Προχωράμε σε φάση ολίσθησης για να αναπροσαρμόσουν οι
προτεραιότητες
(run)
)
```

Η δεύτερη φάση του συστήματος είναι η **φάση ολίσθησης**, η λειτουργία της οποίας υλοποιείται στο module `SHIFT_PHASE`.

Όπως αναφέρθηκε, πριν από την είσοδο σε φάση ολίσθησης εισάγεται ένα fact που να δηλώνει για ποιον σπουδαστή και για πιο μάθημα θα πρέπει να πραγματοποιηθεί η ολίσθηση. Αυτό το fact ονομάζεται `bumpPreferences` και ακολουθείται από τον αριθμό μητρώου του σπουδαστή και τον κωδικό του μαθήματος.

Ο κύριος κανόνας σε αυτή τη φάση βρίσκει τα εργαστηριακά τμήματα του μαθήματος στο οποίο θα γίνει η ολίσθηση και στη συνέχεια μειώνει τις προτιμήσεις του σπουδαστή σε κάθε ένα από αυτά. Για παράδειγμα η προτίμηση 2 θα γίνει 1, η προτίμηση 3 θα γίνει 2 κ.ο.κ.

```
(defrule bumpPreference
  "Ανεβάζουμε την προτίμηση για αυτούς που δεν κατάφεραν να γραφτούν"
  (declare (no-loop TRUE)) ; Για αποφυγή λανθασμένων μειώσεων
  (bumpPreferences ?AM ?courseID)
  ; Αντλούμε πληροφορίες για το εργαστήριο και το απορρίπτουμε εάν είναι για άλλο μάθημα
  (MAIN::labInfo (labID ?labID) (courseID ?courseID))
  ?m <- (MAIN::studentPreference (AM ?AM) (labID ?labID) (PREFERENCE ?pref))
  (test (> ?pref 1))
  =>
  (modify ?m (PREFERENCE (- ?pref 1)))
)
```

Επιπλέον η συγκεκριμένη φάση περιλαμβάνει έναν βοηθητικό κανόνα ο οποίος πυροδοτείται μόνο όταν τελειώσει η ολίσθηση. Αυτός ο κανόνας αφαιρεί το fact '`bumpPreferences`' που δημιουργήθηκε πριν την είσοδο στην φάση ολίσθησης.

```
(defrule removeBumpPreferences
  "Αφαιρεί το fact bumpPreferences όταν η χρησιμότητα του τελειώσει."
  (declare (salience -1))
  ?r <- (bumpPreferences ?AM ?courseID)
  =>
  (retract ?r)
)
```

Τέλος υπάρχει μια βοηθητική φάση με τις λειτουργίες που πρέπει να εκτελεστούν στην περίπτωση **επιτυχημένης εγγραφής**. Αυτές υλοποιούνται στο module

```
POST_REGISTRATION_TASKS.
```

Η πρώτη λειτουργία είναι η αφαίρεση όλων των υπόλοιπων προτιμήσεων του σπουδαστή για το εργαστηριακό μάθημα στο οποίο γράφτηκε. Αυτό εξασφαλίζει ότι ο σπουδαστής δεν θα διπλογραφτεί (δηλαδή δεν θα εγγραφεί σε δύο εργαστηριακά τμήματα του ίδιου μαθήματος) και επιπλέον βελτιώνει την ταχύτητα εκτέλεσης της κλήρωσης αφού μειώνει τα patterns που θα πρέπει να ελεγχθούν.

```
(defrule removeOtherLabsOfSameCourse
  "Αφαίρεση των δηλώσεων για τα υπόλοιπα εργαστηριακά τμήματα του μαθήματος, αφού ο
  σπουδαστής έχει γραφτεί σε ένα"
  (MAIN::Registration (AM ?AM) (courseID ?courseID) (successful TRUE)) ; Αν ο σπουδαστής
  έχει ήδη γραφτεί κάπου από το σύστημα
  (MAIN::labInfo (labID ?labID) (courseID ?courseID)) ; Συσχέτιση του μαθήματος με τα
  εργαστήρια του
  ?r <- (MAIN::studentPreference (AM ?AM) (labID ?labID)) ; Βρίσκουμε τη σχετική εγγραφή
  και την μαρκάρουμε για να την αφαιρέσουμε
  =>
  (retract ?r)
)
```

Η δεύτερη λειτουργία αφορά την αποφυγή εγγραφής σε εργαστηριακά τμήματα διαφορετικών μαθημάτων τα οποία διδάσκονται τις ίδιες ώρες. Αυτό υλοποιείται με την εύρεση των εργαστηριακών τμημάτων τα οποία έχουν συγκρουόμενες ώρες με αυτό στο οποίο γράφτηκε ο σπουδαστής. Στη συνέχεια ελέγχεται αν ο σπουδαστής έχει κάποιο από αυτά τα τμήματα σαν πρώτη προτίμηση. Αν ναι τότε αυτό διαγράφεται από τις προτιμήσεις του και ο έλεγχος μεταβαίνει σε φάση ολίσθησης ώστε τα υπόλοιπα να ανεβούν κατά μια θέση.

```
(defrule conflict
  "Αν έχει δηλώσει 2 τμήματα την ίδια ώρα και γράφτηκε στο 1 τότε σβήνουμε το άλλο από την
  δήλωση και ανεβάζουμε τα άλλα"
  (MAIN::labHours (labID ?labID) (labDay ?lday) (labStartTime ?lst) (labEndTime ?let))
  (MAIN::Registration (AM ?AM) (labID ?labID) (successful TRUE))
  (MAIN::labHours (labID ?labID2) (labDay ?lday) (labStartTime ?lst) (labEndTime ?let)) ;
  Αν υπάρχει άλλο εργαστήριο (ίδια ώρα και ημέρα
  (MAIN::labInfo (labID ?labID2) (courseID ?courseID)) ; Συσχετισμός του labID του
  συγκρουόμενου εργαστηρίου με το courseID
  ?r <- (MAIN::studentPreference (AM ?AM) (labID ?labID2) (PREFERENCE 1) (INITIAL-
  PREFERENCE ?initialPref)) ; Και ο σπουδαστής το έχει στη λίστα του με προτίμηση 1
  =>
  (retract ?r) ; Διαγράφουμε τη δήλωση του σπουδαστή για το συγκεκριμένου τμήμα του
  δευτέρου μαθήματος
  (addRegistration ?AM ?labID2 ?courseID ?initialPref "false" "conflict")
  (assert (SHIFT PHASE::bumpPreferences ?AM ?courseID)) ; Προσθέτουμε κατάλληλο fact για να
  γνωρίζει το σύστημα ότι πρόκειται για σύγκρουση και να ανεβάσει προαιρεδότητες
  (focus SHIFT PHASE) ; Προχωράμε σε φάση ολίσθησης για να αναπροσαρμοστούν οι
  προτεραιότητες
  (run)
)
```

5.4 Προτεραιότητες Εγγραφής/Ειδικές Προτεραιότητες

Μια ακόμα λειτουργία που καλείται να πραγματοποιεί το σύστημα κληρώσεων είναι η παροχή προτεραιότητας σε συγκεκριμένες ομάδες σπουδαστών. Η επιλογή των σπουδαστών αυτών εξαρτάται από πολιτική που εισάγει ο διαχειριστής μέσω του Περιβάλλοντος Δήλωσης Προτιμήσεων (περισσότερα για αυτό υπάρχουν στο σχετικό κεφάλαιο). Οι προτεραιότητες αυτές ονομάζονται Ειδικές Προτεραιότητες (Ε.Π.) και οι σπουδαστές που έχουν τέτοια προτεραιότητα ανήκουν σε Ομάδα Ειδικής Προτεραιότητας (Ο.Ε.Π.).

Η υλοποίηση αυτής της λειτουργίας μπορεί να πραγματοποιηθεί με δύο τρόπους:

1. Δημιουργία custom στρατηγικής στο Έμπειρο Σύστημα που να τοποθετεί τους κανόνες που αφορούν σπουδαστές με Ε.Π. πρώτους στην ατζέντα και έτσι να πυροδοτούνται πρώτοι.
2. Αντί να εισάγονται όλοι οι σπουδαστές ταυτόχρονα στο Έμπειρο Σύστημα, να εισάγονται πρώτα οι σπουδαστές με Ε.Π. και αφού εγγραφούν τότε να εισάγονται οι υπόλοιποι.

Ο τρόπος που επιλέχθηκε είναι ο δεύτερος, γιατί είναι πιο απλός στην υλοποίηση. Έτσι οι σπουδαστές με Ε.Π. μπαίνουν πρώτοι στη διαδικασία κληρώσεων και ανταγωνίζονται μόνο μεταξύ τους για τις διαθέσιμες θέσεις. Οι Ε.Π. χωρίζονται σε επίπεδα, δηλαδή οι σπουδαστές που ανήκουν σε μια Ο.Ε.Π. μπαίνουν στη διαδικασία κληρώσεων μόνο όταν τελειώσουν οι σπουδαστές που ανήκουν σε Ο.Ε.Π. προηγούμενων επιπέδων.

6 Περιβάλλον Δήλωσης Προτιμήσεων

6.1 Εισαγωγή

Το Ε.Σ.Κ. μπορεί να αποτελεί την καρδιά αυτής της πτυχιακής, αλλά εξίσου σημαντική είναι και η ανάπτυξη ενός Περιβάλλοντος Δηλώσεων Προτιμήσεων (Π.Δ.Π.). Το περιβάλλον αυτό αναπτύχθηκε για να δώσει τη δυνατότητα εύκολης δημιουργίας και υποβολής δηλώσεων, παρουσίασης των αποτελεσμάτων και διαχείρισης των παραμέτρων του Ε.Σ.Κ.

Όπως αναφέρθηκε στην εισαγωγή το Π.Δ.Π. αυτό θα είναι διαδικτυακό και θα χρησιμοποιεί γνωστές και ευρέως χρησιμοποιούμενες τεχνολογίες στην ανάπτυξη του. Για αυτό το λόγο ως γλώσσα υλοποίησης επιλέχθηκε η PHP. Περισσότερο τεχνικές λεπτομέρειες όσον αφορά την υλοποίηση του Π.Δ.Π. υπάρχουν στα παρατήματα.

Οι λειτουργίες που παρέχει το Π.Δ.Π. στους χρήστες εξαρτώνται σε μεγάλο βαθμό από το ρόλο και τα δικαιώματα του κάθε χρήστη, αλλά κάποιες λειτουργίες είναι διαθέσιμες για όλους τους χρήστες. Αυτές οι λειτουργίες είναι η Εμφάνιση του Ωρολογίου Προγράμματος, όπου κάθε χρήστης μπορεί να δει το Ωρολόγιο Πρόγραμμα κάθε εξαμήνου με επιπλέον πληροφορίες όπως ξεχωριστό χρώμα για κάθε εργαστηριακό μάθημα και τα ονόματα των διδασκόντων.

