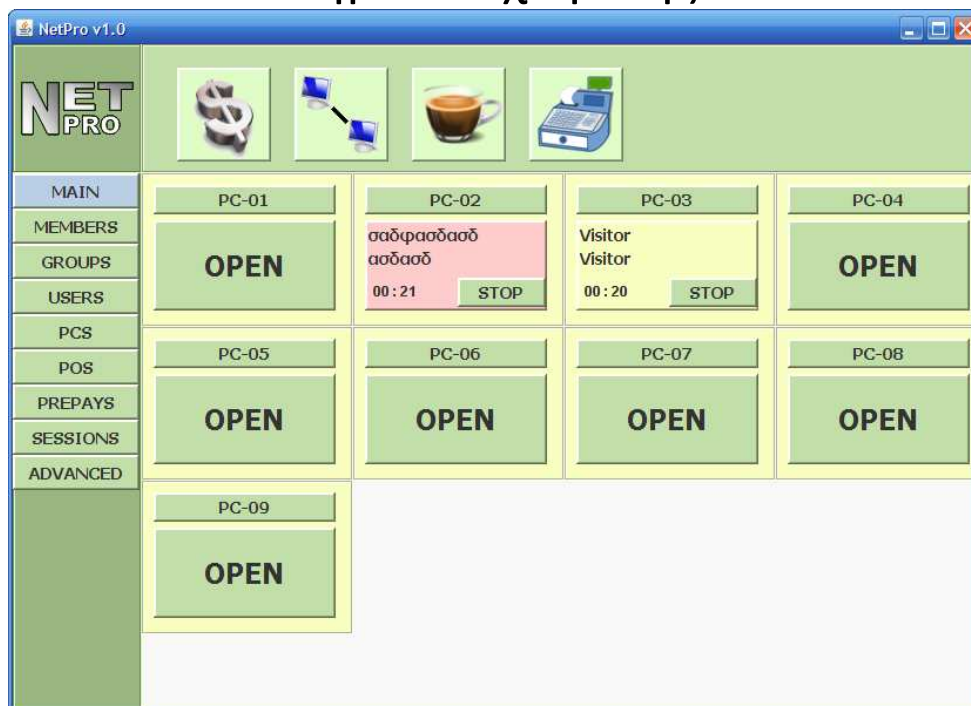


ΑΛΕΞΑΝΔΡΕΙΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ  
ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

“Net-Pro: Σύστημα Διαχείρισης Internet Cafe”



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ  
της  
ΕΛΕΝΗΣ ΧΡΑΠΑΝΑ

Επιβλέπων Καθηγητής: Αδαμίδης Παναγιώτης

Θεσσαλονίκη 2009

# ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ .....	6
ΕΙΣΑΓΩΓΗ.....	10
ΚΕΦΑΛΑΙΟ 1 .....	13
1.1 ΕΜΦΑΝΙΣΗ ΚΑΙ ΕΞΑΠΛΩΣΗ ΤΩΝ INTERNET CAFÉ ΣΤΗΝ ΕΛΛΑΔΑ .....	14
1.2 ΤΑ INTERNET CAFÉ ΣΤΟ ΕΞΩΤΕΡΙΚΟ .....	16
1.3 ΚΑΤΗΓΟΡΙΕΣ ΣΥΓΧΡΟΝΩΝ ΠΡΟΓΡΑΜΜΑΤΩΝ ΧΡΕΩΣΗΣ .....	17
1.4 ΣΥΓΧΡΟΝΑ ΠΡΟΓΡΑΜΜΑΤΑ ΧΡΕΩΣΗΣ .....	18
1.4.1 CAFEPRO .....	18
1.4.2 NETCREDIT .....	20
1.4.3 NETCOUNT.....	21
ΚΕΦΑΛΑΙΟ 2 .....	23
2.1 JAVA – ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ .....	24
2.2 ΑΠΟ ΤΗΝ ΟΑΚ ΣΤΗΝ JAVA.....	24
2.3 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΗΣ JAVA .....	25
2.4 Η ΕΙΚΟΝΙΚΗ ΜΗΧΑΝΗ ΤΗΣ JAVA .....	26
2.5 Ο ΣΥΛΛΕΚΤΗΣ ΣΚΟΥΠΙΔΙΩΝ (GARBAGE COLLECTOR) .....	27
2.6 ΕΠΙΔΟΣΕΙΣ .....	28
2.7 JAVA ΚΑΙ C++.....	28
2.8 Η ΣΥΜΒΟΛΗ ΤΗΣ JAVA ΣΤΟ INTERNET .....	29
2.9 ΝΗΜΑΤΑ.....	29
2.10 JAVA ΚΑΙ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ .....	30
2.11 SOCKETS.....	31
ΚΕΦΑΛΑΙΟ 3 .....	33
3.1 ΟΡΓΑΝΩΣΗ ΤΟΥ INTERNET CAFÉ .....	34
3.1.1 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ INTERNET CAFÉ .....	34
3.1.2 ΡΟΛΟΙ.....	35
3.2 ΚΑΤΑΣΤΑΣΗ ΔΙΑΧΕΙΡΙΣΤΗ – ADMIN MODE .....	36
3.2.1 ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΤΑΣΤΑΣΗΣ ΔΙΑΧΕΙΡΙΣΤΗ .....	36
3.3 ΚΑΤΑΣΤΑΣΗ ΧΡΗΣΤΗ – USER MODE .....	39
3.3.1 ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΤΑΣΤΑΣΗΣ ΧΡΗΣΤΗ.....	39
3.4 ΚΑΤΑΣΤΑΣΗ ΙΣΧΥΡΟΥ ΧΡΗΣΤΗ – POWER USER MODE .....	41
ΚΕΦΑΛΑΙΟ 4 .....	42
4.1 Η ΚΛΑΣΗ OnlineServerData .....	43
4.2 Η ΚΛΑΣΗ Connection .....	43
4.3 Η ΚΛΑΣΗ Server .....	44
4.4 Η ΚΛΑΣΗ Action.....	45

4.5	Η ΚΛΑΣΗ ServerThread.....	46
4.5	ΤΟ ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ .....	46
ΚΕΦΑΛΑΙΟ 5 .....		
5.1	Η ΚΛΑΣΗ Main.....	49
5.2	Η ΚΛΑΣΗ LoginFrame .....	49
5.3	Η ΚΛΑΣΗ MainFrame .....	50
5.4	Η ΚΛΑΣΗ OverviewPanel .....	51
5.4.1	Η ΚΛΑΣΗ AddPrepayDialog .....	52
5.4.2	Η ΚΛΑΣΗ SwitchDialog.....	53
5.4.3	Η ΚΛΑΣΗ AddPosDialog .....	54
5.4.4	Η ΚΛΑΣΗ CloseSessionDialog .....	55
5.4.5	Η ΚΛΑΣΗ RunCommandsDialog .....	55
5.5	Η ΚΛΑΣΗ PcPanel.....	56
5.5.1	Η ΚΛΑΣΗ ClosedInformationPanel .....	57
5.5.1.1	Η ΚΛΑΣΗ ChargeInitDialog .....	58
5.5.2	Η ΚΛΑΣΗ OpenChargeInformationPanel .....	58
5.5.2.1	Η ΚΛΑΣΗ CloseChargeDialog .....	59
5.5.3	Η ΚΛΑΣΗ PrepayChargeInformationPanel .....	60
5.5.4	Η ΚΛΑΣΗ DisabledInformationPanel .....	61
5.6	Η ΚΛΑΣΗ MembersOverviewPanel.....	61
5.6.1	Η ΚΛΑΣΗ MemberAddDialog.....	63
5.6.2	Η ΚΛΑΣΗ MemberEditDialog.....	63
5.7	Η ΚΛΑΣΗ GroupsOverviewPanel .....	64
5.7.1	Η ΚΛΑΣΗ GroupAddDialog .....	65
5.7.2	Η ΚΛΑΣΗ GroupEditDialog .....	66
5.8	ΟΙ ΥΠΟΛΟΙΠΕΣ ΚΛΑΣΕΙΣ ~OverviewPanel .....	66
5.9	Η ΚΛΑΣΗ SessionsOverviewPanel .....	67
5.10	Η ΚΛΑΣΗ AdvancedOverviewPanel .....	68
5.11	Η ΚΛΑΣΗ ColorConfig .....	69
5.12	Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΤΟΥ NET-PRO.....	69
5.13	ΤΑ ΔΙΑΓΡΑΜΜΑΤΑ ΚΛΑΣΕΩΝ .....	74
ΚΕΦΑΛΑΙΟ 6 .....		
6.1	ΚΛΕΙΔΩΜΕΝΗ ΚΑΤΑΣΤΑΣΗ .....	78
6.2	ΚΑΤΑΣΤΑΣΗ ΑΝΟΙΧΤΗΣ ΣΥΝΟΔΟΥ .....	79
ΕΠΙΛΟΓΟΣ .....		
		80
ΠΑΡΑΡΤΗΜΑ Α.....		
		82
ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ .....		
		82
ΒΗΜΑ 1° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ MYSQL.....		
		82
ΒΗΜΑ 2° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ JAVA RUNTIME ENVIRONMENT.....		
		83
ΒΗΜΑ 3° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ NET-PRO SERVER .....		
		83
ΒΗΜΑ 4° – ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΒΑΣΗΣ ΤΟΥ NET-PRO .....		
		83
ΒΗΜΑ 5° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ NET-PRO CLIENT .....		
		83
ΧΡΗΣΗ 84		
1.	ΣΥΝΔΕΣΗ ΣΕ ΚΑΤΑΣΤΑΣΗ ΔΙΑΧΕΙΡΙΣΤΗ .....	84

2. ΣΥΝΔΕΣΗ ΣΕ ΚΑΤΑΣΤΑΣΗ ΑΠΛΟΥ ΚΑΙ ΙΣΧΥΡΟΥ ΧΡΗΣΤΗ .....	85
3. ΟΙ ΛΕΙΤΟΥΡΓΙΕΣ ΤΟΥ OVERVIEW PANEL .....	86
3.1. ΚΛΕΙΣΙΜΟ ΣΥΝΟΔΟΥ ΧΡΗΣΤΗ .....	86
3.2. ΧΡΕΩΣΗ ΜΕ ΠΡΟΠΛΗΡΩΜΗ .....	87
3.3. ΠΩΛΗΣΗ ΠΡΟΪΟΝΤΩΝ .....	89
3.4. ΕΠΟΠΤΕΙΑ ΥΠΟΛΟΓΙΣΤΩΝ ΠΕΛΑΤΗ .....	90
3.5. ΕΝΑΡΞΗ ΝΕΑΣ ΣΥΝΟΔΟΥ ΧΡΗΣΤΗ.....	93
3.6. ΜΕΤΑΦΟΡΑ Η ΑΝΤΑΛΛΑΓΗ.....	94
4. MEMBERS OVERVIEW PANEL.....	95
4.1. ΠΡΟΣΘΗΚΗ – ΜΕΤΑΤΡΟΠΗ .....	96
4.2. ΑΝΑΖΗΤΗΣΗ ΜΕΛΟΥΣ .....	98
4.3. ΔΙΑΓΡΑΦΗ ΜΕΛΟΥΣ .....	99
4.4. ΦΙΛΤΡΟ ΔΙΑΓΡΑΜΜΕΝΩΝ .....	100
5. GROUPS OVERVIEW PANEL.....	100
6. USERS OVERVIEW PANEL .....	101
7. PC'S OVERVIEW PANEL .....	102
8. POS OVERVIEW PANEL.....	104
9. PREPAYS OVERVIEW PANEL .....	105
10. SESSIONS OVERVIEW PANEL .....	107
11. ADVANCED OVERVIEW PANEL .....	109
12. ΟΙ ΕΝΤΟΛΕΣ ΠΡΟΝ ΤΟΝ NET-PRO CLIENT.....	110



# ΠΡΟΛΟΓΟΣ

---

*Λίγα λόγια*

Με αφορμή την κατακόρυφη εξάπλωση των καταστημάτων Internet Café τόσο στην Ελλάδα όσο και στο εξωτερικό, δημιουργήθηκε αυτή η πτυχιακή που αφορά στην ανάπτυξη μιας εφαρμογής για τη διαχείριση ενός τέτοιου καταστήματος. Αυτό περιλαμβάνει τόσο δυνατότητες, όπως η χρονοχρέωση υπολογιστών, όσο και εργαλεία για τη γενικότερη οργάνωση και διαχείριση του καταστήματος. Στις μέρες μας η ύπαρξη ενός τέτοιου συστήματος είναι ζωτικής σημασίας για τις συγκεκριμένες επιχειρήσεις.

Οι βασικότερες λειτουργίες ενός τέτοιου συστήματος περιλαμβάνουν την χρονοχρέωση με απόλυτη ακρίβεια του κάθε υπολογιστή πελάτη με αποτέλεσμα την ευκολότερη διαχείριση του ταμείου από τον υπάλληλο, τον

απόλυτο έλεγχο των συστατικών στοιχείων ενός τέτοιου καταστήματος όπως για παράδειγμα των μελών και των χρηστών, καθώς επίσης δυνατότητες καταγραφής πωλήσεων προϊόντων από το σύστημα.

Η προσθήκη δυνατότητας κλειδώματος των υπολογιστών πελάτη είναι ένα γεγονός που εκτόξευσε τη χρηστικότητα αυτών των συστημάτων με κατακόρυφο τρόπο. Χωρίς αυτή τη δυνατότητα, ο υπάλληλος έπρεπε να έχει μονίμως στραμμένη την προσοχή του στους, τότε ξεκλειδωτους και ελεύθερα προσβάσιμους, υπολογιστές πελάτη έτσι ώστε να αποτρέπει την παράνομη χρήση τους από τους πελάτες. Από τη στιγμή που τα προγράμματα χρέωσης ανέλαβαν αυτή τη δουλειά, η λειτουργικότητα του καταστήματος και η διαχείριση του από τον υπάλληλο γίνεται με μεγαλύτερη ευκολία.

Στα είκοσι περίπου χρόνια ζωής των καταστημάτων Internet Café, απαραίτητο συμπλήρωμά τους αποτελεί ένα σύστημα για τη διαχείρισή τους. Αυτό το σύστημα διαχείρισης πέρασε από πολλά στάδια, ξεκινώντας από ένα απλό τετράδιο που κατέγραφε το ονοματεπώνυμο των πελατών και το χρόνο χρήσης του υπολογιστή που αντιστοιχούσε σ'αυτόν τον πελάτη και καταλήγοντας στην υπάρχουσα σύγχρονη και πολυσύνθετη μορφή. Μιας τέτοιας μορφής είναι και το σύστημα που μελετάμε. Οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξή του είναι η γλώσσα προγραμματισμού Java, το περιβάλλον προγραμματισμού NetBeans και η γλώσσα διαχείρισης βάσεων δεδομένων SQL, με τη χρήση της οποίας δημιουργήθηκε η βάση δεδομένων της εφαρμογής, «internetCafeDB».

# **ABSTRACT**

---

The enormous spread of the Internet Café shops in Greece, as well as in many other countries, led to the need of creating an application for the management of this kind of shops. This includes capabilities such as time-based charges of the customer PC use and tools that help organizing and managing a shop like this. In these days, the existence of such an application is vital for an Internet Café.

The fundamental functions of such a system are the accurate time-based charge of the customer's PCs which results in the easier management of the cashier by the employee, the total control of every element that composes the shop, such as the members and the users, and finally the capability to record the sales of any type of post-on-sale item of the shop.

The addition of the capability to lock the customer computers boosted the use of such applications. Without this capability the employee would have



to pay all of his attention to the customer computers, that used to be unlocked and easily accessed by everyone weather they had the right to use them or not, in order to avoid a possible illegal use of them by the customers. From the moment that the applications undertook this responsibility, the management of an Internet Café became much easier.

During the twenty last years that these shops exist, the Internet Café management applications are an important supplement to them. The management system went through a lot of stages, starting as a simple notepad that included the customers' name and the amount of money that they had to pay up to the current composite form. The application that we are studying in this final year thesis was developed using the Java Programming Language.

# ΕΙΣΑΓΩΓΗ

---

## *Γενική Παρουσίαση*

Προτού περάσουμε στην ανάλυση του αντικειμένου που μελετάμε σε αυτή την εργασία είναι απαραίτητο να προσεγγίσουμε τις αιτίες που οδήγησαν στην ανάγκη για τη δημιουργία του. Η πρόοδος στον τομέα της πληροφορικής ιδιαίτερα τις τρεις τελευταίες δεκαετίες είναι γεωμετρική. Το γεγονός αυτό ήταν καταλυτικό στην διάδοση των υπολογιστών και η χρήση τους, από μία μειονότητα ανθρώπων επεκτάθηκε σε ολόκληρη την κοινωνία και έχει γίνει πλέον τρόπος ζωής και αναπόσπαστο κομμάτι της καθημερινότητας μας.

Φυσικά, αυτό ισχύει για κάθε ηλικιακή κατηγορία. Τις μικρές ηλικίες προσελκύουν τα παιχνίδια που βασίζονται στους υπολογιστές. Εδώ και χρόνια

έχουν ιδρυθεί εταιρίες για αυτό το σκοπό(ανάπτυξη παιχνιδιών) η εξάπλωση των οποίων είναι πρωτοφανής. Παράδειγμα αποτελούν οι NcSoft και Blizzard, δύο από τις μεγαλύτερες εταιρίες του χώρου. Τα χρηματικά ποσά που διακινούνται στον τομέα των παιχνιδιών είναι υπέρογκα. Το αποτέλεσμα που παράγουν, αν αναλογιστούμε τους πόρους πληροφορικής(υλικό, λογισμικό, παροχή συνδέσεων μεγάλης ταχύτητας, ισχυροί servers) που έχουν στη διάθεση τους, μπορεί να εντυπωσιάσει και να προσελκύσει ακόμα κ μεγαλύτερες ηλικίες! Πέρα όμως από το χώρο των παιχνιδιών, είναι δεδομένο ότι οι περισσότεροι ενήλικες έχουν υιοθετήσει τη χρήση του ηλεκτρονικού ταχυδρομείου, του διαδικτύου για διάφορους λόγους, όπως η αναζήτηση πληροφοριών, και για ποικίλες άλλες εφαρμογές πληροφορικής.

Λογική συνέχεια όσων προαναφέρθηκαν είναι και η εμφάνιση των καταστημάτων Internet Café τα οποία δημιουργήθηκαν για να προσφέρουν όλες αυτές τις υπηρεσίες για τις οποίες υπάρχει ζήτηση σε καθημερινή βάση από κάθε ηλικία. Τα καταστήματα αυτά έχουν κατορθώσει να καλύψουν κάθε ψηφιακή ανάγκη που είναι πιθανόν να προκύψει και για το λόγο αυτό παρουσιάζουν τεράστια επιτυχία ως επιχειρήσεις.

Στα κεφάλαια που ακολουθούν περιγράφεται αναλυτικά η εφαρμογή που αναπτύχθηκε στα πλαίσια της πτυχιακής εργασίας αναλύοντας τις λειτουργίες και τις δυνατότητές της. Με αφορμή τη συγκεκριμένη εφαρμογή, κάνουμε μια ιστορική αναδρομή των καταστημάτων Internet Café και έπειτα παρουσιάζουμε τη σύγχρονη φάση αυτών των καταστημάτων τόσο στην Ελλάδα όσο και στο εξωτερικό. Στην συνέχεια γίνεται μία αναφορά σε συστήματα διαχείρισης που χρησιμοποιούνται στις μέρες μας.

Λόγω του ότι η εφαρμογή που εξετάζουμε αναπτύχθηκε σε γλώσσα προγραμματισμού Java, αναφέρονται τα σπουδαιότερα δομικά στοιχεία της γλώσσας αυτής καθώς και ειδικότερα χαρακτηριστικά που χρησιμοποιήθηκαν για να υλοποιηθεί. Τα χαρακτηριστικά τα οποία παρουσιάζουν ιδιαίτερο ενδιαφέρον και έδωσαν ουσιαστικές λύσεις στα προβλήματα που προέκυψαν κατά την ανάπτυξη της εφαρμογής μας είναι τα νήματα(threads) και τα sockets. Μετά την εκτενή ανάλυση όσων προαναφέρθηκαν, γίνεται μία αναφορά στις ομοιότητες της Java με τη C++, στην συμβολή της Java στο

διαδίκτυο και στον τρόπο με τον οποίο η Java μπορεί να ενσωματώσει βάσεις δεδομένων.

Στη συνέχεια προχωράμε στην οργάνωση και τα χαρακτηριστικά που απαρτίζουν ένα κατάστημα Internet Café καθώς και στις λεγόμενες καταστάσεις λειτουργίας που αποτέλεσαν τον βασικό κορμό πάνω στον οποίο στηρίχθηκε το πρόγραμμά μας κατά την ανάπτυξή του. Η εφαρμογή μας είναι οργανωμένη πάνω σε τρεις καταστάσεις λειτουργίας. Την κατάσταση διαχειριστή, την κατάσταση απλού χρήστη και την κατάσταση δυναμικού χρήστη. Κάθε μια από αυτές τις καταστάσεις έχουν διακριτά και κλιμακούμενα δικαιώματα για πρόσβαση στο σύστημα. Έτσι επιτυγχάνεται καλύτερη εποπτεία της επιχείρησης καθώς επίσης και ξεκάθαρα κατανοημένα καθήκοντα για οποιονδήποτε έρχεται σε επαφή με το σύστημα.

Τέλος, το μεγαλύτερο βάρος δίνεται στην εξέταση του συστήματός μας, κομμάτι κομμάτι, επισημαίνοντας όλα τα απαραίτητα στοιχεία, χαρακτηριστικά και λεπτομέρειες που το καθορίζουν. Χωρίζεται σε δύο υποσυστήματα, ένα το οποίο εκτελείται στον κεντρικό υπολογιστή διαχείρισης συστήματος και ένα το οποίο εκτελείται σε κάθε έναν από τους υπολογιστές πελάτη. Το πρώτο από τα δύο έχει και άμεση πρόσβαση στη βάση δεδομένων του καταστήματος. Το δεύτερο δεν έχει άμεση πρόσβαση για λόγους ασφάλειας. Τα δύο αυτά υποσυστήματα επικοινωνούν μεταξύ τους και αλληλοσυμπληρώνονται καλύπτοντας οποιαδήποτε ανάγκη ενδεχομένως παρουσιαστεί. Το υποσύστημα που εκτελείται στον κεντρικό υπολογιστή ονομάζεται Net-Pro Server και είναι το εργαλείο που χρησιμοποιεί ο διαχειριστής και ο υπάλληλος του καταστήματος. Το υποσύστημα που εκτελείται στους υπολογιστές πελάτη ονομάζεται Net-Pro Client. Οι πελάτες μπορεί να έρθουν σε άμεση επαφή με αυτό (Net-Pro Client), αν και το γεγονός αυτό δεν είναι απαραίτητο για την ομαλή λειτουργία του καθολικού μας συστήματος.

# ΚΕΦΑΛΑΙΟ 1

---

*Ta internet café στον παρελθόν και στις μέρες μας*

Στο κεφάλαιο που ακολουθεί γίνεται μία πρώτη γνωριμία με τα καταστήματα Internet Café και το πώς αυτά διαμορφώθηκαν με την πάροδο του χρόνου, ξεκινώντας από τη δεκαετία του '80 που πρωτοεμφανίστηκαν και καταλήγοντας στη σημερινή τους μορφή. Δίνεται μια εικόνα της διάδοσής τους στο εξωτερικό και, τέλος, αναφέρονται μερικά από τα πιο διαδεδομένα συστήματα χρέωσης που εξυπηρετούν τέτοιου είδους καταστήματα και οι δύο κατηγορίες στις οποίες ανήκουν.

## **1.1 ΕΜΦΑΝΙΣΗ ΚΑΙ ΕΞΑΠΛΩΣΗ ΤΩΝ INTERNET CAFÉ ΣΤΗΝ ΕΛΛΑΔΑ**

Τα internet café πρωτοεμφανίστηκαν στην Ελλάδα γύρω στο 1985 με μορφή πολύ διαφορετική από τη σημερινή. Ήταν συνήθως μικρά μαγαζιά ηλεκτρονικών ειδών στεγασμένα σε λίγα τετραγωνικά μέτρα που κυρίως ασχολούνταν με πώληση Υ/Η και λογισμικού και παρείχαν ως επιπλέον υπηρεσία τη χρήση υπολογιστών που ανήκαν στο κατάστημα με χρηματικό αντίτιμο. Η συντριπτική πλειοψηφία των πελατών ήταν μικρής ηλικίας και η κυριότερη χρήση των υπολογιστών ήταν τα ηλεκτρονικά παιχνίδια που είχαν κυκλοφορήσει την περίοδο εκείνη.(Amstrad, commodore 64, spectrum). Με την πάροδο του χρόνου και με την ανάπτυξη της τεχνολογίας εμφανίστηκαν νέοι υπολογιστές με περισσότερες δυνατότητες και με περισσότερα και πιο εντυπωσιακά παιχνίδια. Συγκεκριμένα στα τέλη της δεκαετίας του 80 εμφανίστηκε ο νέος υπολογιστής της commodore, η Amiga commodore 500 η οποία άλλαξε ριζικά τα δεδομένα στην ιστορία των παιχνιδιών και έπαιξε καθοριστικό ρόλο στην ανάπτυξη και εξάπλωση των Internet Cafe.

Σιγά σιγά, τα μαγαζιά ηλεκτρονικών ειδών άρχισαν να μετατρέπονται κ σε ανεξάρτητους και μεγαλύτερους χώρους ψυχαγωγίας με στραμμένο πλέον το ενδιαφέρον στην παροχή παιχνιδιών και όχι στο σέρβις και στην πώληση λογισμικού. Ενδεικτικά αναφέρονται τα παρακάτω παιχνίδια που ήταν τα πρώτα «mass played games» : Street Fighter, Mortal Combat, Sensible Soccer .

Παράλληλα με την ανάπτυξη των Internet Cafe, αναπτύσσονταν συνεχώς και τα καταστήματα ηλεκτρονικών παιχνιδιών τα οποία ευδοκίμούσαν την εποχή εκείνη κ παρείχαν καλύτερα παιχνίδια. Η κύρια διαφορά ανάμεσα στα Internet Cafe και στα καταστήματα ηλεκτρονικών παιχνιδιών ήταν ο τρόπος χρέωσης. Στα καταστήματα ηλεκτρονικών παιχνιδιών για να παίξει κάποιος ένα παιχνίδι έπρεπε να εισάγει ένα κέρμα με το οποίο έπαιρνε κάποια πίστωση. Με κάθε πίστωση μπορούσε να παίξει μέχρι να χάσει. Στα Internet

Cafe η χρέωση ήταν ανάλογη με το χρόνο. Εκτός αυτού στα καταστήματα ηλεκτρονικών παιχνιδιών κάθε παιχνίδι ήταν μια ανεξάρτητη συσκευή και συνήθως για κάθε παιχνίδι υπήρχε μόνο μια συσκευή με αποτέλεσμα να μην μπορούν 2 άτομα να παίζουν ταυτόχρονα το ίδιο παιχνίδι και να δημιουργούνται ατέλειωτες ουρές. Ενδεικτικά, για το παιχνίδι Wonder Boy in Monsterland η ουρά μπορούσε να φτάσει μέχρι και δεκάδες ατόμων και η αναμονή να ξεπεράσει τη μια ώρα! Αντίθετα στα «Internet Cafe» της εποχής μπορούσες σε κάθε σταθμό εργασίας ανά πάσα στιγμή να φορτώσεις το παιχνίδι που ήθελες.

Η νέα γένια των «Internet Cafe» ήρθε με την ανακάλυψη και την ευρεία χρήση των cd. Η νέας γενιάς κονσόλες βασισμένες στην τεχνολογία των cd και με επεξεργαστική δυνατότητα αρκετά μεγάλη πρόσφεραν νέες δυνατότητες στα παιχνίδια τα οποία από δυσδιάστατα εξελίχτηκαν σε τρισδιάστατα. Συγκεκριμένα τα Internet Cafe άρχισαν σταδιακά να αποσύρουν τους παλιούς υπολογιστές κ κονσόλες κ να εξοπλίζονται με τις νέες. Ενδεικτικά οι κυρίως χρησιμοποιούμενες κονσόλες ήταν οι Sony playstation , Sega Saturn και Nintendo 64 με παιχνίδια ευρείας αποδοχής όπως fifa και Mario cart .

Σιγά σιγά άρχισαν να εμφανίζονται ολοένα κ περισσότερα Internet Cafe στα οποία πλέον οι πελάτες δεν ήταν μόνο μικρής ηλικίας. Παράλληλα τα καταστήματα ηλεκτρονικών παιχνιδιών άρχισαν να χάνουν ολοένα και περισσότερο κόσμο και άρχισαν και αυτά να μετατρέπονται σταδιακά σε Internet Cafe.

Με την αναπτυξη των windows 95 και με τη ραγδαία ανάπτυξη των υπολογιστών προσωπικής χρήσης (PC) τα Internet Cafe άρχισαν να εξοπλίζονται με αυτά. Από εκείνη τη στιγμή και μετά άρχισαν να παίρνουν σιγά σιγά τη σημερινή μορφή. Παρόλα αυτά οι υπολογιστές χρησιμοποιούνταν ως ανεξάρτητες μονάδες οι οποίες δεν ήταν συνδεδεμένες μεταξύ τους και τα παιχνίδια που πρόσφεραν ήταν προσωπικού χαρακτήρα(single player). Κάποια από αυτά είναι τα Diablo1, Red alert, heroes of might and magic, age of empires.

