



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Πρόβλεψη Ανεφοδιασμού Μηχανημάτων Αυτόματης Ανάληψης και Κατάθεσης (ATM BNA) Prediction Replacement for ATM and ATM with Deposit (ATM BNA)
Όνοματεπώνυμο Φοιτητή	Δακτυλίδη Σοφία
Πατρώνυμο	Αθανάσιος
Αριθμός Μητρώου	ΜΠΠΛ/ 09053
Επιβλέπων	κα. Μαρία Βίρβου, Καθηγήτρια

Ημερομηνία Παράδοσης	Μήνας Έτος
21/10/2013	Οκτώβριος 2013

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

κα. Μ. Βίρβου
Καθηγήτρια

κ.Γ. Τσιχριντζής
Καθηγητής

κ. Θέμης
Καθηγητής

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω θερμά τους καθηγητές μου και ιδιαίτερα τον κ. Ευθύμιο Αλέπη, για τη πολύτιμη βοήθεια του κατά την συγγραφή της πτυχιακής εργασίας μου.

Επίσης ένα μεγάλο ευχαριστώ την οικογένεια μου και τους φίλους μου για την ψυχική υποστήριξη που μου πρόσφεραν καθ' όλη τη διάρκεια της διεξαγωγής και συγγραφής της αυτής της εργασίας.

Περίληψη

Σκοπός αυτής της διπλωματικής εργασίας είναι η σχεδίαση και η ανάπτυξη ενός Πληροφοριακού Συστήματος επεξεργασίας τραπεζικών συναλλαγών με στόχο την πρόβλεψη τροφοδοσίας των μηχανημάτων αυτόματης ανάληψης και κατάθεσης (ATM BNA). Η κεντρικής εφαρμογής του πληροφοριακού συστήματος σκοπεύει να δώσει στον χρήστη την δυνατότητα να διαχειριστεί μια πληθώρα δεδομένων, μέσα από ένα γραφικό περιβάλλον, με ασφάλεια και αποτελεσματικότητα καθώς και να εξάγει χρήσιμα συμπεράσματα για την ροή του χρήματος. Το όλο σύστημα περιλαμβάνει και την αντίστοιχη βάση δεδομένων στην οποία αποθηκεύονται οι πληροφορίες (ποσά αναλήψεων, ποσά καταθέσεων, αριθμός συναλλαγών).

Σε αυτή τη μελέτη εξετάζονται, αναλύονται και παρουσιάζονται αποτελέσματα για τρία μηχανήματα αυτόματης ανάληψης και κατάθεσης στις περιοχές της Αθήνας, της Θεσσαλονίκης και της Λάρισας για τους μήνες Μάρτιο 2012, Οκτώβριο 2012 και Ιούλιο 2012 αντίστοιχα. Τα στοιχεία που παρουσιάζονται στην εργασία προέκυψαν μέσω συνεντεύξεων στην τράπεζα Citiank και έχουν τροποποιηθεί με σταθερές μεταβλητές ώστε το δείγμα να είναι αντιπροσωπευτικό αλλά ταυτόχρονα να μην δημοσιεύονται τα πραγματικά δεδομένα των συναλλαγών. Για την ανάλυση χρησιμοποιήθηκαν οι μέθοδοι πρόβλεψης του απλού κινούμενου μέσου όρου, του σταθμισμένου κινούμενου μέσου όρου και της απλής εκθετικής εξομάλυνσης καθώς επίσης και τα σφάλματα πρόβλεψης για την αξιοπιστία των μεθόδων.

Για την σχεδίαση της εφαρμογής, χρησιμοποιήθηκε η γλώσσα προγραμματισμού C# για Windows και έχει χρησιμοποιηθεί η τεχνική του αντικειμενοστραφούς προγραμματισμού (object oriented programming). Η όλη εφαρμογή κατασκευάστηκε με τη χρήση του προγράμματος Microsoft Visual Studio 2010 και με την αρχιτεκτονική ανάπτυξης που προσφέρει το μοντέλο MCV (Model View Control). Για την επικοινωνία του χρήστη με τη βάση δεδομένων θα χρησιμοποιηθεί το σύστημα MySql, ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων όπου χρησιμοποιεί την Sql (Structured Query Language), μια διαλογική γλώσσα ερωταπαντήσεων, τέταρτης γενιάς.

Abstract

The purpose of this diploma thesis is the design and development of an information system that will process bank transactions to provide provision for the supply of ATMs in which we can withdraw and deposit cash (ATM, BNA). The main application of information system intends to give the user the ability to manage a large amount of data through a graphical interface , safely and effectively and to export useful conclusions about the flow of money. Moreover, the whole system includes corresponding database that stores information such as withdrawal amounts , deposit amounts , number of transactions , etc.

In this study we examined , analyzed and presented the results for three ATMs that accept deposits and withdrawals in areas of Athens , Thessaloniki and Larissa for March 2012, October 2012 and July 2012 respectively. The data presented in this paper emerged through interviews to the bank of Citibank and modified with fixed variables that the sample is representative yet not published the actual transaction data . For the analysis of the data there used the prediction methods of simple moving average, a weighted moving average and single exponential smoothing. Moreover, the use of prediction errors provide us with the reliability of the methods.

For the design of the application there used the C # programming language for Windows and the technique of object oriented programming. The whole application was built by using the program Microsoft Visual Studio 2010 and the architecture offered by the development model MCV (Model View Control). For communication between the user and the database there will be used the system MySQL, a very fast and powerful database management system which uses Sql (Structured Query Language), a language interactive Q & A , fourth generation .

Περιεχόμενα

1.	Εισαγωγή.....	9
2.	Συστήματα Διαχείρισης Βάσης Δεδομένων.....	11
2.1	Πληροφοριακά Συστήματα.....	11
2.1.1	Πληροφοριακά Συστήματα.....	11
2.1.2	Διαδικασίες στην Ανάπτυξη ενός Πληροφοριακά Συστήματα.....	12
2.2	Βάσεις Δεδομένων.....	13
2.2.1	Βάσεις Δεδομένων.....	13
2.2.2	Ασφάλεια της Βάσης Δεδομένων.....	14
2.2.3	Τεχνικές προσασίας της Βάσης Δεδομένων.....	15
2.2.4	Συσχέτιση.....	17
2.3	Αντικειμενοστραφείς γλώσσες προγραμματισμού.....	18
2.4	Γλώσσα προγραμματισμού - SQL.....	19
3.	Μεθοδολογία.....	20
3.1	Η αξία και ο σκοπός της πρόβλεψης.....	20
3.2	Μέθοδοι Πρόβλεψης.....	21
3.3	Απλός Κινούμενος Μέσος Όρος.....	24
3.4	Σταθμισμένος Κινητός Μέσος Όρος.....	25
3.5	Απλή Εκθετική Εξομάλυνση.....	25
3.6	Σφάλμα Πρόβλεψης.....	26
4.	Αυτόματες Ταμειολογιστικές Μηχανές.....	27
4.1	Ιστορική Αναδρομή.....	29
4.2	Δημοσκοπήσεις.....	30
4.3	Δυνατότητες.....	33
4.4	Κάρτες αυτόματης συναλλαγής.....	34
4.5	Παρουσίαση γενικών στοιχείων και παραμέτρων.....	36
4.6	Παρουσία του τρόπου ανάλυσης των δεδομένων.....	37
4.6.1	Αυτόματη Ταμιακή Μηχανή στη περιοχή της Αθήνας.....	38
4.6.2	Αυτόματη Ταμιακή Μηχανή στη περιοχή της Θεσσαλονίκης.....	46
4.6.3	Αυτόματη Ταμιακή Μηχανή στη περιοχή της Λάρισας.....	54
5.	Ανάπτυξη της εφαρμογής.....	62
5.1	Γενική Περιγραφή εφαρμογής.....	62
5.2	Αναλυτική περιγραφή κάθε καρτέλας.....	66
5.3	Περιγραφή της βάσης δεδομένων.....	87
5.4	Βασικοί κώδικες της εφαρμογής.....	91
5.4.1	Επεξήγηση του κώδικά που βρίσκεται στο Controls\ControlATM.cs.....	91