Μια άλλη λειτουργία που παρέχει το Π.Δ.Π. σε όλους τους χρήστες είναι αυτή της εμφάνισης των Ημερομηνιών Κληρώσεων. Μέσα από αυτή ο χρήστης μπορεί να πάρει πληροφορίες για τις μελλοντικές ή ολοκληρωμένες κληρώσεις όπως την ημερομηνία και ώρα που εκτελέστηκαν/πρόκειται να εκτελεστούν.

Στη συνέχεια ακολουθεί μια επεξήγηση των ρόλων χρηστών και των λειτουργιών στις οποίες έχει πρόσβαση κάθε ρόλος.

6.2 Δικαιώματα χρηστών και πρόσβαση

Το σύστημα πρόσβασης που χρησιμοποιείται από το Π.Δ.Π. χρησιμοποιεί ρόλους χρηστών. Πιο συγκεκριμένα, υπάρχουν κάποιοι προκαθορισμένοι ρόλοι με συγκεκριμένα δικαιώματα και σε κάθε χρήστη μπορούν να ανατεθούν ένας οι περισσότεροι. Έτσι ένας χρήστης μπορεί για παράδειγμα να είναι Καθηγητής αλλά και Διαχειριστής, αποκτώντας πρόσβαση στα λειτουργίες και των δύο.

Οι ρόλοι που υποστηρίζονται από το Περιβάλλον είναι οι εξής:

- **Ανώνυμος:** Αυτόν τον ρόλο τον έχουν όλοι οι χρήστες, ακόμα και αυτοί που δεν έχουν κάνει login στο σύστημα. Παρέχει πρόσβαση σε βασικές λειτουργίες όπως την Εμφάνιση της Αρχικής Σελίδας, τις Ημερομηνίες Κληρώσεων, το Ωρολόγιο Πρόγραμμα και τη Βοήθεια.
- **Σπουδαστής:** Οι σπουδαστές έχουν πρόσβαση σε ένα εξατομικευμένο ωρολόγιο πρόγραμμα (Προσωπικό Ωρολόγιο Πρόγραμμα), στη σελίδα δήλωσης προτιμήσεων κάτω από προϋποθέσεις και στα Αποτελέσματα Κληρώσεων.
- **Καθηγητής:** Ο ρόλος του καθηγητή δίνει πρόσβαση στη σελίδα Επεξεργασίας Εργαστηριακών Τμημάτων, όπου μπορούν να ρυθμιστούν παράμετροι όπως το όνομα του εργαστηρίου, το μέγεθος του, το όνομα του διδάσκοντα κ.α. και μπορεί να εμφανιστεί μια απλή λίστα με τους κληρωμένους σπουδαστές για το συγκεκριμένο τμήμα.
- **Διαχειριστής:** Ο ρόλος του διαχειριστή είναι η Διαχείριση των Ημερομηνιών Κληρώσεων, η Ενεργοποίηση/Απενεργοποίηση Προτεραιοτήτων Εγγραφής και η Εμφάνιση Στατιστικών που αφορούν το σύστημα.

Σημείωση: Οι ρόλοι Σπουδαστής και Καθηγητής θεωρούνται αμοιβαία αποκλειόμενοι και ως εκ τούτου αυτό το ενδεχόμενο δεν έχει δοκιμαστεί κατά την αξιολόγηση του Π.Δ.Π.

6.3 Λειτουργίες για τον Σπουδαστή

Οι λειτουργίες που παρέχονται στον σπουδαστή χωρίζεται σε τρεις βασικές κατηγορίες: Εμφάνιση Προσωπικού Ωρολογίου Προγράμματος, Δήλωση Προτιμήσεων, Εμφάνιση Αποτελεσμάτων Κληρώσεων.

Το Προσωπικό Ωρολόγιο Πρόγραμμα είναι ένας δισδιάστατος πίνακας όπου οι γραμμές αντιπροσωπεύουν δίωρα διδασκαλίας και οι στήλες τις ημέρες της εβδομάδας. Στα κελιά εμφανίζονται τα μαθήματα (θεωρητικά ή εργαστηριακά) που διδάσκονται εκείνες τις ώρες και στα οποία έχει εγγραφεί ο σπουδαστής ή δύναται να εγγραφεί. Για την διευκόλυνση του σπουδαστή οι ώρες διδασκαλίας θεωρίας έχουν χρώμα ανοικτό μπλε ενώ για κάθε εργαστηριακό μάθημα επιλέγεται κάποιο μοναδικό χρώμα και χρησιμοποιείται για όλα τα εργαστηριακά του τμήματα. Επιπλέον ο σπουδαστής έχει τη δυνατότητα να εμφανίσει το Προσωπικό Ωρολόγιο Πρόγραμμα και σε ξεχωριστούς πίνακες με βάση το εξάμηνο. Ακολουθεί screenshot με τη συγκεκριμένη λειτουργία όπου αποτυπώνονται τα περισσότερα από αυτά τα χαρακτηριστικά.

Προσωπικό Ωρολόγιο Πρόγραμμα

Επιστροφή στην κορυφή

36

Η λειτουργία της δήλωσης προτιμήσεων, όπως φαίνεται και από το όνομα, δίνει τη δυνατότητα στον σπουδαστή (εκτός και αν θεωρείται "παλαιός", κάτι που θα εξηγηθεί στην μεθεπόμενη σελίδα) να δει τα διαθέσιμα εργαστηριακά τμήματα, όπως και στο Προσωπικό Ωρολόγιο Πρόγραμμα, με την επιπλέον δυνατότητα να θέσει τις προτιμήσεις του για κάθε ένα από αυτά. Για να επιτραπεί η υποβολή των προτιμήσεων στο σύστημα πρέπει να πληρούνται κάποια κριτήρια:

1. Οι προτιμήσεις πρέπει να ξεκινούν από 1 και να είναι συνεχόμενες για κάθε εργαστηριακό μάθημα. Για παράδειγμα δεν επιτρέπεται ένα εργαστηριακό τμήμα να έχει προτίμηση 1 και κάποιο άλλο (που ανήκει στο ίδιο μάθημα) να έχει προτίμηση 3 χωρίς να υπάρχει η προτίμηση 2 σε κάποιο άλλο τμήμα.
2. Οι προτιμήσεις για τα εργαστηριακά τμήματα που ανήκουν στο ίδιο εργαστηριακό μάθημα πρέπει να είναι μοναδικές.

Για τη διευκόλυνση του σπουδαστή, κατά τη διαδικασία επιλογής προεπιλέγεται αυτόματα η επόμενη διαθέσιμη προτίμηση που επιτρέπεται σε εκείνο το μάθημα (ο σπουδαστής έχει φυσικά τη δυνατότητα να αλλάξει την προτίμηση που προεπιλέχθηκε). Μια επιπλέον δυνατότητα που προσφέρεται είναι το φιλτράρισμα με βάση το εργαστηριακό μάθημα, όπου εμφανίζονται μόνο τα εργαστηριακά τμήματα που ανήκουν στο συγκεκριμένο μάθημα. Τέλος, εκτός από δήλωση μέσα σε πίνακα παρόμοιο με αυτόν του Προσωπικού Ωρολογίου Προγράμματος, δίνεται η δυνατότητα στον σπουδαστή να θέσει τις προτιμήσεις του μέσα από μια Λίστα Εργαστηριακών Τμημάτων.

Η σελίδα δήλωσης προτιμήσεων είναι προσβάσιμη μόνο όταν υπάρχουν μελλοντικές κληρώσεις.

Έχετε συνδεθεί ως

ΔΗΜΟΣΘΕΝΗΣ ΝΕΚΤΑΡΙΟΣ ΝΙΚΟΥΔΗΣ

(Σπουδαστής)

Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών

βάσει Προτιμήσεων

ΤΕΙ

ΑΘΗΝΑΣ

Τμήμα Πληροφορικής

Δήλωση Προτιμήσεων

Αποτελέσματα Κληρώσεων

Προσωπικό Ωρολόγιο Πρόγραμμα

Ημερομηνίες Κληρώσεων

Ωρολόγιο Πρόγραμμα

Σχόλια

Λίστα Εργαστηριακών Τμημάτων ανά Μάθημα

Εμφάνιση Όλων των Μαθημάτων

Μεταγλωττιστές

Τεχνητή Νοημοσύνη

Πίνακας Εργαστηριακών Τμημάτων ανά Μάθημα

Εμφάνιση Όλων των Μαθημάτων

Μεταγλωττιστές

Τεχνητή Νοημοσύνη

Εργαστήρια Χωρίς Παρακολούθηση

Δεν έχετε εγγραφεί ως παλαιός σπουδαστής σε κανένα μάθημα.

Δήλωση Προτιμήσεων

Σημείωση: Μπορείτε να ταξινομήσετε πολλές στήλες ταυτόχρονα κρατώντας πατημένο το SHIFT.

Υπάρχουν μαθήματα, στην παρακάτω λίστα, στα οποία δεν έχετε επιλέξει καμία προτίμηση.

Μάθημα	Όνομα Τμήματος	Ημέρα	Ώρα	Αριθμός Εγγεγραμμένων Σπουδαστών	Αριθμός Σπουδαστών που το έχουν δηλώσει σαν Πρώτη Προτίμηση	Επιλογή Προτίμησης
Μεταγλωττιστές	MET1	Τρίτη	16-18	8/10	0	<div>Αποχώρηση</div>
Τεχνητή Νοημοσύνη	TN1	Δευτέρα	14-16	11/11	2	<div>Καμία</div>
Τεχνητή Νοημοσύνη	TN2	Τρίτη	12-14	7/10	0	<div>Καμία</div>
Τεχνητή Νοημοσύνη	TN3	Τρίτη	14-16	10/10	0	<div>Καμία</div>
Τεχνητή Νοημοσύνη	TN4	Τρίτη	16-18	4/10	0	<div>Καμία</div>

Καταχώρηση

Ακύρωση Αλλαγών

Εγγραφή ως Παλαιός Σπουδαστής

Επιστροφή στην κορυφή

Εικόνα 6.2 Μια τυπική Δήλωση Προτιμήσεων (Λίστα Εργαστηριακών Τμημάτων ανά Μάθημα)

Εκτός από τους σπουδαστές που συμμετέχουν στη διαδικασία κληρώσεων, σε κάποια εργαστηριακά μαθήματα υπάρχει ένα μέρος σπουδαστών που καλούνται "Παλαιοί Σπουδαστές". Το πότε ένας σπουδαστής θεωρείται "παλαιός" είναι μια παράμετρος που δεν αφορά τη συγκεκριμένη πτυχιακή⁶ και εδώ αποτελεί απλά ένα flag στη βάση δεδομένων.


Οι "Παλαιοί Σπουδαστές" δεν έχουν δικαίωμα δήλωσης προτιμήσεων και συμμετοχής στη διαδικασία κληρώσεων για τα εργαστηριακά μαθήματα στα οποία θεωρούνται "παλαιοί". Έχουν όμως δικαίωμα να εγγραφούν σε ένα εικονικό εργαστηριακό τμήμα "για παλαιούς", το οποίο δεν έχει όριο στον αριθμό των εγγεγραμμένων σπουδαστών.