Τα επόμενα χρόνια, με την ανάπτυξη των δικτύων οι υπολογιστές άρχισαν να συνδέονται μεταξύ τους και να θυμίζουν ολοένα κ περισσότερο τα Internet Cafe όπως τα ξέρουμε σήμερα. Πλέον τα παιχνίδια έγιναν πιο συναρπαστικά, η διάδοσή τους αρχίζει να γίνεται μεγαλύτερη και οι πελάτες να γίνονται πλέον σταθεροί. Τα πιο διαδεδομένα παιχνίδια της εποχής είναι το Counter Strike και το Starcraft. Τότε πρωτοεμφανίστηκαν και τα προγράμματα χρέωσης. Παρόλα αυτά ο χαρακτήρας των Internet Cafe παρέμεινε οικογενειακός και η ενσωμάτωση υπηρεσιών Internet ήταν ακόμη σε πρώιμο στάδιο λόγω του δυσβάστακτου κόστους για σύνδεση στο Internet(dial up) καθώς και της μικρής ταχύτητας.

Στα επόμενα χρόνια μετά από συνεχή εξάπλωση των Internet Cafe εμφανίστηκαν τα πρώτα τα οποία παρείχαν πλήρεις υπηρεσίες διαδικτύου και πρόσφεραν συνεχή σύνδεση στο Internet και σε καλές ταχύτητες.

Τελικά ο οικογενειακός χαρακτήρας σιγά σιγά εκλείπει δίνοντας τη θέση του στις πρώτες αλυσίδες καταστημάτων Internet Cafe οι οποίες διαθέτουν καταστήματα σε περισσότερες από μια πόλεις και διαθέτουν αρκετές δεκάδες υπολογιστών το καθένα. Ενδεικτικά αναφέρονται οι αλυσίδες The web, Meganet, Interspot. Πλέον πελάτες όλων των ηλικιών επισκέπτονται τα ιντερνετ καφέ τόσο για τα παιχνίδια όσο και για τις υπηρεσίες που προσφέρει το Internet. Σήμερα τα Internet Cafe είναι τόσο διαδεδομένα που σε κάθε γειτονιά υπάρχει και από ένα.

## **1.2 ΤΑ INTERNET CAFÉ ΣΤΟ ΕΞΩΤΕΡΙΚΟ**

Τα Internet Cafe σε κάποιες χώρες του εξωτερικού όπως για παράδειγμα στην Νότια Κορέα είναι τρόπος ζωής. Δεν υπάρχει καφετέρια που να μην παρέχει πρόσβαση στο Internet. Έχουν εξελιχτεί σε τέτοιο βαθμό που πέραν των προβλεπόμενων υπηρεσιών, παρέχουν και άλλες όπως φαγητό ή ακόμα και διανυκτέρευση με μπάνιο. Υπάρχουν ειδικοί χώροι που μπορεί να



κλείσει μια παρέα απόμων για συγκεκριμένες μέρες οι οποίοι διαθέτουν κρεβάτια και υπολογιστές! Αξίζει να αναφερθεί το παράξενο που συμβαίνει βάσει στατιστικών ότι γύρω στους 5000 φοιτητές ζουν σε Internet Cafe καθώς το κόστος ζωής σε ένα Internet Cafe είναι πολύ μικρότερο από την ενοικίαση σπιτιού. Επίσης υπάρχουν ομάδες οι οποίες ονομάζονται «pro gamers» και σπονσοράρονται από χορηγούς για να παίζουν παιχνίδια και να παίρνουν μέρος σε τουρνουά. Οι απολαβές τους ξεπερνούν τις 100 χιλιάδες ευρώ!!!

### **1.3 ΚΑΤΗΓΟΡΙΕΣ ΣΥΓΧΡΟΝΩΝ ΠΡΟΓΡΑΜΜΑΤΩΝ ΧΡΕΩΣΗΣ**

Τα προγράμματα χρέωσης που χρησιμοποιούν τα Internet Café στις μέρες μας διακρίνονται στις ακόλουθες κατηγορίες:

- Απλά προγράμματα με κάρτες.  
Είναι προγράμματα που εγκαθιστώνται αποκλειστικά και μόνο στον υπολογιστή-ταμείο του καταστήματος. Οι υπολογιστές που προορίζονται για χρήση από τους πελάτες είναι ξεκλειδωμένοι και απόλυτα προσβάσιμοι ανά πάσα στιγμή από αυτούς. Κάθε φορά που ένας πελάτης ζητάει να χρησιμοποιήσει έναν υπολογιστή, ο υπάλληλος του δίνει μια κάρτα που αντιστοιχεί στο συγκεκριμένο υπολογιστή που ουσιαστικά καθορίζει την άδεια χρήσης του υπολογιστή από το συγκεκριμένο πελάτη. Ένα τέτοιου είδους πρόγραμμα χρέωσης χρησιμοποιεί και η μεγάλη αλυσίδα καταστημάτων Internet Café "The Web".
- Προγράμματα που διαθέτουν client-server λειτουργίες έτσι ώστε οι υπολογιστές πελατών να κλειδώνονται και να ξεκλειδώνονται απομακρυσμένα από τον κεντρικό υπολογιστή διαχείρισης του καταστήματος. Είναι σαφώς πιο πολύπλοκα αλλά ταυτόχρονα προσφέρουν και καλύτερο έλεγχο του καταστήματος. Γενικότερα,

διευκολύνουν κατά πολύ τη δουλειά του υπαλλήλου. Κάποια από τα πιο διαδομένα προγράμματα χρέωσης αυτής της κατηγορίας θα παρουσιαστούν παρακάτω. Σε αυτή την κατηγορία ανήκει και η εφαρμογή που αναπτύχθηκε στα πλαίσια της πτυχιακής και παρουσιάζεται στα επόμενα κεφάλαια.

## **1.4 ΣΥΓΧΡΟΝΑ ΠΡΟΓΡΑΜΜΑΤΑ ΧΡΕΩΣΗΣ**

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν μερικά από τα σύγχρονα και πιο διαδεδομένα προγράμματα χρέωσης που χρησιμοποιούνται σήμερα. Αυτά είναι το CafePro, το NetCount και το NetCredit.

### **1.4.1 CAFEPRO**

Είναι κοινά αποδεκτό ως το επικρατέστερο πρόγραμμα χρέωσης που κυκλοφορεί στην αγορά([www.cafepro.com](http://www.cafepro.com)). Περιλαμβάνει τα εξής επιμέρους τμήματα :

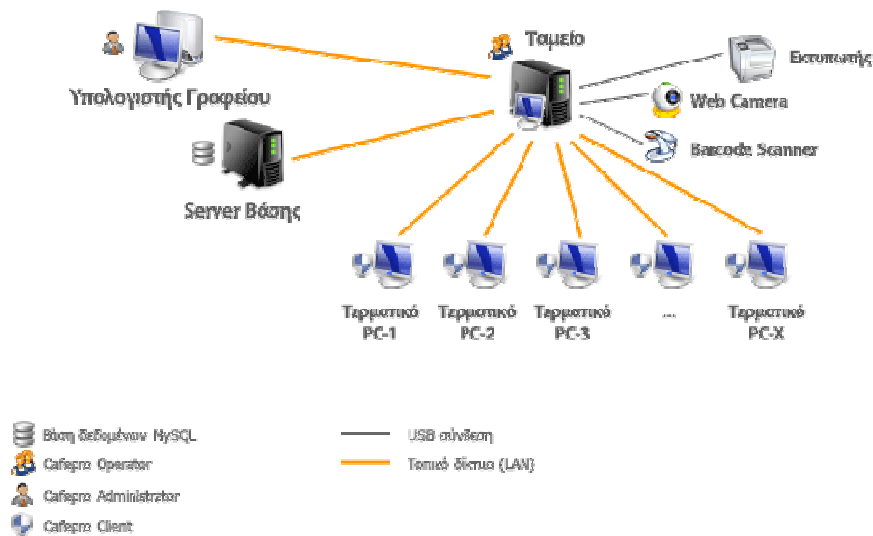
Το **Cafepro Operator** το οποίο κατά βάση το χειρίζονται οι υπάλληλοι της επιχείρησής. Ανοίγουν τους υπολογιστές, χρεώνουν τους πελάτες είτε με υπηρεσίες (χρόνος internet) είτε με προϊόντα (εκτυπώσεις, αναψυκτικά, κ.λπ.), διαχειρίζονται τα μέλη, τις προπληρωμές τους, τις λίστες αναμονής, κ.λπ. Με λίγα λόγια, στο Cafepro Operator περιλαμβάνονται όλες εκείνες οι απαραίτητες εργασίες που πρέπει να γίνονται από τον υπάλληλο της επιχείρησης κατά τη συνήθη λειτουργία του καταστήματος μέσα στη μέρα.

Παράλληλα, υποστηρίζει έλεγχο βαρδιών του προσωπικού ώστε να γίνεται παράδοση/παραλαβή του ταμείου μεταξύ των υπαλλήλων και έτσι να διαπιστώνονται πιθανά λάθη. Θα μπορούσαμε να πούμε ότι παρέχει τις ίδιες δυνατότητες με την κατάσταση χρήστη του NET-PRO, όπως διαπιστώνουμε παρακάτω.

Το **Cafepro Client** σε συνεργασία με το Cafepro Operator, εκτελεί όλες τις απαραίτητες χειροκίνητες ή αυτοματοποιημένες διεργασίες στα τερματικά ώστε τα πάντα να είναι πλήρως διάφανα στον πελάτη. Το κλείδωμα/ξεκλείδωμα των υπολογιστών (εμφάνιση/απόκρυψη της προστατευτικής ασπίδας), τη δυνατότητα για απομακρυσμένη σύνδεση ενός πελάτη απ' ευθείας στο τερματικό με μυστικό κωδικό ή προπληρωμένη κάρτα, την εμφάνιση διαφημιστικών εικόνων όταν το τερματικό είναι κλειδωμένο, την εκκίνηση και το κλείσιμο των υπολογιστών από απόσταση και πολλά άλλα. Επιπρόσθετα, το Cafepro Client αποτελεί την απαραίτητη γέφυρα με το Cafepro Operator για τη διεξαγωγή και πιο σύνθετων διεργασιών, όπως ο προσωπικός φάκελος των μελών που αποθηκεύεται κεντρικά και είναι προσβάσιμος όποιον υπολογιστή κι αν χρησιμοποιεί το μέλος κάθε φορά, αλλά και τη διαχείριση των εκτυπώσεων από τους πελάτες, ενημερώνοντας σε πραγματικό χρόνο το Cafepro Operator για την ακριβή χρέωσή τους, ανάλογα με τον αριθμό και το είδος των εκτυπώσεων που πραγματοποίησαν. Το Cafepro Client είναι το τμήμα του CafePro που το εντάσσει στη δεύτερη κατηγορία που εξετάσαμε προηγουμένως.

Το **Cafepro Administrator** είναι φτιαγμένο αποκλειστικά για τον ιδιοκτήτη του καταστήματος και έχει ως σκοπό την παραμετροποίηση όλων των ρυθμίσεων του προγράμματος χρέωσης. Αντιστοιχεί στην κατάσταση διαχειριστή του NET-PRO που θα εξετάσουμε παρακάτω.[5]

Μια γενική ιδέα για το μοντέλο εγκατάστασης του δίνεται από την παρακάτω εικόνα:

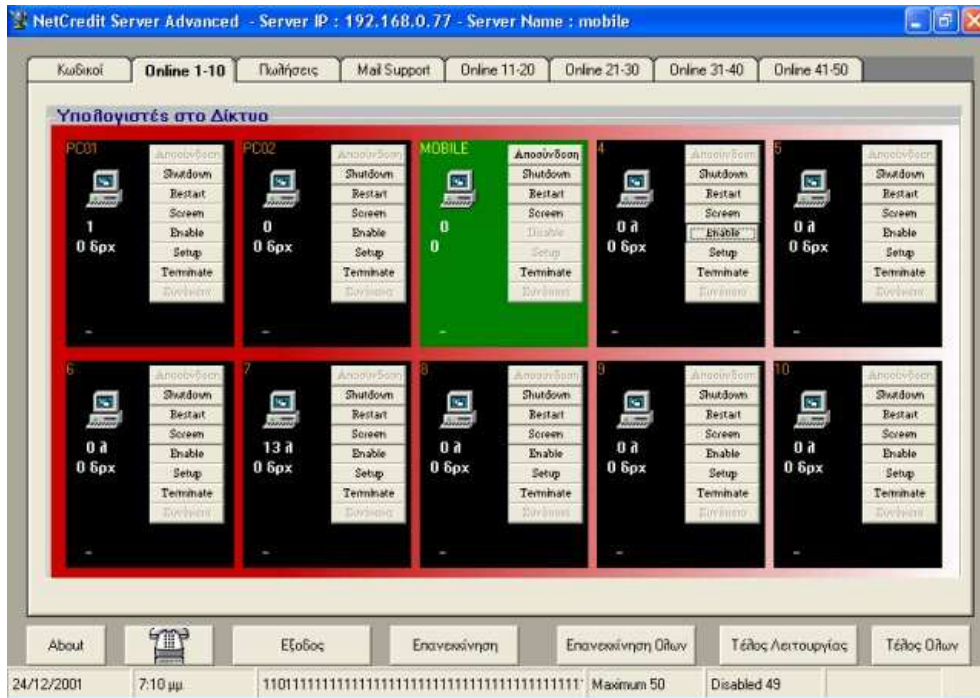


**Εικόνα 1.4.1.1**

## 1.4.2 NETCREDIT

Κυκλοφορεί σε δύο εκδόσεις. (<http://www.lexiconsoftware.gr - /NETCREDIT>) Η απλή έκδοση προορίζεται για μικρά καταστήματα με λίγους υπολογιστές και ανήκει στην πρώτη κατηγορία προγραμμάτων χρέωσης. Η Advanced έκδοση ανήκει στην δεύτερη κατηγορία περιλαμβάνοντας, όπως και το Cafepro, επιμέρους τμήματα Server και Client. Περιλαμβάνει τις λειτουργίες ενός τυπικού προγράμματος χρέωσης.[6]

Στην εικόνα 1.4.2.1 μπορούμε να δούμε τη διασύνδεση του Netcredit και τις βασικές του λειτουργίες.



Εικόνα 1.4.2.1

### 1.4.3 NETCOUNT

Η εφαρμογή Netcount ανήκει και αυτή στην δεύτερη κατηγορία προγραμμάτων χρέωσης. Η κύρια διαφορά της με τα παραπάνω προγράμματα χρέωσης και ταυτόχρονα η κύρια ομοιότητά της με το NET-PRO εντοπίζεται στο γεγονός ότι δεν διαθέτει ξεχωριστά τμήματα για τον ιδιοκτήτη και τους υπαλλήλους αλλά αποτελείται από ένα ενοποιημένο πρόγραμμα. Θα μπορούσαμε να πούμε ότι είναι οργανωμένο με τα κλιμακούμενα δικαιώματα που εξετάζουμε παρακάτω και αφορούν το NET-PRO.

Επιπλέον, είναι το μοναδικό που διαθέτει την πρόσθετη λειτουργία ενός παράθυρου που επιτρέπει την εποπτεία των υπολογιστών ως εικονίδια που αντικατοπτρίζουν τη φυσική τους θέση στο κατάστημα. Όπως είναι φυσικό,

αυτή η απλή, κατά τα άλλα, προσθήκη επιτρέπει στον υπάλληλο να διαχειρίζεται την επιχείρηση με μεγαλύτερη ευκολία.[7]

## **ΚΕΦΑΛΑΙΟ 2**

---

*Χαρακτηριστικά της γλώσσας προγραμματισμού java*

Στο κεφάλαιο που ακολουθεί ασχολούμαστε με την παρουσίαση των σπουδαιότερων στοιχείων της java καθώς επίσης και των χαρακτηριστικών της που βοήθησαν στην υλοποίηση της εφαρμογής NET-PRO.

## **2.1 JAVA – ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ**

Η JAVA είναι μια γλώσσα αντικειμενοστραφούς προγραμματισμού η οποία δημιουργήθηκε από την Sun Microsystems και παρουσιάστηκε της αρχές του 1990. Τα προγράμματα που είναι γραμμένα σε java, όταν μεταγλωττιστούν μετατρέπονται σε Bytecode. Αυτό το χαρακτηριστικό κάνει την JAVA ανεξάρτητη λειτουργικού συστήματος καθώς τα μεταγλωττισμένα προγράμματα τρέχουν πάνω σε κάποια Εικονική Μηχανή(Virtual Machine). Για παράδειγμα, ένα πρόγραμμα java μπορεί να τρέχει και σε Windows και σε Linux. Η Java δεν γεννήθηκε από το πουθενά. Πρωτοσχεδιάστηκε από τους Patrick Naughton, Mike Sheridan και James Gosling το 1991 με την ονομασία «The Green Project». Σκοπός αυτού του project ήταν να δημιουργηθεί μια πλατφόρμα ανάπτυξης λογισμικού για μικροσυσκευές (έως πολύπλοκα συστήματα παραγωγής γραφικών). Τα εργαλεία της εποχής ήταν οι γλώσσες C++ και C. Μετά από κάποιες προσπάθειες συμπέραναν ότι οι υπάρχουσες γλώσσες δεν μπορούσαν να καλύψουν της ανάγκες του project. Ο «πατέρας» της Java, James Gosling, που εργαζόταν εκείνη την εποχή για την Sun, έκανε ήδη πειραματισμούς πάνω στη C++ και είχε παρουσιάσει κατά καιρούς κάποιες πειραματικές γλώσσες ως πρότυπα για το νέο εργαλείο που αναζητούσαν στην Sun. Τελικά, μετά από λίγο καιρό κατέληξαν στη δημιουργία της γλώσσας Oak.[1]

## **2.2 ΑΠΟ ΤΗΝ ΟΑΚ ΣΤΗΝ JAVA**



Η Oak ήταν μία γλώσσα που διατηρούσε μεγάλη συγγένεια με την C++. Παρόλα αυτά είχε πολύ πιο έντονο αντικειμενοστραφή (object oriented) χαρακτήρα σε σχέση με την C++ και χαρακτηριζόταν για την απλότητα της. Σύντομα, οι υπεύθυνοι ανάπτυξης της γλώσσας ανακάλυψαν ότι το όνομα Oak ήταν ήδη κατοχυρωμένο οπότε αποφάσισαν να τη μετονομάσουν σε Java. Η επίσημη εμφάνιση της Java αλλά και του HotJava (πλοηγός με υποστήριξη Java) στη βιομηχανία της πληροφορικής έγινε το Μάρτιο του 1995 όταν η Sun την ανακοίνωσε στο συνέδριο Sun World 1995. Ο πρώτος μεταγλωτιστής (compiler) της ήταν γραμμένος στη γλώσσα C από τον James Gosling. Το 1994, ο A.Van Hoff ξαναγράφει τον μεταφραστή της γλώσσας σε Java (από τα πλέον δύσκολα επιτεύγματα στο χώρο της πληροφορικής είναι να γράψεις έναν μεταφραστή της γλώσσας στην ίδια τη γλώσσα), ενώ το Δεκέμβριο του 1995 πρώτες οι IBM, Borland, Mitsubishi Electronics, Sybase και Symantec ανακοινώνουν τα σχέδια τους, να χρησιμοποιήσουν τη Java για την δημιουργία λογισμικού. Από εκεί και πέρα η Java ακολουθεί μία ανοδική πορεία και είναι πλέον μία από της πιο δημοφιλείς γλώσσες στον χώρο της πληροφορικής. [1]

## **2.3 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΗΣ JAVA**

Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουν με τον ίδιο ακριβώς τρόπο σε Windows, Linux, Unix και Macintosh (σύντομα θα τρέχουν και σε Playstation καθώς και σε παιχνιδιομηχανές) χωρίς να χρειαστεί να ξαναγίνει κάποια μετάφραση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Για να επιτευχθεί αυτό έπρεπε να βρεθεί κάποιος τρόπος έτσι ώστε τα προγράμματα που είναι γραμμένα σε Java

να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή, ανεξάρτητα του είδους επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και του λειτουργικού συστήματος (Windows, Unix, Linux, Unix, MacOS). Κάθε κεντρική μονάδα επεξεργασίας μπορεί και «καταλαβαίνει» διαφορετικό assembly κώδικα. Ο assembly κώδικας που τρέχει σε Windows είναι διαφορετικός από αυτόν που τρέχει σε ένα Macintosh. Η λύση δόθηκε με την ανάπτυξη της Εικονικής Μηχανής (Virtual Machine ή VM ή EM).[1]

## **2.4 Η ΕΙΚΟΝΙΚΗ ΜΗΧΑΝΗ ΤΗΣ JAVA**

Αφού γραφτεί κάποιο πρόγραμμα σε Java τότε γίνεται μετάφραση μέσω του Java compiler (javac), εκείνος με την σειρά του δίνει ένα αριθμό από .class αρχεία (=Bytecode). Το Bytecode είναι η μορφή που παίρνει ο πηγαίος κώδικας της Java μετά τη μετάφραση του προγράμματος. Όταν προσπαθήσουμε λοιπόν να εκτελέσουμε μια εφαρμογή της Java, η εικονική μηχανή που θα είναι εγκατεστημένη στο μηχάνημά μας, θα αναλάβει να διαβάσει τα .class αρχεία και να τα μεταφράσει στη γλώσσα και τις εντολές μηχανής (assembly) που υποστηρίζει το συγκεκριμένο λειτουργικό και ο επεξεργαστής, έτσι ώστε να εκτελεστεί (να σημειώσουμε εδώ ότι αυτό συμβαίνει με την παραδοσιακή Εικονική Μηχανή (Virtual Machine). Πιο σύγχρονες εφαρμογές της εικονικής Μηχανής μπορούν και μεταφράζουν απευθείας το Bytecode, σε εγγενή κώδικα (native code) με αποτέλεσμα να βελτιώνεται η ταχύτητα. Χωρίς αυτό δε θα ήταν δυνατή η εκτέλεση λογισμικού γραμμένου σε Java. Πρέπει να πούμε ότι η εικονική μηχανή είναι ένα λογισμικό για το οποίο, για κάθε είδος λειτουργικού και ανάλογης τεχνολογίας επεξεργαστή, υπάρχει διαφορετική έκδοση. Έτσι υπάρχουν διαθέσιμες εκδόσεις του για Windows, Linux, Unix, Macintosh, κινητά τηλέφωνα, παιχνιδομηχανές κλπ. Οτιδήποτε θέλει να κάνει ο προγραμματιστής (ή ο χρήστης) γίνεται μέσω της εικονικής μηχανής. Αυτό

βοηθάει στο να υπάρχει μεγαλύτερη ασφάλεια στο σύστημα γιατί η εικονική μηχανή είναι υπεύθυνη για την επικοινωνία χρήστη – υπολογιστή. Ο προγραμματιστής δεν μπορεί να γράψει κώδικα ο οποίος θα έχει καταστροφικά αποτελέσματα για τον υπολογιστή γιατί η εικονική μηχανή θα τον ανιχνεύσει και δε θα επιτρέψει να εκτελεστεί. Από την άλλη μεριά ούτε ο χρήστης μπορεί να κατεβάσει «κακό» κώδικα από το δίκτυο και να τον εκτελέσει. Αυτό είναι ιδιαίτερα χρήσιμο για μεγάλα καταναμεμημένα συστήματα όπου πολλοί χρήστες χρησιμοποιούν το ίδιο πρόγραμμα συγχρόνως.[1]

## **2.5 Ο ΣΥΛΛΕΚΤΗΣ ΣΚΟΥΠΙΔΙΩΝ (GARBAGE COLLECTOR)**

Ακόμα μία ιδέα που βρίσκεται πίσω από τη JAVA είναι η ύπαρξη του συλλέκτη σκουπιδιών (Garbage Collector). Συλλογή σκουπιδιών είναι μία κοινή ονομασία που χρησιμοποιείται στον τομέα της πληροφορικής για να δηλώσει την ελευθέρωση τμημάτων μνήμης από δεδομένα που δε χρειάζονται και δε χρησιμοποιούνται άλλο. Αυτή η απελευθέρωση μνήμης στη Java είναι αυτόματη και γίνεται μέσω του συλλέκτη σκουπιδιών. Υπεύθυνη για αυτό είναι και πάλι η εικονική μηχανή η οποία μόλις «καταλάβει» ότι ο σωρός (heap) της μνήμης (στη Java η συντριπτική πλειοψηφία των αντικειμένων αποθηκεύονται στο σωρό σε αντίθεση με τη C++ όπου αποθηκεύονται κυρίως στη στοίβα – stack) κοντεύει να γεμίσει ενεργοποιεί το συλλέκτη σκουπιδιών. Έτσι ο προγραμματιστής δε χρειάζεται να ανησυχεί για το πότε και αν θα ελευθερώσει ένα συγκεκριμένο τμήμα της μνήμης, ούτε και για δείκτες (pointers) που αναφέρονται σε άδειο χώρο μνήμης. Αυτό είναι ιδιαίτερα σημαντικό αν σκεφτούμε ότι ένα μεγάλο ποσοστό κατάρρευσης των προγραμμάτων οφείλονται σε λανθασμένο χειρισμό της μνήμης.[1]

## 2.6 ΕΠΙΔΟΣΕΙΣ

Παρόλο που η εικονική μηχανή προσφέρει όλα αυτά (και όχι μόνο) τα πλεονεκτήματα, η Java είναι πιο αργή σε σχέση με της προγραμματιστικές γλώσσες υψηλού επιπέδου (high-level), C και η C++. Έχει αποδειχτεί ότι η C++ μπορεί να είναι αρκετές φορές γρηγορότερη από τη Java. Ευτυχώς γίνονται φιλότιμες προσπάθειες από τη Sun για τη βελτιστοποίηση της εικονικής μηχανής, ενώ προσφέρονται και καινούργιες υλοποιήσεις από άλλες εταιρίες (IBM) οι οποίες μπορεί σε κάποια σημεία να προσφέρουν καλύτερα και σε κάποια άλλα χειρότερα αποτελέσματα. Επιπλέον με τον ερχομό των JIT (Just In Time) μεταφραστών, οι οποίοι μετατρέπουν το Bytecode απ' ευθείας σε γλώσσα μηχανής, η διαφορά ταχύτητας από τη C++ έχει μικρύνει κατά πολύ. Οι τελευταίες εκδόσεις του «Java Compiler» με την χρήση της τεχνολογίας «Hot Spot» καταφέρνει να παρέχει αξιόλογες επιδόσεις που πλησιάζουν ή και ξεπερνούν σε ταχύτητα τις υπόλοιπες γλώσσες.[1]

## 2.7 JAVA ΚΑΙ C++

Η Java σχετίζεται άμεσα με τη C++. Κληρονομεί τη σύνταξη της από τη C και το μοντέλο αντικειμένων από τη C++. Παρέχει ένα δυναμικό και λογικά δομημένο περιβάλλον που παίρνει τα καλύτερα χαρακτηριστικά από της παλιές γλώσσες προγραμματισμού και παράλληλα προσθέτει νέα χαρακτηριστικά που συμπληρώνουν τα κενά που υπήρχαν πριν την εμφάνισή της. Οι μεγάλες ομοιότητες των δυο αυτών γλωσσών προγραμματισμού έχουν σαν αποτέλεσμα, η γνώση της μιας να συνεπάγεται την εύκολη εκμάθηση της άλλης. Επιπλέον, η Java παρέχει στον προγραμματιστή τον πλήρη έλεγχο,

πράγμα το οποίο αποτελεί βασική αρχή της C και C++. Τέλος, λόγω του ότι οι δύο γλώσσες είναι σχεδιασμένες για να δίνουν λύση σε ένα διαφορετικό σύνολο προβλημάτων, είναι σίγουρο ότι θα συνυπάρχουν για πολλά χρόνια ακόμα.[2]

## **2.8 Η ΣΥΜΒΟΛΗ ΤΗΣ JAVA ΣΤΟ INTERNET**

Η Java έχει ασκήσει μία πολύ σημαντική επίδραση στο Internet λόγω του ότι παράγει όλα τα αντικείμενα εκείνα, τα οποία μπορούν να κυκλοφορούν ελεύθερα στο διαδίκτυο. Τα αντικείμενα αυτά ανήκουν σε δύο μεγάλες κατηγορίες: στις παθητικές πληροφορίες και στα δυναμικά, ενεργά προγράμματα. Όσο πιο μεγάλη ζήτηση υπάρχει για δυναμικά, δικτυωμένα προγράμματα τόσο πιο πολλά είναι και τα σοβαρά προβλήματα που προκύπτουν σε θέματα ασφάλειας και δυνατότητας μεταφοράς. Πριν από τη Java το διαδίκτυο ήταν κλειστό για περίπου τα μισά από αυτά τα αντικείμενα που τώρα πλέον κυκλοφορούν ελεύθερα στο Internet χάρη σε μια νέα μορφή προγράμματος που παρέχει η java, τα γνωστά Applets. [2]

## **2.9 ΝΗΜΑΤΑ**

Ένα από τα πιο ενδιαφέροντα χαρακτηριστικά της java είναι η ενσωματωμένη υποστήριξη που παρέχει για τον πολυνηματικό προγραμματισμό. Ένα πολυνηματικό πρόγραμμα περιέχει δύο ή περισσότερα μέρη τα οποία μπορούν να τρέχουν ταυτόχρονα. Κάθε μέρος του

προγράμματος ονομάζεται νήμα(thread). Κάθε νήμα καθορίζει μια ξεχωριστή διαδρομή εκτέλεσης.

Το πολυνηματικό σύστημα της java βασίζεται στην κλάση Thread και στην αντίστοιχη συνοδευτική διεπαφή Runnable. Η κλάση Thread ενθυλακώνει ένα νήμα της εκτέλεσης. Για τη δημιουργία νέου νήματος είτε θα επεκταθεί η Thread είτε θα υλοποιηθεί η διεπαφή Runnable.