5.4.2	Επεξήγηση του κώδικά που βρίσκεται στο Controls\ControlATMs.cs	94
5.4.3	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Database.cs	95
5.4.4	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Holiday.cs	96
5.4.5	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Holidays.cs.....	97
5.4.6	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Login.cs.....	97
5.4.7	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Main.cs	98
5.4.8	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Reports.cs.....	99
5.4.9	Επεξήγηση του κώδικά που βρίσκεται στο Controls\User.cs.....	100
5.4.10	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Users.cs	101
5.4.11	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Vendor.cs.....	102
5.4.12	Επεξήγηση του κώδικά που βρίσκεται στο Controls\Vendor.cs.....	102
5.4.13	Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelATM.cs	102
5.4.14	Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelATMs.c	103
5.4.15	Επεξήγηση του κώδικά που βρίσκεται στο Model\ Database.cs	104
5.4.16	Επεξήγηση του κώδικά που βρίσκεται στο Model\ Holidays.cs.....	104
5.4.17	Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelLogin.cs	105
5.4.18	Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelMain.cs.....	105
5.4.19	Επεξήγηση του κώδικά που βρίσκεται στο Model\ModelReports.cs	105
5.4.20	Επεξήγηση του κώδικά που βρίσκεται στο Model\ModelVendors.cs.....	108
6.	Συμπέρασμα.....	109
7.	Βιβλιογραφία	110

1. Εισαγωγή

Με το πέρασμα των χρόνων, η έννοια της πληροφορίας αποκτά ολοένα και μεγαλύτερη σημασία καθιστώντας, την υλοποίηση των συστημάτων διαχείρισης των πληροφοριών, ολοένα και πιο σημαντική υπόθεση καθώς τα πληροφοριακά συστήματα αναπτύσσονται ραγδαία τόσο σε επίπεδο λογισμικού (software) όσο και σε επίπεδο υλικού (hardware). Σήμερα, σύγχρονα πληροφοριακά συστήματα υλοποιούνται με γοργούς ρυθμούς τόσο σε ιδιωτικές εταιρείες όσο και στο δημόσιο τομέα, διευκολύνοντας τον πολίτη στις συναλλαγές του και επωμίζοντας σημαντικά κέρδη για τους ιδιώτες.

Τα σύγχρονα πληροφοριακά συστήματα βασίζονται σε ηλεκτρονικούς υπολογιστές όπου συλλέγουν, αποθηκεύουν, αναλύουν και διαχέουν δεδομένα και πληροφορίες. Με τον τρόπο αυτό υποστηρίζουν τις λειτουργίες μιας επιχείρησης και παρέχουν τις πληροφορίες που χρειάζονται στην διοίκηση της για αποτελεσματικότερες αποφάσεις. Ο τρόπος μεταφοράς των δεδομένων βασίζεται σε καινούργια πρωτόκολλα επικοινωνίας και ασύρματα δίκτυα, μειώνοντας αισθητά τον χρόνο μεταφοράς και το ποσοστό τυχόν λανθασμένης μετάδοσής της πληροφορίας.

Το δίκτυο ATM φέρνει τη τράπεζα κοντά σε κάθε πελάτη καθώς δίνει τη δυνατότητα να πραγματοποιούνται συναλλαγές ανάλογες με αυτές που διενεργούνται στα ταμεία των καταστημάτων, με χρεωστική, πιστωτική, προπληρωμένη κάρτα ή ακόμα και χωρίς κάρτα αρκεί να γνωρίζεται το αριθμό του λογαριασμού ή της κάρτας που θέλετε να επηρεάσετε. Τα ATM λειτουργούν 24 ώρες το 24ωρο και η συναλλαγή καταχωρείται στο σύστημα σε πραγματικό χρόνο. Ο χειρισμός των ATM είναι απλός. Εύκολες οδηγίες εμφανίζονται στην οθόνη που καθοδηγούν βήμα-βήμα τις συναλλαγές. Επιπλέον, ηχητικά και οπτικά σήματα προειδοποιούν για τυχόν λάθη ή παραλείψεις, όπως το να ξεχάσει κάποιος να βγάλει την κάρτα από την υποδοχή του ATM ή να παραλάβει τα μετρητά. Περίπου το 95 % των πελατών των τραπεζών επισκέπτονται ένα ATM κάθε χρόνο ενώ ο μέσος πελάτης επισκέπτεται ένα ATM 7,4 φορές το μήνα και η μέση ανάληψη από ATM είναι 60€.

Τα συστήματα επεξεργασίας συναλλαγών μπορεί να επεξεργάζονται τις συναλλαγές με δύο τρόπους, σε πραγματικό χρόνο και σε επεξεργασία δέσης. Στην επεξεργασία σε πραγματικό χρόνο, τα δεδομένα των συναλλαγών επεξεργάζονται από το σύστημα άμεσα, με την εμφάνιση της συναλλαγής, ενώ κατά την επεξεργασία δέσμης τα δεδομένα που παράγονται από τις συναλλαγές συναθροίζονται για ένα χρονικό διάστημα και επεξεργάζονται περιοδικά. Λαμβάνοντας υπόψη ότι ο όγκος των δεδομένων που εισάγονται είναι μεγάλος καθώς αναφερόμαστε σε αναλήψεις, κατάθεση μετρητών, πληρωμές πιστωτικών καρτών ή δανείων, μεταφορές χρημάτων κ.λ.π, τα δεδομένα επεξεργάζονται σε τακτά χρονικά διαστήματα δηλαδή κάθε εργάσιμη ημέρα. Η εφαρμογή απαιτεί μεγάλη αποθηκευτική δυνατότητα, υψηλή ταχύτητα επεξεργασίας και αξιοπιστία συστήματος. Ο προσανατολισμός του συστήματος είναι ιστορικός καθώς η εφαρμογή συλλέγει και διαχειρίζεται παρελθόντα στοιχεία ενώ η είσοδος και έξοδος βασίζεται σε δομημένες φόρμες.

Για τις τράπεζες το ATM ισοδυναμεί με έναν τρόπο καλής και άμεσης εξυπηρέτησης των πελατών της. Για γίνει αυτό δε φτάνει μόνο να έχουν ATM σε πολλές τοποθεσίες αλλά θα πρέπει τα ίδια τα μηχανήματα να έχουν επαρκές υπόλοιπο χρημάτων για τις αναλήψεις των πελατών ή επαρκές ελεύθερο χώρο για να δεχτούν τις καταθέσεις τους και όλα αυτά σε έναν ιδανικό συνδυασμό ώστε να μειώνονται τα λειτουργικά έξοδα. Το ζητούμενο είναι να ακολουθηθεί μια διαδικασία που θα εξασφαλίσει ότι θα παραχθούν όσο τον δυνατόν πιο ακριβείς προβλέψεις, δηλαδή πώς η ακολουθία των παρατηρήσεων θα συνεχιστεί στο μέλλον, αξιοποιώντας στο έπακρο όλη την διαθέσιμη ιστορική πληροφορία. Τα σφάλματα στις προβλέψεις είναι αναπόφευκτα αλλά αν τα δεδομένα είναι υψηλής ποιότητας μπορούμε να εξασφαλίσουμε ακριβείς προβλέψεις, στη προκειμένη περίπτωση τη σωστή μέρα τροφοδοσίας των ATM.

Για την σχεδίαση της εφαρμογής, χρησιμοποιήθηκε η γλώσσα προγραμματισμού C# για Windows και έχει χρησιμοποιηθεί η τεχνική του αντικειμενοστραφούς προγραμματισμού (object oriented programming). Η όλη εφαρμογή κατασκευάστηκε με τη χρήση του προγράμματος Microsoft Visual Studio 2010 και με την αρχιτεκτονική ανάπτυξης που προσφέρει το μοντέλο MCV (Model View Control). Για την επικοινωνία του χρήστη με τη βάση δεδομένων θα χρησιμοποιηθεί το σύστημα MySql, ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων όπου χρησιμοποιεί την Sql (Structured Query Language), μια διαλογική γλώσσα ερωταπαντήσεων, τέταρτης γενιάς.