Η εγγραφή στο τμήμα "παλαιών" είναι άμεση και ο σπουδαστής απλά περνάει από μια σελίδα επιβεβαίωσης επειδή το Π.Δ.Π. δεν παρέχει δυνατότητα αποχώρησης από το συγκεκριμένο τμήμα.

⁶ Αυτή τη στιγμή ως "παλαιός φοιτητής" στο Τμήμα Πληροφορικής του ΤΕΙ Αθήνας ορίζεται κάποιος που συμπλήρωσε τον απαραίτητο αριθμό παρουσιών για κάποιο εργαστηριακό μάθημα αλλά δεν έλαβε προβιβασμό βαθμό.

Έχετε συνδεθεί ως:
ΔΗΜΟΣΘΕΝΗΣ ΝΕΚΤΑΡΙΟΣ ΝΙΚΟΥΔΗΣ
(Σπουδαστής)
Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών
βάσει Προτιμήσεων



Τμήμα Πληροφορικής

Δήλωση Προτιμήσεων

Αποτελέσματα Κληρώσεων

Προσωπικό Ωρολόγιο Πρόγραμμα

Ημερομηνίες Κληρώσεων

Ωρολόγιο Πρόγραμμα

Σχόλια

Λίστα Εργαστηριακών Τμημάτων ανά Μάθημα

Εμφάνιση Όλων των Μαθημάτων
Μεταγλωττιστές
Τεχνητή Νοημοσύνη

Πίνακας Εργαστηριακών Τμημάτων ανά Μάθημα

Εμφάνιση Όλων των Μαθημάτων
Μεταγλωττιστές
Τεχνητή Νοημοσύνη

Εργαστήρια Χωρίς Παρακολούθηση
Δεν έχετε εγγραφεί ως πταλιός σπουδαστής σε κανένα μάθημα.

Δήλωση Προτιμήσεων
Εγγραφή Παλαιού Σπουδαστή

Πρόκειται να εγγραφείτε στο τμήμα παλαιών σπουδαστών του παρακάτω μαθήματος

Τεχνητή Νοημοσύνη

Για να συνεχίσετε πατήστε στο παρακάτω κουμπί

Εγγραφή

Επιστροφή στην κορυφή

Εικόνα 6.8 Σελίδα Επιβεβαίωσης Εγγραφής "Παλαιού Σπουδαστή"

Η τρίτη λειτουργία που παρέχει το σύστημα στον σπουδαστή είναι η Εμφάνιση των Αποτελεσμάτων Κληρώσεων. Σε αυτή τη σελίδα, ο σπουδαστής μπορεί να δει τις ημερομηνίες των κληρώσεων (διαχωρισμένες σε μελλοντικές και ολοκληρωμένες) και τα αποτελέσματα της πιο πρόσφατης κλήρωσης. Τα αποτελέσματα αυτά διαχωρίζονται σε 3 κατηγορίες:


1. Επιτυχείς Εγγραφές
2. Ανεπιτυχείς Εγγραφές σε Εργαστηριακό Μάθημα
3. Ανεπιτυχείς Εγγραφές σε Εργαστηριακά Τμήματα Υψηλότερης Προτεραιότητας

Το 2^ο συμβαίνει μόνο αν μετά το πέρας της κλήρωσης ο σπουδαστής δεν κατάφερε να εγγραφεί σε κανένα εργαστηριακό τμήμα κάποιου μαθήματος.

Στην περίπτωση επιτυχούς εγγραφής, ο σπουδαστής έχει τη δυνατότητα αποχώρησης από το εργαστηριακό τμήμα στο οποίο γράφτηκε. Αυτό του δίνει τη δυνατότητα να επιλέξει νέες προτιμήσεις για τα εργαστηριακά τμήματα εκείνου του μαθήματος για εισαγωγή στην επόμενη κλήρωση. Κατά την αποχώρηση του σπουδαστή από κάποιο εργαστηριακό τμήμα εμφανίζεται σχετική σελίδα επιβεβαίωσης, για την αποφυγή λανθασμένων αποχωρήσεων.

Έχετε συνδεθεί ως
ΔΗΜΟΣΘΕΝΗΣ ΝΕΚΤΑΡΙΟΣ ΝΙΚΟΥΔΗΣ
(Σπουδαστής)
Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών
βάσει Προτιμήσεων



Τμήμα Πληροφορικής

Δήλωση Προτιμήσεων

Αποτελέσματα Κληρώσεων

Προσωπικό Ωρολόγιο Πρόγραμμα

Ημερομηνίες Κληρώσεων

Ωρολόγιο Πρόγραμμα

Σχόλια

Ημ/νίες Κληρώσεων
Μελλοντικές

2011-03-30 21:00:00

Πιο πρόσφατη

2011-03-23 21:00:00

Αποτελέσματα Κληρώσεων

Κλήρωση 2011-03-23 21:00:00

Επιτυχείς Εγγραφές

Μάθημα	Τμήμα	Ημέρα	Ωρα	Αποτέλεσμα	Ενέργεια
Μεταγλωττιστές (Εργαστήριο)	MET'	Τρίτη	16-18	✓	Αποχώρηση

Αποτυχία Εγγραφής σε Εργαστηριακά Μαθήματα

Μάθημα

Τεχνητή Νοημοσύνη (Εργαστήριο)

Μη-Επιτυχείς Εγγραφές σε Τμήματα Υψηλότερης Προτεραιότητας

Μάθημα	Τμήμα	Ημέρα	Ωρα	Λόγος Αποτυχίας
Τεχνητή Νοημοσύνη (Εργαστήριο)	TN'	Δευτέρα	14-16	labFull

Επιστροφή στην κορυφή

Εικόνα 6.3 Αποτελέσματα Κλήρωσης (της πιο πρόσφατης) ενός σπουδαστή

6.4 Λειτουργίες για τον Καθηγητή

Στον αρχικό σχεδιασμό, ο καθηγητής έχει πρόσβαση σε μια βασική λειτουργία: την Επεξεργασία των Τμημάτων για τα μαθήματα που είναι υπεύθυνος. Αυτή η λειτουργία δίνει τη δυνατότητα καθορισμού παραμέτρων για τα εργαστηριακά τμήματα των μαθημάτων αυτών. Τέτοιες παράμετροι μπορούν να είναι τα μεγέθη των εργαστηριακών τμημάτων, τα ονόματα τους και ο διδάσκων καθηγητής κάθε τμήματος. Μια επιπλέον λειτουργία που παρέχεται στον καθηγητή είναι η εμφάνιση των ονομάτων των κληρωμένων σπουδαστών καθώς και των παλαιών σπουδαστών που επέλεξαν να εγγραφούν στο τμήμα Παλαιών.

Στη συγκεκριμένη λειτουργία έχουν πρόσβαση όλοι οι χρήστες με το ρόλο του Καθηγητή, ανεξαρτήτως αν είναι υπεύθυνοι για κάποιο μάθημα ή όχι. Αν κάποιος καθηγητής δεν είναι υπεύθυνος για κανένα μάθημα τότε η λίστα των μαθημάτων του θα είναι κενή και έτσι δεν θα έχει πρόσβαση να δει ή να αλλάξει οτιδήποτε.

Εδώ πρέπει να σημειωθεί ότι στα πλαίσια της συγκεκριμένης εργασίας οι παράμετροι των εργαστηριακών τμημάτων παίρνονται από ανεξάρτητη βάση δεδομένων (πιο συγκεκριμένα από τη βάση δεδομένων της πλατφόρμας Eclass). Έτσι, εφόσον δεν μπορούν να γίνουν αλλαγές στις παραμέτρους των εργαστηριακών τμημάτων μέσω του Π.Δ.Π., η συγκεκριμένη σελίδα μετονομάστηκε σε "Εμφάνιση Κληρωθέντων Σπουδαστών". Η σελίδα αυτή έχει παρόμοια δομή με την προηγούμενη, με την διαφορά ότι επιτρέπει μόνο την εμφάνιση των σπουδαστών που έχουν κληρωθεί σε κάθε εργαστηριακό τμήμα.

Έχετε συνδεθεί ως
Κατερίνα Γεωργιάδη
(Καθηγητής, Διαχειριστής)
Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών

βάσει Προτιμήσεων

Τμήμα Πληροφορικής

Στατιστικά

Επεξεργασία Προτεραιοτήτων

Επεξεργασία Κληρώσεων

Εμφάνιση Κληρωθέντων Σπουδαστών

Ημερησινίες Κληρώσεων

Ωρολόγιο Πρόγραμμα

Σχόλια

Λίστα Εργαστηριακών Τμημάτων ανά Μάθημα

Τεχνητή Νοημοσύνη

Μεταγλωττιστές

Πίνακας Εργαστηριακών Τμημάτων ανά Μάθημα

Τεχνητή Νοημοσύνη

Μεταγλωττιστές

Εμφάνιση Κληρωθέντων Σπουδαστών

Εμφάνιση Παλαιών Σπουδαστών

Σημείωση: Μπορείτε να ταξινομήσετε πολλές στήλες ταυτόχρονα κρατώντας πατημένο το SHIFT.

Όνομα Τμήματος	Ημέρα	Ωρα	Έχουν Κληρωθεί	Μένεθος	Σπουδαστές σε αναμονή που το έχουν δηλώσει σαν Πρώτη Προτεραιότητα	Καθηγητής
TN1	Δευτέρα	14-16	1' (Εμφάνιση)	1'	2	Γεωργιάδη Κατερίνα
TN2	Τρίτη	12-14	7 (Εμφάνιση)	10	0	Σκαλκίδης Ηλίας
TN3	Τρίτη	14-16	10 (Εμφάνιση)	10	0	Σκαλκίδης Ηλίας
TN4	Τρίτη	16-18	4 (Εμφάνιση)	10	0	Σκαλκίδης Ηλίας

Επιστροφή στην κορυφή

Εικόνα 6.4 Εμφάνιση Κληρωθέντων Σπουδαστών

44


6.5 Λειτουργίες για τον Διαχειριστή

Οι λειτουργίες που προσφέρει το Π.Δ.Π. στον διαχειριστή είναι τρεις: Επεξεργασία Κληρώσεων, Επεξεργασία Προτεραιοτήτων Εγγραφής και Εμφάνιση Στατιστικών Στοιχείων.

Η λειτουργία της Επεξεργασίας Κληρώσεων δίνει τη δυνατότητα στον διαχειριστή να εισάγει ημερομηνίες κληρώσεων ή να διαγράφει τις υπάρχουσες. Οι νέες κληρώσεις έχουν τον περιορισμό ότι πρέπει να αφορούν μελλοντικές ημερομηνίες, μιας και αν βρίσκονται στο παρελθόν δεν θα εκτελεστούν ποτέ. Ο ίδιος περιορισμός ισχύει και κατά τη διαγραφή, δηλαδή δεν μπορούν να διαγραφούν κληρώσεις οι οποίες έχουν ολοκληρωθεί, γιατί ενδέχεται να υπάρχουν σπουδαστές που να έχουν εγγραφεί σε τμήματα μέσω αυτών.