Στην εφαρμογή που εξετάζουμε, η Thread χρησιμοποιήθηκε έτσι ώστε να υλοποιηθεί ένα είδος χρονομέτρου που χειρίζεται της ανοιχτές συνόδους μελών. Με τη βοήθεια της μεθόδου sleep(), κάθε ένα λεπτό ανανεώνει της χρόνους της συνόδους μελών και εντοπίζει το αν κάποια σύνοδος έχει λήξει. Επιπλέον, χρησιμοποιήθηκε για να υποστηρίξει την επικοινωνία ανάμεσα στον κεντρικό υπολογιστή στον οποίο τρέχει ο NET-PRO Server και τους υπόλοιπους στους οποίους τρέχει ο NET-PRO Client για το γεγονός ότι τη στιγμή που πραγματοποιείται η συγκεκριμένη διαδικασία της επικοινωνίας δεν πρέπει να διακόπτεται η εκτέλεση του κύριου προγράμματος.[3]

## 2.10 JAVA ΚΑΙ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Είναι πλέον αυτονόητο ότι οι περισσότερες εφαρμογές χρειάζονται την υποστήριξη μιας βάσης δεδομένων για να υλοποιήσουν το επιθυμητό αποτέλεσμα. Η java μπορεί να συνεργαστεί με ιδιαίτερη ευκολία με μια βάση δεδομένων.

Αρχικά είναι καλό να αναφέρουμε ότι στο πρόγραμμά μας η βάση δεδομένων αναπτύχθηκε σε mySql. Για να συνδεθούμε στη mySql από java θα πρέπει να ενσωματώσουμε στο πρόγραμμά της την κατάλληλη βιβλιοθήκη. Αυτή η βιβλιοθήκη είναι το πρόγραμμα οδήγησης που ονομάζεται **JDBC** (Java Database Connectivity) Στη συνέχεια θα πρέπει να χρησιμοποιήσουμε το πακέτο sql της java για να πραγματοποιήσουμε τη σύνδεση. Σε μία εφαρμογή,

όταν παρουσιάζεται η ανάγκη να συνδεόμαστε στη βάση δεδομένων από πολλά σημεία του προγράμματος είναι προτιμότερο να δημιουργήσουμε μία ξεχωριστή κλάση η οποία να πραγματοποιεί ακριβώς αυτή τη διαδικασία και στη συνέχεια να καλούμε την κατάλληλη μέθοδο στην αρχή της `main()`. Στην εφαρμογή που μελετάμε η κλάση για την οποία μιλάμε ονομάζεται `Database`.

Η μέθοδος της `Java` που πραγματοποιεί τη σύνδεση είναι στην περίπτωση μας η `connect()` και παίρνει σαν ορίσματα τη διεύθυνση στην οποία είναι καταχωρημένη η βάση δεδομένων(είτε αυτή είναι καταχωρημένη τοπικά είτε απομακρυσμένα), και το `username` και `password` τα οποία χρειάζονται για να αποκτηθεί πρόσβαση στη βάση. Αυτή η μέθοδος φορτώνει τον οδηγό που αναφέραμε, με την κλήση της μεθόδου `forName()` η οποία παίρνει σαν όρισμα τον οδηγό. Στη συνέχεια, παίρνει τη σύνδεση με τη βάση μέσω της κλήσης της μεθόδου `getConnection()` και με ορίσματα, τα ορίσματα της `connect()`, και την αποθηκεύει σε ένα αντικείμενο τύπου `Connection`. Τέλος, με την `createStatement()`, δημιουργεί μία δήλωση μέσω της οποίας θα μπορούμε να αποστέλλουμε ερωτήματα στη βάση.

Με αυτό τον τρόπο, καλώντας μετά σε οποιοδήποτε σημείο του κωδικά μας τις `execute()`, `executeQuery()` και `executeUpdate()`, τα ερωτήματα με τη μορφή `String` που δημιουργούμε στη `Java`, αποστέλλονται στη βάση, εκτελούνται και επιστρέφουν αποτελέσματα **όπου αυτό είναι επιθυμητό**. Τα αποτελέσματα που επιστρέφονται είναι τύπου `ResultSet`.

## 2.11 SOCKETS

Ένα socket είναι το ένα άκρο της αμφίδρομης επικοινωνίας μεταξύ δύο εφαρμογών που αποστέλλουν πληροφορίες η μία στην άλλη. Με άλλα λόγια, μέσω των sockets κάθε εφαρμογή έχει τη δυνατότητα πρόσβασης σε δίκτυο και μετάδοσης δεδομένων. Το πακέτο `net` της `Java` παρέχει όλες τις

απαραίτητες κλάσεις για την υλοποίηση των Sockets, οι σημαντικότερες από τις οποίες είναι η `ServerSocket` και η `Socket`. Η πρώτη παρέχει ένα socket στην πλευρά του server. Ένα τέτοιου είδους socket περιμένει για αιτήματα και μόλις αντιληφθεί ένα, πραγματοποιεί τις διαδικασίες που προβλέπονται σε κάθε περίπτωση και επιστρέφει κάποιο αποτέλεσμα αν αυτό είναι επιθυμητό. Η δεύτερη παρέχει ένα socket στην πλευρά του client. Είναι υπεύθυνο για να αποστέλλει αιτήματα στον server και ενδεχομένως να περιμένει κάποιο αποτέλεσμα.

Τα βήματα που πρέπει να ακολουθηθούν έτσι ώστε να γίνει δυνατή μία τέτοια επικοινωνία είναι συγκεκριμένα για κάθε μία από τις δύο πλευρές. Στην πλευρά του server θα πρέπει να δημιουργηθεί ένα στιγμιότυπο του `ServerSocket` το οποίο θα δηλώνει σε ποιο port και σε ποια ip διεύθυνση (ή host name) θα περιμένει αιτήματα καθώς επίσης και έναν αριθμό που θα χαρακτηρίζει το συγκεκριμένο socket. Στη συνέχεια με τη μέθοδο `accept()` περιμένει για αιτήματα έτσι ώστε μόλις τα αντιληφθεί, να πραγματοποιήσει σύνδεση και να τα ικανοποιήσει. Στην πλευρά του client χρησιμοποιώντας τη μέθοδο `connect()` το στιγμιότυπο της κλάσης `Socket` συνδέεται στην άλλη πλευρά και, αφού συνδεθεί, μπορεί να στείλει κάποιο αίτημα με τη βοήθεια του πακέτου `io` της Java.

Σε επόμενα κεφάλαια θα αναλυθεί ο τρόπος με τον οποίο υλοποιήθηκε η επικοινωνία μεταξύ των επιμέρους εφαρμογών του NET-PRO, Server και Client χρησιμοποιώντας το πακέτο `net` της Java.[4]



## **ΚΕΦΑΛΑΙΟ 3**

---

### *Χαρακτηριστικά Internet Café – Καταστάσεις λειτουργίας*

Στο Κεφάλαιο που ακολουθεί αναλύουμε τα ιδιαίτερα χαρακτηριστικά που απαρτίζουν ένα Internet Café και πρέπει οπωσδήποτε να λαμβάνονται υπόψη κατά την οργάνωση του. Τα χαρακτηριστικά αυτά, τα οποία κατηγοριοποιούνται σε δομικά στοιχεία και σε ρόλους, πρέπει να παρατεθούν καθώς οδηγούν στην πλήρη κατανόηση των λειτουργιών που θα πρέπει να διαθέτει ένα πρόγραμμα χρέωσης και των απαιτήσεων που θα πρέπει να καλύπτει. Στην συνέχεια θα γίνει η πρώτη αναφορά στη εφαρμογή NET-PRO με μία εισαγωγή στα κλιμακούμενα δικαιώματα και τις καταστάσεις λειτουργίας του(modes).

### 3.1 ΟΡΓΑΝΩΣΗ ΤΟΥ INTERNET CAFÉ

Τα δομικά στοιχεία που συγκροτούν ένα Internet Café είναι οι υπολογιστές πελατών (Clients), ο κεντρικός υπολογιστής διαχείρισης του καταστήματος (Server), και οι εξειδικευμένες υπηρεσίες που προσφέρει (Services).

#### 3.1.1 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ INTERNET CAFÉ

- **Υπολογιστές πελατών.** Αποτελούν το σημαντικότερο δομικό στοιχείο του καταστήματος. Είναι οι υπολογιστές που παρέχουν στον πελάτη ένα μεγάλο εύρος υπηρεσιών. Η επιλογή του πλήθους τους καθορίζει τη διαθεσιμότητα των υπολογιστών αυτών και την δυνατότητα να ανατεθούν, στον καθένα ξεχωριστά, εξειδικευμένες αρμοδιότητες, ανάλογα με της δυνατότητες του. Για παράδειγμα, ένας υπολογιστής με μεγάλη επεξεργαστική ισχύ, μεγάλη RAM και καλή κάρτα γραφικών μπορεί να χρησιμοποιηθεί συγκεκριμένα για «βαριά» παιχνίδια και εφαρμογές. Ομοίως ένας υπολογιστής με όχι τόσο καλές επιδόσεις μπορεί να εξειδικευτεί για περιήγηση στο διαδίκτυο ή για email, υπηρεσίες οι οποίες δεν προϋποθέτουν μεγάλες απαιτήσεις.
- **Κεντρικός υπολογιστής διαχείρισης καταστήματος.** Η ανάγκη ύπαρξής του ταυτίζεται απόλυτα και προέρχεται από την ανάγκη για χρήση του λογισμικού που θα αναλάβει την καθολική διαχείριση του καταστήματος. Μέσω μιας εφαρμογής, ο κεντρικός υπολογιστής αναλαμβάνει την χρέωση, καθορίζει το διάστημα κατά το οποίο ο υπολογιστής θα είναι διαθέσιμος για έναν πελάτη και εντοπίζει

οποιοσδήποτε αλλαγές στην ομαλή λειτουργία των υπολογιστών και της μεταξύ τους συνεργασίας.

- **Υπηρεσίες καταστήματος.** Κάθε Internet Café προσφέρει ένα μεγάλο εύρος υπηρεσιών. Κάποιες από αυτές της υπηρεσίες είναι η πλοήγηση στο διαδίκτυο, το email, το downloading και τα παιχνίδια, διαδικτυακά και μη.

### 3.1.2 ΡΟΛΟΙ

Οι ρόλοι (Roles) που αφορούν τα άτομα τα οποία είτε θα προσφέρουν είτε θα απολαμβάνουν της υπηρεσίες ενός τέτοιου καταστήματος είναι ο Ιδιοκτήτης (Owner), ο υπάλληλος (employee) και ο πελάτης (Customer).

- **Ιδιοκτήτης καταστήματος.** Ο ιδιοκτήτης ενός Internet Café, καθώς και ο ιδιοκτήτης οποιουδήποτε άλλου καταστήματος, είναι το άτομο που παίρνει τις κρίσιμες αποφάσεις όσον αφορά το κατάστημα, έχει τα πλήρη δικαιώματα και αναλαμβάνει την κεντρική διαχείριση. Επιπλέον ενδέχεται να κληρονομήσει μέρος από τα δικαιώματα του σε υπαλλήλους. Ο ιδιοκτήτης κάνει είσοδο στην εφαρμογή NET-PRO πάντα **σε κατάσταση διαχειριστή (Admin Mode)** για την οποία θα μιλήσουμε παρακάτω.
- **Υπάλληλος καταστήματος.** Ο υπάλληλος παίρνει αποφάσεις λιγότερο κρίσιμες και περισσότερο καθημερινές και έχει συγκεκριμένα καθήκοντα, ανάλογα πάντα με τα δικαιώματα που έχει πάρει από τον ιδιοκτήτη. Θεωρητικά, καταστήματα του είδους που μελετάμε, μπορούν να υπάρξουν χωρίς απαραίτητα τη συμμετοχή του υπαλλήλου. Πρακτικά όμως αυτό δεν είναι δυνατό. Ο υπάλληλος κάνει είσοδο στην εφαρμογή NET-PRO είτε σε **κατάσταση απλού** είτε **ισχυρού χρήστη (User – Power User Mode)**

- **Πελάτης καταστήματος.** Οι πελάτες κατηγοριοποιούνται σε τακτικούς και έκτακτους. Οι τακτικοί, έπειτα από τη διαδικασία καταχώρησης των στοιχείων τους και εγγραφής τους, ονομάζονται **μέλη** και συνήθως απολαμβάνουν κάποια προνόμια, όπως πχ ειδικές εκπτώσεις στη χρέωση. Οι έκτακτοι πελάτες ονομάζονται **επισκέπτες**. Όπως είναι φυσικό, οι πελάτες δεν έχουν καμία άμεση αλληλεπίδραση με την εφαρμογή.

## **3.2 ΚΑΤΑΣΤΑΣΗ ΔΙΑΧΕΙΡΙΣΤΗ – ADMIN MODE**

Όταν λέμε ότι γίνεται είσοδος στο NET-PRO σε κατάσταση διαχειριστή εννοούμε ότι το πρόγραμμα λειτουργεί χαρίζοντας πρόσβαση στο 100% των λειτουργιών που μπορεί να εκτελέσει. Προκειμένου να περάσει το πρόγραμμα σε αυτήν την κατάσταση θα πρέπει να δοθεί το μοναδικό **συνθηματικό του διαχειριστή (Admin Password)** κατά τη **σύνδεση στο σύστημα (System Login)**. Ως συμπέρασμα από τα προηγούμενα, η κατάσταση διαχειριστή είναι φτιαγμένη αποκλειστικά για τον ιδιοκτήτη. Έχει ως σκοπό την παραμετροποίηση των ρυθμίσεων όλων των τμημάτων που απαρτίζουν την εφαρμογή. Είναι το εργαλείο που επιτρέπει την αυτοματοποίηση της επίβλεψης και του ελέγχου για το πώς λειτουργεί το κατάστημα, καθιστώντας εύκολη τη διαπίστωση ενδεχόμενων αδυναμιών στην παραγωγικότητα και στο προσωπικό (υπάλληλοι).

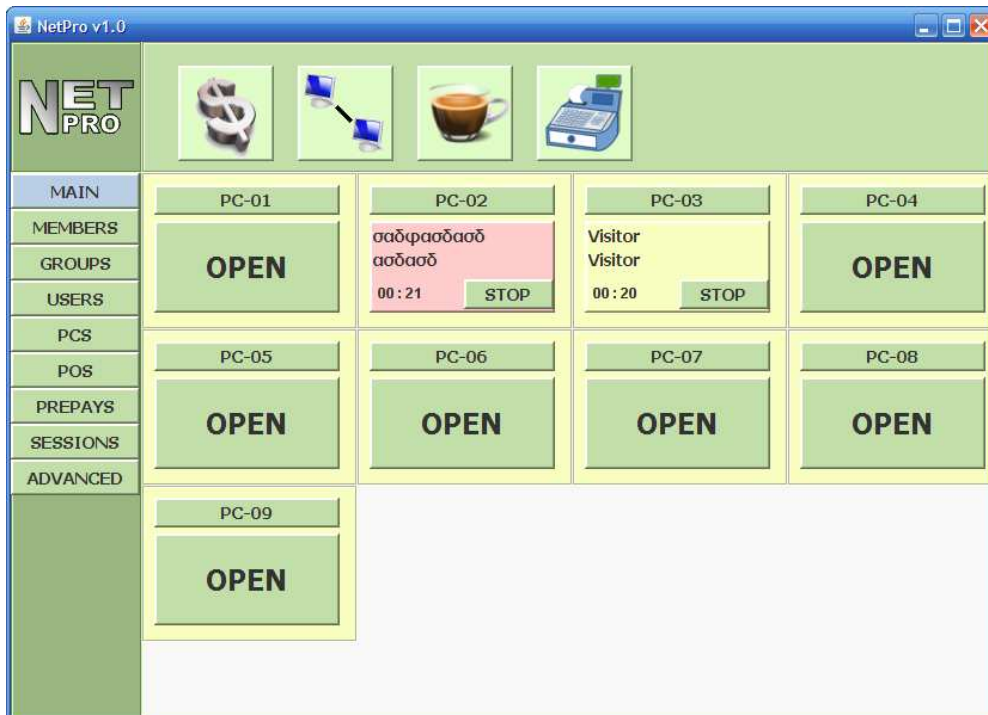
### **3.2.1 ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΤΑΣΤΑΣΗΣ ΔΙΑΧΕΙΡΙΣΤΗ**

Ο ιδιοκτήτης του καταστήματος, μέσω του Admin Mode έχει τη δυνατότητα

- να εισάγει ένα καινούργιο μέλος(Member) στο σύστημα και να το τροποποιήσει ή να το διαγράψει. Επιπλέον, μπορεί να κάνει αναζήτηση(Search) με βάση κάποια από τα στοιχεία του πελάτη.
- να ορίσει κατηγορίες μελών(Groups) και να εντάξει κάθε μέλος σε μία από αυτές. Με βάση την κατηγορία στην οποία ανήκει το κάθε μέλος, έχει κάποια κλιμακούμενα προνόμια. Για παράδειγμα, υπάρχει η δυνατότητα κάποιου ποσοστού έκπτωσης στη χρέωση του.
- να αποκλείσει τη χρήση των υπολογιστών πελάτη από κάποιο μέλος για συγκεκριμένο χρονικό διάστημα μετατρέποντας το σε ανενεργό(Inactive)
- να αποφασίσει για τις χρεώσεις(Charges) που επιθυμεί, καθώς επίσης και να καθορίσει ειδικές εκπτώσεις(Discounts) για κάθε group καθώς και προσφορές που παρακινούν τον πελάτη να αγοράσει περισσότερο χρόνο για χρήση των υπολογιστών.
- να εισάγει έναν καινούργιο χρήστη(User) στο σύστημα δίνοντας του συγκεκριμένα προνόμια κ δικαιώματα πρόσβασης. Πρακτικά, το σύστημα δίνει τη δυνατότητα στο διαχειριστή να παραχωρήσει στο χρήστη σχεδόν το σύνολο των δικαιωμάτων που διαθέτει και ο ίδιος επιτρέποντας του την πρόσβαση σε κατάσταση ισχυρού χρήστη (Power User Mode). Αυτό μπορεί να γίνει αν ο διαχειριστής, εισάγοντας το χρήστη στο σύστημα, του δώσει τα λεγόμενα δικαιώματα διαχειριστή(Admin Rights).
- να τροποποιήσει ή και να διαγράψει τα στοιχεία του χρήστη.
- να παρακολουθεί την κίνηση των χρημάτων με λεπτομέρειες για το ποιος χρήστης ήταν υπεύθυνος για το κατάστημα τη στιγμή της συναλλαγής, ποιος πελάτης συμμετέχει στη συναλλαγή και τον ακριβή χρόνο της συναλλαγής. Αυτό είναι δυνατόν μέσω της καταγραφής των συνόδων χρήστη (User Sessions).

- να παρακολουθεί ποια από τα μέλη αγοράζουν χρόνο για χρήση των υπολογιστών και τα λεπτομερή ποσά που διακινούνται. Επιπλέον να τα κατηγοριοποιεί σε σκοπό την εξαγωγή συμπερασμάτων και στατιστικών, πχ ποιος πελάτης έχει αγοράσει τον περισσότερο χρόνο τον τελευταίο μήνα.
- να εισάγει και να διαγράφει υπολογιστές πελάτη από το σύστημα.
- να τροποποιεί τα στοιχεία(π.χ. ip διεύθυνση) των υπολογιστών πελάτη στο σύστημα
- Να κρατάει backups με τη βάση δεδομένων του συστήματος
- Να εισάγει χρεώσεις των υλικών ειδών που πουλάει(εκτυπώσεις, cd's, USB sticks)

Όπως βλέπουμε στην εικόνα 3.2.1.1 ο κεντρικός κορμός των λειτουργιών του NET-PRO σε Admin Mode είναι οι: Main, Members, Users, Pc's, Pos, Prepays, Sessions, Advanced



Εικόνα 3.2.1.1

### **3.3 ΚΑΤΑΣΤΑΣΗ ΧΡΗΣΤΗ – USER MODE**

Κάθε υπάλληλος-χρήστης για να έχει πρόσβαση σε αυτό το τμήμα του συστήματος καλείται να δώσει το όνομά του και τον προσωπικό κωδικό του, ο οποίος έχει καταχωρηθεί κατά την εγγραφή του στο σύστημα από τον διαχειριστή (Administrator) . Με αυτό τον τρόπο αποκλείεται η είσοδος στο σύστημα σε μη εξουσιοδοτημένους χρήστες. Μέσω της κατάστασης χρήστη ικανοποιούνται όλες οι καθημερινές λειτουργίες όπως οι προπληρωμές, το μέτρημα του χρόνου κατά τον οποίο παρέχονται οι υπηρεσίες στους πελάτες και η χρέωση προϊόντων όπως εκτυπώσεις και CD, DVD. Οι λειτουργίες της κατάστασης χρήστη παρουσιάζονται συνοπτικά παρακάτω.

#### **3.3.1 ΛΕΙΤΟΥΡΓΙΕΣ ΚΑΤΑΣΤΑΣΗΣ ΧΡΗΣΤΗ**

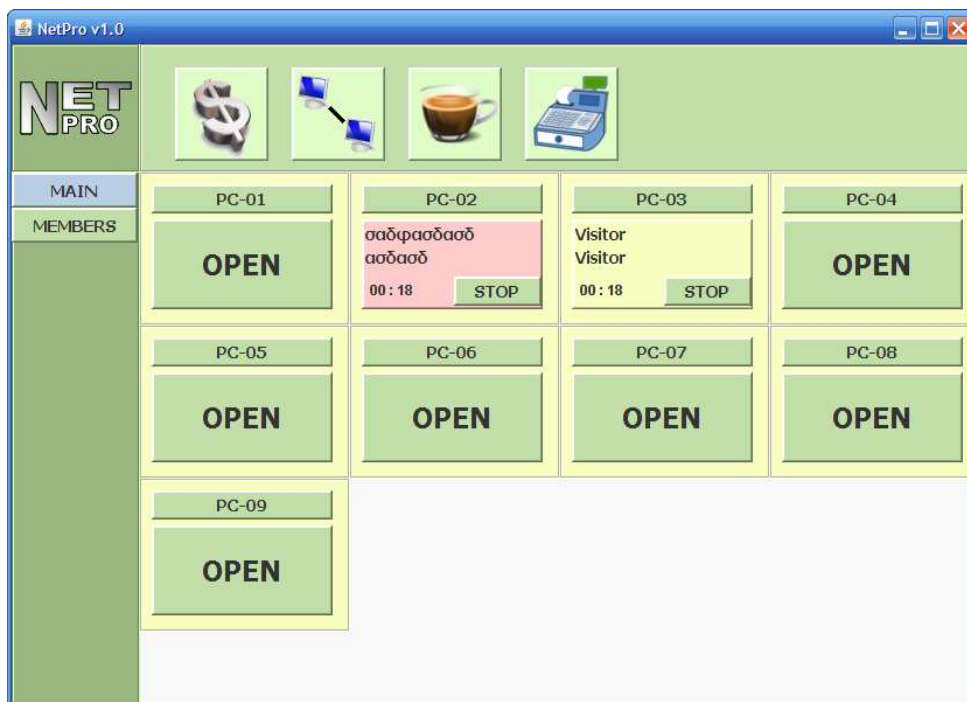
Ο υπάλληλος του καταστήματος, μέσω της κατάστασης χρήστη έχει τη δυνατότητα

- Να εισάγει καινούργιο μέλος στο σύστημα και να τροποποιεί τα στοιχεία του. Επιπλέον, να αναζητά κάποιο μέλος με βάση τα στοιχεία του. Σε αντίθεση με την κατάσταση διαχειριστή , η κατάσταση χρήστη δεν δίνει το δικαίωμα στον υπάλληλο να διαγράψει ένα μέλος. Αυτό αφήνεται αποκλειστικά στην κρίση του διαχειριστή.
- Γενικότερης διαχείρισης των συνόδων για τους υπολογιστές που ανοίγουν οι υπάλληλοι στους πελάτες. Αυτό περιλαμβάνει ένα

κατάλογο με τους υπολογιστές και πληροφορίες που τους αφορούν, όπως το αν χρησιμοποιούνται από κάποιον πελάτη και από ποιον.

- Προπληρωμής. Αφορά στην διαδικασία κατά την οποία ο πελάτης, πριν χρησιμοποιήσει οποιαδήποτε υπηρεσία που παρέχει το κατάστημα και χωρίς απαραίτητα να σκοπεύει άμεσα να το κάνει, καταβάλλει κάποιο χρηματικό ποσό. Το ποσό αυτό, μέσω του συστήματος θα καταχωρηθεί στη βάση και ο πελάτης θα μπορεί να το εκμεταλλευτεί οποτεδήποτε και με οποιονδήποτε τρόπο επιθυμεί.

Βλέποντας την εικόνα 3.3.1.1 παρατηρούμε ότι το πρόγραμμα μέσω της κατάστασης χρήστη δίνει στον χρήστη πρόσβαση μόνο στις λειτουργίες «Main» και «Members».



Εικόνα 3.3.1.1



### **3.4 ΚΑΤΑΣΤΑΣΗ ΙΣΧΥΡΟΥ ΧΡΗΣΤΗ – POWER USER MODE**

Στην περίπτωση που ο πελάτης που συνδέεται στο σύστημα διαθέτει τα δικαιώματα διαχειριστή, περνάμε σε κατάσταση ισχυρού χρήστη. Η κατάσταση αυτή όπως έχει προαναφερθεί παρέχει την πρόσβαση σε όλες σχεδόν της λειτουργίες του NET-PRO. Θα μπορούσαμε να πούμε ότι ο ισχυρός χρήστης ανήκει ακριβώς ένα σκαλί χαμηλότερα από τον διαχειριστή στην ιεραρχία των δικαιωμάτων πρόσβασης του NET-PRO. Η διαφορά της με την κατάσταση διαχειριστή εντοπίζεται στο ότι ο ισχυρός χρήστης δεν έχει δικαίωμα να ορίσει την πρωταρχική χρέωση(Initial charge).

## ΚΕΦΑΛΑΙΟ 4

---

### *Η επικοινωνία μεταξύ Net-Pro Server και Client*

Στο κεφάλαιο που ακολουθεί περιγράφουμε τις κλάσεις της εφαρμογής μας που χρησιμοποιούνται για να πραγματοποιηθεί η επικοινωνία μεταξύ των δύο υποεφαρμογών Net-Pro Server και Net-Pro Client. Όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, η λειτουργία αυτή είναι υλοποιήσιμη με τη βοήθεια του πακέτου Net της Java. Οι κλάσεις αυτές έχουν ενσωματωθεί και στις δύο υποεφαρμογές. Με τον τρόπο αυτό επιτυγχάνεται η αποστολή και η εξυπηρέτηση αιτημάτων και από τις πλευρές της επικοινωνίας.

## 4.1 Η ΚΛΑΣΗ `OnlineServerData`

Η κλάση `OnlineServerData` περιλαμβάνει μόνο κάποιες `static` μεταβλητές καθώς επίσης και τις μεθόδους πρόσβασής τους. Αυτές είναι οι `path`, `logErrorLines`, `socketTimeout`, `connectTries`, `serverPort`, `clientPort`, `theFrame`.

Η `path` κρατάει τη διεύθυνση από την οποία τρέχει η εφαρμογή μας. Η `logErrorLines` είναι μία μεταβλητή τύπου `Boolean` η οποία μας πληροφορεί για το αν τα `Exceptions` που είναι πιθανό να προκύψουν θα γράφονται σε αρχείο. Η `socketTimeout` καθορίζει το πόσο χρόνο θα περιμένει ο `client` να συνδεθεί στο `server`. Η `connectTries` το πόσες προσπάθειες θα κάνει ο `client` να συνδεθεί στο `server`. Η `serverPort` αποθηκεύει το `port` στο οποίο θα στέλνονται τα αιτήματα και η `clientPort`, σε ποιο `port` θα στέλνονται οι απαντήσεις των αιτημάτων, όπου αυτό είναι αναγκαίο.

Οι τιμές αυτών των μεταβλητών καθορίζονται από το διαχειριστή του συστήματος και αποθηκεύονται στο αρχείο `config.txt`. Στην αρχή της εκτέλεσης τόσο του `Net-Pro Server` όσο και του `Net-Pro Client` το σύστημα διαβάζει τις τιμές αυτές από το αρχείο. Αν δεν τις βρει τότε παίρνει κάποιες `default` τιμές που έχουμε ορίσει εμείς.

## 4.2 Η ΚΛΑΣΗ `Connection`

Η κλάση `Connection` περιλαμβάνει δύο χαρακτηριστικά. Το `host`, τύπου `String`, που αποτελεί το όνομα του υπολογιστή που εξυπηρετεί τα αιτήματα ή την `ip` διεύθυνση του, και το `port` του. Η κλάση αυτή χρησιμοποιώντας τα

παραπάνω χαρακτηριστικά υλοποιεί τη σύνδεση του υπολογιστή που στέλνει αιτήματα σε αυτόν που θα τα εξυπηρετήσει. Αυτό πραγματοποιείται με τη βοήθεια της μεθόδου `connect()` η οποία δοκιμάζει να συνδεθεί στο `server` τόσες φορές όσες ακριβώς προβλέπονται από την τιμή της μεταβλητής `connectTries` που εξετάσαμε προηγουμένως. Κάθε φορά που θα χρειαστεί να στείλουμε ένα αίτημα από οποιαδήποτε από τις δύο υποεφαρμογές μας στην άλλη, θα πρέπει να δημιουργήσουμε ένα αντικείμενο τύπου `Connection`.

### **4.3 Η ΚΛΑΣΗ `Server`**

Η κλάση `Server` δημιουργήθηκε για να εξυπηρετήσει την αποδοχή των αιτημάτων για σύνδεση. Στην ουσία δημιουργεί ένα νήμα στο οποίο καλείται η `accept()` η λειτουργία της η οποίας είναι να περιμένει για αιτήματα για σύνδεση στον `server`. Η χρήση νήματος είναι απαραίτητη για την υλοποίηση ταυτόχρονων συνδέσεων σε ένα `socket`. Κάθε σύνδεση απαιτεί και ένα καινούργιο νήμα. Μόλις η `accept()` ανιχνεύσει κάποιο αίτημα τότε δημιουργείται ένα αντικείμενο τύπου `ServerThread` την ιδιότητα του οποίου θα αναλύσουμε στη συνέχεια. Αξίζει, στο σημείο αυτό, να αναφερθεί ότι ο όρος `server` που χρησιμοποιήθηκε εδώ δεν έχει την ίδια ερμηνεία με τον `NET-PRO Server`. `Server` είναι οποιοσδήποτε από τους δύο υπολογιστές που επικοινωνούν, δεχτεί ένα αίτημα. Έτσι οποιαδήποτε από τις δύο υποεφαρμογές του συστήματος μας μπορεί να συμπεριφερθεί ως `server` ανάλογα με τη λειτουργία που εκτελείται. Για το λόγο αυτό άλλωστε η κλάση `server` υπάρχει αυτούσια και στις δύο υποεφαρμογές.