Η C# έχει σχεδιαστεί για τη δημιουργία μίας ποικιλίας από εφαρμογές που τρέχουν στο πλαίσιο .NET. Η C# είναι απλή, ισχυρή, ασφαλής γλώσσα διατηρώντας την εκφραστικότητα των γλωσσών τύπου C. Το Microsoft Visual Studio υποστηρίζει τη Visual C# με ένα πλήρες πρόγραμμα επεξεργασίας κώδικα, διορθωτή, πρότυπο του έργου, σχεδιαστές κωδικοποιημένους οδηγούς ένα ισχυρό και εύκολο στη χρήση πρόγραμμα εντοπισμού σφαλμάτων και άλλα εργαλεία. Το .NET είναι μία κλάση βιβλιοθήκης στο πλαίσιο που παρέχει πρόσβαση σε πολλές υπηρεσίες του λειτουργικού συστήματος και άλλες χρήσιμες κλάσεις που επιταχύνουν τον κύκλο ανάπτυξης σημαντικά. Όλες οι μεταβλητές και οι μέθοδοι συμπεριλαμβανομένης της μεθόδου Main είναι έγκλειστες μέσα σε ορισμούς τάξεων. Μία κλάση μπορεί να κληρονομήσει απ ευθείας από τη μία κατηγορία γονέα, αλλά μπορεί να εφαρμόσει οποιοδήποτε αριθμό διεπαφών. Στη C# η διαδικασία της δημιουργίας (build) είναι απλή σε σύγκριση με τη C και C++ και πιο ευέλικτη από ότι στη Java. Δεν υπάρχουν ξεχωριστά αρχεία κεφαλίδας και καμία απαίτηση ότι οι μέθοδοι και οι τύποι πρέπει να δηλώνονται σε μία συγκεκριμένη σειρά. Ένα C# αρχείο μπορεί να καθορίζει τον αριθμό των τάξεων, των structs, τις διαδικασίες (interfaces) και τα γεγονότα.

Το MCV (Model View Control) μοντέλο είναι ένα κλασσικό σχεδιαστικό πρότυπο που χρησιμοποιείται συχνά από εφαρμογές που χρειάζονται την δυνατότητα να διατηρούν πολλαπλές διαφορετικές παρουσιάσεις των ίδιων δεδομένων. Το MCV πρότυπο βασίζεται στον σαφή διαχωρισμό των αντικειμένων σε τρεις κατηγορίες, τα μοντέλα (model) για να διατηρούν τα δεδομένα, τις όψεις (view) για την παρουσίαση όλων ή μέρους των δεδομένων και τους ελεγκτές (controllers) για να διαχειρίζονται τις ενέργειες που επηρεάζουν τα μοντέλα ή τις όψεις. Επομένως πολλές όψεις και πολλοί controller μπορούν να αλληλεπιδρούν με το ίδιο μοντέλο. Ένα γεγονός προκαλεί τον controller να αλλάξει το μοντέλο ή την όψη ή και τα δύο. Οπότε ένας controller αλλάζει τα δεδομένα ή τις ιδιότητες κάποιου μοντέλου, ανανεώνεται και όλες οι όψεις που εξαρτώνται από αυτό. Με τον ίδιο τρόπο όταν ένας controller αλλάζει μία όψη, η όψη αντλεί δεδομένα από το μοντέλο για να ανανεωθεί. Η αρχιτεκτονική MCV χαρακτηρίζεται από τέσσερα βασικά πλεονεκτήματα. Σαφήνεια σχεδιασμού που βοηθάει στην υλοποίηση και τη συντήρηση του προγράμματος. Αποδοτικότητα σχεδιασμού καθώς ο τρόπος με τον οποίο σχεδιάζεται το πρόγραμμα επιτρέπει τόσο στον χρήστη όσο και στον προγραμματιστή να αντικαταστήσει τα συστατικά της εφαρμογής κατά βούληση. Οι αλλαγές σε κάποιο μέρος του προγράμματος δεν συνδέονται με άλλα μέρη, διευκολύνοντας την αποτελεσματική. Πολλαπλές όψεις αφού η εφαρμογή μπορεί να εμφανίσει την κατάσταση ενός μοντέλου με πολλούς διαφορετικούς τρόπους. Όλες οι όψεις χρησιμοποιούν τα ίδια ακριβώς δεδομένα αλλά τα παρουσιάζουν με τελείως διαφορετικό τρόπο. Ευκολία ανάπτυξης, καθώς οι controllers και οι όψεις μπορούν να αναπτύσσονται καθ όλη την διάρκεια ανάπτυξης του μοντέλου. Όλες οι παλιές όψεις και οι παλιοί controllers μπορούν να χρησιμοποιούνται όσο η κοινή διεπιφάνεια διατηρείται.

Η ανάπτυξη και διαχείριση μιας βάσεων δεδομένων επιτυγχάνεται με τη βοήθεια ειδικών γλωσσών προγραμματισμού, οι οποίες ονομάζονται γλώσσες ερωτοαπαντήσεων. Είναι γλώσσες μη διαδικαστικές, τέταρτης γενιάς, στις οποίες δεν απαιτείται να οριστεί η διαδικασία πρόσβασης στη ΒΔ. Ο χρήστης απλώς διατυπώνει τις απαιτήσεις με απλές εντολές και το σύστημα διαχείρισης ΒΔ αναλαμβάνει να τις ικανοποιήσει. Η SQL (Structured Query Language) είναι η περισσότερο διαδεδομένη γλώσσα ανάπτυξης και διαχείρισης ΒΔ.

2. Συστήματα Διαχείρισης Βάσης Δεδομένων

2.1 Πληροφοριακά Συστήματα

2.1.1 Πληροφοριακά Συστήματα

Το κεφάλαιο αυτό στηρίζεται στα βιβλία του Ramakrishnan R., Gehrke J.2002, Συστήματα Διαχείρισης Βάσεων Δεδομένων, του Δημήτρης Φωλίνης,2006, Ολοκληρωμένα πληροφοριακά συστήματα διαχείρισης επιχειρηματικών πόρων, των Ι. Ε. Σαμουηλίδης, Μ. Ανδρουλάκης, Α. Αραμπατζή, Δ. Ασκούνης, Ν. Κοσματόπουλος, 2000, Συστήματα Διοίκησης και Πληροφοριών, έκδοση Εθνικού Μετσοβίου Πολυτεχνείου καθώς επίσης και στις ακόλουθες ιστοσελίδες στο διαδίκτυο el.wikipedia.org, www.ops.gr/Ergorama, www.mou.gr, www2.e-yliko.gr, infolab.cs.unipi.gr/preclass/courses, dide.flo.sch.gr/Pline και www.dblab.upatras.gr.

Στην εποχή μας, οι περισσότεροι μεγάλοι οργανισμοί ,ιδιωτικοί και μη ,είναι δέσμιοι των Πληροφοριακών τους Συστημάτων. Αποτελούν για αυτούς την ραχοκοκαλιά πάνω στην οποία στηρίζονται ώστε να εξασφαλίζουν την ομαλή λειτουργία τους και να εκπληρώσουν τους στόχους τους. Παράλληλα η παγκοσμιοποίηση της αγοράς και ο ανταγωνισμός, σε αυτή, έφερε την υλοποίηση πληροφοριακών συστημάτων σε εξέχουσα θέση, στην βιομηχανία της πληροφορίας η οποία αναζητά καινούργιες τεχνικές και μοντέλα υλοποίησης. Ο βασικότερος λόγος ανάπτυξης ΠΣ σε οργανισμούς είναι το να γίνει ο τελευταίος πιο αποδοτικός στη λειτουργία του. Έτσι καταφέρνει να μειώσει το κόστος λειτουργίας του , να αυξήσει τα έσοδα του ενώ παράλληλα μπορεί να μειώσει τις ανάγκες του σε ανθρώπινο δυναμικό και να μετασχηματιστεί σε ευέλικτο οργανισμό ικανό να παραμείνει ζωντανός επιχειρηματικά. Με όλα τα παραπάνω ο οργανισμός οδηγείται στην απόκτηση ανταγωνιστικού πλεονεκτήματος έναντι άλλων, βελτιώνει τις υπηρεσίες του και τελικά παρέχει καλύτερες και περισσότερες υπηρεσίες στον πολίτη.