Έχετε συνδεθεί ως
Κατερίνα Γεωργιάδη
(Καθηγητής, Διαχειριστής)
[Αποσύνδεση](#)

Σύστημα Εργαστηριακών Εγγραφών
βάσει Προτιμήσεων



Τμήμα Πληροφορικής

Στατιστικά

Επεξεργασία Προτεραιοτήτων

Επεξεργασία Κληρώσεων

Εμφάνιση Κληρωθέντων Σπουδαστών

Ημερομηνίες Κληρώσεων

Ορολόγιο Πρόγραμμα

Σχόλια

Επεξεργασία Κληρώσεων


Διαγραφή Μελλοντικών Κληρώσεων

Ημερομηνία και Ώρα	Επιλογή
03/30/11 21:00:00	<input type="checkbox"/>

Διαγραφή Επιλεγμένων

Εισαγωγή Νέας Κλήρωσης

Οι κληρώσεις προτείνεται να απέχουν τουλάχιστον 24 ώρες μεταξύ τους.



Εισαγωγή Κλήρωσης Ακύρωση

Επιστροφή στην κορυφή

Εικόνα 6.5 Επεξεργασία Κληρώσεων

Η Επεξεργασία Προτεραιοτήτων Εγγραφής είναι η δεύτερη λειτουργία που προσφέρεται στον διαχειριστή. Όπως είχε αναφερθεί σε προηγούμενα κεφάλαια, το Σύστημα Κληρώσεων έχει τη δυνατότητα να δώσει προτεραιότητα σε συγκεκριμένες ομάδες σπουδαστών όταν οι θέσεις των εργαστηριακών τμημάτων δεν επαρκούν για όλους. Ο καθορισμός αυτών των ομάδων γίνεται μέσα από τη συγκεκριμένη ιστοσελίδα. Όπως αναφέρθηκε στο κεφάλαιο του Ε.Σ.Κ., ο όρος που χρησιμοποιείται για αυτές τις ομάδες είναι Ομάδα Ειδικής Προτεραιότητας (Ο.Ε.Π.).

Η δημιουργία τέτοιων προτεραιοτήτων γίνεται υπό τη μορφή SQL ερωτημάτων στην βάση δεδομένων των σπουδαστών. Πιο συγκεκριμένα ο διαχειριστής σχηματίζει ένα SQL ερώτημα το οποίο επιστρέφει ένα σύνολο σπουδαστών που έχουν τα επιλεγμένα χαρακτηριστικά και αυτοί οι σπουδαστές μπαίνουν στην συγκεκριμένη Ο.Ε.Π.

Οι σπουδαστές που ανήκουν σε μια Ο.Ε.Π. ανταγωνίζονται μεταξύ τους για τις διαθέσιμες θέσεις. Οι διαφορετικές Ο.Ε.Π. δεν έχουν ίσες προτεραιότητες μεταξύ τους, αλλά χωρίζονται σε επίπεδα προτεραιοτήτων. Για παράδειγμα οι σπουδαστές που ανήκουν στην Ο.Ε.Π. με προτεραιότητα 1 θα εγγραφούν πρώτοι στις διαθέσιμες θέσεις ενώ οι σπουδαστές που ανήκουν στην Ο.Ε.Π. με προτεραιότητα 2 θα εγγραφούν σε όσες θέσεις έμειναν ελεύθερες μετά την εγγραφή των πρώτων.

Οι σπουδαστές που δεν ανήκουν σε καμία Ο.Ε.Π. θεωρείται ότι ανήκουν σε μια εικονική Ο.Ε.Π. με προτεραιότητα 9 (δηλαδή τελευταία).

Έχετε συνδεθεί ως
Κατερίνα Γεωργιάδη
(Καθηγήτρια, Διαχειριστής)
Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών
βάσει Προτιμήσεων



Τμήμα Πληροφορικής

Στατιστικά

Επεξεργασία Προτεραιοτήτων

Επεξεργασία Κληρώσεων

Εμφάνιση Κληρωθέντων Σπουδαστών

Ημερομηνίες Κληρώσεων

Ορολόγιο Πρόγραμμα

Σχόλια

Επεξεργασία Προτεραιοτήτων Εγγραφής

Προτεραιότητα	Ενεργοποιημένη	Όνομα
1	<input checked="" type="checkbox"/>	Άτομα με κινητικά προβλήματα
2	<input checked="" type="checkbox"/>	Εργαζόμενοι σπουδαστές
3	<input type="checkbox"/>	
4	<input type="checkbox"/>	
5	<input type="checkbox"/>	
6	<input type="checkbox"/>	
7	<input type="checkbox"/>	
8	<input type="checkbox"/>	
9	<input checked="" type="checkbox"/>	Οι υπόλοιποι σπουδαστές

Για προχωρημένους

Καταχώρηση

Ακύρωση Αλλαγών

Επιστροφή στην κορυφή

Εικόνα 6.6 Επεξεργασία Προτεραιοτήτων Εγγραφής/Ειδικών Προτεραιοτήτων

Η τελευταία λειτουργία που προσφέρεται στον διαχειριστή είναι αυτή της Εμφάνισης Στατιστικών Στοιχείων (Ε.Σ.Σ.). Η συγκεκριμένη λειτουργία παρέχει στατιστικά στοιχεία στον διαχειριστή που αφορούν τη συνολική λειτουργία του συστήματος ή τα αποτελέσματα των κληρώσεων.

Τα στοιχεία που αφορούν αποτελέσματα κληρώσεων εμφανίζονται σε δύο μορφές: ανά προτίμηση και ανά μάθημα. Στη εμφάνιση ποσοστών επιτυχίας ανά προτίμηση εμφανίζονται οι επιτυχείς εγγραφές με βάση το επίπεδο προτίμησης. Για παράδειγμα αν κάποιος σπουδαστής δεν κατάφερε να εγγραφεί στην 1η και στην 2η προτίμηση του αλλά γράφτηκε στην 3η, τότε στο επίπεδο 1 και 2 θα μετρηθεί ως ανεπιτυχής και στο επίπεδο 3 ως επιτυχής. Τα ποσοστά επιτυχίας ανά μάθημα εμφανίζουν έναν παρόμοιο πίνακα, μόνο που αντί για επίπεδα προτιμήσεων υπάρχουν τα εργαστηριακά μαθήματα.

Να σημειωθεί ότι όλοι οι πίνακες μπορούν να ταξινομηθούν με βάση οποιαδήποτε στήλη. Η προεπιλεγμένη ταξινόμηση γίνεται με βάση τις Ανεπιτυχείς Εγγραφές, καθώς αποτελούν ένα από τα πιο χρήσιμα στατιστικά στοιχεία του συστήματος.

Κλήρωση 2011-03-23 21:00:00

Σπουδαστές που δεν γράφτηκαν σε κανένα τμήμα, εμφάνιση κατά μάθημα

* Οι σπουδαστές που δεν είχαν ορίσει καμία προτίμηση δεν προσμετρώνται στον παραπάνω πίνακα

Αποκρυψη			
Εργαστηριακό Τμήμα	Επιτυχείς Εγγγραφές	Ανεπιτυχείς Εγγγραφές	Σύνολο
Τεχνητή Νοημοσύνη	32 (86.5%)	5 (13.5%)	37
Αλληλεπίδραση Ανθρώπου-Υπολογιστή	15 (88.2%)	2 (11.8%)	17
Τεχνολογία Λογισμικού	10 (90.9%)	1 (9.1%)	11
Σήματα και Συστήματα	3 (75.0%)	1 (25.0%)	4
Τηλεπικοινωνίες	7 (87.5%)	1 (12.5%)	8
Δίκτυα Υπολογιστών II	9 (90.0%)	1 (10.0%)	10
Μεθοδολογίες Ανάπτυξης Εφαρμογών	9 (90.0%)	1 (10.0%)	10
Δικτυακός Προγραμματισμός	9 (90.0%)	1 (10.0%)	10
Αριθμητική Ανάλυση & Εφαρμογές I	6 (100.0%)	0 (0.0%)	6
Δομές Δεδομένων	12 (100.0%)	0 (0.0%)	12
Ψηφιακή Σχεδίαση	1 (100.0%)	0 (0.0%)	1
Αντικειμενοστραφής Προγραμματισμός	13 (100.0%)	0 (0.0%)	13

50

7 Αξιολόγηση

7.1 Δοκιμαστικά εργαλεία και παραδοχές

Έχοντας αναπτύξει πλέον το σύστημα προχωράμε στη φάση της αξιολόγησης. Η φάση της αξιολόγησης θα περιλαμβάνει σε πρώτη φάση την διεξαγωγή δοκιμαστικών σεναρίων και αξιολόγηση των αποτελεσμάτων. Ως σενάριο θεωρούμε μια κατάσταση που θα εισαχθεί στο Ε.Σ.Κ. και περιλαμβάνει τις δηλώσεις προτιμήσεων ενός ή περισσότερων σπουδαστών σε ένα ή περισσότερα εργαστηριακά τμήματα (τα οποία μπορεί να ανήκουν ή να μην ανήκουν στα ίδια εργαστηριακά μαθήματα).

Για την διευκόλυνση της διεξαγωγής δοκιμών αναπτύχθηκε ένα απλό εργαλείο σε γλώσσα PHP, το οποίο δίνει τη δυνατότητα παραγωγής σεναρίων, δημιουργώντας αυτόματα δηλώσεις προτιμήσεων ενός ή περισσότερων σπουδαστών, σε ένα ή περισσότερα εργαστηριακά τμήματα. Αυτό γίνεται αφού ληφθεί υπόψη η υπάρχουσα κατάσταση του συστήματος (πχ. ποιοί σπουδαστές έχουν δηλώσει το μάθημα για το οποίο θα δημιουργηθούν σενάρια), οι περιορισμοί που υπάρχουν και οι παράμετροι του σεναρίου που πρόκειται να δημιουργηθεί. Όλες οι παράμετροι καθορίζονται από το δημιουργό του σεναρίου.

Ο αριθμός των σπουδαστών μπορεί να είναι είτε μεγαλύτερος είτε μικρότερος από την χωρητικότητα του εργαστηριακού τμήματος. Υπάρχει ένας περιορισμός, στο ότι ο αριθμός των δηλώσεων που θα δημιουργηθούν δεν μπορεί να υπερβαίνει τον αριθμό των σπουδαστών που είναι εγγεγραμμένοι στο εργαστηριακό μάθημα. Η παραπάνω διαδικασία αυτή μπορεί να επαναληφθεί όσες φορές θέλουμε ώστε να δημιουργηθούν πιο σύνθετα σενάρια, δηλαδή σενάρια με πολλά εργαστηριακά μαθήματα και πολλά εργαστηριακά τμήματα.