## 4.4 Η ΚΛΑΣΗ Action

Η κλάση Action περιλαμβάνει τα χαρακτηριστικά name και returned. Μέσω των sockets στη συγκεκριμένη εφαρμογή μεταφέρουμε αντικείμενα τύπου Action. Για να γίνει αυτό (κανονικά μέσω των sockets μεταφέρονται μόνο Strings) κάνουμε implement το interface Serializable. Επιπλέον, απαιτείται να δώσουμε μια συγκεκριμένη final τιμή στο id του αντικειμένου τύπου Action υπερβαίνοντας τη μεταβλητή serialVersionUID. Σε διαφορετική περίπτωση υπάρχει το ενδεχόμενο να προκύψουν λάθη καθώς το id αυτό μεταβάλλεται σε κάθε compile που γίνεται στο αντικείμενο. Το χαρακτηριστικό name υποδηλώνει το είδος του αιτήματος που μεταφέρεται. Υπάρχουν συγκεκριμένα είδη αιτημάτων στην εφαρμογή μας. Αυτά είναι τα παρακάτω:

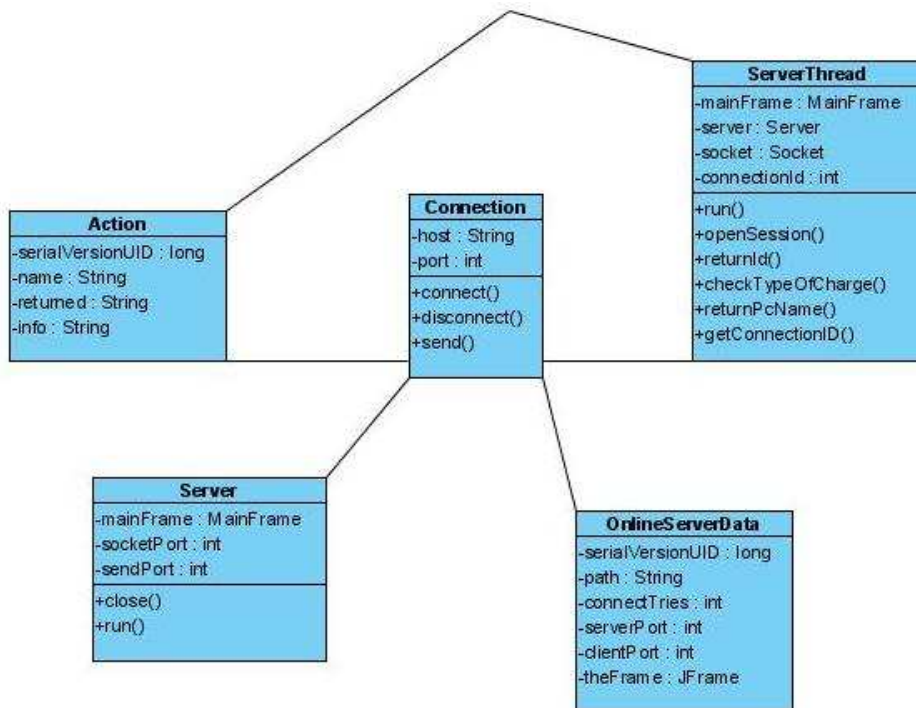
- OpenClient : Με το άνοιγμα μιας συνόδου μέλους "ξεκλειδώνει" τον υπολογιστή πελάτη
- CloseClient : Με το κλείσιμο μιας συνόδου μέλους "κλειδώνει" τον υπολογιστή πελάτη
- Run : Ζητάει από τον υπολογιστή πελάτη να εκτελέσει μία εντολή όπως πχ τη notepad.exe
- RequestCheck : Ζητάει από τον υπολογιστή πελάτη να ενημερώσει για το αν τρέχει ο NET-PRO Client και υπάρχει επικοινωνία
- ReplyCheck : Δέχεται απάντηση από τον υπολογιστή πελάτη για το αν τρέχει ο NET-PRO Client και υπάρχει επικοινωνία
- RequestTime : Ο υπολογιστής πελάτη με ανοιχτή σύνοδο μέλους ζητάει από τον κεντρικό υπολογιστή το χρόνο της συνόδου.
- ReplyTime : Ο υπολογιστής πελάτη με ανοιχτή σύνοδο μέλους δέχεται ως απάντηση από τον κεντρικό υπολογιστή το χρόνο της συνόδου.
- Shutdown : Ζητάει από τον υπολογιστή πελάτη να τερματίσει
- Restart : Ζητάει από τον υπολογιστή πελάτη να κάνει επανεκκίνηση

- RequestOpen : Ο υπολογιστής πελάτη ζητάει από τον κεντρικό υπολογιστή την άδεια να ξεκινήσει σύνοδο μέλους
- DenyOpen : Ο κεντρικός υπολογιστής αρνείται να ξεκινήσει σύνοδο μέλους στον υπολογιστή πελάτη από τον οποίο δέχτηκε το αίτημα RequestOpen

## **4.5 Η ΚΛΑΣΗ ServerThread**

Η κλάση ServerThread ενσωματώνει ένα νήμα το οποίο εκτελεί την εξής λειτουργία : Διαβάζει από το socket το αίτημα που έχει φτάσει στο server. Το αίτημα αυτό είναι ένα αντικείμενο τύπου Action. Από την τιμή της μεταβλητής name του Action καταλαβαίνει ποια λειτουργία θα πρέπει να εκτελέσει και την εκτελεί. Αφού την εκτελέσει αποστέλλει απάντηση αν αυτό είναι επιθυμητό και κλείνει τη σύνδεση. Για παράδειγμα, αν αποσταλεί ένα Action που για name έχει την τιμή "Shutdown", θα εκτελέσει τερματισμό στον υπολογιστή πελάτη. Αν ληφθεί αντικείμενο που δεν είναι τύπου Action, θα απορριφτεί.

## **4.5 ΤΟ ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ**



## **ΚΕΦΑΛΑΙΟ 5**

---

### *Λειτουργίες NET – PRO SERVER*

Στο κεφάλαιο που ακολουθεί αναπτύσσουμε λεπτομερώς όλες τις λειτουργίες που διαθέτει το πρόγραμμα χρέωσης NET-PRO. Περιγράφουμε τη δομή της βάσης δεδομένων που το υποστηρίζει. Επιπλέον αναφερόμαστε στους περιορισμούς και στους ελέγχους που έχουν εισαχτεί κατά την ανάπτυξη της εφαρμογής με σκοπό να "προλάβουν" και να αποτρέψουν πιθανά λάθη. Με αυτόν τον τρόπο διασφαλίζεται η απόλυτα σωστή λειτουργία του συστήματος.



## 5.1 Η ΚΛΑΣΗ Main

Είναι η κλάση που διαθέτει τη μέθοδο `main()` και συνεπώς είναι το πρώτο σημείο εκτέλεσης του προγράμματος. Αρχικά γίνεται σύνδεση με τη βάση δεδομένων του Net-Pro. Στη συνέχεια διαβάζονται από το αρχείο `Config.txt` οι τιμές των μεταβλητών που σχετίζονται με τα `sockets`, δηλαδή με την επικοινωνία μεταξύ του κεντρικού υπολογιστή και των υπολογιστών πελάτη. Τέλος, καλείται το `LoginFrame`, το οποίο είναι το αρχικό σημείο της διεπαφής που οδηγεί τον χρήστη στον κυρίως πρόγραμμα μετά από μία σειρά ελέγχων. Το `LoginFrame` αναλύεται λεπτομερώς στη συνέχεια.

## 5.2 Η ΚΛΑΣΗ LoginFrame

Το `LoginFrame` πραγματοποιεί την είσοδο του χρήστη στο σύστημα. Με τον όρο «χρήστης» εννοούμε είτε το διαχειριστή του καταστήματος ή τον απλό ή το δυναμικό χρήστη. Προτού επιτευχθεί η είσοδος αυτή πρέπει να γίνει μία σειρά από ελέγχους. Με την αρχικοποίηση του `LoginFrame` πραγματοποιούνται τα εξής:

- Το σύστημα ελέγχει αν υπάρχει ανοιχτή σύνοδος χρήστη μέσω ερωτήματος στη βάση. Το ερώτημα αυτό ψάχνει συγκεκριμένα για το αν υπάρχει σύνοδος με το πεδίο `date_closed` να είναι ίσο με μηδέν. Αν υπάρχει, σημαίνει ότι όντως υπάρχει ανοιχτή σύνοδος, τότε θα πάρει το `username` του χρήστη και θα περάσει στον επόμενο έλεγχο.

- Χρησιμοποιώντας το `username` του χρήστη θα ελέγξει αν ο χρήστης αυτός είναι ανενεργός ή διαγραμμένος. Αν είναι, η ανοιχτή σύνοδος θα κλείσει αυτόματα τοποθετώντας στην τιμή του πεδίου `date_closed` την τρέχουσα ώρα και στη συνέχεια θα ενημερώσει το χρήστη ότι δεν έχει πλέον τα απαραίτητα δικαιώματα εισόδου στο σύστημα. Αν όλα έχουν πάει καλά μέχρι εδώ, θα εμφανιστεί το `username` του χρήστη στο κατάλληλο `textfield`.

Αφότου ο χρήστης εισάγει και τον κωδικό του και πατήσει το κουμπί «οκ» θα εκτελεστεί η μέθοδος `loginProcess()`. Η μέθοδος αυτή εξετάζει αν τα στοιχεία είναι σωστά, αν ο χρήστης είναι ενεργός, αν δεν είναι διαγραμμένος, αν δεν είναι ο διαχειριστής του καταστήματος και, στην περίπτωση που όλα αυτά ισχύουν :

- Αν υπάρχει ανοιχτή σύνοδος χρήστη, θα ανοίξει ένα `MainFrame` με την υπάρχουσα ανοιχτή σύνοδο.
- Αν δεν υπάρχει, θα δημιουργήσει μία καινούρια σύνοδο καλώντας τη μέθοδο `sessionInit()` και στη συνέχεια θα ανοίξει ένα `MainFrame` με αυτή τη σύνοδο.

### **5.3 Η ΚΛΑΣΗ `MainFrame`**

Η κλάση `MainFrame`, η οποία αποτελεί το κεντρικό σημείο διαχείρισης από το οποίο εκτελείται το σύνολο των λειτουργιών της εφαρμογής, έχει ως χαρακτηριστικά τα `userSessionId` που είναι το `id` της ανοιχτής συνόδου, `userId` που είναι το `id` του χρήστη που χειρίζεται την παρούσα σύνοδο,

username που είναι το όνομα χρήστη του και adminRights που δείχνει το αν έχει δικαιώματα διαχειριστή ή όχι.

Με τη δημιουργία ενός αντικειμένου τέτοιου τύπου, ενεργοποιούνται ή απενεργοποιούνται κάποιες από τις δυνατότητες (buttons) του Net-Pro ανάλογα με τα δικαιώματα πρόσβασης. Επιπλέον δημιουργείται ένα νέο OverviewPanel το οποίο περιέχει τα PcPanels (ένα για κάθε υπολογιστή πελάτη) και αναλύεται στη συνέχεια. Ταυτόχρονα, ανοίγεται ένα socket στην πλευρά του server το οποίο περιμένει για αιτήματα σύνδεσης. Τέλος, εκτελείται ένα νήμα το οποίο καλεί τη μέθοδο updatePcs() του OverviewPanel.

## 5.4 Η ΚΛΑΣΗ OverviewPanel

Η κλάση OverviewPanel έχει ως χαρακτηριστικό το MainFrame από το οποίο δημιουργήθηκε. Το γεγονός αυτό είναι ένα από τα πολλά παραδείγματα σύνδεσης που παρουσιάζονται στην εφαρμογή μας. Κατά τη δημιουργία της, η OverviewPanel καλεί τη μέθοδο init() η οποία είναι υπεύθυνη για τη δημιουργία των αντικειμένων PcPanel και την τοποθέτησή τους σε ένα πίνακα τύπου PcPanel. Επιπλέον χρησιμοποιείται ένα νήμα το οποίο, καλώντας τη μέθοδο checkClient(), ελέγχει ανά τακτά χρονικά διαστήματα της τάξεως των 30 δευτερολέπτων το αν οι υπολογιστές πελάτη είναι συνδεδεμένοι στον κεντρικό και ο Net-Pro Client εκτελείται ομαλά. Αυτό πραγματοποιείται στέλνοντας αιτήματα requestCheck και checkClient, που εξετάσαμε στο προηγούμενο κεφάλαιο, μεταξύ των Net-Pro Server και Client. Άλλη μία μέθοδος αυτής της κλάσης, που αναφέρθηκε και προηγουμένως λόγω του ότι καλείται και από το mainframe, είναι η updatePcs() η οποία ανανεώνει τις πληροφορίες των PcPanels.

Εκτός από τις παραπάνω ενέργειες που συμβαίνουν κατά την αρχικοποίηση, παρουσιάζονται και μέθοδοι που εκτελούνται ανάλογα με τις ενέργειες που θα πραγματοποιήσει ο χρήστης. Οι μέθοδοι αυτές απλά δημιουργούν νέα αντικείμενα τα οποία εξετάζουμε στη συνέχεια.

#### **5.4.1 Η ΚΛΑΣΗ AddPrepayDialog**

Η κλάση αυτή διαθέτει ως χαρακτηριστικό το `mainframe` που περιέχει το `OverviewPanel` από το οποίο κλήθηκε. Οι μέθοδοι που εμφανίζονται εδώ είναι μέθοδοι που καλούνται μετά από συγκεκριμένες ενέργειες του χρήστη. Ουσιαστικά στο σημείο αυτό της εφαρμογής μπορεί να εισαχθεί χρόνος για μελλοντική χρήση υπολογιστή πελάτη στο λογαριασμό ενός μέλους.

Η `usernameBtnActionPerformed()` δημιουργεί ένα αντικείμενο τύπου `MemberSearchDialog()` με τιμή «by username» στο χαρακτηριστικό του «`typeOfSearch`». Εδώ αναζητείται το επιθυμητό μέλος με βάση το `username` του. Αφού η αναζήτηση ολοκληρωθεί, το `username` στέλνεται στο κατάλληλο `textfield` του `AddPrepayDialog` και συμπληρώνεται το ονοματεπώνυμο και το `id` του μέλους.

Η `nameBtnActionPerformed()` δημιουργεί ένα αντικείμενο τύπου `MemberSearchDialog()` με τιμή «by surname» στο χαρακτηριστικό του «`typeOfSearch`». Εδώ αναζητείται το επιθυμητό μέλος με βάση το επώνυμό του. Αφού η αναζήτηση ολοκληρωθεί, το ονοματεπώνυμο του μέλους στέλνεται στο κατάλληλο `textfield` του `AddPrepayDialog` και συμπληρώνεται το `username` και το `id` του.

Η `idBtnActionPerformed()` αναζητά στη βάση το ονοματεπώνυμο και το `username` του μέλους το οποίο έχει ως `id` το `id` που εισήγαγε ο χρήστης στο κατάλληλο `textfield`. Στη συνέχεια συμπληρώνει και τα υπόλοιπα `textfields` με τις τιμές που πήρε από τη βάση.

Καθεμιά από τις παραπάνω μεθόδους καλεί τη μέθοδο `fillTheTextflds()` η οποία αναζητεί στον πίνακα `seconds_left` για το διαθέσιμο χρόνο που έχει το συγκεκριμένο μέλος στο λογαριασμό του και ενημερώνει τα κατάλληλα πεδία του `PrepayChargeDialog` .

Οι μέθοδοι `moneyBtnActionPerformed()` και `hoursBtnActionPerformed()` κατά την κλήση τους υπολογίζουν το χρόνο και το χρηματικό ποσό που αντιστοιχούν στις τιμές του χρηματικού ποσού ή του χρόνου που εισήγαγε ο χρήστης αντίστοιχα. Εδώ χρησιμοποιείται και η μέθοδος `round()` η οποία πραγματοποιεί τις κατάλληλες στρογγυλοποιήσεις.

Με το πάτημα του κουμπιού «οκ» του `PrepayChargeDialog`, τα χρήματα εισάγονται στο λογαριασμό του μέλους και ενημερώνονται οι πίνακες `member_prepays`, `seconds_left` και `user_sessions` της βάσης δεδομένων.

#### **5.4.2 Η ΚΛΑΣΗ `SwitchDialog`**

Η κλάση αυτή έχει ως στόχο τη μεταφορά μιας συνόδου μέλους από έναν υπολογιστή πελάτη σε έναν άλλο. Έχει ως χαρακτηριστικό το `OverviewPanel` από το οποίο δημιουργήθηκε. Κατά τη δημιουργία ενός αντικειμένου τέτοιου τύπου, εκτελείται ένας αλγόριθμος ο οποίος απομονώνει τους υπολογιστές πελάτη που έχουν ανοιχτές συνόδους μέλους και στη συνέχεια τους παρουσιάζει στο χρήστη μέσω ενός `combobox`. Πλέον ο χρήστης μπορεί να επιλέξει σε ποιον υπολογιστή πελάτη θα μεταφέρει τη σύνοδο. Όταν λοιπόν κληθεί η μέθοδος `switchBtnActionPerformed()`, υπάρχουν δύο ενδεχόμενα, η μεταφορά της συνόδου σε έναν υπολογιστή πελάτη που δεν έχει ανοιχτή σύνοδο ή η ανταλλαγή συνόδων. Αυτός είναι και ο πρωταρχικός έλεγχος της συνόδου. Στην πρώτη περίπτωση, στην ουσία κλείνει η σύνοδος μέλους με μηδέν στο πεδίο `"money_created"` του πίνακα `"member_sessions"` και ανοίγει μια καινούργια, με τιμή στο πεδίο

"date\_opened", την τιμή της συνόδου που έκλεισε. Στη δεύτερη περίπτωση, οι εγγραφές του πίνακα "member\_sessions" ανταλλάσσουν τις τιμές που έχουν στα πεδία "member\_id" και "date\_opened" .

### 5.4.3 Η ΚΛΑΣΗ AddPosDialog

Η κλάση AddPosDialog έχει ως χαρακτηριστικό ένα αντικείμενο τύπου MainFrame όπως και οι προηγούμενες κλάσεις που εξετάσαμε. Μόλις δημιουργηθεί θα ανακτήσει από τη βάση όλα τα ονόματα των προϊόντων προς πώληση που διαθέτει το κατάστημα για να τα παρουσιάσει στο χρήστη μέσω του κατάλληλου combobox. Διαθέτει τις μεθόδους quantityTextFld-KeyReleased(), posComboBoxItemStateChanged() και okBtnActionPerformed().

Η quantityTextFld-KeyReleased() εκτελείται όταν ο χρήστης πληκτρολογήσει μία ποσότητα του προϊόντος προς πώληση. Βλέποντας την τιμή του αντικείμενου από τη βάση δεδομένων και υπολογίζοντας τη συνολική τιμή ανάλογα με την ποσότητα, την παρουσιάζει στο κατάλληλο textfield.

Η posComboBoxItemStateChanged() εκτελείται κάθε φορά που ο χρήστης επιλέγει κάποιο προϊόν από τη λίστα και ο ρόλος της δεν είναι άλλος από το να μηδενίσει τα textfields που εμφανίζουν την ποσότητα και την τιμή έτσι ώστε να αποφευχθεί πιθανό λάθος.

Η okBtnActionPerformed() εκτελείται όταν ο χρήστης επιλέξει να ολοκληρώσει την πώληση. Η μέθοδος αυτή ενημερώνει τρεις πίνακες της βάσης δεδομένων. Στο πεδίο "quantity" του πίνακα "pos" αφαιρείται η ποσότητα των προϊόντων που πωλήθηκαν έτσι ώστε να περιέχει το τρέχον απόθεμα. Στον πίνακα pos\_sale προστίθεται μία νέα εγγραφή που αντιπροσωπεύει τη συγκεκριμένη αγοραπωλησία ενώ ταυτόχρονα ενημερώνεται και το πεδίο "money\_created" του πίνακα "user\_sessions" .

Συγκεκριμένα προστίθεται το χρηματικό ποσό που προέκυψε από την πώληση του προϊόντος στην υπάρχουσα τιμή του πεδίου αυτού.

#### **5.4.4 Η ΚΛΑΣΗ CloseSessionDialog**

Η κλάση CloseSessionDialog έχει και αυτή ως χαρακτηριστικό ένα αντικείμενο τύπου MainForm. Όταν ο χρήστης επιλέξει να τερματίσει τη σύννοδό του εισάγοντας τον χρηματικό ποσό που έχει δημιουργήσει, θα κληθεί η μέθοδος okBtnActionPerformed(). Αυτή θα ενημερώσει τη βάση τοποθετώντας στην εγγραφή που έχει ως τιμή πεδίου "date\_closed" το 0, την τρέχουσα ώρα και στο πεδίο "money\_closed" το χρηματικό ποσό που εισήγαγε ο χρήστης. Αφού ολοκληρωθεί αυτή η διαδικασία το πρόγραμμα θα τερματιστεί.

#### **5.4.5 Η ΚΛΑΣΗ RunCommandsDialog**

Η κλάση RunCommandsDialog έχει ως χαρακτηριστικό ένα αντικείμενο τύπου OverviewPanel. Με τη δημιουργία ενός αντικειμένου RunCommandsDialog στέλνεται ένα ερώτημα στη βάση το οποίο ζητάει τα ονόματα των υπολογιστών πελάτη για να τα παρουσιάσει στο χρήστη στο κατάλληλο combobox. Οι μέθοδοι της κλάσης αυτής παρουσιάζονται παρακάτω.

Η λειτουργία που εκτελεί η μέθοδος okBtnActionPerformed() είναι να στέλνει μία εντολή προς εκτέλεση στον υπολογιστή πελάτη που έχει επιλεγεί από το χρήστη. Συγκεκριμένα, συνδέεται στον υπολογιστή πελάτη, ο οποίος

τη συγκεκριμένη στιγμή από άποψης socket λειτουργεί ως server , και στη συνέχεια στέλνει την εντολή που θέλει καλώντας τη μέθοδο send() της κλάσης Connection. Το αίτημα που στέλνει στο σημείο αυτό είναι το αίτημα "Run" ως όρισμα της μεθόδου send() .Το δεύτερο απαραίτητο όρισμα που πρέπει να εμπερικλείεται είναι ένα String με την εντολή. Αυτό το String το έχει πληκτρολογήσει ο χρήστης στη διασύνδεση της κλάσης. Αν δεν καταφέρει να επικοινωνήσει με την εφαρμογή Net-Pro Client που εκτελείται στον υπολογιστή πελάτη θα ενημερώσει το χρήστη με κατάλληλο μήνυμα.

Με τη μέθοδο restartPcBtnActionPerformed() και πάλι θα σταλεί ένα αίτημα "Run" στον υπολογιστή πελάτη με τη διαφορά ότι αυτή τη φορά το δεύτερο όρισμα θα έχει την τιμή "shutdown -r -f -t 00" που είναι η εντολή που εκτελεί την επανεκκίνηση σε έναν υπολογιστή.

Ομοίως και η μέθοδος shutdownPcBtnActionPerformed() με τη διαφορά ότι αυτή τη φορά το δεύτερο όρισμα θα έχει την τιμή "shutdown -f -t 00" που είναι η εντολή που εκτελεί το κλείσιμο σε έναν υπολογιστή.

Και στις τρεις παραπάνω μεθόδους, αν ο Net-Pro Server δεν καταφέρει να επικοινωνήσει με την εφαρμογή Net-Pro Client που εκτελείται στον υπολογιστή πελάτη θα ενημερώσει το χρήστη με κατάλληλο μήνυμα.

## **5.5 Η ΚΛΑΣΗ PcPanel**

Η κλάση PcPanel έχει ως χαρακτηριστικά το mainframe και το αντίστοιχο overviewpanel από τα οποία δημιουργείται, το id του υπολογιστή πελάτη, το όνομά του, την κατάσταση του και την ip διεύθυνσή του. Στο δομητή του , δημιουργεί ένα αντικείμενο τύπου connection για να μπορεί να στείλει αιτήματα στον Net-Pro Client του συγκεκριμένου υπολογιστή πελάτη όποτε υπάρξει αυτή η ανάγκη. Επιπλέον, δημιουργεί ένα JPanel το οποίο αργότερα θα μετατραπεί σε τύπου Closed, OpenedCharge, PrepayCharge ή



disabledInformationPanel ανάλογα με την τιμή που έχει το state του υπολογιστή πελάτη. Αυτές οι κλάσεις όμως περιγράφονται στα επόμενα κεφάλαια οπότε και αυτή η διαδικασία θα γίνει απόλυτα κατανοητή.

Η μέθοδος `init()` του `PcPanel`, στέλνοντας ερώτημα στη βάση για το αν υπάρχει εγγραφή στον πίνακα `member_sessions` με το `id` του υπολογιστή πελάτη και τιμή στο πεδίο `date_closed = 0` δίνει ανάλογη τιμή στη βοηθητική μεταβλητή `typeOfCharge` και καλεί την `initInformationPanel()`. Αυτή με τη σειρά της χρησιμοποιώντας τη μεταβλητή `typeOfCharge` καθορίζει τον τύπο του `informationPanel` όπως αναφέραμε παραπάνω. Επιπλέον, η μέθοδος `init()` διαθέτει αλγόριθμο ο οποίος υπολογίζει με βάση το χρόνο που έχει απομείνει στην προπληρωμένη χρέωση πότε θα πρέπει να τερματίσει τη σύνοδο μέλους(δηλαδή όταν εξαντληθεί ο προπληρωμένος χρόνος του μέλους) καλώντας τη μέθοδο `stop()`. Όπως αναφέρθηκε προηγουμένως η μέθοδος `init()` καλείται από το `OverviewPanel` μέσα σε ένα νήμα.

### **5.5.1 Η ΚΛΑΣΗ `ClosedInformationPanel`**

Η κλάση αυτή έχει μοναδικό χαρακτηριστικό της το `pcPanel` από το οποίο δημιουργήθηκε. Εδώ είναι απαραίτητο να αναφερθεί ότι κάθε αντικείμενο τύπου `PcPanel` διαθέτει και ένα `informationPanel` το οποίο είναι είτε τύπου `ClosedInformationPanel` είτε ενός από τους τύπους που εξετάζουμε παρακάτω. Στην κλάση αυτή υλοποιείται η μέθοδος `openBtnActionPerformed()` με μοναδική λειτουργία της τη δημιουργία ενός αντικειμένου `chargeInitDialog`.

### **5.5.1.1 Η ΚΛΑΣΗ ChargeInitDialog**

Η κλάση αυτή υλοποιεί το άνοιγμα μιας συνόδου μέλους στον υπολογιστή πελάτη τον οποίο αντιπροσωπεύει το PcPanel που διαθέτει το ClosedInformationPanel από το οποίο δημιουργήθηκε. Επομένως, δέχεται ως όρισμα το συγκεκριμένο αντικείμενο τύπου PcPanel. Υλοποιεί τις μεθόδους που μελετάμε στη συνέχεια.

Η idBtnActionPerformed() διαβάζει το id του πελάτη που εισήγαγε ο χρήστης και με τη βοήθεια ερωτήματος στη βάση δεδομένων εμφανίζει το ονοματεπώνυμο και το username του μέλους. Οι αλγόριθμοι που χρησιμοποιούν αυτή καθώς και οι μέθοδοι nameBtnActionPerformed() και usernameBtnActionPerformed() είναι όμοιοι με τους αλγόριθμους που περιγράψαμε στις αντίστοιχες μεθόδους της κλάσης PrepayAddDialog().

Ιδιαίτερο ενδιαφέρον παρουσιάζει η μέθοδος startBtnActionPerformed() η οποία υλοποιεί την έναρξη καινούργιας συνόδου μέλους. Αρχικά, από την τιμή του προπληρωμένου χρόνου που έχει υπολογιστεί με την εμφάνιση των στοιχείων του μέλους στη διασύνδεση της κλάσης(με οποιονδήποτε από τους 3 παραπάνω τρόπους) υπολογίζεται το αν η χρέωση θα είναι ανοιχτή ή προπληρωμένη. Στη συνέχεια, προστίθεται μία εγγραφή στον πίνακα member\_sessions με τιμή 0 για το πεδίο date\_closed η οποία υποδηλώνει ότι η σύνοδος δεν έχει κλείσει. Έπειτα καλείται η μέθοδος initInformationPanel() του pcPanel. Τέλος, καλείται η μέθοδος sendRequestOpenToClient() η οποία υλοποιεί την αποστολή αιτήματος "openClient" με δεύτερο όρισμα "prepayCharge" ή "openCharge" αν η χρέωση είναι ελεύθερη ή προπληρωμένη αντίστοιχα.

### **5.5.2 Η ΚΛΑΣΗ OpenChargeInformationPanel**

Η κλάση `OpenChargeInformationPanel`, με όρισμα το `rcPanel` μέσω του οποίου δημιουργήθηκε, παρουσιάζει το ονοματεπώνυμο του μέλους που έχει ανοίξει σύνοδο με ανοιχτή χρέωση στο χρήστη. Επιπλέον διαθέτει χρονόμετρο που δείχνει το χρόνο που πέρασε από τη στιγμή έναρξης της συνόδου. Ο χρόνος αυτός υπολογίζεται με ένα νήμα του `overviewPanel` το οποίο ανανεώνει την ένδειξη αυτή κάθε ένα λεπτό σε όλα τα `rcPanels` με ανοιχτή σύνοδο. Στη συγκεκριμένη περίπτωση όπως είναι λογικό το μέτρημα είναι αυξητικό.