Ένα σύγχρονο ΠΣ πρέπει να πλήρη μια δέσμη προδιαγραφών που προέρχονται, άμεσα, από τις ανάγκες της σύγχρονης κοινωνίας. Έτσι πρέπει να:

- Είναι Ευέλικτο
- Υποστηρίζει ποικιλία τεχνολογιών, ικανοτήτων ή γνώσης
- Περιλαμβάνει πολλαπλά μοντέλα
- Αντανακλά επιχειρησιακές διαδικασίες λήψης αποφάσεων
- Είναι ευαίσθητο σε αλλαγές
-

Τα σύγχρονα ΠΣ κατασκευάζονται και ενσωματώνουν μία πλειάδα καινούριων τεχνολογιών. Στο σύνολο τους οι τελευταίες ταξινομούνται σε δύο μεγάλες κατηγορίες, στους υπολογιστές και Συστήματα (hardware) και στο λογισμικό Π.Σ. (software). Η κατηγορία του υλικού περιλαμβάνει όλα εκείνα τα μηχανήματα που χρησιμοποιούν το Πληροφοριακό Σύστημα ή ένα τμήμα αυτού. Η καρδιά όμως του ΠΣ βρίσκεται στο λογισμικό και στις τεχνικές που χρησιμοποιεί στην επεξεργασία-ανάλυση των δεδομένων που περιέχει.

2.1.2 Διαδικασίες στην Ανάπτυξη ενός Πληροφοριακά Συστήματα

Για την αποδοτική και ορθή υλοποίηση ενός πληροφοριακού συστήματος απαιτείται μια συγκεκριμένη αλληλουχία βημάτων , που πρέπει να τηρούνται ώστε να μην προκύψουν προβλήματα και δυσλειτουργίες τόσο κατά την σχεδίαση όσο και κατά την λειτουργία. Τα βήματα αυτά, που αποτελούν και τα στάδια ανάπτυξης ενός πληροφοριακού συστήματος είναι τα εξής:

➤ **ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΔΙΑΔΙΚΑΣΙΩΝ (Systems Analysis)**

Στη φάση αυτή αποτυπώνονται και αναλύονται οι κατηγορίες των δεδομένων, οι μεταξύ τους σχέσεις καθώς και οι λειτουργικές διαδικασίες της επιχείρησης. Η φάση αυτή αποτελεί ένα από τα σημαντικότερα και δυσκολότερα βήματα κατά τον σχεδιασμό, καθώς πρέπει να αναλυθεί κάτι το μη σαφώς καθορισμένο και στατικό, με σκοπό να δημιουργηθεί μία ΒΔ που από τη φύση της δεν έχει επίσης προκαθορισμένο τρόπο αξιοποίησης.

➤ **ΣΧΕΔΙΑΣΜΟΣ (Systems Design)**

Αφορά τον λογικό, και φυσικό σχεδιασμό του συστήματος καθώς επίσης και τις εναλλακτικές λύσεις που μπορούν να ακολουθηθούν κατά την υλοποίηση του. Στην φάση αυτή γίνεται και ο καταμερισμός των ρόλων των χρηστών που θα χρησιμοποιούν μελλοντικά το σύστημα

➤ **ΟΛΟΚΛΗΡΩΣΗ - ΚΛΕΙΣΙΜΟ (Completion)**

Αναφέρεται στην τεχνική φάση κατασκευής του πληροφοριακού συστήματος , δηλαδή το προγραμματισμό των εφαρμογών που θα χρησιμοποιεί καθώς και τον έλεγχο (Testing) της ομαλής λειτουργίας του.

➤ **ΜΕΤΑΠΤΩΣΗ (migration, conversion)**

Αφού ολοκληρωθούν τα προηγούμενα βήματα με επιτυχία ακολουθεί το στάδιο της μετάπτωσης όπου το σύστημα τίθεται σε πιλοτική λειτουργία ή ακόμη σε παράλληλη λειτουργία με άλλα ή άλλο πληροφοριακό σύστημα.

➤ **ΠΑΡΑΓΩΓΗ ΚΑΙ ΣΥΝΤΗΡΗΣΗ (Production and Maintenance)**

Αποτελεί το τελευταίο και ίσως το πιο μεγάλο χρονικά βήμα στην κατασκευή ενός πληροφοριακού συστήματος αφού μέχρι το τέλος λειτουργίας του γίνεται μια συνεχή του διόρθωση, ακόμη και παραγωγή νέων τμημάτων του (επέκταση) καθώς και συντήρηση του σε σημεία που εντοπίστηκαν προβλήματα και βλάβες

Το βασικότερο χαρακτηριστικό ενός λογισμικού είναι:

- η ποιότητά
- η συντηρησιμότητα: Το Λογισμικό πρέπει να έχει την ικανότητα να ανταποκρίνεται στις καινούργιες απαιτήσεις του πελάτη.
- Η εξάρτηση: Δεν πρέπει να επηρεάζεται η αξιοπιστία και η ασφάλεια του λογισμικού από την αλληλοσύνδεση με άλλα συστήματα.
- Η αποδοτικότητα: Το Λογισμικό πρέπει να είναι αποδοτικό όσον αφορά τη χρήση πόρων.
- Η χρησιμοποιησιμότητα: Το σύστημα πρέπει να συνοδεύεται από πλήρες εγχειρίδιο χρήσης και τεκμηρίωση για ευκολία του χρήστη.

2.2 Βάσεις Δεδομένων

2.2.1 Βάσεις Δεδομένων

Το κεφάλαιο αυτό στηρίζεται στα βιβλία των Elmasri & Navathe, 2001, Θεμελιώσεις Αρχές Συστημάτων Βάσεων Δεδομένων, του Χρήστος Σκουρλάς, 2000, Σχεσιακές Βάσεις Δεδομένων και του C.J. Date, 1996, Εισαγωγή στα Συστήματα Βάσεων Δεδομένων.

Ο συνεχώς αυξανόμενος ρυθμός των δεδομένων μιας επιχείρησης ή ενός οργανισμού αλλά και η ανάγκη οργάνωσης των δεδομένων ανάλογα με τον τύπο τους, ώστε αυτά να είναι τόσο εύκολα στην άντληση όσο και στην επεξεργασία οδήγησαν στην ιδέα της δημιουργίας των βάσεων δεδομένων.

Με την έννοια βάση δεδομένων εννοούμε μια ολοκληρωμένη συλλογή δεδομένων που συσχετίζονται δηλαδή μία δεξαμενή πληροφοριών με υψηλό βαθμό οργάνωσης, που μπορούν να χρησιμοποιούν και να ενημερώνουν σύμφωνα με τις ανάγκες τους οι διάφορες εφαρμογές του οργανισμού ή της επιχείρησης. Η ΒΔ περιέχει δεδομένα τα οποία μπορεί να χρησιμοποιηθούν από περισσότερους του ενός χρήστες και το σημαντικότερο, τα δεδομένα που αντιστοιχούν σε κάθε χρήστη είναι δυνατόν να αλληλεπικαλύπτονται με διάφορους τρόπους, με αυτά των υπολοίπων χρηστών. Συνοψίζοντας μία ΒΔ είναι μία συλλογή δεδομένων με τα εξής βασικά χαρακτηριστικά :

- Τα δεδομένα είναι κοινά για όλες τις εφαρμογές και έχουν υψηλό βαθμό οργάνωσης.
- Τα προγράμματα εφαρμογών είναι ανεξάρτητα από την δομή των δεδομένων.
- Τα δεδομένα αποθηκεύονται μία μόνο φορά, ανεξάρτητα αν χρησιμοποιούνται από περισσότερες εφαρμογές.
- Ικανοποιεί τις ανάγκες των εφαρμογών, χωρίς όμως οι εφαρμογές αυτές να καθορίζουν αναγκαστικά τη δομή ή το περιεχόμενό της

Οι συνήθεις λειτουργίες που τελούνται σε ένα σύστημα ΒΔ είναι οι εξής :

- Προσθήκη νέων αρχείων στην βάση δεδομένων
- Εισαγωγή νέων δεδομένων σε υπάρχοντα αρχεία
- Ανάκληση δεδομένων από υπάρχοντα αρχεία
- Ενημέρωση δεδομένων σε υπάρχοντα αρχεία
- Διαγραφή δεδομένων από υπάρχοντα αρχεία
- Αφαίρεση υπαρχόντων αρχείων, κενών ή μη ,από την βάση δεδομένων

Η διαχείριση ή καλύτερα η διοίκηση των δεδομένων γίνεται με την βοήθεια ενός συστήματος διοίκησης / διαχείριση βάσεων δεδομένων γνωστό και ως DATABASE MANAGEMENT SYSTEM – DBMS. Στην ουσία πρόκειται για μια αυτοτελής συλλογή από τμήματα λογισμικού (προγράμματα) για την δημιουργία, επεξεργασία και την συντήρηση βάσεων δεδομένων. Η βάση δεδομένων και το λογισμικό διαχείρισης αυτής ονομάζονται με μια λέξη Σύστημα Βάσεως Δεδομένων.