Η διεξαγωγή της αξιολόγησης γίνεται ως εξής: Υπάρχει ένας σπουδαστής τον οποίο ελέγχουμε. Χρησιμοποιώντας αυτόν τον σπουδαστή, δηλώνουμε προτιμήσεις σε κάποιο ή κάποια εργαστηριακά τμήματα ανάλογα με τις ανάγκες του σεναρίου. Στη συνέχεια, ενδέχεται να χρησιμοποιηθεί το εργαλείο παραγωγής σεναρίων που αναφέρθηκε παραπάνω, για την εισαγωγή προτιμήσεων άλλων σπουδαστών, στο ίδιο ή σε διαφορετικά τμήματα. Τέλος εκτελείται η κλήρωση και τα αποτελέσματα αντλούνται από τη σελίδα "Αποτελέσματα Κληρώσεων" του ΠΔΠ για τον συγκεκριμένο σπουδαστή, ώστε να παρατεθούν προς αξιολόγηση.

Α' Φάση Αξιολόγησης

7.2 Απλή περίπτωση εγγραφής σπουδαστή σε μη-γεμάτο τμήμα (Σενάριο 1)

Σαν πρώτο σενάριο έχουμε μια απλή περίπτωση όπου ο σπουδαστής που ελέγχουμε δηλώνει μόνο ένα εργαστηριακό τμήμα (MET1, το οποίο ανήκει στο μάθημα Μεταγλωττιστές) με προτίμηση 1. Επιπλέον, στο συγκεκριμένο εργαστηριακό τμήμα ο αριθμός των σπουδαστών που το έχουν δηλώσει είναι μικρότερος από την χωρητικότητα του τμήματος.

Δήλωση Προτιμήσεων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Προτίμηση
MET1	Τρίτη	16-18	1

Αναμενόμενο Αποτέλεσμα:

Ο σπουδαστής θα εγγραφεί με επιτυχία στο τμήμα MET1.

Αποτελέσματα Κλήρωσης:

Επιτυχείς Εγγραφές:

MET1

Αποτυχία Εγγραφής σε Εργαστηριακά Μαθήματα:

--

Αποτυχία Εγγραφής σε Τμήματα Υψηλότερης Προτεραιότητας:

--	--

7.3 Ολική αποτυχία εγγραφής σε μάθημα (Σενάριο 2)

Σε αυτό το σενάριο έχουμε μια δήλωση παρόμοια με αυτή του Σεναρίου 1. Το σενάριο διαφέρει στο ότι το εργαστηριακό τμήμα MET1 έχει δηλωθεί σαν πρώτη προτίμηση από άλλους 82 σπουδαστές (το τμήμα έχει χωρητικότητα 15 σπουδαστές). Το MET1 έχει χωρητικότητα 10 σπουδαστών.

Δήλωση Προτιμήσεων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Προτίμηση
MET1	Τρίτη	16-18	1

Αναμενόμενο Αποτέλεσμα:

Λόγω του μεγάλου ανταγωνισμού για εγγραφή στο τμήμα, η πιθανότητα να εγγραφεί ο σπουδαστής στο MET1 είναι μικρή (~1.22%). Για αυτό το λόγο, το πιο πιθανό αποτέλεσμα είναι ότι ο σπουδαστής δεν θα επιτύχει να εγγραφεί στο MET1. Επιπλέον, αφού δεν έχει δηλώσει προτίμηση σε κανένα άλλο τμήμα για το συγκεκριμένο μάθημα, το πιθανότερο είναι να μην επιτύχει να εγγραφεί σε κανένα τμήμα του μαθήματος Μεταγλωττιστές.

Αποτελέσματα Κλήρωσης:

Επιτυχείς Εγγραφές:

--

Αποτυχία Εγγραφής σε Εργαστηριακά Μαθήματα:

Μεταγλωττιστές

Αποτυχία Εγγραφής σε Τμήματα Υψηλότερης Προτεραιότητας:

MET1	labFull
------	---------

7.4 Αποτυχία εγγραφής στην πρώτη επιλογή (Σενάριο 3)

Αυτό το σενάριο είναι παρόμοιο με το σενάριο 2, με την διαφορά ότι ο σπουδαστής δηλώνει, εκτός από το εργαστηριακό τμήμα MET1, και το τμήμα MET2 με προτίμηση 2. Για την απλοποίηση του σεναρίου, στο MET1 υπάρχουν άλλοι 82 σπουδαστές που το έχουν δηλώσει με προτίμηση 1, όπως στο σενάριο 2, αλλά το MET2 το έχει δηλώσει μόνο ο σπουδαστής που ελέγχουμε. Το MET1 έχει χωρητικότητα 10 σπουδαστών.

Δήλωση Προτιμήσεων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Προτίμηση
MET1	Τρίτη	16-18	1
MET2	Τρίτη	18-20	2

Αναμενόμενο Αποτέλεσμα:

Λόγω του μεγάλου ανταγωνισμού για εγγραφή στο τμήμα, η πιθανότητα να εγγραφεί ο σπουδαστής στο MET1 είναι μικρή (~1.22%). Για αυτό το λόγο, το πιο πιθανό αποτέλεσμα είναι ότι ο σπουδαστής δεν θα επιτύχει να εγγραφεί στο MET1. Επειδή όμως έχει δηλώσει το MET2 με προτίμηση 2, θα εγγραφεί εκεί με επιτυχία.

Αποτελέσματα Κλήρωσης:

Επιτυχείς Εγγραφές:

MET2

Αποτυχία Εγγραφής σε Εργαστηριακά Μαθήματα:

--

Αποτυχία Εγγραφής σε Τμήματα Υψηλότερης Προτεραιότητας:

MET1	labFull
------	---------

7.5 Αποτυχία εγγραφής λόγω σύγκρουσης ωρών (Σενάριο 4)

Αυτό το σενάριο δοκιμάζει το δεύτερο κριτήριο αποτυχίας εγγραφής - σύγκρουση ωρών. Ο σπουδαστής που ελέγχουμε δηλώνει σαν πρώτη προτίμηση τα εργαστηριακά τμήματα MET1 και TN2 (το TN2 ανήκει σε διαφορετικό μάθημα, Τεχνητή Νοημοσύνη). Τα δύο αυτά τμήματα διδάσκονται την ίδια ημέρα και ώρα και οι δηλώσεις σε αυτά είναι λιγότερες των διαθέσιμων θέσεων έτσι ώστε να επιτρέπουν την εγγραφή του σπουδαστή.

Δήλωση Προτιμήσεων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Προτίμηση
MET1	Τρίτη	16-18	1
TN2	Τρίτη	16-18	1

Αναμενόμενο Αποτέλεσμα:

Ο σπουδαστής θα εγγραφεί με επιτυχία μόνο σε ένα από τα δύο τμήματα. Το εργαστήριο στο οποίο θα γίνει τελικά η εγγραφή εξαρτάται από την στρατηγική που χρησιμοποιεί το Ε.Σ.Κ. για να επιλέξει ποια θα εκτελεστεί πρώτη. Επειδή η στρατηγική που χρησιμοποιείται στα συγκεκριμένα σενάρια είναι η random, δηλαδή τυχαία επιλογή, το αποτέλεσμα δεν μπορεί να προβλεφθεί.

Αποτελέσματα Κλήρωσης:

Επιτυχείς Εγγραφές:

TN2

Αποτυχία Εγγραφής σε Εργαστηριακά Μαθήματα:

Μεταγλωττιστές

Αποτυχία Εγγραφής σε Τμήματα Υψηλότερης Προτεραιότητας:

MET1	conflict
------	----------

Β' Φάση Αξιολόγησης

7.6 Δοκιμή με πραγματικούς σπουδαστές

Έχοντας εξασφαλίσει την ορθή λειτουργία του συστήματος σε απλές δοκιμαστικές περιπτώσεις, προχωράμε στη δεύτερη φάση αξιολόγησης, η οποία περιλαμβάνει τη συμμετοχή πραγματικών σπουδαστών στη διαδικασία.

Στα πλαίσια αυτής της δοκιμής, ζητήθηκε από περίπου 70 σπουδαστές του ΤΕΙ Αθήνας να συμμετάσχουν στη διαδικασία αξιολόγησης, δηλώνοντας της προτιμήσεις τους στα τμήματα όπου θα ήθελαν ιδανικά να εγγραφούν. Η δοκιμή περιλάμβανε δύο κληρώσεις, οι οποίες απείχαν μια εβδομάδα μεταξύ τους (ημερομηνίες 23/03/2011 και 30/03/2011), δίνοντας την ευκαιρία στους σπουδαστές να χρησιμοποιήσουν, εκτός από τη λειτουργία Δήλωσης Προτιμήσεων, την λειτουργία Αποχώρησης. Επιπλέον, οι χωρητικότητες των εργαστηριακών τμημάτων της δοκιμής μειώθηκαν κατά 50% σε σχέση με τις πραγματικές, για να αντισταθμιστεί το γεγονός ότι οι σπουδαστές που θα συμμετείχαν θα ήταν λιγότεροι από τη συνολική χωρητικότητα για την οποία είχαν σχεδιαστεί τα πραγματικά τμήματα.

Ο κύριος στόχος της δοκιμής ήταν η άντληση στατιστικών στοιχείων από τα αποτελέσματα των κληρώσεων. Υπήρχε ωστόσο, και ένας αριθμός από δευτερεύοντες στόχους, των οποίων τα αποτελέσματα καταγράφηκαν, αλλά δεν θα παρουσιαστούν στις παρακάτω παραγράφους. Κάποιοι από αυτούς ήταν: αξιολόγηση και βελτίωση της ευχρηστίας του συστήματος, εύρεση τεχνικών προβλημάτων/σφαλμάτων, παρατήρηση της στάσης των χρηστών προς τις νέες διαδικασίες κ.α.

Η διαδικασία αξιολόγησης σε αυτή τη φάση είναι διαφορετική από αυτή της Α' Φάσης. Τα αποτελέσματα αντλούνται από της σελίδες "Εμφάνιση Κληρωμένων Σπουδαστών" και "Εμφάνιση Στατιστικών Στοιχείων" του καθηγητή και του διαχειριστή αντίστοιχα. Για τον περιορισμό των αποτελεσμάτων η αξιολόγηση θα επικεντρωθεί στα εργαστηριακά τμήματα ενός συγκεκριμένου μαθήματος. Το μάθημα που επιλέχθηκε ήταν η Τεχνητή Νοημοσύνη, καθώς είχε τον μεγαλύτερο αριθμό δηλώσεων.

Η μορφή της παρουσίασης θα είναι η εξής: Αρχικά θα παρατεθεί η αρχική κατάσταση όπου φαίνονται τα ονόματα και οι χωρητικότητες των τμημάτων, καθώς και ο αριθμός των σπουδαστών που τα έχουν δηλώσει ως πρώτη προτίμηση. Στη συνέχεια θα παρατεθεί ένας πίνακας με στατιστικά στοιχεία που να αφορούν τα ποσοστά εγγραφής σε κάθε εργαστηριακό τμήμα αυτού του μαθήματος. Τέλος θα παρατεθεί μια εικόνα με τα συνολικά στατιστικά στοιχεία της συγκεκριμένης κλήρωσης.