Επιπλέον, εδώ υλοποιείται η μέθοδος `chooseColor()` η οποία δεσμεύει από την κλάση `ColorConfig` την τιμή της κατάλληλης static μεταβλητής για να εμφανίσει το `backgroundColor`.

Τέλος, υπάρχει η δυνατότητα τερματισμού της συνόδου με τη βοήθεια της μεθόδου `stopBtnActionPerformed()` η οποία δημιουργεί ένα αντικείμενο τύπου `CloseChargeDialog` το οποίο εξετάζουμε στη συνέχεια.

### **5.5.2.1 Η ΚΛΑΣΗ `CloseChargeDialog`**

Η κλάση αυτή κατά τη δημιουργία της ελέγχει αν η χρέωση είναι ανοιχτή ή προπληρωμένη. Αν είναι προπληρωμένη, εμφανίζει μηδέν ως ποσό κλεισίματος. Αν είναι ανοιχτή, καλεί τη μέθοδο `calculatePriceOfOpenCharge()` η οποία υπολογίζει το ποσό κλεισίματος χρησιμοποιώντας έναν αλγόριθμο στην υλοποίηση της, ο οποίος λαμβάνει υπόψιν τη χρέωση ανά ώρα από τον πίνακα `open_charge` και την έκπτωση που πιθανόν να έχει το `group` στο οποίο ανήκει το μέλος, και στη συνέχεια το εμφανίζει. Εδώ καλείται και η μέθοδος `round()` για να πραγματοποιήσει τα απαραίτητες στρογγυλοποιήσεις.

Όταν κληθεί η μέθοδος `closeSessionBtnActionPerformed()`, θα καλέσει με τη σειρά της τη βοηθητική μέθοδο `closeCharge()` η οποία υλοποιεί τα εξής:

- Αν η χρέωση είναι ανοιχτή ενημερώνει την αντίστοιχη εγγραφή του πίνακα `member_sessions` θέτοντας στο πεδίο `date_closed` την τρέχουσα ώρα και στο πεδίο `money_created` το ποσό που υπολόγισε η μέθοδος `calculatePriceOfOpenCharge()`. Στη συνέχεια ενημερώνει και το πεδίο `money_created` του πίνακα `user_sessions` αθροίζοντας στην υπάρχουσα τιμή το ποσό κλεισίματος.
- Αν η χρέωση είναι ανοιχτή ενημερώνει το πεδίο `date_closed` του πίνακα `member_sessions`. Στη συνέχεια αφαιρεί το χρόνο που διήρκεσε η σύνοδος από την τιμή του πεδίου `seconds` του πίνακα `seconds_left` της εγγραφής που έχει στο πεδίο `id`, το `id` του μέλους στο οποίο ανήκε η σύνοδος.

Αφού ολοκληρωθεί αυτή η διαδικασία, η `closeSessionBtnActionPerformed()` καλεί τη μέθοδο `sendRequestCloseToClient()`, η οποία θα στείλει αίτημα "CloseClient" στο Net-Pro Client μέσω του socket.

### 5.5.3 Η ΚΛΑΣΗ `PrepayChargeInformationPanel`

Η κλάση `PrepayChargeInformationPanel`, με όρισμα το `pcPanel` μέσω του οποίου δημιουργήθηκε, παρουσιάζει το ονοματεπώνυμο του μέλους που έχει ανοίξει σύνοδο με προπληρωμένη χρέωση στο χρήστη. Επιπλέον διαθέτει χρονόμετρο που δείχνει τον προπληρωμένο χρόνο που απομένει στο λογαριασμό του χρήστη. Το χρονόμετρο αυτό ανανεώνεται με τον τρόπο που παρουσιάσαμε στο κεφάλαιο 5.5.2 με τη διαφορά ότι εδώ ο χρόνος μειώνεται. Τα παραπάνω προκύπτουν με την κλήση της βοηθητικής μεθόδου `init()` που υλοποιείται στη συγκεκριμένη κλάση.

Επιπλέον, η κλάση διαθέτει τη μέθοδο `stopBtnActionPerformed()` η οποία δημιουργεί ένα `CloseChargeDialog()` για τον τερματισμό της συνόδου. Συγκεκριμένα, το `CloseChargeDialog()` δημιουργείται από τη `myStopBtnActionPerformed()` που καλείται από τη `stopBtnActionPerformed()`.

#### **5.5.4 Η ΚΛΑΣΗ DisabledInformationPanel**

Η κλάση αυτή λειτουργεί απλά και μόνο ως ένδειξη ότι ο συγκεκριμένος υπολογιστής πελάτη είναι υπό επισκευή ή ότι για οποιονδήποτε άλλο λόγο δεν είναι σε θέση να φιλοξενήσει κάποια σύνοδο μέλους.

### **5.6 Η ΚΛΑΣΗ MembersOverviewPanel**

Το `MembersOverviewPanel` έχει ως χαρακτηριστικό το `mainFrame` από το οποίο δημιουργήθηκε. Κατά την αρχικοποίησή του δημιουργεί έναν πίνακα και καλεί τη μέθοδο `updateData()` για να τον συμπληρώσει με τα στοιχεία των μελών του καταστήματος. Ειδικότερα, οι ενέργειες που εκτελεί η `updateData()` είναι οι εξής:

- Ελέγχει αν το `showDeletedCheckBox` είναι επιλεγμένο έτσι ώστε να αποφασίσει αν θα ανακτήσει από τις βάση τα μέλη που έχουν διαγραφεί, δηλαδή τις εγγραφές που έχουν στο πεδίο `deleted` την τιμή `true`.

- Στη συνέχεια ελέγχει αν το `searchPanel` είναι ενεργοποιημένο. Αν δεν είναι, εμφανίζει κανονικά τα μέλη στον πίνακα. Αν είναι, επιλέγει μόνο τις εγγραφές που ταιριάζουν στα κριτήρια που έχει εισάγει ο χρήστης.

Με τον τρόπο αυτό η `updateData()` χρησιμοποιείται και στις εξής περιπτώσεις :

- στην περίπτωση που ο χρήστης αλλάξει το κριτήριο της αναζήτησης
- κάθε φορά που θα αλλάζει το πρότυπο που αναζητά
- όταν θελήσει να τερματίσει την αναζήτηση και να εμφανιστεί ολόκληρη η λίστα με τα μέλη όπως εμφανίζεται και κατά τη δημιουργία του αντικειμένου.

Πράγματι, οι μέθοδοι που καλούνται σε αυτές τις τρεις περιπτώσεις, οι οποίες είναι αντίστοιχα οι `searchComboBoxActionPerformed()`, `searchTextFldKey-Released()` και `CloseSearchBtnActionPerformed()`, δεν κάνουν τίποτε άλλο από το να καλούν την `updateData()` (Με εξαίρεση την τελευταία η οποία κάνει `null` όλα τα `components` του `searchPanel` πριν την καλέσει).

Επιπλέον, σε αυτή την κλάση υλοποιούνται και οι μέθοδοι :

- `addMemberBtnActionPerformed()` η οποία δημιουργεί ένα αντικείμενο `MemberAddDialog` για την προσθήκη νέου μέλους
- `editMemberBtnActionPerformed()` η οποία δημιουργεί ένα αντικείμενο `MemberEditDialog` για την μετατροπή των στοιχείων του μέλους που έχει επιλέξει ο χρήστης από τον πίνακα. Προτού το δημιουργήσει ελέγχει αν υπάρχει επιλεγμένο μέλος και αν δεν υπάρχει ενημερώνει το χρήστη με μήνυμα λάθους.
- `deleteMemberBtnActionPerformed()` ή οποία διαγράφει εικονικά το επιλεγμένο μέλος. Προτού το διαγράψει ελέγχει αν υπάρχει επιλεγμένο μέλος και αν δεν υπάρχει ενημερώνει το χρήστη με μήνυμα λάθους.

Όταν μιλάμε για εικονική διαγραφή εννοούμε ότι οι εγγραφές δεν διαγράφονται από τη βάση, απλά το πεδίο `deleted` του πίνακα `members` αυτών των εγγραφών παίρνει την τιμή `true` έτσι ώστε να θεωρούνται διαγραμμένες από το σύστημα.

### **5.6.1 Η ΚΛΑΣΗ `MemberAddDialog`**

Η κλάση `MemberAddDialog` υλοποιεί την προσθήκη νέου μέλους, προσθέτοντας μία εγγραφή στον πίνακα `members` με τα στοιχεία που εισήγαγε ο χρήστης στη διασύνδεση της κλάσης. Το `id` του μέλους είναι αυτόματα ο επόμενος αριθμός από το μεγαλύτερο `id` που έχει μέχρι τώρα κάποιο μέλος στο κατάστημα. Πριν δημιουργηθεί η καινούργια εγγραφή στον πίνακα `members`, θα πρέπει να ελεγχθεί:

- αν το `username` που έχουμε σκοπό να δώσουμε στο καινούργιο μέλος είναι καταχωρημένο σε άλλο.
- αν το `id` και το ονοματεπώνυμο είναι κενά

### **5.6.2 Η ΚΛΑΣΗ `MemberEditDialog`**

Η κλάση αυτή έχει ως χαρακτηριστικά το `membersOverviewPanel`. Δίνει τη δυνατότητα μετατροπής του αριθμού ταυτότητας ενός μέλους, του τηλεφώνου, της διεύθυνσης, του `username`, του `group` στο οποίο ανήκει καθώς και του προπληρωμένου χρόνου. Αυτός είναι ο τρόπος να βάζει ο

χρήστης χρόνο δωρεάν στο λογαριασμό του μέλους. Αν εισάγει χρόνο από αυτό το σημείο του προγράμματος δεν θα ενημερωθεί ούτε ο πίνακας `member_sessions` ούτε ο `user_sessions` παραμόνο ο `seconds_left`.

Όταν κληθεί η μέθοδος `okBtnActionPerformed()` θα ενημερωθούν τα πεδία `id_card_number`, `group_id`, `telephone`, `address`, `username` του πίνακα `members`. Επιπλέον, ενημερώνεται και το πεδίο `seconds` του πίνακα `seconds_left`. Φυσικά, προηγούνται οι απαραίτητοι έλεγχοι που είναι το να μην υπάρχουν άδεια πεδία στη διασύνδεση της κλάσης και να μην υπάρχει άλλο μη διαγραμμένο μέλος με το ίδιο `username`.

## 5.7 Η ΚΛΑΣΗ `GroupsOverviewPanel`

Το `GroupsOverviewPanel` έχει ως χαρακτηριστικό το `mainFrame` από το οποίο δημιουργήθηκε. Κατά την αρχικοποίησή του δημιουργεί έναν πίνακα και καλεί τη μέθοδο `updateData()` για να τον συμπληρώσει με τις κατηγορίες μελών του καταστήματος. Ειδικότερα, οι ενέργειες που εκτελεί η `updateData()` είναι οι εξής:

- Ελέγχει αν το `showDeletedCheckBox` είναι επιλεγμένο έτσι ώστε να αποφασίσει αν θα ανακτήσει από τη βάση τις κατηγορίες μελών που έχουν διαγραφεί, δηλαδή τις εγγραφές που έχουν στο πεδίο `deleted` την τιμή `true`.
- Στη συνέχεια ελέγχει αν το `searchPanel` είναι ενεργοποιημένο. Αν δεν είναι, εμφανίζει κανονικά τις κατηγορίες μελών στον πίνακα. Αν είναι, επιλέγει μόνο τις εγγραφές που ταιριάζουν στα κριτήρια που έχει εισάγει ο χρήστης.

Με τον τρόπο αυτό η `updateData()` χρησιμοποιείται και στις εξής περιπτώσεις :



- στην περίπτωση που ο χρήστης αλλάξει το κριτήριο της αναζήτησης
- κάθε φορά που θα αλλάζει το πρότυπο που αναζητά
- όταν θελήσει να τερματίσει την αναζήτηση και να εμφανιστεί ολόκληρη η λίστα με τις κατηγορίες μελών όπως εμφανίζεται και κατά τη δημιουργία του αντικειμένου.

Πράγματι, οι μέθοδοι που καλούνται σε αυτές τις τρεις περιπτώσεις, οι οποίες είναι αντίστοιχα οι `searchComboBoxActionPerformed()`, `searchTextFldKeyReleased()` και `CloseSearchBtnActionPerformed()`, δεν κάνουν τίποτε άλλο από το να καλούν την `updateData()` (Με εξαίρεση την τελευταία η οποία κάνει null όλα τα components του `searchPanel` πριν την καλέσει).

Επιπλέον, σε αυτή την κλάση υλοποιούνται και οι μέθοδοι :

- `addGroupBtnActionPerformed()` η οποία δημιουργεί ένα αντικείμενο `GroupAddDialog` για την προσθήκη νέας κατηγορίας μέλους
- `editGroupBtnActionPerformed()` ή οποία δημιουργεί ένα αντικείμενο `GroupEditDialog` για την μετατροπή των στοιχείων της κατηγορίας μελών που έχει επιλέξει ο χρήστης από τον πίνακα. Προτού το δημιουργήσει ελέγχει αν υπάρχει επιλεγμένη κατηγορία και αν δεν υπάρχει ενημερώνει το χρήστη με μήνυμα λάθους.
- `deleteGroupBtnActionPerformed()` η οποία διαγράφει εικονικά την επιλεγμένη κατηγορία. Προτού τη διαγράψει ελέγχει αν υπάρχει επιλεγμένη κατηγορία και αν δεν υπάρχει ενημερώνει το χρήστη με μήνυμα λάθους.

### **5.7.1 Η ΚΛΑΣΗ `GroupAddDialog`**

Η κλάση `GroupAddDialog` υλοποιεί την προσθήκη νέας κατηγορίας μέλους, προσθέτοντας μία εγγραφή στον πίνακα `groups` με τα στοιχεία που εισήγαγε ο χρήστης στη διασύνδεση της κλάσης.

### **5.7.2 Η ΚΛΑΣΗ `GroupEditDialog`**

Η κλάση αυτή έχει ως χαρακτηριστικά το `groupsOverviewPanel` και το `id` της κατηγορίας μελών που ο χρήστης είχε επιλέξει ακριβώς πριν από τη δημιουργία ενός τέτοιου αντικειμένου. Δίνει τη δυνατότητα μετατροπής του ονόματος μιας κατηγορίας μελών ή της έκπτωσής της. Με άλλα λόγια, όταν κληθεί η μέθοδος `okBtnActionPerformed()` θα ενημερωθούν τα πεδία `name` και `discount` του πίνακα `groups`. Φυσικά, προηγούνται οι απαραίτητοι έλεγχοι που είναι το να μην υπάρχουν άδεια πεδία στη διασύνδεση της κλάσης και να μην υπάρχει άλλο μη διαγραμμένο `group` με το ίδιο όνομα.

## **5.8 ΟΙ ΥΠΟΛΟΙΠΕΣ ΚΛΑΣΕΙΣ `~OverviewPanel`**

Ίδια υλοποίηση με τις παραπάνω έχουν και οι κλάσεις `usersOverviewPanel`, `pcOverviewPanel`, `posOverviewPanel`, `prepayChargesOverviewPanel` με κάποιες μικρές διαφορές. Οι κλάσεις `posOverviewPanel`, `pcOverviewPanel` και `prepayChargesOverviewPanel` δεν χρειάζονται το `searchPanel` στο οποίο πραγματοποιούνται οι αναζητήσεις. Οι κλάσεις για την προσθήκη και τη μετατροπή χρησιμοποιούνται και από αυτές τις κλάσεις.

## 5.9 Η ΚΛΑΣΗ SessionsOverviewPanel

Το SessionsOverviewPanel έχει ως χαρακτηριστικό του το mainframe. Παρουσιάζει τρεις πίνακες της βάσης, τους member\_sessions, user\_Sessions και pos\_sessions ανάλογα με το ποία από τις μεθόδους init() και load() θα κληθεί κάθε φορά, μετά από ενέργεια του χρήστη. Συγκεκριμένα οι μέθοδοι αυτές είναι:

- initTableForMemberSessions()
- loadTableForMemberSessions()
- initTableForUserSessions()
- loadTableForUserSessions()
- initTableForPosSessions()
- loadTableForPosSessions()

Επιπλέον, και στους τρεις πίνακες υπάρχει η δυνατότητα να απομονωθούν οι εγγραφές για ένα μόνο μέλος ή χρήστη ή για ένα συγκεκριμένο χρονικό διάστημα, πράγμα το οποίο υλοποιούν οι παρακάτω μέθοδοι:

- filterByUserBtnActionPerformed()
- filterByMemberBtnActionPerformed()
- filterByDateBtnActionPerformed()
- filterByUserBtn1ActionPerformed()
- filterByMemberBtn1ActionPerformed()

- `filterByDate1BtnActionPerformed()`
- `filterByPosNameBtnActionPerformed()`
- `filterByDateBtn2ActionPerformed()`

## 5.10 Η ΚΛΑΣΗ **AdvancedOverviewPanel**

Και εδώ το `MainFrame` υπάρχει ως χαρακτηριστικό. Με τη βοήθεια των μεθόδων `pBtnActionPerformed()`, `oBtnActionPerformed()`, `vBtnActionPerformed()`, μπορούμε να αλλάξουμε τις τιμές `red`, `green`, `blue` του πίνακα `colors` της βάσης δεδομένων για την προπληρωμένη, την ανοιχτή χρέωση καθώς και τη χρέωση του `visitor`. Με τη `resetBtnActionPerformed()` μπορούμε να επαναφέρουμε τα αρχικά (default) χρώματα.

Με τη μέθοδο `changePasswdBtnActionPerformed`, υλοποιείται η διαδικασία με την οποία ο διαχειριστής του συστήματος αλλάζει τον κωδικό του.

Τέλος, οι μέθοδοι `pathBtnActionPerformed()` και `backupBtnActionPerformed()` υλοποιούν τη δημιουργία αντιγράφων ασφαλείας της βάσης δεδομένων. Μέσω της πρώτης ο χρήστης επιλέγει το `directory` στο οποίο θα αποθηκεύσει το αντίγραφο. Η δεύτερη, χρησιμοποιώντας το αντικείμενο της Java "Process" και τη μέθοδο `exec()`, εκτελεί την εντολή `"mysqldump --opt -hlocalhost -uusername -password[password] internet_cafe_db"` που αντιγράφει τη βάση δεδομένων και στην συνέχεια την αποθηκεύει στην τοποθεσία που προσδιορίστηκε από την `pathBtnActionPerformed()`.

## 5.11 Η ΚΛΑΣΗ ColorConfig

Η κλάση αυτή έχει ως μοναδικό σκοπό το χειρισμό των χρωμάτων που χρησιμοποιούνται στα PcPanels ως ένδειξη του είδους της χρέωσης. Με τις μεθόδους :

- `setPrepayChargeInformationPanelBgColor(r,g,b)`
- `setOpenChargeInformationPanelBgColor(r,g,b)`
- `setVisitorInformationPanelBgColor(r,g,b)`
- `setFreeInformationPanelBgColor(r,g,b)`

δίνεται η δυνατότητα να αλλάξουμε αυτά τα χρώματα. Για παράδειγμα, αυτές οι μέθοδοι θα κληθούν στο `advancedOverviewPanel` από τις μεθόδους για τη μετατροπή των χρωμάτων που διαθέτει.

Επιπλέον, η μέθοδος `resetAllBgColors()` επαναφέρει τα αρχικά (default) χρώματα.

## 5.12 Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΤΟΥ NET-PRO

Στην ενότητα αυτή θα προχωρήσουμε στην ανάλυση της Βάσης Δεδομένων που αναπτύχθηκε για να υποστηρίξει το Σύστημα Διαχείρισης Internet Café NET-PRO.

Ο κορμός της Βάσης αποτελείται από τους πίνακες `members`, `users`, `groups`, `pcs` και `pos` οι οποίοι παρουσιάζονται παρακάτω:

**Πίνακας: groups**

### Πεδία

id(int): ο αριθμητικός κωδικός του γκρουπ

name(varchar): το όνομα του γκρουπ

discount(int): η έκπτωση στη χρέωση για τα μέλη που ανήκουν σ'αυτό το γκρουπ

deleted(bool): αν το γκρουπ είναι διεγραμμένο ή όχι

### **Πίνακας: members**

#### Πεδία

id(int): ο αριθμητικός κωδικός του μέλους

surname(varchar): το επίθετο του μέλους

name(varchar): το όνομα του μέλους

id\_card\_number(varchar): ο αριθμός ταυτότητας του μέλους

telephone(varchar): το τηλέφωνο του μέλους

address(varchar): η διεύθυνση του μέλους

username(varchar): το όνομα χρήστη του μέλους

group\_id(int): ο αριθμητικός κωδικός του γκρουπ στο οποίο ανήκει το μέλος ως ξένο κλειδί σε αυτόν τον πίνακα

date\_registered(int): η ημερομηνία εγγραφής του μέλους

active(bool): αν το μέλος είναι ενεργό ή όχι

deleted(bool): αν το μέλος είναι διεγραμμένο ή όχι

### **Πίνακας: users**

#### Πεδία

id(int): ο αριθμητικός κωδικός του χρήστη

username(varchar): το όνομα χρήστη

passwd(varchar): ο κωδικός του

date\_registered(int): η ημερομηνία εγγραφής του

admin\_rights(bool): αν έχει τα δικαιώματα διαχειριστή ή όχι

active(bool): αν ο χρήστης είναι ενεργός ή όχι

deleted(bool): αν ο χρήστης είναι διεγραμμένος ή όχι

### **Πίνακας: pcs**

#### Πεδία

id(int): ο αριθμητικός κωδικός του υπολογιστή πελάτη

ip\_address(varchar): η ip διεύθυνση του υπολογιστή πελάτη

name(varchar): το όνομα του υπολογιστή πελάτη

state(int): η κατάσταση του υπολογιστή πελάτη. Αν δηλαδή είναι διαθέσιμος(ready) ή υπο επισκευή(on service)

deleted(deleted): αν ο υπολογιστής πελάτη είναι διεγραμμένος ή όχι

### **Πίνακας: pos**

#### Πεδία

id(int) : ο αριθμητικός κωδικός του προϊόντος της πώληση

name(varchar) : το όνομα του προϊόντος της πώληση

price(float) : η τιμή του προϊόντος της πώληση

quantity(int) : το απόθεμα του προϊόντος της πώληση στο κατάστημα

Οι υπόλοιποι πίνακες τη βάση παρουσιάζονται παρακάτω:

### **Πίνακας: user\_sessions**

#### Πεδία

id(int): ο αριθμητικός κωδικός της συνόδου χρήστη

user\_id(int): ο αριθμητικός κωδικός του χρήστη ως ξενο κλειδί σ' αυτό τον πίνακα

date\_opened(int): η ημερομηνία έναρξης της συνόδου

date\_closed(int): η ημερομηνία λήξης της συνόδου

money\_created(float): τα χρήματα που δημιουργήθηκαν κατά τη διάρκεια της συνόδου χρήστη υπολογισμένα από το σύστημα

money\_closed(float): τα χρήματα που καταμέτρησε ο χρήστης από το ταμείο

### **Πίνακας: seconds\_left**

#### Πεδία

member\_id(int): ο αριθμητικός κωδικός του μέλους ως ξένο κλειδί σ'αυτό τον πίνακα

seconds(int): τα δευτερολεπτα που έχει καταχωρημένα στο λογαριασμο του ένα μέλος

### **Πίνακας: prepay\_charges**

#### Πεδία

id(int): ο αριθμητικός κωδικός της προπληρωμένης χρέωσης

money(float): το ποσό χρημάτων ανα τα δευτερόλεπτα του πεδίου seconds

seconds(int): τα δευτερόλεπτα

deleted(bool): αν η χρέωση έχει διαγραφεί ή όχι

### **Πίνακας: member\_prepays**

#### Πεδία

id(int): ο αριθμητικός κωδικός της προπληρωμής που κάνει το μέλος

member\_id(int): ο αριθμητικός κωδικος του μέλους ως ξένο κλειδί σε αυτό τον πίνακα

user\_session\_id(int): ο αριθμητικός κωδικός της συνόδου χρήστη κατά την οποία λαμβάνει χώρα η προπληρωμή

money(float): το ποσό χρημάτων που εισάγεται στη συγκεκριμένη προπληρωμή

seconds(int): ο χρόνος σε δευτερόλεπτα που αντιστοιχεί στο ποσό χρημάτων που εισάγεται κατά την προπληρωμή

date(int): η ημερομηνία κατά την οποία πραγματοποιήθηκε η προπληρωμή

### **Πίνακας: member\_sessions**

#### Πεδία

id(int):ο αριθμητικός κωδικός της συνόδου μέλους

member\_id(int):ο αριθμητικός κωδικός του μέλους για το οποίο δημιουργείται η σύνοδος ως ξένο κλειδί σ'αυτό τον πίνακα



pc\_id(int): ο αριθμητικός κωδικός του υπολογιστή στον οποίο λαμβάνει χώρα η συνόδος ως ξένος κλειδί σ'αυτό τον πίνακα

user\_session\_id(int): ο αριθμητικός κωδικός της συνόδου χρήστη κατά την οποία **γίνεται η έναρξη** της συνόδου μέλους ως ξένο κλειδί σε αυτό τον πίνακα

date\_opened(int) : ο χρόνος έναρξης της συνόδου μέλους σε δευτερόλεπτα

date\_closed(int) : ο χρόνος λήξης της συνόδου μέλους σε δευτερόλεπτα

money\_created(float) : το ποσό χρημάτων που χρεώθηκε το μέλος

type\_of\_charge(int) : το είδος της χρέωσης που χαρακτηρίζει τη συγκεκριμένη σύνοδο μέλους. Δηλαδή το αν ήταν ανοιχτή ή προπληρωμένη

### **Πίνακας: open\_charge**

#### Πεδία

id(int) : ο αριθμητικός κωδικός της ανοιχτής χρέωσης

fromsecond(int) : ο αριθμός 0 που υποδηλώνει ότι η χρέωση ισχύει από της 12 το βράδυ

tosecond(int) : ο αριθμός 86400 που υποδηλώνει ότι η χρέωση ισχύει μέχρι

price(float) : η τιμή ανά ώρα

deleted(bool) : αν η χρέωση είναι διεγγραμμένη ή όχι

### **Πίνακας: pos\_sales**

#### Πεδία

id(int):ο αριθμητικός κωδικός της πώλησης

pos\_id(int): ο αριθμητικός κωδικός του προϊόντος το οποίο πωλείται, ως ξένο κλειδί σε αυτό τον πίνακα

user\_session\_id(int): ο αριθμητικός κωδικός της συνόδου χρήστη κατά την οποία κατά την οποία πραγματοποιείται η πώληση ως ξένο κλειδί σε αυτό τον πίνακα

member\_id(int): ο αριθμητικός κωδικός του μέλους στο οποίο πραγματοποιείται η πώληση ως ξένο κλειδί σε αυτό τον πίνακα

quantity(int) : ο αριθμός αντιγράφων που πωλούνται

date\_sold(int) : η ημερομηνία κατά την οποία πραγματοποιείται η πώληση

## Πίνακας: colors

### Πεδία

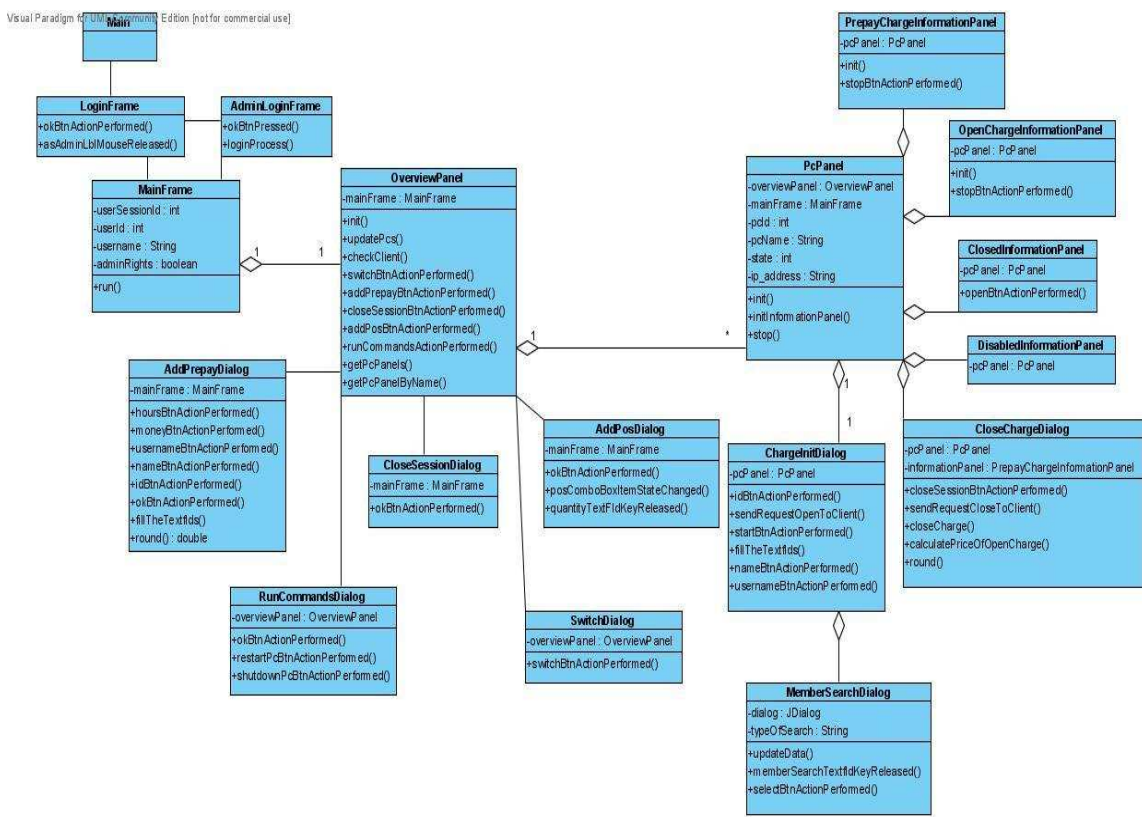
id(int) : ο αριθμητικός κωδικός του χρώματος