Ανάλογα με τις απαιτήσεις του πληροφοριακού συστήματος αλλά και με βάση τις προδιαγραφές που θέτει κανείς στον σχεδιασμό του, οι κατηγορίες των χρηστών μεταβάλλονται. Σε γενικές γραμμές , μπορεί κανείς να κατατάξει του χρήστες των DBMS στις ακόλουθες κατηγορίες:

- Διαχειριστές (Database Administrators)
Υπεύθυνοι για την διαχείριση της Βάσης. Τα άτομα που βοηθούν τους χρήστες να επιλέγουν μεθόδους πρόσβασης και ταυτόχρονα είναι υπεύθυνοι για την ασφάλεια και την ακεραιότητα της βάσης καθώς και για την απόδοση του Συστήματος.
- Σχεδιαστές (Database Designers)
Υπεύθυνοι για τον Σχεδιασμό και Ανάπτυξη της Βάσης
- Αναλυτές Συστημάτων και Προγραμματιστές (Application Programmers / Systems Analysts)
Αναπτύσσουν τις Εφαρμογές
- Τελικοί Χρήστες (End-Users)
Κάνουν χρήση των εφαρμογών
- Tool Developers
Αυτοί που αναπτύσσουν εργαλεία για καλύτερη χρήση των DBMS

- Operators and Maintenance Personnel
Βοηθούν τον Διαχειριστή και είναι υπεύθυνοι για τη συντήρηση της Βάσης.

2.2.2 Ασφάλεια της Βάσης Δεδομένων

Μία ΒΔ περιέχει συνήθως πολύτιμες πληροφορίες, απαραίτητες για την λειτουργία της επιχείρησης στην οποία ανήκει. Είναι λοιπόν προφανές ότι η προστασία από τυχόν ατύχημα ή εσκεμμένη ενέργεια των πληροφοριών αυτών αποτελεί σημαντικό πρόβλημα.

Οι πιθανοί κίνδυνοι που απειλούν την ασφάλεια της ΒΔ είναι πολλών μορφών. Υπάρχουν καταρχήν οι φυσικές καταστροφές από φωτιά, πλημμύρες ή προβλήματα ηλεκτρικής φύσεως που μπορεί να συνεπάγονται την απώλεια των πληροφοριών. Ακόμα και αν η ίδια η ΒΔ δεν πάθει τίποτα, είναι πολύ πιθανό σε περιπτώσεις σαν αυτές να διακοπεί η λειτουργία της για τόσο χρόνο, ώστε να δημιουργηθούν προβλήματα. Υπάρχουν ακόμα περιπτώσεις ανθρώπινων λαθών όπως π.χ. να πάθει ζημιά κάποιος δίσκος. Η απώλεια πληροφοριών από αμέλεια είναι επίσης πιθανή. Υπάρχουν τέλος κίνδυνοι για την ασφάλεια των πληροφοριών από τυχαία ή και σκόπιμη κακή χρήση του μηχανογραφικού συστήματος.

Οι τρεις βασικές κακές χρήσεις του συστήματος είναι η παράνομη απόκτηση δεδομένων, η παράνομη τροποποίηση των δεδομένων ή της δομής των δεδομένων και η παράνομη τροποποίηση των προγραμμάτων της ΒΔ.

1) Παράνομη απόκτηση των δεδομένων

Παράνομη απόκτηση των δεδομένων από την ΒΔ έχουμε όταν κάποιος χρήστης που δεν είναι εξουσιοδοτημένος προσπελάει τα δεδομένα. Αν μάλιστα τα δεδομένα δεν προστατεύονται (π.χ. με λέξεις-κλειδιά) τότε μπορεί να χρησιμοποιηθεί και το ίδιο το σύστημα της ΒΔ για τη μη εξουσιοδοτημένη αυτή προσπέλαση.

Ακόμα όμως και σε περιπτώσεις που η ΒΔ προσφέρει κάποια προστασία των δεδομένων, είναι πάλι δυνατόν να αποκτήσει κάποιος μη εξουσιοδοτημένος χρήστης τα δεδομένα, γράφοντας ειδικά προγράμματα που θα προσπελάσουν απευθείας τα αρχεία της ΒΔ. Είναι όμως προφανές ότι αυτό είναι αρκετά δύσκολο μια και ο μη εξουσιοδοτημένος χρήστης θα πρέπει π.χ. να ανακαλύψει πρώτα τη συγκεκριμένη δομή των φυσικών αρχείων της ΒΔ.

Η παγίδευση είναι μία ακόμη μέθοδος που χρησιμοποιείται για την παράνομη προσπέλαση στα δεδομένα. Στην περίπτωση αυτή παγιδεύονται τα σήματα ανάμεσα στον Η/Υ και κάποιον νόμιμο χρήστη. Με τον τρόπο αυτό είναι δυνατή η προσπέλαση στις πηγές των πληροφοριών με την χρησιμοποίηση των εξουσιοδοτήσεων του νόμιμου χρήστη.

2) Παράνομη τροποποίηση των δεδομένων

Η παράνομη τροποποίηση των δεδομένων μιας ΒΔ είναι επίσης δυνατό να γίνει με πολλούς τρόπους. Ένας από αυτούς είναι να τροποποιηθούν παράνομα τα δεδομένα πριν καταχωρηθούν στην ΒΔ. Ένας άλλος τρόπος είναι να χρησιμοποιηθεί το ίδιο το σύστημα για να κάνει τις τροποποιήσεις. Αυτό είναι δυνατόν όταν δεν υπάρχει αρκετή προστασία των δεδομένων στα φυσικά αρχεία.

Ένας τρίτος τρόπος είναι να γραφούν ειδικά προγράμματα που να τροποποιούν απευθείας τα δεδομένα στα αρχεία. Τα προγράμματα όμως αυτά είναι συνήθως περίπλοκα, μια και πρέπει να τροποποιηθεί με προσοχή, στην περίπτωση αυτή ολόκληρη η δομή της ΒΔ. Σε αντίθετη περίπτωση, το ίδιο το σύστημα της ΒΔ θα διαγνώσει πιθανώς τη μεταβολή.

Τέλος, παράνομη τροποποίηση των δεδομένων είναι πιθανό να γίνει και με την μέθοδο της παγίδευσης. Στην περίπτωση αυτή παγιδεύονται τα σήματα μεταξύ κεντρικού Η/Υ και του τερματικού χρήστη, τροποποιούνται και στη συνέχεια τροφοδοτούνται οι τροποποιημένες πληροφορίες στον Η/Υ.

3) Παράνομη τροποποίηση των προγραμμάτων της ΒΔ

Η παράνομη τροποποίηση των προγραμμάτων μιας ΒΔ είναι η περίπτωση κατά την οποία μεταβάλλονται χωρίς εξουσιοδότηση βασικά εσωτερικά προγράμματα της ΒΔ. Μια συνηθισμένη τέτοια περίπτωση είναι η παράνομη τροποποίηση των προγραμμάτων της ΒΔ, που εξασφαλίζουν την προστασία της, μέσα από κλειδιά (passwords). Με τον τρόπο αυτό γίνεται δυνατό να έχουν πρόσβαση στα δεδομένα της ΒΔ μη εξουσιοδοτημένοι χρήστες. Μια

άλλη περίπτωση είναι να τροποποιηθούν χωρίς εξουσιοδότηση τα προγράμματα που δημιουργούν ή τυπώνουν τις πληροφορίες.