7.7 Αποτελέσματα Πρώτης Κλήρωσης (23/03/2011)

Πριν την κλήρωση:

Πίνακας Εργαστηριακών Τμημάτων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Αριθμός Κληρωμένων Σπουδαστών	Μέγεθος	Σπουδαστές σε αναμονή που το έχουν δηλώσει σαν Πρώτη Προτίμηση
TN1	Δευτέρα	14-16	0	11	16
TN2	Τρίτη	12-14	0	10	5
TN3	Τρίτη	14-16	0	10	9
TN4	Τρίτη	16-18	0	10	3

Μετά την κλήρωση:

Πίνακας Εργαστηριακών Τμημάτων:

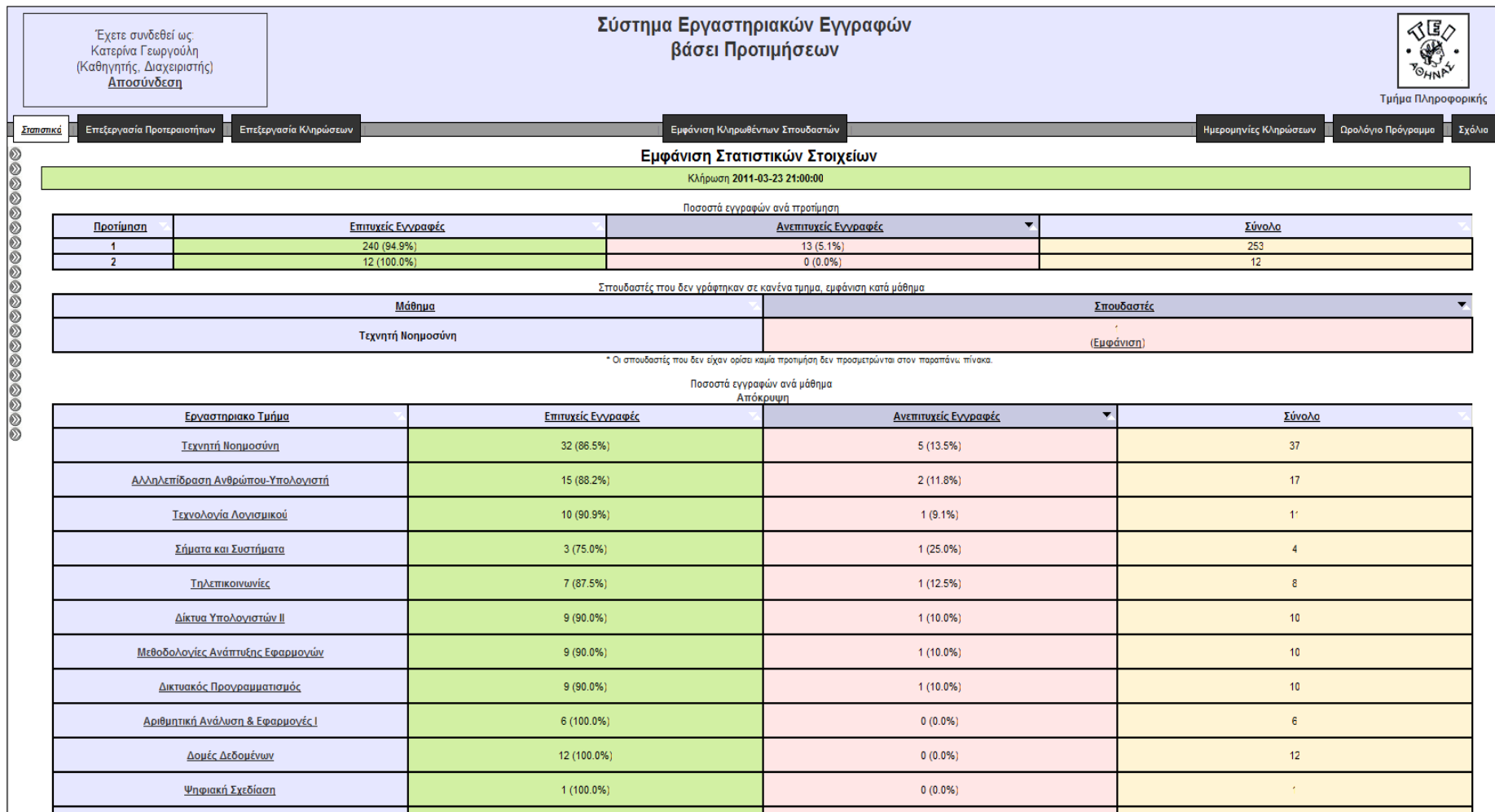
Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Αριθμός Κληρωμένων Σπουδαστών	Μέγεθος	Σπουδαστές σε αναμονή που το έχουν δηλώσει σαν Πρώτη Προτίμηση
TN1	Δευτέρα	14-16	11	11	0
TN2	Τρίτη	12-14	7	10	0
TN3	Τρίτη	14-16	10	10	0
TN4	Τρίτη	16-18	4	10	0

Εγγραφές ανά Εργαστηριακό Τμήμα:

Εργαστηριακό Τμήμα	Επιτυχείς Εγγραφές	Ανεπιτυχείς Εγγραφές
TN1	11	5
TN2	7	0
TN3	10	0
TN4	4	0

Σπουδαστές που δήλωσαν τουλάχιστον μια προτίμηση, αλλά δεν γράφτηκαν σε κανένα εργαστηριακό τμήμα:

Εργαστηριακό Μάθημα	Σπουδαστές
Τεχνητή Νοημοσύνη	1



Εικόνα 7.1 Εμφάνιση Στατιστικών Στοιχείων μετά από την Πρώτης Κλήρωση (23/03/2011)

7.8 Αποτελέσματα Δεύτερης Κλήρωσης (30/03/2011)

Πριν την κλήρωση:

Πίνακας Εργαστηριακών Τμημάτων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Αριθμός Κληρωμένων Σπουδαστών	Μέγεθος	Σπουδαστές σε αναμονή που το έχουν δηλώσει σαν Πρώτη Προτίμηση
TN1	Δευτέρα	14-16	11	11	2
TN2	Τρίτη	12-14	7	10	0
TN3	Τρίτη	14-16	10	10	0
TN4	Τρίτη	16-18	4	10	0

Μετά την κλήρωση:

Πίνακας Εργαστηριακών Τμημάτων:

Εργαστηριακό Τμήμα	Ημέρα	Ώρα	Αριθμός Κληρωμένων Σπουδαστών	Μέγεθος	Σπουδαστές σε αναμονή που το έχουν δηλώσει σαν Πρώτη Προτίμηση
TN1	Δευτέρα	14-16	11	11	0
TN2	Τρίτη	12-14	8	10	0
TN3	Τρίτη	14-16	10	10	0
TN4	Τρίτη	16-18	4	10	0

Εγγραφές ανά Εργαστηριακό Τμήμα:

Εργαστηριακό Τμήμα	Επιτυχείς Εγγραφές	Ανεπιτυχείς Εγγραφές
TN1	0	2
TN2	1	0

Σπουδαστές που δήλωσαν τουλάχιστον μια προτίμηση, αλλά δεν γράφτηκαν σε κανένα εργαστηριακό τμήμα:

Εργαστηριακό Μάθημα	Σπουδαστές
Τεχνητή Νοημοσύνη	1

Έχετε συνδεθεί ως:
Κατερίνα Γεωργιάδη
(Καθηγήτριας, Διαχειρίστρας)
Αποσύνδεση

Σύστημα Εργαστηριακών Εγγραφών
βάσει Προτιμήσεων

Τμήμα Πληροφορικής

Στατιστικά

Επεξεργασία Προτεραιοτήτων

Επεξεργασία Κληρώσεων

Εμφάνιση Κληρωθέντων Σπουδαστών

Ημερομηνίες Κληρώσεων

Οραλόγιο Πρόγραμμα

Σχόλια

Εμφάνιση Στατιστικών Στοιχείων

Κλήρωση 2011-03-30 21:00:00

Ποσοστά εγγραφών ανά προτίμηση

Προτίμηση	Επιτυχείς Εγγραφές	Ανεπιτυχείς Εγγραφές	Σύνολο
1	18 (78.3%)	5 (21.7%)	23
2	4 (100.0%)	0 (0.0%)	4

Σπουδαστές που δεν γράφτηκαν σε κανένα τμήμα, εμφάνιση κατά μάθημα

Μάθημα	Σπουδαστές
Τεχνητή Νοημοσύνη	1 (Εμφάνιση)

* Οι σπουδαστές που δεν είχαν ορίσει καμία προτίμηση δεν προσμετρώνται στον παραπάνω πίνακα.

Ποσοστά εγγραφών ανά μάθημα
Απόκρυψη

Εργαστηριακό Τμήμα	Επιτυχείς Εγγραφές	Ανεπιτυχείς Εγγραφές	Σύνολο
Τεχνητή Νοημοσύνη	1 (33.3%)	2 (66.7%)	3
Δίκτυα Υπολογιστών II	1 (50.0%)	1 (50.0%)	2
Αλληλεπίδραση Ανθρώπου-Υπολογιστή	1 (50.0%)	1 (50.0%)	2
Εισαγωγή στον Παράλληλο Υπολογισμό	2 (66.7%)	1 (33.3%)	3
Αριθμητική Ανάλυση & Εφαρμογές I	1 (100.0%)	0 (0.0%)	1
Δομές Δεδομένων	1 (100.0%)	0 (0.0%)	1
Ψηφιακή Σχεδίαση	1 (100.0%)	0 (0.0%)	1

Εικόνα 7.2 Εμφάνιση Στατιστικών Στοιχείων μετά από την Δεύτερη Κλήρωση (30/03/2011)

7.9 Συμπεράσματα Β' Φάσης Αξιολόγησης

Το πρώτο συμπέρασμα που μπορούμε να εξάγουμε από τις παραπάνω κληρώσεις είναι ότι στην συντριπτική πλειοψηφία τους, οι σπουδαστές κατάφεραν να εγγραφούν στις πρώτες τους προτιμήσεις. Από τους σπουδαστές που δεν κατάφεραν να εγγραφούν στις πρώτες τους προτιμήσεις, όλοι εκτός από δύο (ένας σε κάθε κλήρωση), γράφτηκαν στις δεύτερες. Αντίστοιχο συμπέρασμα μπορεί να εξαχθεί και για το σύνολο των μαθημάτων, όπου στην κατά την πρώτη κλήρωση το 94.9% των δηλώσεων⁷ κληρώθηκαν στην πρώτη προτίμηση.