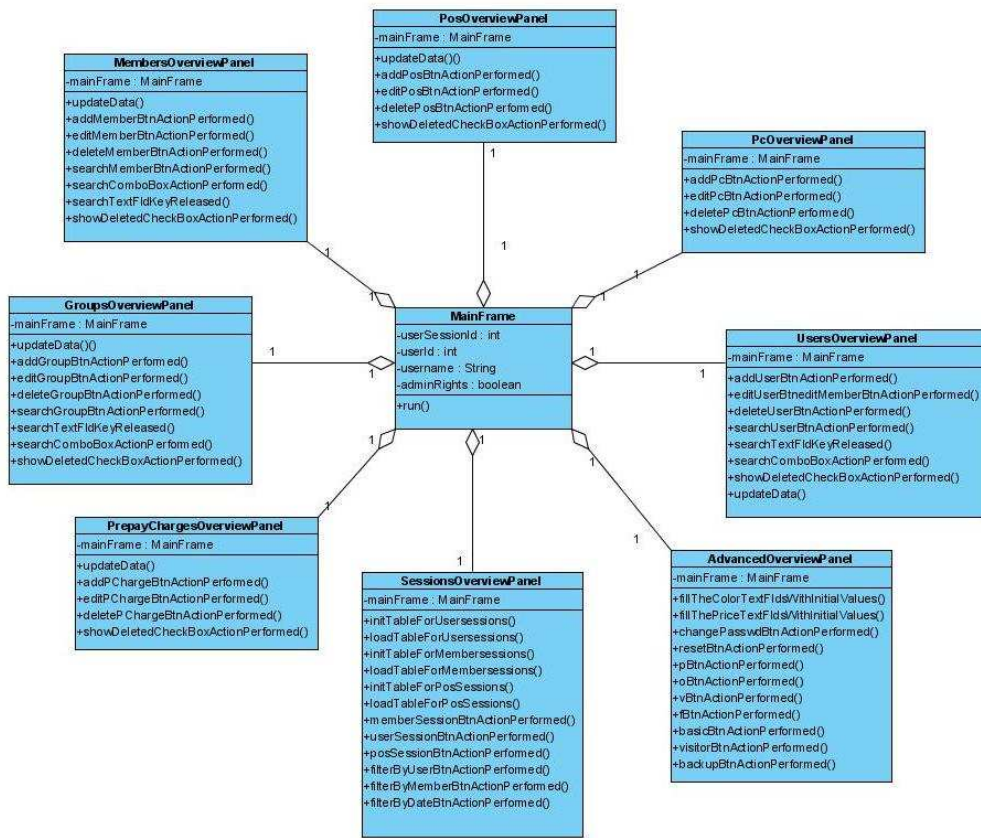
red(int) : ο αριθμός που αντιστοιχεί στην κόκκινη απόχρωση

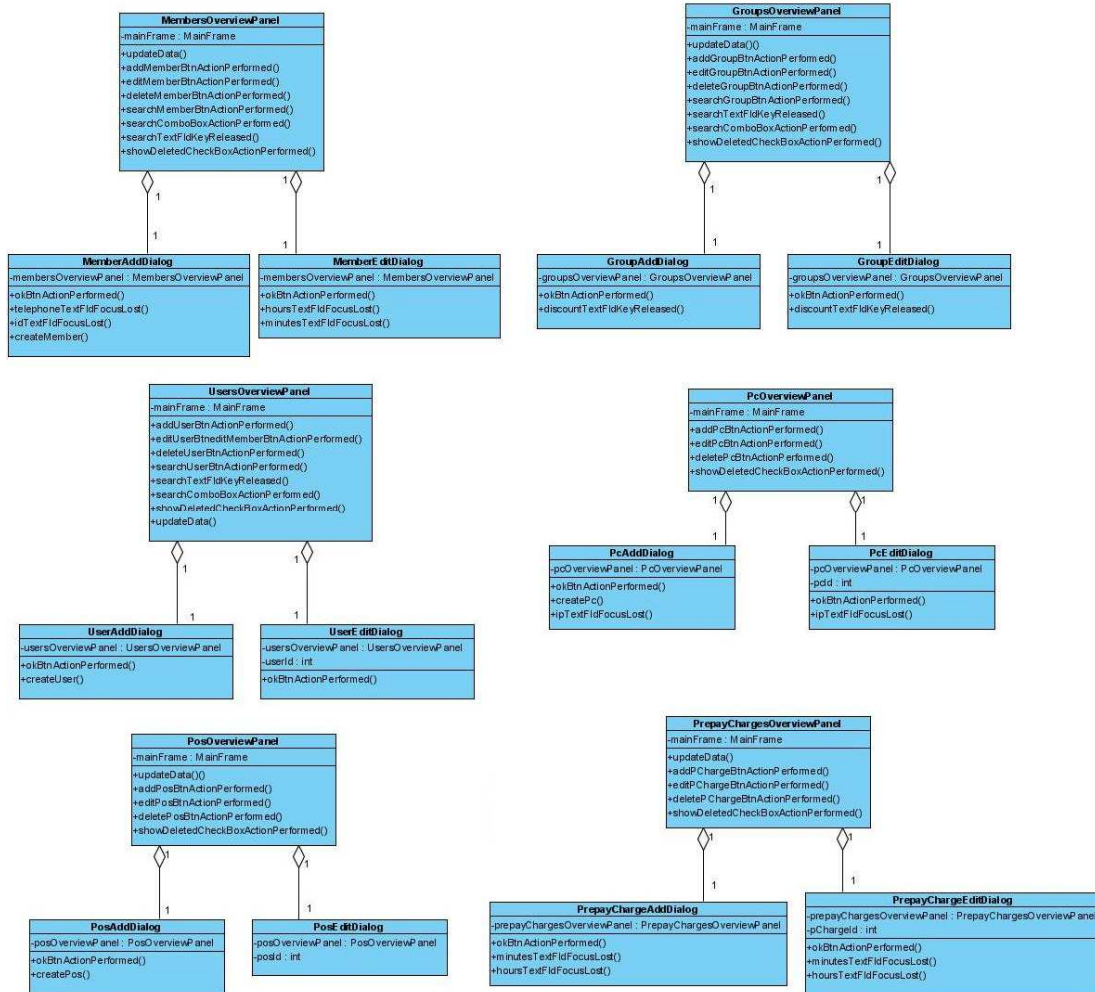
green(int) : ο αριθμός που αντιστοιχεί στην πράσινη απόχρωση

blue(int) : ο αριθμός που αντιστοιχεί στη μπλε απόχρωση

## 5.13 ΤΑ ΔΙΑΓΡΑΜΜΑΤΑ ΚΛΑΣΕΩΝ







## **ΚΕΦΑΛΑΙΟ 6**

---

### *Οι λειτουργίες του Net-Pro Client*

Στο κεφάλαιο αυτό εξετάζουμε με λεπτομέρεια όλες τις λειτουργίες που εκτελούνται στην πλευρά των υπολογιστών πελάτη σε κάθε έναν από τους οποίους εκτελείται η υποεφαρμογή Net-Pro Client. Με λίγα λόγια, ο λόγος ύπαρξης αυτής της υποεφαρμογής εξυπηρετεί το σκοπό, να αποτρέπεται η πρόσβαση σε μη εξουσιοδοτημένους πελάτες κλειδώνοντας και ξεκλειδώνοντας τους υπολογιστές πελατών κατά το κλείσιμο και το άνοιγμα συνόδων μελών αντίστοιχα.

## 6.1 ΚΛΕΙΔΩΜΕΝΗ ΚΑΤΑΣΤΑΣΗ

Κλειδωμένη είναι η κατάσταση κατά την οποία δεν υπάρχει ανοιχτή σύνοδος μέλους σε έναν υπολογιστή πελάτη(Εικόνα 6.1). Στην κατάσταση αυτή είναι ενεργό το PcClosedFrame. Το frame αυτό αποκλείει τη χρήση του υπολογιστή πελάτη καθώς εκτείνεται σε όλη την οθόνη(full screen), δεν υπάρχει δυνατότητα μεγιστοποίησης, ελαχιστοποίησης και κλεισίματος, ενώ και τα πλήκτρα που μπορούν να το απενεργοποιήσουν(ctrl-alt-delete, windows key) είναι απενεργοποιημένα. Από την κατάσταση αυτή μπορούμε να περάσουμε στην κατάσταση ανοιχτής συνόδου μέλους με δύο τρόπους.

- Να αποσταλεί αίτημα "OpenClient" από τον NET-PRO Server.
- Να αποσταλεί αίτημα "RequestOpen" από τον NET-PRO Client προς τον NET-PRO Server και στη συνέχεια να αποστείλει αίτημα ο NET-PRO Server "OpenClient" ως απάντηση στο "RequestOpen"

Το "RequestOpen" θα αποσταλεί, όταν ένα μέλος ζητήσει να ανοιχτεί σύνοδος στον λογαριασμό του συμπληρώνοντας το username και το password του στο LoginPanel του NET-PRO Client που τρέχει σε ένα συγκεκριμένο υπολογιστή πελάτη. Το panel αυτό φαίνεται στην εικόνα 6.2. Μόλις σταλεί το αίτημα, ο NET-PRO Server θα ελέγξει τα στοιχεία του μέλους στη βάση και με την προϋπόθεση ότι αυτά είναι έγκυρα και ότι υπάρχει προπληρωμένος χρόνος στο λογαριασμό του θα απαντήσει με αίτημα "OpenClient". Ο NET-PRO Client θα περάσει σε κατάσταση ανοιχτής συνόδου μέλους.

## 6.2 ΚΑΤΑΣΤΑΣΗ ΑΝΟΙΧΤΗΣ ΣΥΝΟΔΟΥ

Είναι η κατάσταση κατά την οποία υπάρχει ανοιχτή σύνοδος μέλους στον υπολογιστή πελάτη. Αυτό σημαίνει ότι ένας πελάτης χρησιμοποιεί τον συγκεκριμένο υπολογιστή είτε με ελεύθερη χρέωση είτε με προπληρωμή. Το PcClosedFrame είναι απενεργοποιημένο για όλη τη διάρκεια της συνόδου ενώ παράλληλα ενεργοποιείται το PcOpenedFrame. Εδώ συναντάμε δύο λειτουργίες.

Η πρώτη είναι η εμφάνιση του χρόνου. Αυτός ο χρόνος είναι είτε ο χρόνος που έχει παρέλθει από την έναρξη της συνόδου(σε περίπτωση ελεύθερης χρέωσης) είτε ο χρόνος που απομένει μέχρι το τέλος της συνόδου(σε περίπτωση προπληρωμής). Αυτό υλοποιείται με τη βοήθεια της διαδικασίας αποστολής αιτημάτων "RequestTime" και "ReplyTime" ανάμεσα στις δύο υποεφαρμογές. Με τη βοήθεια ενός νήματος τα αιτήματα αυτά ανταλλάσσονται κάθε ένα λεπτό.

Η δεύτερη είναι η λειτουργία "Logout". Η λειτουργία αυτή είναι ενεργοποιημένη μόνο στην περίπτωση προπληρωμένης συνόδου μέλους. Ουσιαστικά, το μέλος που έχει προπληρώσει κάποιο χρόνο και έχει ανοιχτή σύνοδο μπορεί να επιλέξει να τερματίσει τη σύνοδο αυτή από τον υπολογιστή πελάτη. Στην περίπτωση αυτή, ο NET-PRO Client στέλνει αίτημα "PcClose" στον NET-PRO Server. Αυτός με τη σειρά του, αφού ενημερώσει τη βάση δεδομένων για τη σύνοδο που πρόκειται να κλείσει, στέλνει αίτημα "RequestClose" πίσω στον NET-PRO Client. Το PcClosedFrame ενεργοποιείται και η διαδικασία ολοκληρώνεται.

# ΕΠΙΛΟΓΟΣ

---

## *Συμπεράσματα*

Στα είκοσι περίπου χρόνια ζωής των καταστημάτων Internet Café, απαραίτητο συμπλήρωμά τους αποτελεί ένα σύστημα για τη διαχείρισή τους. Αυτό το σύστημα διαχείρισης πέρασε από πολλά στάδια, ξεκινώντας από ένα απλό τετράδιο που κατέγραφε το ονοματεπώνυμο των πελατών και το χρόνο χρήσης του υπολογιστή που αντιστοιχούσε σε αυτό τον πελάτη και καταλήγοντας στην υπάρχουσα σύγχρονη και πολυσύνθετη μορφή.

Η εφαρμογή αυτή που αναπτύχθηκε στα πλαίσια της πτυχιακής ήρθε για να δώσει τη λύση σε όλα τα υπάρχοντα προβλήματα οργάνωσης ενός καταστήματος τέτοιου είδους με τη μορφή που συναντάται σήμερα. Από την πολύ απλή ηλεκτρονική καταχώρηση των ημερήσιων εσόδων μέχρι την πολύπλοκη επικοινωνία μεταξύ εφαρμογών με χρήση του δικτύου.



Μελετήσαμε τους λόγους εμφάνισης αυτών των καταστημάτων, τη διαδρομή που ακολούθησαν από τη στιγμή που δημιουργήθηκαν μέχρι σήμερα, διαπιστώσαμε την σύγχρονη τεράστια εξάπλωσή τους και συγκεντρώσαμε τα επικρατέστερα υπάρχοντα συστήματα για τη διαχείριση τους. Τέλος, συνδυάζοντας και επαυξάνοντας τις δυνατότητες τους, δημιουργήσαμε ένα καινούργιο σύστημα το οποίο εκπληρώνει απόλυτα το σκοπό του.

Στη συνέχεια περιγράψαμε την οργάνωση, τα χαρακτηριστικά και τους ρόλους ενός Internet Café, στη μορφή που συναντάται σήμερα, καθώς και τις έννοιες και τη χρησιμότητα των καταστάσεων πρόσβασης στο σύστημα που αναπτύξαμε.

Τέλος, προχωρήσαμε στην ανάλυση του βήμα προς βήμα. Αναφέραμε όλα τα απαραίτητα στοιχεία της γλώσσας προγραμματισμού στην οποία αναπτύχθηκε και μιλήσαμε για τις δύο εφαρμογές από τις οποίες αποτελείται και τις λειτουργίες που καθεμία από αυτές πραγματοποιεί.

# ΠΑΡΑΡΤΗΜΑ Α

*Εγκατάσταση και χρήση*

---

## ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

### **ΒΗΜΑ 1<sup>ο</sup> – ΕΓΚΑΤΑΣΤΑΣΗ ΤΗΣ MYSQL**

Εκτελέστε το setup.exe του αρχείου mysql.zip που βρίσκεται μέσα στο CD. Επιλέξτε `typical` να πραγματοποιηθεί τυπική εγκατάσταση (Typical installation). Αφού ολοκληρωθεί η εγκατάσταση της mysql προχωρήστε στην παραμετροποίηση του mysql server. Επιλέξτε τα παρακάτω:

Configuration type -> Detailed Configuration

Server type -> Developer machine

Database Usage -> Multifunctional Database

---

Επιλέξτε ως default character set το Manual selected default character set / collaboration. Στο security settings εισάγετε το username και το password της επιλογής σας. Ξετικάρτε το Modify Security Settings και προχωρήστε στην εκτέλεση.

---

## **ΒΗΜΑ 2° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ JAVA RUNTIME ENVIRONMENT**

Εκτελέστε το setup.exe του αρχείου jruntime.zip που βρίσκεται μέσα στο CD.

## **ΒΗΜΑ 3° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ NET-PRO SERVER**

Η εγκατάσταση πρέπει να γίνει μόνο στον κεντρικό υπολογιστή του καταστήματος. Εκτελέστε το αρχείο setup.exe που βρίσκεται μέσα στο φάκελο Net-Pro Server. Αφού επιλέξετε το φάκελο προορισμού που επιθυμείτε πραγματοποιήστε την εγκατάσταση.

---

## **ΒΗΜΑ 4° – ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΒΑΣΗΣ ΤΟΥ NET-PRO**

Κάντε κλικ στο κουμπί «Έναρξη». Επιλέξτε «Εκτέλεση». Με την εντολή cd πηγαίνετε στο directory που βρίσκεται ο φάκελος Net-Pro Server. Στη συνέχεια πληκτρολογήστε την εντολή:

```
Mysql -uroot -pPASSWORD < internetCafeDB.sql
```

Όπου PASSWORD πληκτρολογήστε τον κωδικό που εισάγατε κατά την παραμετροποίηση.

---

## **ΒΗΜΑ 5° – ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ NET-PRO CLIENT**

Η εγκατάσταση πρέπει να γίνει σε κάθε έναν από τους υπολογιστές πελατών. Εκτελέστε το αρχείο setup.exe που βρίσκεται μέσα στο φάκελο Net-Pro Server. Αφού επιλέξετε το φάκελο προορισμού που επιθυμείτε πραγματοποιήστε την εγκατάσταση. Ανοίξτε το αρχείο config.txt που βρίσκεται μέσα στο φάκελο προορισμού και καταχωρείστε ως ip\_address την ip διεύθυνση του κεντρικού υπολογιστή του καταστήματος. Προαιρετικά μπορείτε να τροποποιήσετε τα πεδία server\_port και client\_port. Αν

πραγματοποιήσετε αυτή την ενέργεια, θα πρέπει να εισάγετε τις ίδιες τιμές αντίστροφα στο αντίστοιχο αρχείο του κεντρικού υπολογιστή του καταστήματος.

Για παράδειγμα:

Αρχείο config.txt του Net-Pro Client:

```
[server_port]=48002
```

```
[client_port]=48003
```

Αντίστοιχο αρχείο του Net-Pro Server:

```
[server_port]=48003
```

```
[client_port]=48002
```

Στη συνέχεια πηγαίνετε στο φάκελο reg και κάντε δεξί κλικ στο αρχείο disableWindowsKey.reg. Επιλέξτε «Merge». Επανεκκινήστε τον υπολογιστή σας.

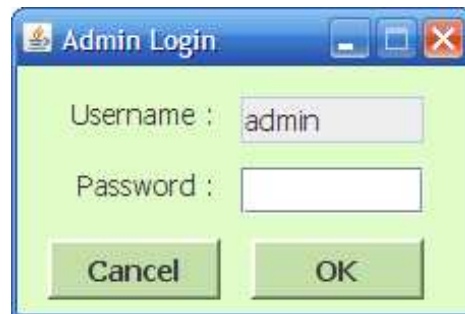
---

## ΧΡΗΣΗ

### 1. ΣΥΝΔΕΣΗ ΣΕ ΚΑΤΑΣΤΑΣΗ ΔΙΑΧΕΙΡΙΣΤΗ

Είναι το αρχικό και απαραίτητο στάδιο για την είσοδο του διαχειριστή στο σύστημα. Ο διαχειριστής στο παράθυρο που ανοίγει επιλέγει "Login as admin" και μεταφέρεται σε περιβάλλον ειδικό για την σύνδεση του. Στη βάση του συστήματος υπάρχει ένας πίνακας που ονομάζεται users. Ο πίνακας αυτός κατά την αρχική εγκατάσταση του προγράμματος τροφοδοτείται με μια εγγραφή που έχει ως εξ'ορισμού (default) τιμές, όνομα χρήστη (username) το "admin" και κωδικό(password) επίσης το "admin". Το όνομα χρήστη θα παραμείνει αυστηρά το ίδιο για όλη τη διάρκεια κατά την οποία χρησιμοποιείται το NET-PRO. Ο κωδικός φυσικά θα πρέπει να μετατραπεί σε αυτό που θα επιλέξει ο διαχειριστής. Μόλις ο διαχειριστής εισάγει τα στοιχεία του και πατήσει το κουμπί «ok» ολοκληρώνεται η διαδικασία της σύνδεσης και ο διαχειριστής εισάγεται στο κυρίως πρόγραμμα σε κατάσταση διαχειριστή. Είναι απαραίτητο να σημειώσουμε ότι ο διαχειριστής μπαίνοντας σε admin mode δεν έχει το δικαίωμα να προσθέτει σε πελάτες προπληρωμένο χρόνο ή

να ανοίγει συνεδρίες για αυτούς. Στην παρακάτω εικόνα παρουσιάζεται το παράθυρο σύνδεσης στην κατάσταση διαχειριστή.



Εικόνα 1.1

## 2. ΣΥΝΔΕΣΗ ΣΕ ΚΑΤΑΣΤΑΣΗ ΑΠΛΟΥ ΚΑΙ ΙΣΧΥΡΟΥ ΧΡΗΣΤΗ

Ο υπάλληλος και χρήστης του συστήματος, διαθέτει και αυτός ένα λογαριασμό και ειδικότερα μια εγγραφή στον πίνακα "users" της βάσης δεδομένων. Μόλις εισάγει το όνομα χρήστη και τον κωδικό του μπορεί να συνδεθεί στο σύστημα σε κατάσταση απλού ή ισχυρού χρήστη ανάλογα με το αν διαθέτει τα απαραίτητα δικαιώματα ή όχι. Αυτό που παρουσιάζει ιδιαίτερο ενδιαφέρον είναι η περίπτωση κατά την οποία ο διαχειριστής πιθανόν να διαγράψει τον χρήστη που έχει ανοιχτή συνεδρία στο σύστημα. Το πρόβλημα αυτό έχει αντιμετωπιστεί συνεχίζοντας κανονικά τη συνεδρία του συγκεκριμένου χρήστη μέχρι αυτός να πατήσει έξοδο από το πρόγραμμα. Αν ακολουθήσει την τυπική έξοδο με κλείσιμο της συνεδρίας(Session close) τότε τα χρήματα τα οποία εισάγει κατά το κλείσιμο θα αποθηκευτούν κανονικά στην βάση δεδομένων. Αν ακολουθήσει έξοδο χωρίς κλείσιμο συνεδρίας τότε η εφαρμογή θα κλείσει από μόνη της τη συνεδρία με μηδενικό ποσό χρημάτων στο κατάλληλο πεδίο. Όταν ο χρήστης προσπαθήσει να εισαχτεί ξανά στο σύστημα θα ενημερωθεί με ένα μήνυμα ότι είναι διαγραμμένος και δεν έχει κανένα δικαίωμα πρόσβασης. Στην παρακάτω εικόνα παρουσιάζεται το παράθυρο σύνδεσης στην κατάσταση απλού και ισχυρού χρήστη.



Εικόνα 2 1

---

### **3. ΟΙ ΛΕΙΤΟΥΡΓΙΕΣ ΤΟΥ OVERVIEW PANEL**

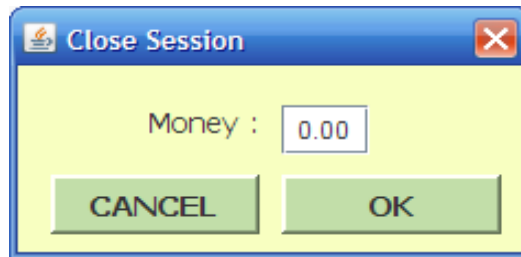
Το πρώτο πράγμα που θα αντικρύσει ο χρήστης αφού συνδεθεί στο σύστημα είναι το Overview Panel. Εδώ παρουσιάζονται με λεπτομέρεια οι υπολογιστές πελατών που διαθέτει το κατάστημα δίνοντας πληροφορίες για την κατάσταση τους και εκτελούνται οι λειτουργίες της προπληρωμής (Prepay Charge), της ανοιχτής χρέωσης (Open Charge), της πώλησης προϊόντων (Pos sale), της μεταφοράς συνόδου πελάτη από έναν υπολογιστή σε έναν άλλο (Switch) και του κλεισίματος συνόδου χρήστη (Close Session).

---

#### **3.1. ΚΛΕΙΣΙΜΟ ΣΥΝΟΔΟΥ ΧΡΗΣΤΗ**

Μόλις ο χρήστης πατήσει σε αυτή την επιλογή θα ανοίξει το παράθυρο για το κλείσιμο της συνόδου του. Εκεί είναι αναγκαία προϋπόθεση να εισαχθούν τα χρήματα κλεισίματος συνόδου. Ο χρήστης θα πρέπει να κάνει καταμέτρηση του ταμείου του και να υπολογίσει με αυτό τον τρόπο τα χρήματα που παρήγαγε κατά τη διάρκεια της συνόδου του. Τα χρήματα αυτά προέρχονται είτε από εισαγωγές προπληρωμής είτε από ανοιχτές χρεώσεις είτε από πωλήσεις προϊόντων. Ο σκοπός ύπαρξης αυτής της λειτουργίας εξυπηρετεί την ανάγκη του διαχειριστή να ελέγχει τις εισροές χρημάτων στην επιχείρησή του. Μόλις ολοκληρωθεί η διαδικασία το πρόγραμμα θα τερματιστεί και όταν ξανανοίξει θα δημιουργηθεί νέα σύνοδος χρήστη. Στην εικόνα 3.1.1 φαίνεται

το παράθυρο στο οποίο ο χρήστης εισάγει το ποσό χρημάτων και τερματίζει τη σύνοδο του.



Εικόνα 3.1.1

---

### **3.2. ΧΡΕΩΣΗ ΜΕ ΠΡΟΠΛΗΡΩΜΗ**

Χρέωση με προπληρωμή ονομάζεται η διαδικασία κατά την οποία μετά από επιθυμία του πελάτη εισάγεται στο λογαριασμό του ένα ποσό χρημάτων το οποίο θα μπορεί να εκμεταλλευτεί οποιαδήποτε στιγμή επιθυμεί και όχι απαραίτητα εξολοκλήρου σε μια σύνοδο μέλους. Η λειτουργία αυτή ολοκληρώνεται μετά από δυο στάδια.

Το πρώτο είναι η εισαγωγή του χρήστη που μας αφορά και η επαλήθευση του. Η διαδικασία αυτή όπως έχει σχεδιαστεί από το NET-PRO είναι απλή για το χρήστη και προσφέρει την άνεση και στο χρήστη αλλά και στο διαχειριστή να την πραγματοποιήσουν έχοντας παραμόνο ένα οποιοδήποτε στοιχείο για τον πελάτη μεταξύ του αριθμητικού κωδικού μέλους, ονοματεπωνύμου, και ονόματος μέλους(member username). Αν ο πελάτης γνωρίζει τον κωδικό μέλους τότε αυτός εισάγεται στο σύστημα και αμέσως μετά παράγονται τα υπόλοιπα στοιχεία του μέλους από τη βάση δεδομένων και επαληθεύεται η ταυτότητα του. Αν δεν τον γνωρίζει, υπάρχει η δυνατότητα της αναζήτησης του με βάση είτε το επίθετο είτε το username του.

Στην εικόνα 3.2.1 φαίνεται το παράθυρο προπληρωμής και στην εικόνα 3.2.2 ο τρόπος με τον οποίο γίνεται αναζήτηση του μέλους με βάση το επίθετο του.

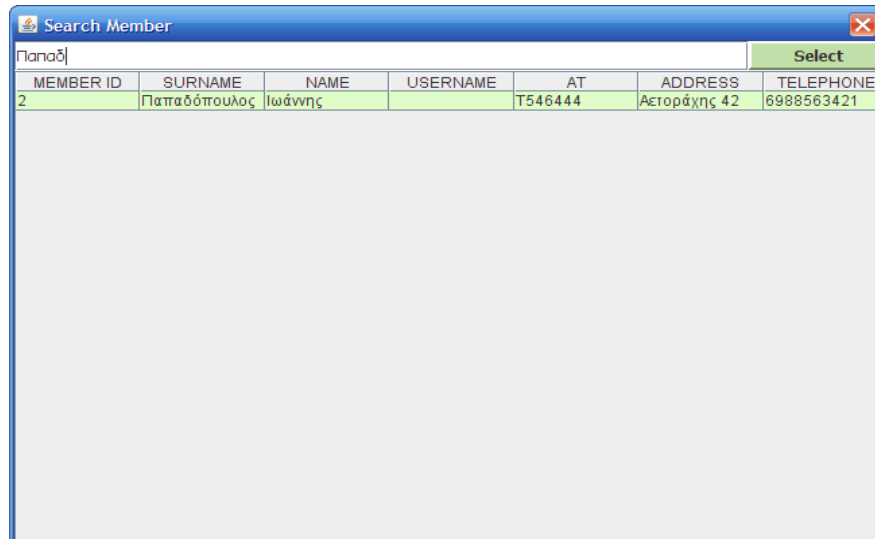
Dialog box titled "Add Prepay" with the following fields and controls:

- ID:  [...]
- Surname-Name:  [...]
- Username:  [...]
- Time left: 00:00
- Section: Add money/hours :
  - money:  [...]
  - hours:  :  [...]
- Buttons: CANCEL, OK

Εικόνα 3.2.1

Το δεύτερο βήμα είναι η εισαγωγή χρημάτων στο λογαριασμό του χρήστη είτε με μορφή πόσου χρημάτων είτε με μορφή χρονικού διαστήματος. Ένα μέλος έχει τη δυνατότητα να τροφοδοτήσει το λογαριασμό του με δυο διαφορετικούς τρόπους οι οποίοι θα φέρουν το ίδιο ακριβώς αποτέλεσμα. Για παράδειγμα μπορεί να ζητήσει να μπουν 5 ευρώ ή 5 ώρες. (Το παράδειγμα είναι τυχαίο και δε σημαίνει ότι η χρέωση του καταστήματος είναι απαραίτητα 1 ευρώ την ώρα). Το γεγονός αυτό καθιστά την όλη διαδικασία ιδιαίτερα λειτουργική καθώς για παράδειγμα ένα μέλος μπορεί πολύ εύκολα να ζητήσει να εισαχθούν στο λογαριασμό του ακριβώς μια ώρα και 43 λεπτά!