2.2.3 Τεχνικές προστασίας της Βάσης Δεδομένων

1) Λέξεις-Κλειδιά (Passwords)

Οι λέξεις-κλειδιά είναι η πιο διαδεδομένη τεχνική που χρησιμοποιείται για την προστασία της ΒΔ. Για να μπορέσει ο χρήστης να προσπελάσει δεδομένα που προστατεύονται με αυτόν τον τρόπο θα πρέπει να δώσει πρώτα τις καθορισμένες λέξεις-κλειδιά.

Η τεχνική αυτή προστασίας μπορεί να χρησιμοποιηθεί είτε για ολόκληρη την ΒΔ είτε για ένα μικρό υποσύνολό της, ακόμα και για ένα πεδίο κάποιας εγγραφής. Είναι ακόμη δυνατόν η προστασία αυτή να συνδεθεί με ορισμένο τρόπο προσπέλασης, όπως ανάγνωση, ενημέρωση ή διαγραφή.

Το κύριο πλεονέκτημα των λέξεων-κλειδιά είναι ότι εγκαθίστανται και χρησιμοποιούνται εύκολα. Το βασικό τους μειονέκτημα είναι ότι είναι δύσκολο να κρατηθούν μυστικά και έτσι για να είναι αποτελεσματικά πρέπει να αλλάζουν τακτικά. Υπάρχει ακόμα η περίπτωση να ξεχαστούν και γι' αυτό θα πρέπει να προβλέπεται κάποιος ασφαλής τρόπος να παρακάμπτονται σε τέτοιου είδους περιπτώσεις.

Τύποι Ακεραιότητας

- Nulls: Το null είναι ένας κανόνας που εφαρμόζεται σε μία συγκεκριμένη στήλη και επιτρέπει ή αποτρέπει εισαγωγές ή ενημερώσεις γραμμών που έχουν μηδενική τιμή σε αυτή τη στήλη.
- Unique Column Values: Ο κανόνας της μοναδικής τιμής ορίζεται σε μία στήλη ή σύνολο στηλών και επιτρέπει την εισαγωγή ή ενημέρωση μιας γραμμής μόνο αν περιέχει μία μοναδική τιμή σε αυτήν ή αυτές τις στήλες.
- Primary Key Values: Κάθε πίνακας μπορεί να έχει το πολύ ένα πρωτεύον κλειδί το οποίο και προσδιορίζει μονοσήμαντα τις γραμμές του. Από την εφαρμογή του περιορισμού δεν επιτρέπεται η είσοδος διπλών τιμών καθώς και nulls. Παρόλο που δεν απαιτείται, κάθε πίνακας πρέπει να διαθέτει το δικό του πρωτεύον κλειδί.
- Referential Integrity: Είναι ένας κανόνας που ορίζεται σε ένα κλειδί ενός πίνακα και εξασφαλίζει ότι οι τιμές αυτού του κλειδιού ταιριάζουν με τις τιμές σε έναν συσχετιζόμενο πίνακα.
- Foreign Key: Διαφορετικοί πίνακες μιας σχεσιακής ΒΔ μπορούν να συσχετιστούν μέσω κοινών στηλών. Οι κανόνες «αναφορικής ακεραιότητας» είναι αυτοί που κανονίζουν και εξασφαλίζουν τη λειτουργία αυτών των συσχετίσεων.

2) Κωδικοποίηση των δεδομένων

Η κωδικοποίηση των δεδομένων (data encryption) είναι μια δεύτερη τεχνική προστασίας της ΒΔ. Με τη μέθοδο αυτή τα δεδομένα φυλάσσονται και μεταδίδονται σε κωδικοποιημένη μορφή. Για παράδειγμα, μπορεί να κωδικοποιηθεί η δυαδική απεικόνιση των δεδομένων υψώνοντάς την στο τετράγωνο ή προσθέτοντας μια σταθερά.

Άλλος τρόπος κωδικοποίησης είναι να γίνει αντιμετάθεση χαρακτήρων ή να χρησιμοποιηθεί ένα άλλο αλφάβητο. Η κωδικοποίηση των δεδομένων είναι συνήθως εύκολη στον προγραμματισμό και ιδιαίτερα αποτελεσματική σε περίπτωση παγίδευσης και παράνομης προσπέλασης με ειδικά προγράμματα. Ο τρόπος κωδικοποίησης πρέπει να φυλάγεται καλά και ο συγκεκριμένος κώδικας να αλλάζει συχνά. Τα σχετικά προγράμματα κωδικοποίησης και αποκωδικοποίησης πρέπει επίσης να φυλάγονται προσεκτικά.

Το βασικό μειονέκτημα της κωδικοποίησης είναι ότι απασχολεί σημαντικά την κεντρική μονάδα επεξεργασίας. Αυτό συμβαίνει γιατί τα δεδομένα πρέπει να κωδικοποιούνται και να αποκωδικοποιούνται κάθε φορά που χρησιμοποιούνται. Ένας τρόπος να ξεπεραστεί το πρόβλημα αυτό είναι να κωδικοποιούνται μερικά μόνο από τα πεδία του πίνακα, τα οποία διαλέγονται κατάλληλα.

3) Έλεγχος δραστηριότητας

Ο έλεγχος δραστηριότητας (activity monitor) είναι ένας τρίτος τρόπος προστασίας της ΒΔ. Ο έλεγχος αυτός γίνεται με τη βοήθεια ειδικού ημερολογίου ελέγχου (audit log) που

διατηρείται από το σύστημα. Στο ημερολόγιο αυτό καταγράφονται η ημερομηνία και ο χρόνος προσπέλασης, το όνομα του προγράμματος, τα δεδομένα που προσπελάστηκαν, ο τρόπος προσπέλασης κλπ. Το ημερολόγιο πρέπει να εξετάζεται προσεκτικά σε τακτά χρονικά διαστήματα και αν διερευνούνται ιδιαίτερα οι προσπελάσεις σε ευαίσθητα δεδομένα. Υπάρχει βέβαια πάντοτε η πιθανότητα να μπορέσει κάποιος μη εξουσιοδοτημένος χρήστης να προσπελάσει τα δεδομένα χωρίς αυτό να καταγραφεί στο ημερολόγιο. Μια μέθοδος για να ξεπεραστεί το πρόβλημα αυτό είναι να ξαναδημιουργείται κατά τακτά χρονικά διαστήματα η ΒΔ με βάση τις εγγραφές αυτές.

Οι παραπάνω τεχνικές έχουν αποδειχθεί αποτελεσματικές σε περιπτώσεις παράνομης προσπέλασης ή τροποποίησης των δεδομένων. Δεν είναι όμως αρκετές για να προστατέψουν το σύστημα από την παράνομη τροποποίηση των προγραμμάτων της ΒΔ. Για την αποτελεσματική προστασία τους τα προγράμματα αυτά θα πρέπει να κρατούνται με τρόπο που να τα εξασφαλίζει από παράνομες παρεμβάσεις. Θα πρέπει ακόμα να τεκμηριώνονται σχολαστικά όλες οι νόμιμες τροποποιήσεις. Το μήκος των προγραμμάτων θα πρέπει να ελέγχεται περιοδικά για να διαπιστώνεται αν υπάρχουν προσθήκες που δεν δικαιολογούνται από τον φάκελο της τεκμηρίωσης.

Αν τέλος τα προγράμματα έχουν αγοραστεί απευθείας από κάποιο προμηθευτή, μια καλή μέθοδος προστασίας είναι να χρησιμοποιούνται κατά διαστήματα καινούριες κόπιες των προγραμμάτων αυτών. Με τον τρόπο αυτό εξασφαλίζεται το ότι δεν έχει γίνει «τοπική» παράνομη παρέμβαση στα προγράμματα.