Ωστόσο, είναι σημαντικό να σημειωθεί ότι οι δύο σπουδαστές που αναφέρθηκαν παραπάνω, δεν κατάφεραν να εγγραφούν σε κανένα τμήμα του μαθήματος Τεχνητή Νοημοσύνη. Οι λόγοι που μπορεί να οδήγησαν σε κάτι τέτοιο είναι δύο: η μη-δήλωση όλων των δυνατών προτιμήσεων (πχ. ο σπουδαστής δηλώνει το TN1 ως πρώτη προτίμηση, χωρίς τη δήλωση άλλων εργαστηριακών τμημάτων ως δεύτερη, τρίτη κτλ.), ή η ύπαρξη περισσότερων σπουδαστών από τη συνολική χωρητικότητα των εργαστηριακών τμημάτων του μαθήματος. Στη συγκεκριμένη δοκιμή το δεύτερο όπως μπορούμε να δούμε δεν συνέβη, καθώς υπήρχαν κενές θέσεις σε ορισμένα τμήματα, άρα μπορούμε να συμπεράνουμε ότι ο λόγος της μη-εγγραφής τους ήταν ο πρώτος.

⁷ Το ποσοστό δεν αναφέρεται σε αριθμό σπουδαστών, αλλά δηλώσεων, καθώς ένας σπουδαστής μπορεί να έχει δηλώσεις σε πολλαπλά μαθήματα.

8 Τελικά συμπεράσματα και μελλοντικοί στόχοι

Σύμφωνα με τους στόχους που τέθηκαν στην εισαγωγή, υλοποιήθηκαν τόσο το Έμπειρο Σύστημα Κληρώσεων όσο και το Περιβάλλον Δήλωσης Προτιμήσεων. Σύμφωνα με τα σενάρια που δοκιμάστηκαν στις δύο φάσεις αξιολόγησης, το σύστημα φαίνεται να λειτουργεί ικανοποιητικά στις περισσότερες δοκιμαστικές περιπτώσεις.

Είναι σημαντικό να σημειωθεί ότι, αν και χρησιμοποιήθηκαν δοκιμασμένες και δημοφιλείς τεχνικές κατά την ανάπτυξη του συστήματος, η χρήση του σε πραγματικό περιβάλλον θα απαιτούσε μια πιο επαγγελματική προσέγγιση. Για παράδειγμα κάποιοι τομείς που θα μπορούσαν να βελτιωθούν είναι: η αισθητική και η ευχρηστία του ΠΔΠ, η διαλειτουργικότητα με άλλες πλατφόρμες πέραν του Eclass/Claroline, η δημιουργία κατάλληλων προγραμματιστικών διεπαφών (API) που να επιτρέπουν τη σύνδεση άλλων εφαρμογών με το σύστημα, η δοκιμή σε περισσότερα περιβάλλοντα ώστε να εξασφαλιστεί η καλή λειτουργία του ανεξάρτητα από το λογισμικό/υλικό στο οποίο βρίσκεται.

Ένα πρόβλημα το οποίο δεν μπορεί να επιλυθεί αυτή τη στιγμή από το σύστημα είναι όταν οι δηλώσεις των σπουδαστών δεν περιλαμβάνουν όλα τα διαθέσιμα εργαστηριακά τμήματα, με αποτέλεσμα κάποιοι σπουδαστές να μην εγγράφονται σε κανένα τμήμα ενώ υπάρχει διαθέσιμη χωρητικότητα. Μια μελλοντική λύση για αυτό το πρόβλημα θα ήταν η προσέγγιση των υπόλοιπων σπουδαστών και εργαστηριακών τμημάτων ως μέρος ενός προβλήματος Bin Packing. Η τοποθέτηση των σπουδαστών που δεν κληρώθηκαν σε κανένα εργαστηριακό τμήμα θα μπορούσε πλέον να γίνει με κατάλληλο αλγόριθμο Bin Packing, ώστε να επιτευχθεί η βέλτιστη κατανομή.

9 Βιβλιογραφία

Baldi, Mauro Maria, Teodor Gabriel Crainic, Guido Perboli, και Roberto Tadei. «The Generalized Bin Packing Problem.» *CIRRELT*, 21 May 2010.

Coffman, E. G., M. R. Garey, και D. S. Johnson. *Approximation Algorithms for Bin-Packing -- An Updated Survey*. In *Algorithm Design for Computer System Design*, ed. by Ausiello, Lucertini, and Serafini. Springer-Verlag, 1984.

Feigenbaum, Edward, και συν. «JTEC Panel on Knowledge-Based Systems in Japan.» Final Report, 1993, Chapter 1.

Forgy, Charles L. *Rete: A fast algorithm for the many pattern/many object pattern match problem*. Elsevier Science B.V, 1982.

Friesen, D. K., και M. A. Langston. «Variable Sized Bin Packing.» *SIAM Journal on Computing* Volume 15, αρ. Issue 1 (1986): 222-230.

Kang, Jangha, και Sungsoo Park. *European Journal of Operational Research* Volume 147, Issue 2, June 2003: 365-372.

Riley, Gary. *CLIPS: A Tool for Building Expert Systems*. <http://clipsrules.sourceforge.net/>.

Stefik, Mark. *Introduction to Knowledge Systems*. 1995.

Βλαχάβας, Ιωάννης, Πέτρος Κεφαλάς, Νίκος Βασιλειάδης, και Ηλίας Σακελλαρίου. *Τεχνητή Νοημοσύνη*. Θεσσαλονίκη: Β. Γκιούρδας Εκδοτική, 2006.

Παράρτημα Α

Διαχείριση Δεδομένων

Ένας από τους βασικούς τεχνικούς στόχους που τέθηκαν για το σύστημα είναι η παροχή ενός υψηλού βαθμού φορητότητας. Πιο συγκεκριμένα, ο στόχος όσον αφορά τη διαχείριση δεδομένων είναι να υπάρχει όσο το δυνατό μεγαλύτερη ανεξαρτησία του συστήματος από τη δομή των συστημάτων που βρίσκονται από κάτω του. Δηλαδή το σύστημα πρέπει να έχει τη δυνατότητα να λειτουργήσει με δεδομένα που μπορεί να προέρχονται από τελείως ανεξάρτητες και διαφορετικές πηγές (πχ. να μπορεί να γίνει αυθεντικοποίηση μέσω ενός LDAP server αλλά οι δηλώσεις μαθημάτων να ανακτώνται μέσα από μια τυπική βάση δεδομένων).

Για την επίτευξη αυτού του στόχου χρησιμοποιείται ένα ενδιάμεσο interface που ονομάζεται DataHandler και αναλαμβάνει όλες τις διαδικασίες ανάκτησης και αποθήκευσης δεδομένων δίνοντας πρόσβαση σε αυτά μέσω των συναρτήσεων της. Οι κλάσεις που υλοποιούν αυτό το interface μπορούν να ανακτήσουν τα δεδομένα που χρειάζονται με οτιδήποτε τρόπο. Στα πλαίσια της συγκεκριμένης εργασίας έχει αναπτυχθεί μια απλή κλάση που υλοποιεί αυτό το interface και αντλεί τα δεδομένα της μέσα από δύο βάσεις δεδομένων: μια εσωτερική βάση που κρατάει τα δεδομένα που αφορούν το σύστημα κληρώσεων (πχ. προτιμήσεις σπουδαστών, ημερομηνίες κληρώσεων, αποτελέσματα κτλ.) και μια γενική βάση που περιέχει πληροφορίες για το περιβάλλον της εφαρμογής (πληροφορίες για τα εργαστηριακά τμήματα, το ωρολόγιο πρόγραμμα κτλ.). Σαν γενική βάση στη συγκεκριμένη εργασία χρησιμοποιείται η βάση δεδομένων της πλατφόρμας Eclass του Τμήματος Πληροφορικής του ΤΕΙ Αθήνας.

Η παραπάνω φιλοσοφία χρησιμοποιείται και στο Έμπειρο Σύστημα Κληρώσεων (ΕΣΚ), αλλά και στο Περιβάλλον Δήλωσης Προτιμήσεων (ΠΔΠ). Για να καταστεί πιο κατανοητό, στην επόμενη σελίδα ακολουθούν μερικές δηλώσεις συναρτήσεων από το DataHandler του ΠΔΠ (υλοποιημένο σε γλώσσα PHP) μαζί με την τεκμηρίωση τους σε μορφή phpDoc. Να σημειωθεί ότι οι συναρτήσεις που παρουσιάζονται παρακάτω αποτελούν ένα μικρό μόνο μέρος του συνολικού DataHandler. Η πλήρης τεκμηρίωση βρίσκεται μέσα στον φάκελο phpDoc.

Συνάρτηση που επιστρέφει εργαστηριακά τμήματα:

```
/**
 * Επιστρέφει ένα δισδιάστατο πίνακα με εργαστηριακά τμήματα και πληροφορίες
 * για αυτά, φιλτραρισμένα με βάση διάφορες παραμέτρους, οι οποίες μπορούν
 * να συνδυαστούν. Οι πληροφορίες που περιέχονται αφορούν τη θέση στο
 * ωρολόγιο πρόγραμμα, τον τύπο, το εξάμηνο στο οποίο διδάσκονται κ.τ.λ.
 * και είναι οι εξής:
 * <ol><li>NAME -> Όνομα του εργαστηριακού τμήματος.</li>
 * <li>DAY -> Το ID της ημέρας στην οποία διδάσκεται.</li>
 * <li>dayName -> Το όνομα της ημέρας στην οποία διδάσκεται.</li>
 * <li>tfrom -> Η ώρα έναρξης του εργαστηρίου.</li>
 * <li>tto -> Η ώρα λήξης του εργαστηρίου.</li>
 * <li>courseType -> Ο τύπος του μαθήματος (συνήθως "Εργαστήριο").</li>
 * <li>labID -> Ο μοναδικός κωδικός του εργαστηριακού τμήματος.</li>
 * <li>courseID -> Ο μοναδικός κωδικός του μαθήματος στο οποίο ανήκει.</li>
 * <li>courseTaughtSemester -> Το εξάμηνο στο οποίο διδάσκεται.</li>
 * </ol>
 * @param String $AM Επιστροφή μόνο των τμημάτων στα οποία είναι γραμμένος ο
 * σπουδαστής με αυτό τον αριθμό μητρώου. Αν είναι null τότε δεν υπάρχει
 * φιλτράρισμα με βάση τον αριθμό μητρώου.
 * @param String $courseID Επιστροφή μόνο των τμημάτων που ανήκουν στο
 * συγκεκριμένο μάθημα. Αν είναι null τότε δεν υπάρχει φιλτράρισμα με βάση
 * τον κωδικό μαθήματος.
 * @param int $courseTaughtSemester Επιστροφή μόνο των τμημάτων που ανήκουν
 * σε μάθημα που διδάσκονται στο συγκεκριμένο εξάμηνο. Αν είναι null τότε
 * δεν υπάρχει φιλτράρισμα με βάση το εξάμηνο.
 * @return mixed Δισδιάστατος πίνακας με εργαστηριακά τμήματα και
 * πληροφορίες που τα αφορούν. Η μια διάσταση είναι απλά ένα αριθμητικό
 * index, ενώ η άλλη είναι associative array με τις διάφορες πληροφορίες. Η
 * θέση με δείκτη numStudents περιέχει τον αριθμό των φοιτητών που έχουν
 * κληρωθεί σε αυτό το εργαστήριο, ή έχουν γραφτεί μόνιμα με κάποιον τρόπο.