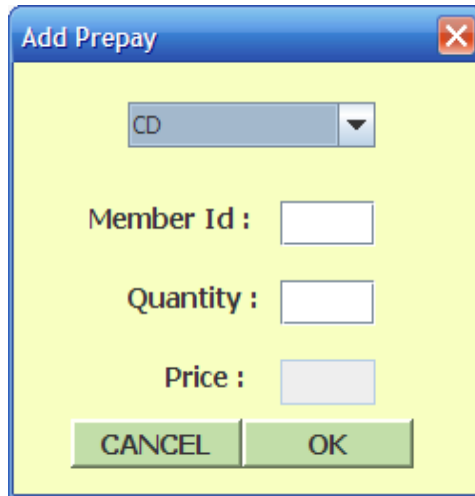
Εικόνα 3.2.2

---

### 3.3. ΠΩΛΗΣΗ ΠΡΟΪΟΝΤΩΝ

Στην εφαρμογή που αναπτύξαμε, θεωρήσαμε απαραίτητο, όλα τα έσοδα που μπορούν να προκύψουν ανεξαρτήτως τρόπου να εισάγονται στη βάση δεδομένων και να καταγράφονται λεπτομερώς. Έτσι προστέθηκε μια επιπλέον λειτουργία, η οποία δεν έχει καμιά σχέση με συνοδούς μελών αλλά παρόλα αυτά καταγραφεί απαραίτητα στοιχεία για την ορθή λειτουργία ενός καταστήματος Internet Café. Ένα τέτοιο κατάστημα εκτός από τις υπηρεσίες internet που προσφέρει είναι πιθανόν να ασχολείται και με πωλήσεις προϊόντων όπως cd, εκτυπώσεις κτλ. Αυτά τα προϊόντα εντάσσονται στη συγκεκριμένη λειτουργία. Ο χρήστης μετά από παρότρυνση του μέλους μπορεί να πουλήσει εκείνη τη στιγμή πολλά αντίγραφα ταυτόχρονα. Τέλος καταγράφεται ο νέος τρέχων αριθμός αντιγράφων του προϊόντος καθώς έχουν μειωθεί από την πώληση.

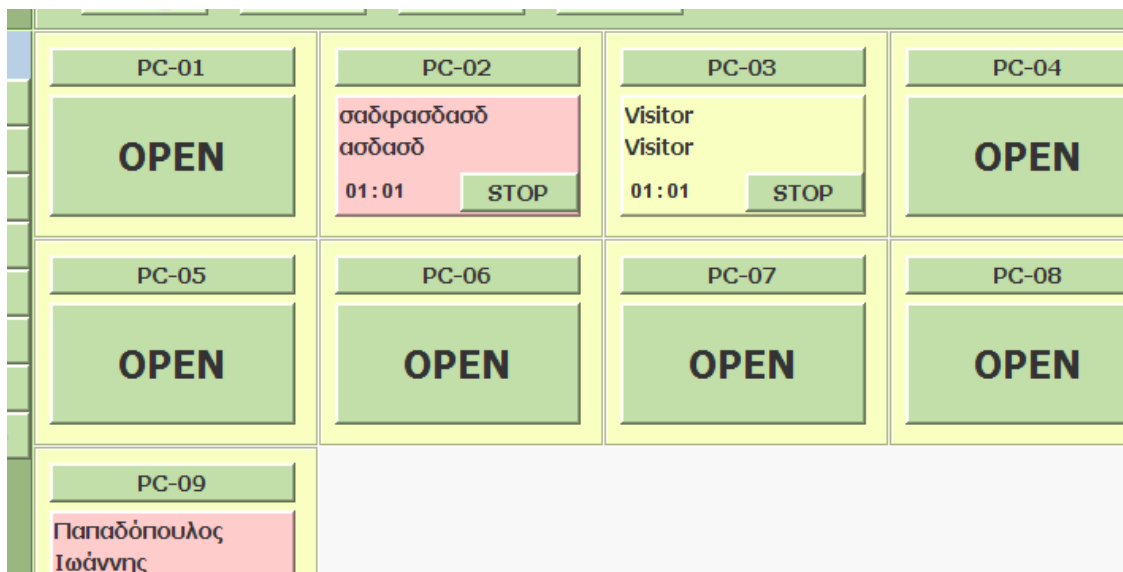
Το παράθυρο στο οποίο ο χρήστης χειρίζεται τη λειτουργία της πώλησης προϊόντων παρουσιάζεται στην εικόνα 3.3.1



Εικόνα 3.3.1

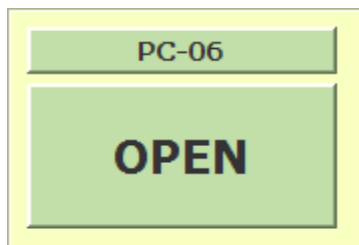
### 3.4. ΕΠΟΠΤΕΙΑ ΥΠΟΛΟΓΙΣΤΩΝ ΠΕΛΑΤΗ

Η εικόνα 3.4.1 παρουσιάζει το Panel που περιέχει μικρότερα Panels με πληροφορίες για κάθε έναν από τους υπολογιστές πελάτη του καταστήματος. Κάθε Panel αρχικά δείχνει την κατάσταση του υπολογιστή. Ένας υπολογιστής μπορεί να είναι κλειστός(να μην υπάρχει σύνδεσος μέλους σε αυτόν), μπορεί να είναι ανοιχτός(να υπάρχει σύνδεσος μέλους), ή ακόμα και να είναι υπό συντήρηση(on service).



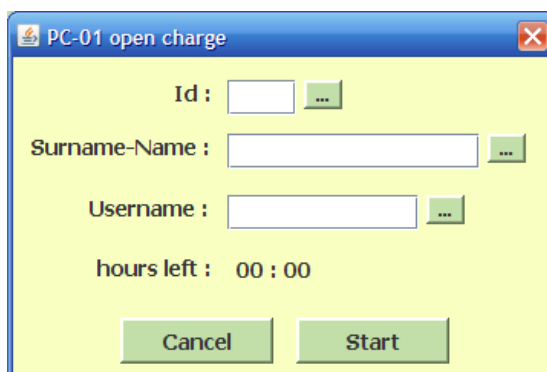
Εικόνα 3.4.1

Σε περίπτωση που είναι κλειστός παρουσιάζει ένα κουμπί που δίνει τη δυνατότητα στο χρήστη να ανοίξει καινούργια σύνοδο μέλους. Αυτό φαίνεται στην εικόνα 3.4.2.

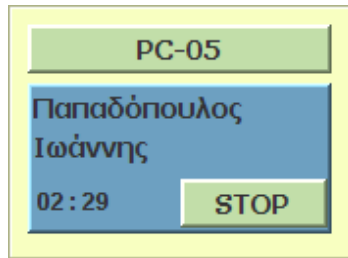


Εικόνα 3.4.2

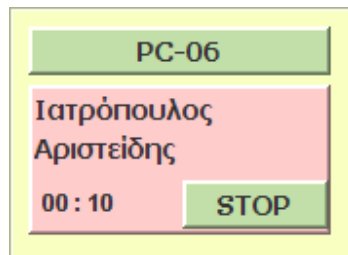
Μόλις ο χρήστης πατήσει αυτό το κουμπί ανοίγει ένα νέο παράθυρο(εικόνα 3.4.3) το οποίο δημιουργεί νέα σύνοδο μέλους και θα αναλυθεί παρακάτω. Μετά την ολοκλήρωση της διαδικασίας αυτής το Panel που αντιστοιχεί στο συγκεκριμένο υπολογιστή αλλάζει μορφή και παίρνει μια από αυτές που παρουσιάζονται στις εικόνες 3.4.4 και 3.4.5.



Εικόνα 3.4.3



Εικόνα 3.4.4



Εικόνα 3.4.5

Η πρώτη εικόνα δείχνει τη μορφή του Panel για έναν υπολογιστή στον οποίο υπάρχει ανοιχτή σύνοδος μέλους με προπληρωμή. Το Panel αυτό ονομάζεται Panel προπληρωμένης χρέωσης(Prepay Charge Panel). Παρουσιάζονται τα στοιχεία του μέλους, πόσος χρόνος του απομένει και μια επιλογή για τερματισμό της συνόδου. Ο χρόνος μετράει αντίστροφα, υποδεικνύει το πόσος χρόνος απομένει μέχρι τον αναγκαστικό τερματισμό της συνόδου από το σύστημα και η ένδειξη του ανανεώνεται κάθε ένα λεπτό. Αν ο χρήστης δεν επιλέξει ποτέ τερματισμό της συνόδου, τότε το πρόγραμμα θα τερματίσει αυτόματα τη σύνοδο όταν ο χρόνος φτάσει στο μηδέν.

Η δεύτερη εικόνα δείχνει τη μορφή του Panel για έναν υπολογιστή στον οποίο υπάρχει ανοιχτή συνοδος μέλους με ανοιχτή χρέωση. Το Panel αυτό ονομάζεται Panel προπληρωμένης χρέωσης(Open Charge Panel). Παρατηρούμε ότι το χρώμα είναι διαφορετικό πράγμα το οποίο έχει ρυθμιστεί έτσι από το διαχειριστή με σκοπό να διακρίνονται οι διαφορετικού τύπου συνοδοί. Επιπλέον ο χρόνος εδώ μετράει με αύξοντα ρυθμό και είναι

ενδεικτικός του ποσού χρόνου πέρασε από τη στιγμή που δημιουργήθηκε η συγκεκριμένη συνοδός χρήστη. Στη συγκεκριμένη περίπτωση της ανοιχτής χρέωσης, ο μονός τρόπος να τερματιστεί μια συνοδός μέλους είναι με την παρέμβαση του χρήστη. Δεν δίνεται η δυνατότητα να γίνει αυτόματα.

Αξίζει να σημειωθεί ότι αν στο λογαριασμό ενός μέλους που βρίσκεται σε ανοιχτή σύνοδο με προπληρωμή προστεθεί επιπλέον χρηματικό ποσό, τότε το Panel που παρουσιάζει τη συγκεκριμένη σύνοδο θα ενημερωθεί και θα δείχνει τον νέο χρόνο που απομένει μέχρι τον τερματισμό.

---

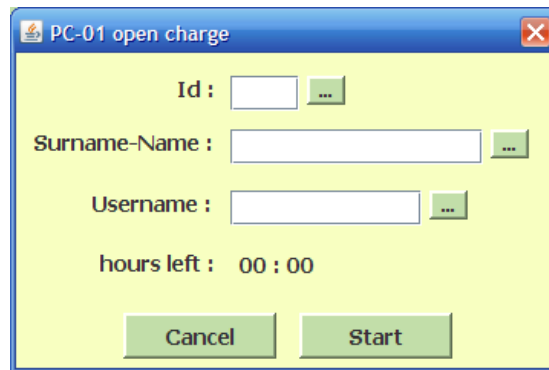
### **3.5. ΕΝΑΡΞΗ ΝΕΑΣ ΣΥΝΟΔΟΥ ΧΡΗΣΤΗ**

Στο κομμάτι αυτό της εφαρμογής πραγματοποιείται η διαδικασία της αρχικοποίησης συνόδων για τους πελάτες που το επιθυμούν. Αφού ο πελάτης ζητήσει να καθίσει σε κάποιο υπολογιστή ο υπάλληλος θα πατήσει στο κουμπί “Open” **του συγκεκριμένου υπολογιστή** και θα ανοίξει το κατάλληλο παράθυρο. Εκεί ο υπάλληλος(χρήστης) θα εισάγει τα στοιχεία του μέλους με τον ίδιο τρόπο που περιγράψαμε στο κεφάλαιο της προπληρωμής. Η διαφορά με την διαδικασία της προπληρωμής εντοπίζεται στο γεγονός ότι δεν υπάρχει η δυνατότητα προσθήκης χρόνου. Αντίθετα, υπάρχει απλά μια ενημέρωση για το υπόλοιπο του χρόνου που έχει το συγκεκριμένο μέλος στο λογαριασμό του. Στο σημείο αυτό το σύστημα θα αποφασίσει αυτόματα για το είδος της χρέωσης της συνόδου(ανοιχτή ή προπληρωμένη). Αν το υπόλοιπο του χρόνου είναι μηδενικό τότε το Panel που θα ανοίξει θα είναι το Open Charge Panel, δηλαδή η χρέωση θα είναι ανοιχτή. Αν το υπόλοιπο του χρόνου δεν είναι μηδενικό, θα ανοίξει το Prepay Charge Panel, δηλαδή η χρέωση θα είναι προπληρωμένη.

Όπως είναι φυσικό οι απαραίτητοι έλεγχοι προηγούνται της διαδικασίας αυτής. Για παράδειγμα θα πρέπει να πιστοποιηθεί ότι το μέλος υπάρχει στη βάση δεδομένων της εφαρμογής και ότι δεν είναι διαγραμμένο ή ανενεργό. Με την ολοκλήρωση της εφαρμογής ο πίνακας “member\_sessions” της βάσης

δεδομένων τροφοδοτείται με μια επιπλέον εγγραφή η οποία στο πεδίο "date\_closed" θα πάρει ως αρχική τιμή 0.

Στην εικόνα 3.5.1 παρουσιάζεται το παράθυρο όπου πραγματοποιείται η διαδικασία που περιγράψαμε.



Εικόνα 3.5.1

---

### 3.6. ΜΕΤΑΦΟΡΑ Η ΑΝΤΑΛΛΑΓΗ

Κατά τη διάρκεια μιας συνόδου μέλους είναι πιθανό να υπάρξει ανάγκη ο πελάτης να μεταφερθεί σε άλλο υπολογιστή. Αυτό μπορεί να γίνει, για παράδειγμα, αν ο πελάτης χρειαστεί μια υπηρεσία η οποία δεν παρέχεται από τον υπολογιστή που χρησιμοποιεί τώρα. Το NET-PRO έχει προβλέψει την ανάγκη αυτή διαθέτοντας την λειτουργία της μεταφοράς σε άλλον υπολογιστή. Αφού ο πελάτης ζητήσει να μεταφερθεί ο χρήστης θα ανοίξει το παράθυρο στο οποίο πραγματοποιείται αυτή η λειτουργία.

Το παράθυρο αυτό διαθέτει μια λίστα στην αριστερή πλευρά με τους υπολογιστές που έχουν ανοιχτές συνόδους μελών και μόνο και άλλη μια στην δεξιά πλευρά με όλους τους υπολογιστές. Η πρώτη είναι η λίστα προέλευσης και η δεύτερη η λίστα προορισμού. Στην λίστα προέλευσης, δεν υπάρχει κανένα νόημα να εμφανιστούν όλοι οι υπολογιστές του καταστήματος καθώς δεν νοείται μεταφορά από έναν υπολογιστή που είναι «κλειστός». Ο υπάλληλος – χρήστης θα επιλέξει τον υπολογιστή στον οποίο ήταν ανοιχτή η

σύνοδος του συγκεκριμένου μέλους από τη λίστα προέλευσης και έναν υπολογιστή στον οποίο θα μεταφερθεί το μέλος από τη λίστα προορισμού.

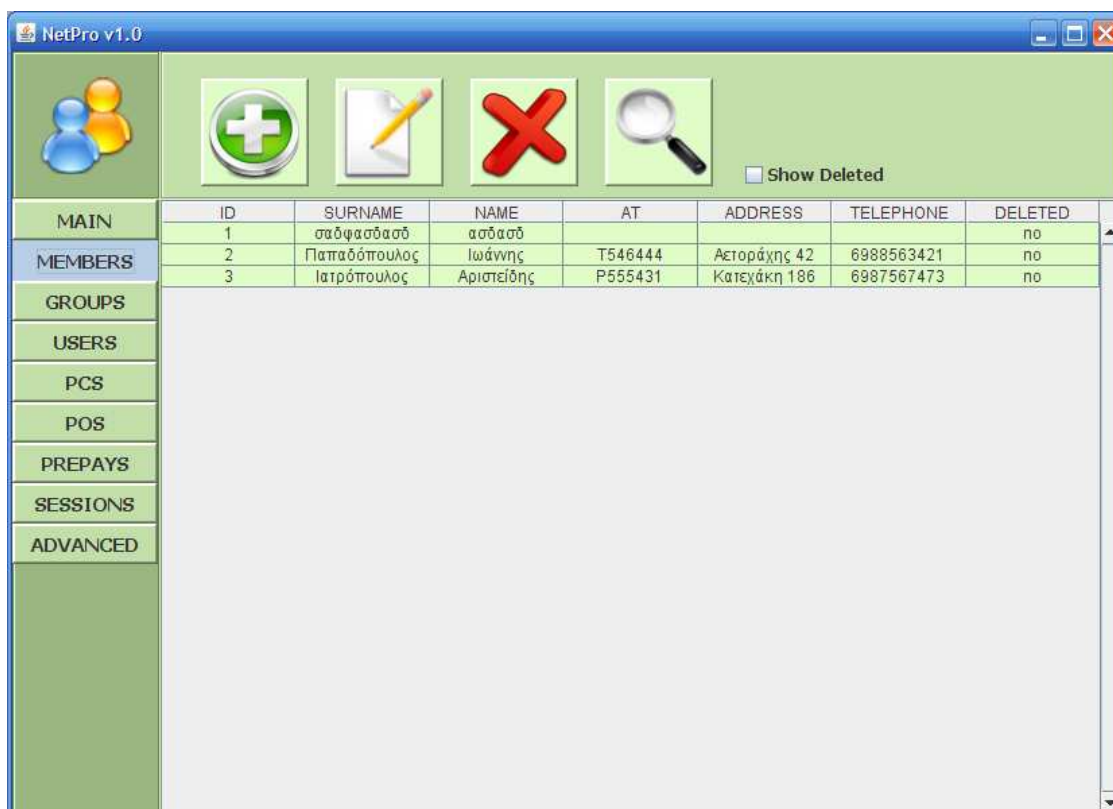
Μόλις πατήσει το κουμπί “οκ” θα πραγματοποιηθεί η μεταφορά.

Όμοια με τη μεταφορά, πραγματοποιείται και η διαδικασία της ανταλλαγής. Λιγότερο συχνά μεν, χωρίς να αποκείται η πιθανότητα δε, δύο μελή που έχουν ανοιχτές συνόδους μπορεί να θελήσουν να καθίσουν ο ένας στη θέση του άλλου.

---

#### **4. MEMBERS OVERVIEW PANEL**

Το Member Overview Panel περιέχει την επισκόπηση των μελών του καταστήματος. Όπως βλέπουμε και στην εικόνα 4.1, παρουσιάζει τα προσωπικά στοιχεία του κάθε μέλους όπως ο αριθμητικός κωδικός του(id), το επίθετο και το όνομα του, τη διεύθυνση, τον αριθμό ταυτότητας και το τηλέφωνο του και τέλος το αν κάποιο μέλος είναι ανενεργό ή όχι.



Εικόνα 4.1

#### 4.1. ΠΡΟΣΘΗΚΗ – ΜΕΤΑΤΡΟΠΗ

Πατώντας στο κουμπί "add" του Member Overview Panel θα ανοίξει ο διάλογος χρήστη για την εισαγωγή νέου μέλους στο σύστημα. Όπως παρατηρούμε στην εικόνα 4.1.1, ο αριθμητικός κωδικός που θα αποδοθεί στο νέο μέλος αρχικά "προτείνεται" από το σύστημα. Με αλλά λόγια, στο πλαίσιο που δέχεται τον αριθμητικό κωδικό, τοποθετείται ο επόμενος αριθμητικός κωδικός από αυτόν που έχει αποδοθεί στο τελευταίο μέλος που έκανε εγγραφή. Παρόλα αυτά ο χρήστης έχει τη δυνατότητα να τον αλλάξει με κάποιον άλλο με την προϋπόθεση να μην ανήκει σε άλλο μέλος.



The image shows a software dialog box titled "Add Member". It features a light yellow background and a blue title bar with a close button. The dialog contains the following fields and controls:

- Id :** A text input field containing the number "4".
- Surname :** An empty text input field.
- Name :** An empty text input field.
- Id Card Number :** An empty text input field.
- Telephone :** An empty text input field.
- Address :** An empty text input field.
- Group :** A dropdown menu with "Default" selected.
- Username :** An empty text input field.
- activate :** An unchecked checkbox.
- Buttons:** "Cancel" and "OK" buttons at the bottom.

Εικόνα 4.1.1

Το επόμενο σημείο που αξίζει να αναφερθεί εδώ είναι ότι τα απαραίτητα στοιχεία που πρέπει να συμπληρωθούν είναι ο αριθμητικός κωδικός, το επίθετο και το όνομα ενώ για το γκρουπ και το αν είναι active το μέλος, οι εξορισμού τιμές είναι "default" και "false" αντίστοιχα.

Για να πραγματοποιηθεί η διαδικασία της μετατροπής μέλους θα πρέπει αρχικά ο χρήστης να επιλέξει το μέλος από την επισκόπηση και στη συνέχεια να πατήσει το κουμπί "edit". Αμέσως θα εμφανιστεί ο ίδιος διάλογος χρηστη που εμφανίζεται και στην προσθήκη μέλους με τη διαφορά ότι τα πεδία που περιλαμβάνει θα είναι ήδη συμπληρωμένα αυτόματα με τις παλιές τιμές του μέλους. Εδώ το σύστημα δίνει το δικαίωμα στον χρήστη να αλλάξει τις τιμές μόνο των πεδίων με τη διεύθυνση, το τηλέφωνο, το όνομα χρήστη, το γκρουπ στο οποίο θα ανήκει και το αν θα είναι ενεργό ή όχι. Με την ολοκλήρωση της διαδικασίας αυτής ενημερώνεται και η βάση δεδομένων. Οι έλεγχοι που γίνονται στις παραπάνω διαδικασίες αφορούν το αν ο αριθμητικός κωδικός και

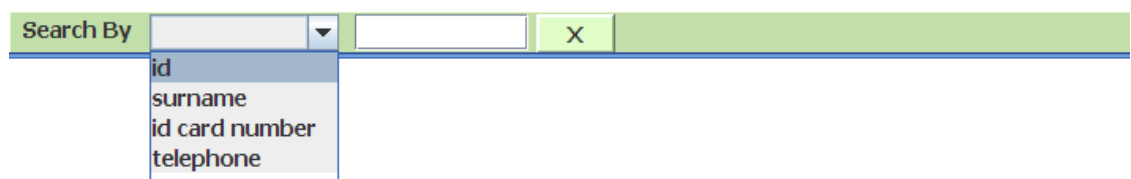
το όνομα χρήστη υπάρχει. Αυτά τα δυο στοιχεία θα πρέπει να είναι μοναδικά για το κάθε μέλος. Επιπλέον, είναι απαραίτητο να πιστοποιηθεί από το σύστημα ότι κανένα από τα απαραίτητα πεδία δεν είναι κενό.

---

## 4.2. ΑΝΑΖΗΤΗΣΗ ΜΕΛΟΥΣ

Σε ένα κατάστημα τα εγγεγραμμένα μέλη μπορεί να ξεπεράσουν ακόμα και τα εκατό. Όπως λοιπόν γίνεται κατανοητό το να ψάχνει ο χρήστης ένα μέλος από τη λίστα της επισκόπησης δεν είναι λειτουργικό και μπορεί να γίνει ιδιαίτερα κουραστικό. Για το λόγο αυτό έχει ενσωματωθεί στο σύστημα η λειτουργία της αναζήτησης μέλους. Όταν ο χρήστης επιλέξει "Search" θα ανοίξει το Panel που φαίνεται στην εικόνα 4.2.1. Ο χρήστης θα πρέπει να επιλέξει κάποιο από τα πεδία που του προσφέρονται για αναζήτηση. Αυτά είναι το επίθετο και ο αριθμητικός κωδικός. Μόλις συμπληρώσει το στοιχείο που θέλει να αναζητήσει, η λίστα θα φιλτραρισθεί και θα εμφανίσει μόνο τα μέλη που ταχτοποιούνται στο συγκεκριμένο στοιχείο.

Στην αναζήτηση με βάση το επίθετο, ο χρήστης δεν είναι απαραίτητο να εισάγει το πλήρες επίθετο του μέλους. Για παράδειγμα, αν το πρώτο γράμμα του επιθέτου είναι το 'X' και ο χρήστης εισάγει αυτό το γράμμα και μόνο, θα εμφανιστούν στη λίστα όλα τα μέλη των οποίων το επίθετο ξεκάνει από 'X'.



Εικόνα 4.2.1

### **4.3. ΔΙΑΓΡΑΦΗ ΜΕΛΟΥΣ**

Σε αυτό το σημείο θα πρέπει να αναλυθεί ο τρόπος με τον οποίο πραγματοποιείται η διαδικασία της διαγραφής τόσο στα μελή όσο και σε οποιαδήποτε άλλη οντότητα της συγκεκριμένης εφαρμογής που επιδέχεται διαγραφή.

Στα υπάρχοντα προγράμματα χρέωσης για καταστήματα internet cafe έχει διαπιστωθεί πρόβλημα στην γενικότερη εποπτεία του καταστήματος μετά από διαγραφές οντοτήτων από το σύστημα. Πιο συγκεκριμένα, διαγράφοντας ένα μέλος από το σύστημα δημιουργούνται τα εξής προβλήματα:

1. Οι κινήσεις χρημάτων που αφορούν το συγκεκριμένο μέλος διαγράφονται επίσης ακόμα και αν έχουν πραγματοποιηθεί την ίδια μέρα με την διαγραφή. Αυτό δημιουργεί μπέρδεμα στην καταμέτρηση του ταμείου από το σύστημα.
2. Πέρα από την σωστή καταμετρήσει του ταμείου, οι κινήσεις χρημάτων που πραγματοποιήθηκαν από το συγκεκριμένο μέλος είναι απαραίτητο να μένουν για πάντα στο σύστημα άσχετα από το αν το μέλος έχει διαγραφεί η όχι. Αυτό, φυσικά, δεν είναι δυνατό αν το μέλος διαγραφεί οριστικά από το σύστημα.

Για τους παραπάνω λόγους όταν θα μιλάμε για διαγραφή οποιασδήποτε οντότητας του NET-PRO, δε θα εννοούμε οριστική διαγραφή, αλλά έναν δείκτη που θα χρησιμοποιείται για να ξεχωρίζει τις διαγραμμένες από τις μη διαγραμμένες οντότητες. Ο δείκτης αυτός ουσιαστικά είναι ένα πεδίο "deleted" που υπάρχει σε κάθε πινάκα του οποίου οι έγγραφες επιδέχονται διαγραφή.

Στην περίπτωση της διαγραφής μελών που εξετάζουμε, από τη στιγμή που ένα μέλος διαγραφεί και μετά, το πεδίο "deleted" στην εγγραφή του παίρνει την τιμή "false" και θεωρείται πλέον οριστικά διαγραμμένο και αποκλεισμένο από οποιαδήποτε δραστηριότητα του συστήματος. Παρόλα αυτά η εγγραφή του στη βάση δεδομένων δεν σβήνεται. Αυτή η εικονική διαγραφή έχει το αποτέλεσμα που κάθε ιδιοκτήτης τέτοιου καταστήματος επιθυμεί. Δηλαδή, τη σωστή εποπτεία του internet café ανεξάρτητα από τις οποίες διαγραφές.

#### **4.4. ΦΙΛΤΡΟ ΔΙΑΓΡΑΜΜΕΝΩΝ**

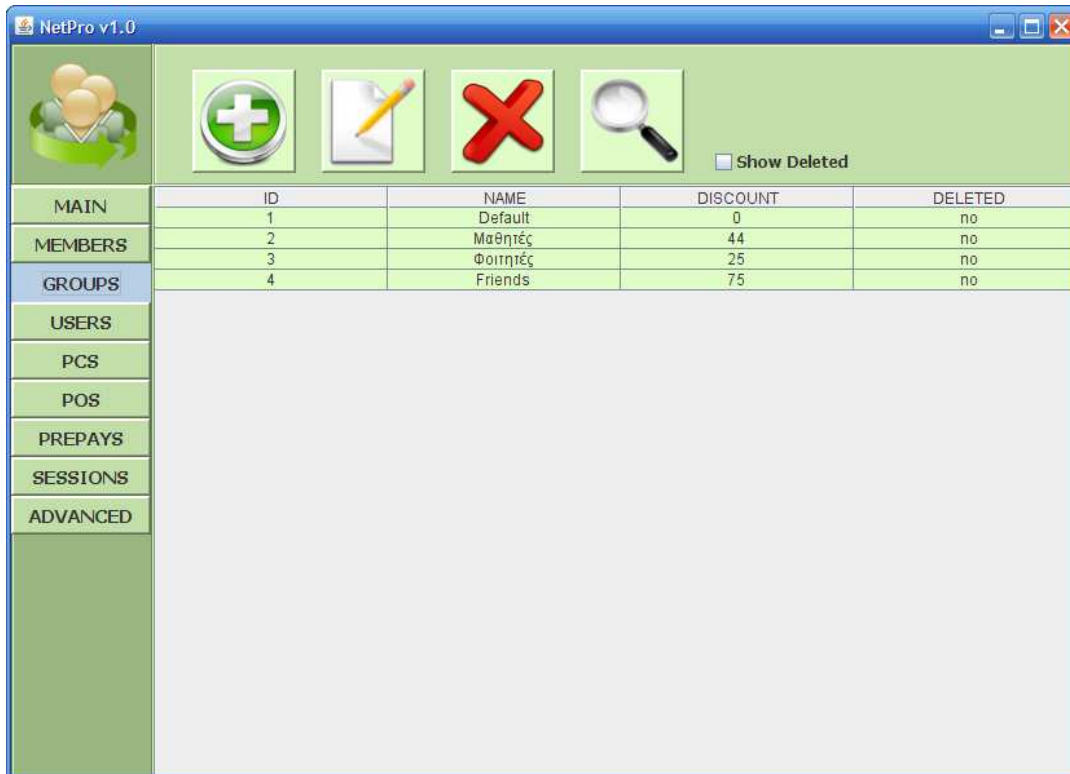
Αναφέρεται στο φιλτράρισμα που χρησιμοποιείται για να εμφανίζονται στη λίστα είτε όλα τα μελή, διαγραμμένα και μη, είτε μόνο τα διαγραμμένα. Είναι ένας τρόπος για να βλέπει ο διαχειριστής του συστήματος τα στοιχεία των μελών που έχουν διαγραφεί.