Μια ακόμα τεχνική για να διαπιστωθεί ότι δεν υπάρχουν παράνομες τροποποιήσεις των προγραμμάτων της ΒΔ, είναι να ελέγχονται τα προγράμματα αυτά με τη βοήθεια ενός μικρού αριθμού από test data. Στην περίπτωση αυτή τα αποτελέσματα συγκρίνονται με τα αναμενόμενα. Τυχόν διαφορές πιθανόν να σημαίνουν παράνομη παρέμβαση στα προγράμματα. Η σύγκριση αυτή είναι δυνατόν να γίνει αυτόματα, με ειδικά προγράμματα ελέγχου. Αρκεί φυσικά να εξασφαλιστούν και τα ειδικά αυτά προγράμματα από παράνομες τροποποιήσεις.

Πρέπει πάντως να σημειωθεί ότι καμία τεχνική προστασίας της ΒΔ δεν είναι ασφαλής. Η καλύτερη προστασία είναι βέβαια να μην κρατούνται πολύ ευαίσθητα δεδομένα σε ΒΔ που μπορούν να προσπελαστούν από πολλούς χρήστες. Αν αυτό δεν είναι εφικτό, πρέπει τουλάχιστον τέτοιου είδους δεδομένα να τοποθετούνται χωριστά, να γίνονται όσο το δυνατόν λιγότερα αντίγραφα και να φυλάγονται με μεγάλη προσοχή.

2.2.4 Συσχέτιση

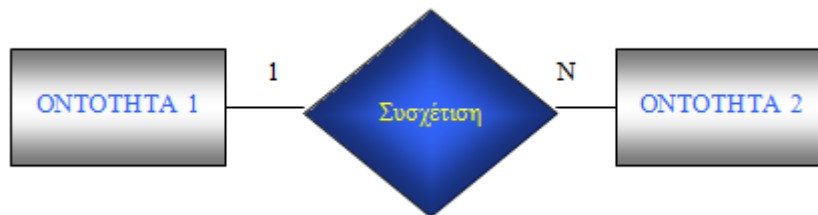
Συσχέτιση ονομάζεται ο λογικός σύνδεσμος ανάμεσα σε δύο πίνακες. Όταν ορίζεται μία συσχέτιση, το σχεσιακό μοντέλο πληροφορείται ότι θα υπάρξει μία σύνδεση ενός πρωτεύοντος κλειδιού με ένα ξένο κλειδί (σε έναν άλλο πίνακα). Άρα για να υφίσταται μία συσχέτιση χρειάζονται δύο κλειδιά το πρωτεύον και το ξένο κλειδί. Οι τύποι των συσχετίσεων είναι οι εξής :

1:1 συνήθως χρησιμοποιείται όταν θέλουμε να χωρίσουμε έναν μεγάλο πίνακα σε δύο.



Διάγραμμα 3.2: Συσχέτιση τύπου 1:1

1:N είναι ο πιο γνωστός τύπος. Στον τύπο αυτό υπάρχει μία κύρια εγγραφή που συνδέεται με καμία, μία ή περισσότερες εγγραφές σε έναν άλλο πίνακα.



Διάγραμμα 3.3: Συσχέτιση τύπου 1:N

M:N ο καλύτερος τρόπος υλοποίησης μιας τέτοιας συσχέτισης είναι να διασπαστεί σε δύο συσχετίσεις 1:M και 1:N, αντίστοιχα.



Διάγραμμα 3.4: Συσχέτιση τύπου M:N

Σε μία συσχέτιση ορίζονται οι εξής οντότητες :

- Ξένο κλειδί (Foreign Key): Η στήλη ή ομάδα στηλών που περιλαμβάνεται στον ορισμό του περιορισμού αναφορικής ακεραιότητας που αναφέρεται σε ένα κλειδί αναφοράς.
- Κλειδί Αναφοράς (Referenced Key): Το μοναδικό ή πρωτεύον κλειδί του ίδιου ή διαφορετικού πίνακα που αναφέρεται από ένα ξένο κλειδί.

- Εξαρτώμενος ή πίνακας-παιδί (Dependent or child table): Ο πίνακας που περιλαμβάνει το ξένο κλειδί. Οπότε, είναι ο πίνακας που είναι εξαρτώμενος από τις τιμές που υπάρχουν στο μοναδικό ή πρωτεύον κλειδί.
- Αναφερόμενος ή πίνακας-γονιός (Referenced or parent table): Ο πίνακας στον οποίο αναφέρεται το ξένο κλειδί του πίνακα-παιδιού. Με βάση αυτού του πίνακα το αναφερόμενο κλειδί αποφασίζεται κατά πόσο επιτρέπονται ειδικές εισαγωγές ή ενημερώσεις στον πίνακα-παιδί.

2.3 Αντικειμενοστραφείς γλώσσες προγραμματισμού

Οι αντικειμενοστραφείς γλώσσες προγραμματισμού παρέχουν πολλά πλεονεκτήματα στην ανάπτυξη λογισμικού, που έχουν σχέση με την επαναχρησιμοποίηση των αντικειμένων και την μεγαλύτερη ελαστικότητα της γλώσσας. Τα χαρακτηριστικά του αντικειμενοστραφή προγραμματισμού είναι τρία ενθυλάκωση, πολυμορφισμός και κληρονομικότητα.

➤ Ενθυλάκωση:

Είναι ένας μηχανικός που προγράμματος που συνδέει τον κώδικα με τα δεδομένα που χειρίζεται και τα κρατάει και τα δύο ασφαλή από εξωτερικές παρεμβολές και κακομεταχείριση. Ο κώδικας και τα δεδομένα μπορεί να είναι ιδιωτικά (private), όπου μπορούν να προσπελαστούν μόνο από άλλα μέρη του αντικειμένου ή δημόσια (public), όπου μπορούν να προσπελαστούν από τμήμα του προγράμματος που βρίσκεται έξω από το αντικείμενο. Η βασική μονάδα ενθυλάκωσης είναι η κλάση (class). Μία κλάση ορίζει τη μορφή θα έχει ένα αντικείμενο της, δηλαδή καθορίζει τα δεδομένα και τον κώδικα που θα δρα επί αυτών των δεδομένων. Τόσο τα δεδομένα όσο και ο κώδικας ονομάζονται μέλη της κλάσης, Κατά αντιστοιχία τα δεδομένα ονομάζονται μεταβλητές μέλους ενώ ο κώδικας μέθοδοι μέλους.

➤ Πολυμορφισμός:

Είναι το χαρακτηριστικό που επιτρέπει στο ίδιο όνομα μεθόδου να προκαλεί την εκτέλεση διαφορετικού κώδικα ανάλογα με τον τύπο του αντικειμένου στο οποίο καλείται. Με αυτό τον τρόπο μειώνεται η πηλοπλοκότητα του κώδικα.

➤ Κληρονομικότητα:

Είναι η διαδικασία κατά την οποία ένα αντικείμενο μπορεί να πάρει τις ιδιότητες ενός άλλου αντικειμένου. Επομένως πετυχαίνουμε την ιεραρχική δομή των κλάσεων και ταυτόχρονα κάθε αντικείμενο πρέπει να ορίζει μόνο εκείνες τις ιδιότητες που το κάνουν μοναδικό μέσα στη κλάση του και όχι αυτές που κληρονομεί.

2.4 Γλώσσα προγραμματισμού - SQL

Όπως κάθε γλώσσα προγραμματισμού, έτσι και η SQL αποτελείται από τις εντολές με τα ορίσματά τους τις οποίες χρησιμοποιεί ο χρήστης με καθορισμένους κανόνες σύνταξης, για να δημιουργήσει το πρόγραμμά του. Η SQL χρησιμοποιείται για να εξυπηρετήσει κάποιον από τους εξής σκοπούς :

- Τον ορισμό της δομής της ΒΔ
- Τη μεταβολή του περιεχομένου της ΒΔ
- Τον καθορισμό περιορισμών ακεραιότητας
- Την αναζήτηση δεδομένων στη βάση

Επιπλέον ως τυπική γλώσσα περιγραφής της ΒΔ πρέπει να περιλαμβάνει τις ακόλουθες βασικές λειτουργίες :

- Να προσδιορίζει τις βασικές ομάδες υποσυνόλων των δεδομένων.
- Να δίνει ένα μοναδικό όνομα στις ομάδες αυτές.
- Να προσδιορίζει που ανήκουν τα διάφορα δεδομένα ή ομάδες δεδομένων καθώς και τη ακολουθία τους.
- Να προσδιορίζει τους τύπους των δεδομένων που χρησιμοποιούνται σαν κλειδιά.
- Να προσδιορίζει τις λογικές σχέσεις μεταξύ των δεδομένων.
- Να δίνει όνομα στις λογικές αυτές σχέσεις.
- Να προσδιορίζει τον πιθανό τρόπο κωδικοποίησης των δεδομένων που χρησιμοποιεί ο χρήστης ή το πρόγραμμα εφαρμογών.
- Να προσδιορίζει το μήκος των δεδομένων.
- Να προσδιορίζει το πεδίο τιμών που μπορεί να πάρει μία μεταβλητή.
- Να προσδιορίζει τον τρόπο ελέγχου της ορθότητας των δεδομένων.
- Να προσδιορίζει τους ελέγχους προσπέλασης των δεδομένων.