 */
abstract public function getLabs($AM = null, $courseID = null, $courseTaughtSemester
= null);
```

Συνάρτηση που επιστρέφει τις τρέχουσες προτιμήσεις ενός σπουδαστή.

```
/**
 * Επιστρέφει έναν πίνακα με τις προτιμήσεις του δοθέντα σπουδαστή.
 * @param String $AM Ο αριθμός μητρώου του σπουδαστή.
 * @return mixed Επιστρέφει ένα μονοδιάστατο associative array, με τα
 * ονόματα των δεικτών να είναι κωδικοί εργαστηριακών τμημάτων και τα
 * περιεχόμενα να είναι οι αντίστοιχες προτιμήσεις για αυτά τα τμήματα.
 */
abstract public function getCurrentPreferences($AM);
```

Συνάρτηση που επιστρέφει πληροφορίες για διάφορες κληρώσεις:

```
/**
 * Επιστρέφει τα ID και τις ημερομηνίες των κληρώσεων που είναι μέσα
 * στα πλαίσια της παραμέτρου. Επίσης επιστρέφει αν κάποια/ες από αυτές
 * είναι αυτή τη στιγμή σε εξέλιξη (δηλαδή τρέχει το σύστημα κληρώσεων για
 * αυτές).
 * @param String $context Δηλώνει ποιές κληρώσεις θέλουμε να εμφανίσουμε.
 * Μπορεί να έχει τις τιμές all, past (ή συνώνυμα το completed) ή future.
 * Επίσης μπορεί να χρησιμοποιηθεί η τιμή last που επιστρέφει μόνο την
 * πιο πρόσφατη ολοκληρωμένη κλήρωση.
 * @param int $lotID Μας επιτρέπει να φιλτράρουμε τα αποτελέσματα ώστε να
 * επιστρέφουν πληροφορίες μόνο για τη συγκεκριμένη κλήρωση με αυτό το
 * lotID. Χρήσιμο όταν θέλουμε να ανακτήσουμε πληροφορίες (πχ. ημ/νία) για
 * μια συγκεκριμένη κλήρωση όπου το lotID είναι γνωστό.
 * @return mixed Επιστρέφει δισδιάστατο πίνακα που περιέχει τα ID, τις
 * ημερομηνίες των κληρώσεων και το αν αυτές είναι αυτή τη στιγμή σε
 * εξέλιξη.
 */
abstract public function getLotteries($context = 'all');
```

Παράρτημα Β

Οδηγίες Εγκατάστασης και Ελάχιστες Απαιτήσεις

Οι απαιτήσεις συστήματος όσον αφορά το **λογισμικό** (εγκατεστημένα προγράμματα ή βιβλιοθήκες) είναι οι εξής:

- PHP για το interface των χρηστών.
- MySQL αν χρησιμοποιηθεί ο προεπιλεγμένος DataHandler που επικοινωνεί άμεσα με τη βάση δεδομένων.
- Java Standard Edition για να μπορεί να τρέξει το σύστημα κληρώσεων.

Οι απαιτήσεις σε **υλικό** (hardware) είναι:

- Επεξεργαστής: 1GHz για την ολοκλήρωση κληρώσεων σε λογικά χρονικά πλαίσια.
- Μνήμη: 512MB ελεύθερη μνήμη κατά τη διάρκεια διεξαγωγής της κλήρωσης.

Οδηγίες Εγκατάστασης:

1. Αντιγραφή των αρχείων του interface σε κατάλογο που βρίσκεται εντός του web root (είναι δηλαδή προσβάσιμος από μέσω http).
2. Εγκατάσταση των κατάλληλων βάσεων δεδομένων σύμφωνα με τις οδηγίες του 6.2.
3. Αλλαγή δικαιωμάτων έτσι ώστε ο χρήστης του web server να έχει πρόσβαση ανάγνωσης και εγγραφής στους καταλόγους libs/Smarty/cache και libs/Smarty/templates_c.
4. Ρύθμιση των παραμέτρων στο config.php και στο lottery_system/config.properties.
5. Δημιουργία μιας Scheduled Task (Windows) ή Cron Job (Linux) που να τρέχει το PHP script LotteryCronJob.php ανά σύντομα χρονικά διαστήματα. Να σημειωθεί ότι όταν δεν υπάρχει κλήρωση που πρέπει να εκτελεστεί το script εκτελεί μόνο ένα query και κλείνει, άρα καταναλώνει πολύ λίγους πόρους.

Παράρτημα Γ

Επεκτασιμότητα του Περιβάλλοντος Δήλωσης Προτιμήσεων

Το Π.Δ.Π. είναι ένα αρκετά μεγάλο project που καλείται να περιλάβει έναν μεγάλο αριθμό λειτουργιών. Για να διευκολυνθεί η διαδικασία της ενσωμάτωσης αυτών των λειτουργιών, το Π.Δ.Π. αυτό αναπτύχθηκε σε μορφή μονάδων (modules). Αυτό σημαίνει ότι για κάθε λειτουργία αναπτύχθηκε μια ξεχωριστή μονάδα/module και στη συνέχεια ενσωματώθηκε στο σύστημα με τον τρόπο που θα αναλυθεί παρακάτω.

Κάθε μονάδα/module πρέπει να βρίσκεται σε δικό του κατάλογο, ο οποίος πρέπει να βρίσκεται μέσα στον κεντρικό κατάλογο modules. Κατά την εκτέλεση του Περιβάλλοντος τα περιεχόμενα του καταλόγου modules σαρώνονται και σε κάθε υποκατάλογο αναζητούνται δύο αρχεία:

- **ModuleDescr.php**: Αυτό το αρχείο περιλαμβάνει μεταβλητές που περιγράφουν το module και τη θέση/σειρά του στο κεντρικό menu. Μιας και το αρχείο είναι ρηρ υπάρχει η δυνατότητα ορισμού επιπλέον προϋποθέσεων για την εμφάνιση/πρόσβαση στο module, αλλά αυτό δεν είναι καλή πρακτική και ο έλεγχος προτείνεται να γίνεται μέσα στο ίδιο το module.
- **ΌνομαModule.php**: Αυτό το αρχείο πρέπει να έχει το ίδιο όνομα με τον κατάλογο στον οποίο βρίσκεται και να δηλώνει μια κλάση πάλι με το ίδιο όνομα. Η κλάση αυτή πρέπει να υλοποιεί το interface IModule, το οποίο περιέχει τη συνάρτηση run() η οποία θα εκτελεστεί όταν ο χρήστης επιλέξει το συγκεκριμένο module.

Εκτός από τα παραπάνω, το Π.Δ.Π. δίνει τη δυνατότητα χρησιμοποίησης της βιβλιοθήκης Smarty, για διαχωρισμό του επιπέδου της παρουσίασης από τον κώδικα υλοποίησης⁸ του module. Αυτό γίνεται με την χρήση templates, τα οποία συνήθως βρίσκονται σαν υποκατάλογοι στον κατάλογο templates.

Αυτό καθιστά το Π.Δ.Π. ιδιαίτερα επεκτάσιμο, αφού μπορούν να ενσωματωθούν νέες λειτουργίες με ελάχιστη έως καθόλου μετατροπή του υπάρχοντος κώδικα.

⁸ Ελεύθερη μετάφραση του Αγγλικού όρου Business Logic.

Παράρτημα Δ

Γλωσσάρι

Δήλωση Προτιμήσεων

Περιλαμβάνει όλες τις προτιμήσεις για όλα τα Εργαστηριακά Μαθήματα που θέτει ένας σπουδαστής. Όταν χρησιμοποιείται ο πληθυντικός αφορά τις δηλώσεις προτιμήσεων περισσότερων του ενός σπουδαστών.

Διαχειριστής Συστήματος

Πρόσωπο που διαχειρίζεται γενικά χαρακτηριστικά όπως ημερομηνίες/ώρες κληρώσεων, ομάδες ειδικών προτεραιοτήτων και έχει πρόσβαση σε στατιστικά που απορρέουν από τη διαδικασία κληρώσεων.

Έμπειρο Σύστημα Κληρώσεων (Ε.Σ.Κ.)

Το σύστημα που αναλαμβάνει να διεκπεραιώσει τις κληρώσεις. Βασίζεται στην τεχνολογία των Εμπείρων Συστημάτων. Περισσότερες πληροφορίες στο αντίστοιχο κεφάλαιο.

Εργαστηριακό Μάθημα

Είναι το εργαστηριακό μέρος ενός μαθήματος (το θεωρητικό μέρος μπορεί να αναφέρεται ως Θεωρητικό Μάθημα ή απλά Θεωρία). Ένα Εργαστηριακό μάθημα περιλαμβάνει έναν αριθμό από Εργαστηριακά Τμήματα.

Εργαστηριακό Τμήμα

Ένα από τα τμήματα που περιλαμβάνεται σε εργαστηριακό μάθημα. Έχει συγκεκριμένη χωρητικότητα, θέση στο ωρολόγιο πρόγραμμα και καθηγητή.

Καθηγητής

Πρόσωπο που διαχειρίζεται ένα ή περισσότερα Εργαστηριακά Μαθήματα. Έχει δικαίωμα να ορίζει παραμέτρους των Εργαστηριακών Τμημάτων που ανήκουν σε αυτό όπως πχ. το όνομα και τη χωρητικότητα.

Κλήρωση

Η διαδικασία τοποθέτησης των σπουδαστών σε εργαστηριακά τμήματα με βάση τις προτιμήσεις τους και την πολιτική του συστήματος.

Ομάδα Ειδικής Προτεραιότητας (Ο.Ε.Π.)

Ένα σύνολο από σπουδαστές που έχουν προτεραιότητα στην κατοχύρωση θέσεων στα εργαστηριακά τμήματα σε σχέση με τους υπόλοιπους.

Περιβάλλον Δήλωσης Προτιμήσεων (Π.Δ.Π.)

Δικτυακή εφαρμογή που επιτρέπει στους σπουδαστές να δημιουργούν τη Δήλωση Προτιμήσεων τους και επιπλέον παρέχει διαχειριστικές δυνατότητες σε Καθηγητές και Διαχειριστές. Περισσότερες πληροφορίες στο αντίστοιχο κεφάλαιο.

Προτίμηση

Η σειρά με την οποία θέλει ένας σπουδαστής να εγγραφεί σε ένα Εργαστηριακό Τμήμα.

Σπουδαστής

Χρήστης που είναι εγγεγραμμένος σε τουλάχιστον ένα Εργαστηριακό Μάθημα και έχει δικαίωμα να πραγματοποιήσει Δήλωση Προτιμήσεων ή εγγραφή σε τμήμα παλαιών σπουδαστών.

Ωρολόγιο Πρόγραμμα

Δισδιάστατος πίνακας που περιέχει τις ώρες και τις ημέρες που διδάσκονται τα θεωρητικά και Εργαστηριακά Μαθήματα. Το ωρολόγιο πρόγραμμα είναι ορατό από το Περιβάλλον Δήλωσης Προτιμήσεων.