#### **5. GROUPS OVERVIEW PANEL**

Το Groups Overview Panel περιέχει την επισκόπηση των γκρουπ του καταστήματος. Η ύπαρξη του εξυπηρετεί στην ομαδοποίηση των μελών, έτσι ώστε κάθε ομάδα να έχει διαφορετική έκπτωση(discount) στις χρεώσεις. Με την εγκατάσταση του προγράμματος εισάγεται στη βάση δεδομένων ένα εξορισμού (default) γκρουπ του οποίου η έκπτωση είναι μηδενική. Από εδώ και στο εξής για κάθε καινούργιο μέλος αν δεν επιλεγθεί γκρουπ από το χρήστη, αυτό μπαίνει αυτόματα από το σύστημα στο default γκρουπ. Το Groups Overview Panel είναι ορατό μόνο σε κατάσταση διαχειριστή και ισχυρού χρήστη.

Η επισκόπηση περιλαμβάνει τον αριθμητικό κωδικό του γκρουπ, το όνομα του και την έκπτωση του. Οι λειτουργίες του είναι η πρόσθεση, η τροποποίηση και η διαγραφή των γκρουπ. Επιπλέον, και εδώ υπάρχει φίλτρο διαγραμμένων.

Η εικόνα 5.1 παρουσιάζει το Groups Overview Panel και τις ενσωματωμένες λειτουργίες του.



Εικόνα 5.1

## 6. USERS OVERVIEW PANEL

Το Users Overview Panel περιέχει την επισκόπηση των χρηστών του καταστήματος. Πρόσβαση και διαχείριση του συγκεκριμένου τμήματος της εφαρμογής διαθέτει μόνο ο διαχειριστής και ο ισχυρός χρήστης όπως συμβαίνει και με το Groups Overview Panel.

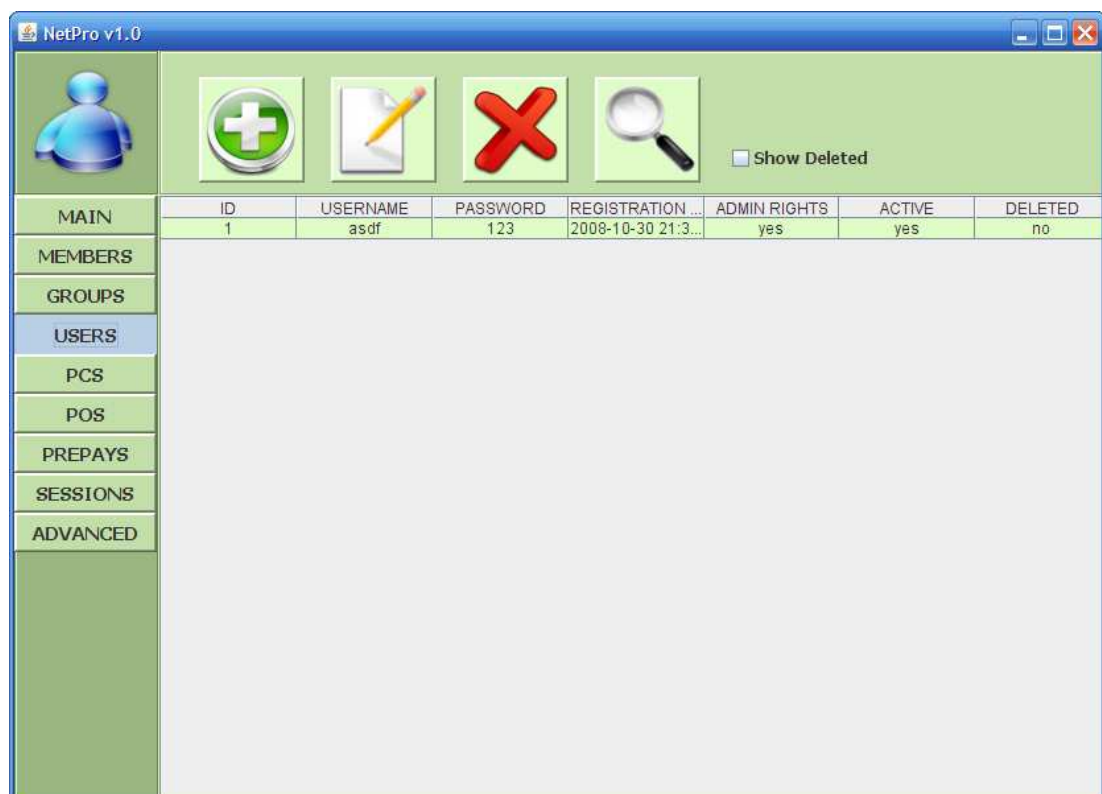
Ο διαχειριστής θεωρείται και αυτός από το σύστημα ως ένας χρήστης. Μάλιστα, ορίζεται και αυτός (όπως και το default group) κατά την εγκατάσταση της εφαρμογής. Επειδή όμως θα ήταν λάθος να έχουν τη δυνατότητα οι ισχυροί χρήστες να βλέπουν τον κωδικό του, η συγκεκριμένη εγγραφή παραλείπεται από την επισκόπηση.

Στο συγκεκριμένο κομμάτι της εφαρμογής, ο διαχειριστής ή ένας χρήστης που διαθέτει τα κατάλληλα δικαιώματα (admin\_rights), μπορεί να δει τις πληροφορίες των χρηστών όπως για παράδειγμα το όνομα χρήστη, τον

κωδικό, την ημερομηνία εγγραφής του, το αν έχει δικαιώματα διαχειριστή ή όχι, το αν έχει διαγραφεί ή όχι κτλ.

Τέλος, οι διαθέσιμες λειτουργίες που ενσωματώνονται και σε αυτό το Panel είναι η πρόσθεση, η διαγραφή, η τροποποίηση και η αναζήτηση χρήστη καθώς και το φίλτρο διαγραμμένων. Η αναζήτηση κάποιου χρήστη γίνεται με βάση το όνομα χρήστη ή του αριθμητικού κωδικού του.

Η εικόνα 6.1 παρουσιάζει το Users Overview Panel και τις ενσωματωμένες λειτουργίες του.



Εικόνα 6.1

## 7. PC'S OVERVIEW PANEL

Το Pc's Overview Panel περιέχει την επισκόπηση των υπολογιστών πελάτη του καταστήματος. Πρόσβαση και διαχείριση του συγκεκριμένου τμήματος της

εφαρμογής διαθέτει μόνο ο διαχειριστής και ο ισχυρός χρήστης όπως συμβαίνει και με το Groups Overview Panel και Users Overview Panel.

Εδώ ο διαχειριστής και ο ισχυρός χρήστης μπορούν να εισάγουν υπολογιστές πελάτη στο σύστημα. Επιπλέον, υπάρχει η δυνατότητα να μετατραπούν οι τιμές κάποιων από τα πεδία μέσω της λειτουργίας Edit. Οι λειτουργίες της διαγραφής και του φίλτρου διαγραμμένων συναντώνται και εδώ με τη μορφή που τις γνωρίσαμε προηγουμένως. Η δυνατότητα της αναζήτησης δε συναντάται στην επισκόπηση υπολογιστών. Δεν υπάρχει νόημα να κάνουμε αναζήτηση ακριβώς επειδή ο αριθμός των υπολογιστών θα είναι περιορισμένος και αποκλείεται η περίπτωση να προκύψει τέτοια ανάγκη.

Όταν προσθέσουμε έναν καινούργιο υπολογιστή πελάτη στο σύστημα και καταχωρηθεί στη βάση δεδομένων μας, ενδιαφέρον παρουσιάζει το γεγονός ότι θα εμφανιστεί ένα επιπλέον Pc Panel στην εποπτεία υπολογιστών πελάτη. Αυτό σημαίνει ότι ένας ακόμα υπολογιστής θα είναι έτοιμος προς χρήση από τους πελάτες και τα μέλη του καταστήματος. Αντίστοιχα, αν διαγράψουμε έναν υπολογιστή πελάτη από το σύστημα, η διαγραφή αυτή θα επηρεάσει και την εποπτεία υπολογιστών πελάτη.

Αξιοσημείωτο είναι το γεγονός ότι το πεδίο "ip\_address" του πίνακα pcs περιέχει την ip διεύθυνση του υπολογιστή πελάτη και είναι απαραίτητο για την επικοινωνία της υποεφαρμογής NET-PRO Server που εκτελείται στον κεντρικό υπολογιστή του καταστήματος με την υποεφαρμογή NET-PRO Client που εκτελείται σε κάθε έναν από τους υπολογιστές πελάτη.

Η εικόνα 7.1 παρουσιάζει το Pc's Overview Panel και τις ενσωματωμένες λειτουργίες του.

The screenshot shows the NetPro v1.0 application window. At the top, there are three icons: a green plus sign, a pencil on a notepad, and a red X. Below these icons is a checkbox labeled "Show Deleted". The main area contains a table with the following data:

	ID	IP ADDRESS	PC NAME	STATE	DELETED
MAIN	1	192.168.1.11	PC-01	0	no
MEMBERS	2	192.168.1.12	PC-02	0	no
	3	192.168.1.13	PC-03	0	no
GROUPS	4	192.168.1.14	PC-04	0	no
	5	192.168.1.15	PC-05	0	no
USERS	6	192.168.1.16	PC-06	0	no
	7	192.168.1.17	PC-07	0	no
PCS	8	192.168.1.18	PC-08	0	no
	9	192.168.1.19	PC-09	0	no
POS					
PREPAYS					
SESSIONS					
ADVANCED					

The sidebar on the left contains the following menu items: MAIN, MEMBERS, GROUPS, USERS, PCS, POS, PREPAYS, SESSIONS, and ADVANCED.

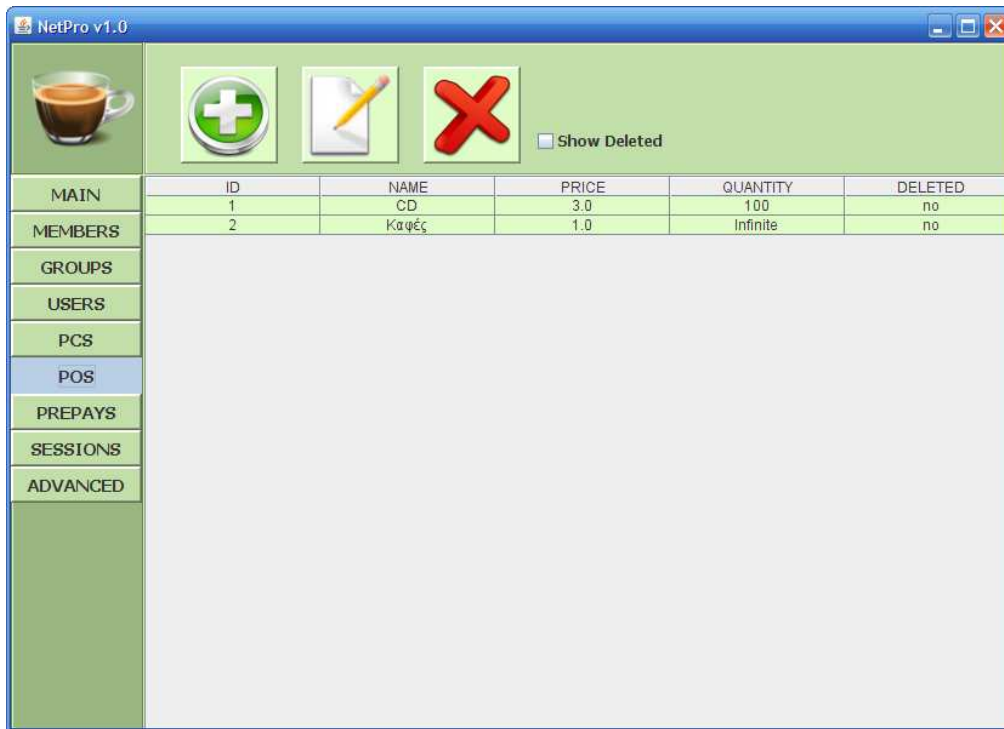
Εικόνα 7.1

## 8. POS OVERVIEW PANEL

Το Pos Overview Panel περιέχει την επισκόπηση όλων των προϊόντων προς πώληση που διαθέτει το κατάστημα. Διαθέτει επίσης τις λειτουργίες της προσθήκης, της μετατροπής και της διαγραφής που εξετάσαμε προηγουμένως, καθώς και του φίλτρου διαγραμμένων. Πρόσβαση σε αυτό έχουν μόνο ο διαχειριστής και ο δυναμικός χρήστης. Πιο συγκεκριμένα το Pos Overview Panel παρουσιάζει τα περιεχόμενα του πίνακα pos της βάσης δεδομένων. Με τη λειτουργία Add, δημιουργείται μία επιπλέον εγγραφή σε αυτό τον πίνακα και με τον τρόπο αυτό προσθέτουμε ένα ακόμη προϊόν προς πώληση για το κατάστημα. Με τη λειτουργία Edit μπορούμε να μετατρέψουμε τα πεδία price και quantity του πίνακα pos, δηλαδή να αλλάξουμε την τιμή ή και την διαθέσιμη ποσότητα οποιουδήποτε προϊόντος. Η διαγραφή πραγματοποιείται με τη λογική που περιγράψαμε και στην επισκόπηση μελών.



Η εικόνα 8.1 παρουσιάζει το Pos Overview Panel και τις ενσωματωμένες λειτουργίες του.



Εικόνα 8.1

## 9. PREPAYS OVERVIEW PANEL

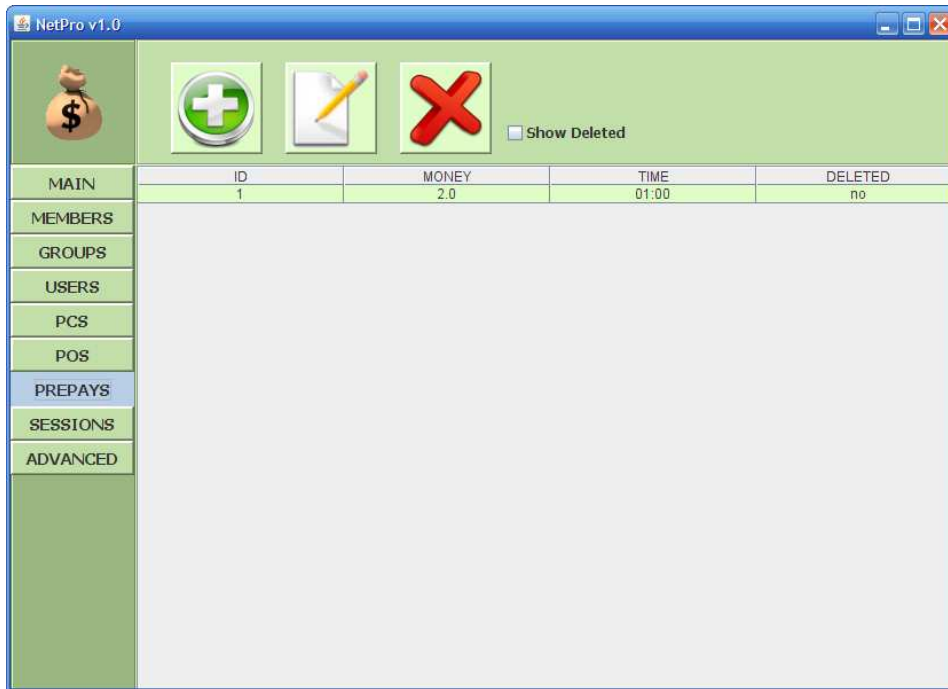
Το Prepays Overview Panel περιέχει την επισκόπηση των προπληρωμών που έχουν πραγματοποιηθεί. Στην ουσία προβάλλονται τα περιεχόμενα του πίνακα `prepay_charges` της βάσης δεδομένων. Τα πεδία αυτού του πίνακα και συνεπώς τα περιεχόμενα της επισκόπησης προπληρωμών, είναι ο αριθμητικός κωδικός της προπληρωμής, το χρηματικό ποσό για την αγορά συγκεκριμένου χρόνου, ο χρόνος αυτός, ο οποίος στη βάση είναι καταχωρημένος σε δευτερόλεπτα ενώ στην επισκόπηση έχει τη μορφή «ω:λλ» έπειτα από τις απαραίτητες μετατροπές. Τέλος, και το αναγνωριστικό `deleted` το οποίο αποτελεί πεδίο του πίνακα, τύπου `boolean`.

Οι λειτουργίες της πρόσθεσης, της μετατροπής, της διαγραφής και του φίλτρου διαγραμμένων εμφανίζονται και εδώ, ενώ οι διαδικασίες που

εκτελούν προγραμματιστικά είναι αντίστοιχες με αυτές που εξετάσαμε προηγουμένως. Αξίζει, δε να αναφερθεί ότι εδώ κάνοντας Edit έχουμε τη δυνατότητα να μετατρέψουμε μόνο τα πεδία money και seconds του πίνακα prepay\_charges δηλαδή μόνο το χρηματικό ποσό ή και το χρόνο που αντιστοιχεί σε αυτό το ποσό.

Το πιο σημαντικό όμως που πρέπει να αναφέρουμε είναι ότι είτε κατά την πρόσθεση είτε κατά τη μετατροπή μιας προπληρωμής η εφαρμογή δεν επιτρέπει να εισαχθεί διπλοεγγραφή ούτε για το πεδίο money ούτε για το πεδίο seconds. Αν αυτό ήταν επιτρεπτό θα δημιουργούνταν η εφαρμογή μας δε θα εκτελούνταν σωστά. Το τι θα συνέβαινε μπορεί να γίνει κατανοητό με ένα παράδειγμα. Ας πούμε ότι υπάρχει καταχωρημένη στον πίνακα prepay\_charges μία εγγραφή που έχει στο πεδίο money την τιμή 2 και στο πεδίο seconds την τιμή 3600. Αυτό ουσιαστικά σημαίνει ότι 1 ώρα παροχής υπηρεσιών στο κατάστημα, κοστίζει 2 ευρώ. Αν προσθέσουμε μία εγγραφή που να έχει στο πεδίο seconds την ίδια τιμή με την προηγούμενη εγγραφή που αναφέραμε και διαφορετική τιμή στο πεδίο money, πχ 3 ευρώ, τότε θα οδηγηθούμε σε λάθος επειδή είτε πληρώνοντας 2 είτε 3 ευρώ ο χρόνος που θα αγοράζει ένας πελάτης θα είναι ο ίδιος.

Η εικόνα 9.1 παρουσιάζει το Prepays Overview Panel και τις ενσωματωμένες λειτουργίες του.



Εικόνα 9.1

## 10. SESSIONS OVERVIEW PANEL

Το Sessions Overview Panel περιλαμβάνει την επισκόπηση των συνόδων του καταστήματος. Εμπερικλείει την επισκόπηση των συνόδων μελών, την επισκόπηση των συνόδων χρηστών και την επισκόπηση των πωλήσεων προϊόντων που πραγματοποιήθηκαν στο κατάστημα.

Η επισκόπηση των συνόδων μελών ουσιαστικά αποτελεί την παρουσίαση του πίνακα `member_sessions` της βάσης δεδομένων. Δίνει την ευχέρεια στο διαχειριστή και στο δυναμικό χρήστη να έχουν μία γενική εικόνα της κίνησης στο κατάστημα. Ειδικότερα, έχουν επίβλεψη του ποια μέλη χρησιμοποίησαν τις υπηρεσίες του καταστήματος για συγκεκριμένο χρονικό διάστημα, πόσα έσοδα είχε το κατάστημα παρέχοντας αυτή την υπηρεσία στο συγκεκριμένο μέλος, κατά τη σύνοδο ποιου χρήστη παράχθηκε αυτή η υπηρεσία.

Η επισκόπηση των συνόδων χρηστών ουσιαστικά αποτελεί την παρουσίαση του πίνακα `user_sessions` της βάσης δεδομένων. Εδώ, οποιοσδήποτε έχει τα απαραίτητα δικαιώματα πρόσβασης, αποκτά μία γενική εικόνα για το ποιος

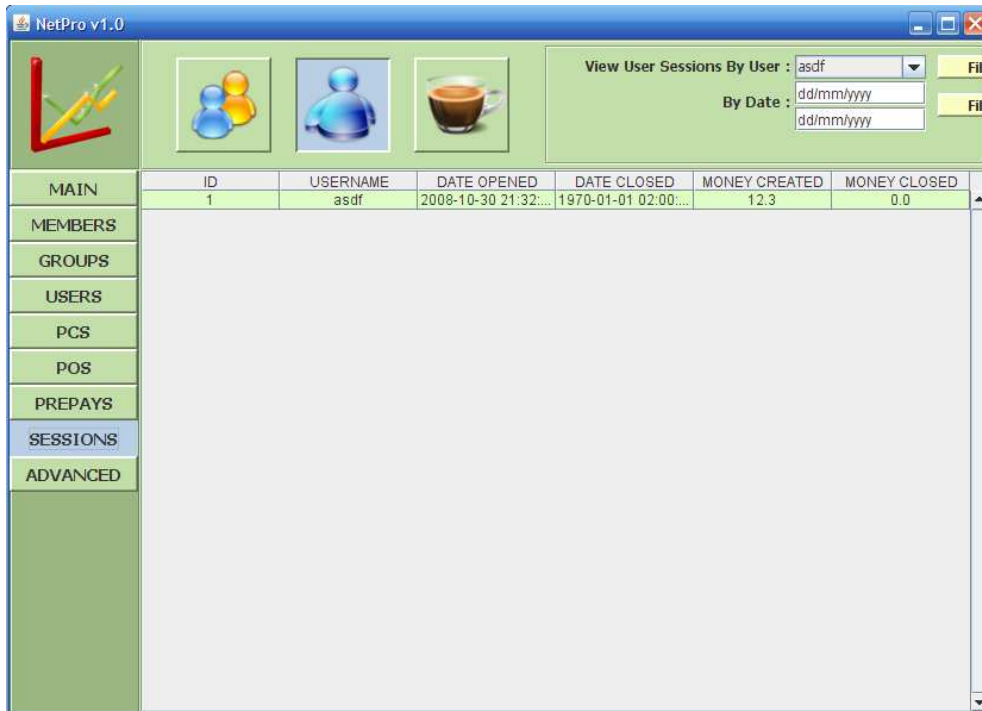
χρήστης δούλεψε τότε και για το χρηματικό ποσό που παρήγαγε το μαγαζί κατά τη διάρκεια αυτή.

Ιδιαίτερο ενδιαφέρον παρουσιάζει η ύπαρξη των πεδίων `money_created` και `money_closed` του πίνακα `user_sessions` σε αυτή την επισκόπηση. Επιτρέπει στο διαχειριστή να διαπιστώσει αν όλα πήγαν καλά κατά το κλείσιμο του ταμείου ή αν παρουσιάστηκαν λάθη. Το πεδίο `money_created` υπολογίζεται από σύστημα ανανεώνοντας την τιμή του αυξητικά ανάλογα με τις προπληρωμές και τις ανοιχτές χρεώσεις που πραγματοποιούνται κατά τη διάρκεια μιας συνόδου χρήστη. Από την άλλη, η τιμή του πεδίου `money_closed` καταχωρείται στη βάση κατά το κλείσιμο του ταμείου και είναι η τιμή που θα εισάγει ο χρήστης αφού μετρήσει τα έσοδα που είχε. Αν ο χρήστης μετρήσει σωστά, οι τιμές των πεδίων `money_created` και `money_closed` θα είναι ίδιες, πράγμα το οποίο θα σημαίνει ότι όλη η διαδικασία πραγματοποιήθηκε όπως έπρεπε.

Η επισκόπηση των πωλήσεων προϊόντων ουσιαστικά αποτελεί την παρουσίαση του πίνακα `ros_sessions` της βάσης δεδομένων. Περιλαμβάνει πληροφορίες όπως ποιο προϊόν πωλήθηκε, σε ποια ποσότητα, σε ποιον πελάτη, ποια ήταν η συνολική τιμή και κατά τη διάρκεια ποιας συνόδου χρήστη πραγματοποιήθηκε η συναλλαγή.

Σε κάθε μία από αυτές τις επισκοπήσεις υπάρχει η δυνατότητα της εισαγωγής φίλτρων έτσι ώστε να υπάρχει καλύτερη, ευκολότερη και πιο συγκεντρωτική εποπτεία. Εφαρμόζοντας τα φίλτρα αυτά, το σύστημα εκτελεί τα απαραίτητα ερωτήματα `Select` και επιστρέφει πιο συγκεκριμένα αποτελέσματα όπως π.χ τις συνόδους μελών που πραγματοποιήθηκαν κατά τη διάρκεια μιας συγκεκριμένης συνόδου χρήστη και μόνο.

Η εικόνα 4.10.1 παρουσιάζει το `Sessions Overview Panel` και τις ενσωματωμένες λειτουργίες του.



Εικόνα 10.1

## 11. ADVANCED OVERVIEW PANEL

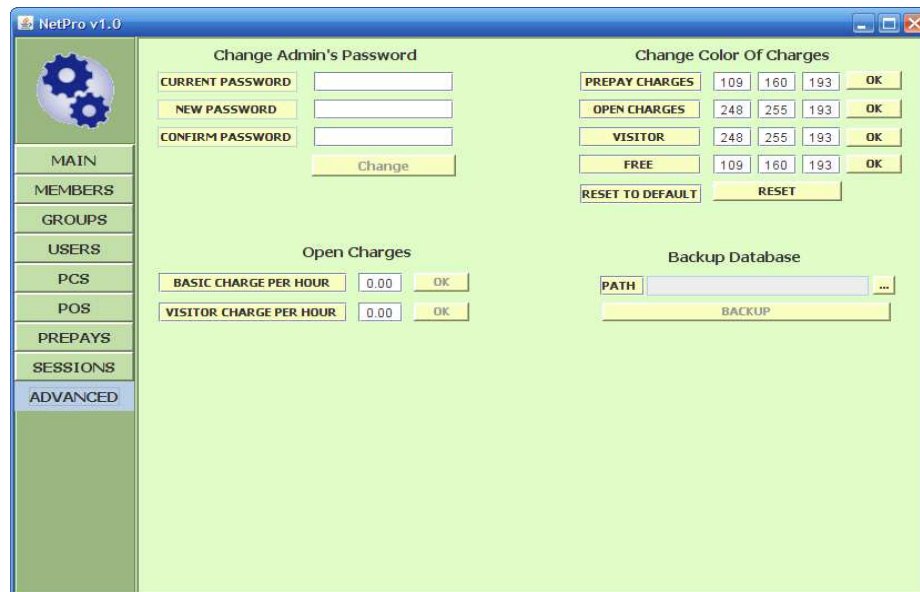
Το Advanced Overview Panel περιέχει τις γενικές ρυθμίσεις και την παραμετροποίηση της εφαρμογής. Εδώ ο διαχειριστής έχει τη δυνατότητα να αλλάξει τον κωδικό του. Αφού ταυτιστεί το παλιό password στη βάση ελέγχοντας το πεδίο passwd της εγγραφής με το username "admin" του πίνακα users τότε το πεδίο ενημερώνεται και αποκτά την καινούργια τιμή.

Εκτός αυτού, δυνατότητες κοινές και για το διαχειριστή και για τον ισχυρό χρήστη είναι η αλλαγή των ενδεικτικών χρωμάτων στα PcPanels. Οι αλλαγές καταχωρούνται στον πίνακα colors της βάσης δεδομένων.

Επιπλέον, υπάρχει η δυνατότητα αλλαγής την ανοιχτής χρέωσης. Εισάγοντας μια διαφορετική χρέωση, ανανεώνεται το πεδίο price του πίνακα open\_charge πράγμα που σημαίνει ότι αλλάζει η χρέωση ανά ώρα για τα μέλη και για τους επισκέπτες αντίστοιχα.

Τέλος, είτε ο διαχειριστής είτε ο δυναμικός χρήστης μπορούν να δημιουργούν αντίγραφα ασφαλείας της βάσης δεδομένων έτσι ώστε να μην υπάρχει κίνδυνος να χαθούν δεδομένα σε καμία περίπτωση.

Η εικόνα 11.1 παρουσιάζει το Advanced Overview Panel και τις ενσωματωμένες λειτουργίες του.



Εικόνα 11.1

## 12. ΟΙ ΕΝΤΟΛΕΣ ΠΡΟΝ ΤΟΝ NET-PRO CLIENT

Στην κεντρική επισκόπηση του NET-PRO Server υπάρχει η δυνατότητα αποστολής αιτημάτων στους υπολογιστές πελατών. Αυτή η λειτουργία λαμβάνει χώρα στο RunCommanndsDialog, είναι εφαρμογή των Sockets της Java και χρησιμοποιεί τις κλάσεις του μελετήσαμε στο προηγούμενο κεφάλαιο. Σε αυτή τη λειτουργία έχουν πρόσβαση όλες οι κατηγορίες χρηστών. Επιλέγοντας τον υπολογιστή πελάτη που επιθυμούν, ανοίγει ένα socket προς τον NET-PRO Client(ο NET-PRO Client λειτουργεί ως server τη συγκεκριμένη στιγμή), και αποστέλλονται αιτήματα προς τον υπολογιστή πελάτη. Συγκεκριμένα τα αιτήματα που μπορούν να αποσταλούν με αυτό τον τρόπο είναι τα Run, Shutdown και Restart.

## Βιβλιογραφία

---

[1]-<http://el.wikipedia.org/wiki/Java>

[2]-[http://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/Java_(programming_language))

[3]-**Οδηγός της Java 2 – Herbert Schildt**

[4]-<http://java.sun.com/docs/books/tutorial/networking/sockets/>

[5]-<http://www.cafepro.com/>

[6]-<http://www.lexiconsoftware.gr/default.asp?URL=/netcredit.asp>

[7]- <http://netcount.gr/>

---