Για την υλοποίηση των χρήσεων αυτών, η SQL διαθέτει τα εξής τμήματα:

A. Τη Γλώσσα ορισμού Δεδομένων (Data Description Language (DDL))

Οι γλώσσες που χρησιμοποιούνται για την περιγραφή της βάσης δεδομένων

- 1) Γλώσσες περιγραφής του υπομοντέλου
- 2) Γλώσσες περιγραφής του λογικού μοντέλου
- 3) Γλώσσες φυσικής περιγραφής των δεδομένων

B. Τη Γλώσσα Χειρισμού Δεδομένων (Data Manipulation Language (DML))

Οι γλώσσες που επιτρέπουν τον χειρισμό των δεδομένων και την υποβολή ερωτημάτων (query language)

- 1) Ειδικές λειτουργίες για άνοιγμα, αναζήτηση, ενημέρωση και κλείσιμο αρχείων και εγγράφων
- 2) Εισαγωγή-Διαγραφή-Αλλαγή-Ταξινόμηση-Στατιστικά στοιχεία

Μία βάση δεδομένων επιτρέπει την αποθήκευση, την αναζήτηση, την ταξινόμηση και την ανάκτηση δεδομένων. Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, ώστε να μπορούν να εργαστούν πολλοί χρήστες ταυτόχρονα, παρέχει γρήγορη πρόσβαση και διασφάλιση εισαγωγής μόνο σε πιστοποιημένους χρήστες. Η MySQL χαρακτηρίζεται από μεγάλη ταχύτητα και χαμηλό κόστος καθώς διατίθεται δωρεάν με άδεια ανοικτού κώδικα. Μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά συστήματα όπως Unix και Microsoft Windows, επιβεβαιώνοντας τη μεταφερισιμότητα της.

3. Μεθοδολογία

3.1 Η αξία και ο σκοπός της πρόβλεψης

Το κεφάλαιο αυτό στηρίζεται στα βιβλία Jeffrey Jarrett, 2002, Μέθοδοι προβλέψεων για οικονομικές – επιχειρηματικές αποφάσεις, της Ζαχαροπούλου Χρυσούλα, 2003, Στατιστική μέθοδοι – εφαρμογές, των Κωνσταντίνος Ρόντος και Ευστράτιος Παπάνης, 2006, Στατιστική Έρευνα-Μέθοδοι και Εφαρμογές, σε διαλέξεις των Δριτσάκης Νικόλαος και Πετράκης Ανδρέας για το μάθημα Στατιστικές Μέθοδοι με Εντατική Χρήση Η/Υ του Πανεπιστημίου Μακεδονίας, Μονάδα Προβλέψεων & Στρατηγικής του Εθνικού Μετσόβιου Πολυτεχνείου, Διαχείριση και πρόβλεψη ζήτηση, για το μάθημα Διοίκηση Παραγωγής και Συστημάτων Υπηρεσιών του Εθνικού Μετσόβιου Πολυτεχνείου, Operations Management, ICBS Business College, σε άρθρα των Γρηγόρη Κιοσέογλου και Ανδρέα Δημητρίου, Ανάλυση προβλέψεων: Στατιστική Μεθοδολογία για την αξιολόγηση μοντέλων πρόβλεψης με ποιοτικά δεδομένα στις κοινωνικές επιστήμες καθώς και στις ακόλουθες ιστοσελίδες στο διαδίκτυο www.markets.com/el/education/technical-analysis, r.investing.com/education, library.tee.gr/digital/kma/kma_m1274.pdf (σημειώσεις στη Διαχείριση Αποθεμάτων).

Στη διοίκηση και τη διαχείριση των μεγάλων οργανισμών, η ανάγκη για σχεδιασμό και έλεγχο είναι ιδιαίτερα σημαντική. Ο χρόνος για τον οποίο ξοδεύει μια επιχείρηση για να καταλήξει με μία απόφαση κυμαίνεται από αρκετά χρόνια έως λίγες ώρες ή ακόμα και μέρες. Οι επιχειρήσεις έχουν ανάγκη να προγραμματίσουν τις ενέργειες τους αρκετό καιρό πριν την εκδήλωση της ζήτησης, επομένως εμφανίζεται η ανάγκη πρόβλεψης της ζήτησης. Για τον παραπάνω σκοπό αναπτύχθηκαν μέθοδοι με τις οποίες μπορεί να εκτιμηθεί η ποσότητα και ο χρόνος στον οποία αναμένεται να ζητηθεί το προϊόν ώστε να γίνει ο προγραμματισμός της παραγωγής ή των προμηθειών.

Πρόβλεψη θεωρείτε η εκτίμηση μελλοντικών γεγονότων που θα χρησιμοποιήσει μια επιχείρηση για τον προγραμματισμό των σχεδίων της όσον αφορά τις διαχειριστικές και διοικητικές αποφάσεις. Η συγκεκριμένη άποψη αντιμετωπίζεται από μία μορφή αμφισβήτησης ως προς την αξιοπιστία και την αποτελεσματικότητα. Ο βασικότερος λόγος για την κριτική είναι οι εσφαλμένες απαιτήσεις των χρηστών των προβλέψεων. Η πρόβλεψη δεν είναι υποκατάστατο της προφητείας, επομένως τα σφάλματα στις προβλέψεις είναι αναπόφευκτα. Πολλές εταιρίες δεν ασχολούνται με την πρόβλεψη γιατί εξακολουθούν να πιστεύουν πως ούτε το μέλλον επιφυλάσσει σημαντικές αλλαγές, ούτε ότι θα υπάρξει αρκετός χρόνος στο μέλλον που θα επιτρέπει σε μια εταιρία να αντιδράσει στις αλλαγές των γεγονότων. Άλλες εταιρίες δεν χρησιμοποιούν την πρόβλεψη γιατί αγνοούν την ανάγκη της καθώς δεν παρουσιάζουν υπερβολικά αποθέματα ή απώλειες πωλήσεων. Ιδιαίτερα σημαντικό, από πρακτικής απόψεως, είναι να κατανοήσουν οι χρήστες των προβλέψεων τα ρεαλιστικά πλεονεκτήματα αλλά και τα όρια των μεθόδων προβλέψεων και να τα λάβουν υπόψη τους όταν τις χρησιμοποιούν στον σχεδιασμό και στην λήψη αποφάσεων. Χαρακτηριστικό της ζήτησης των προϊόντων και των υπηρεσιών, είναι η αβεβαιότητα, επομένως όλες οι εταιρίες λειτουργούν μέσα σε μια ατμόσφαιρα αβεβαιότητας σε ότι αφορά τα μελλοντικά γεγονότα. Οι διαρκώς μεταβαλλόμενες επιχειρηματικές συνθήκες ως αποτέλεσμα του παγκόσμιου ανταγωνισμού και των ραγδαίων τεχνολογικών αλλαγών ασκούν πιέσεις στις επιχειρήσεις για όσο το δυνατόν πιο ακριβείς προβλέψεις. Οι προβλέψεις χρειάζονται προκειμένου να καθορίζει μια επιχείρηση τις πηγές που θα χρειαστεί, να προγραμματίσει τις υπάρχουσες πηγές και να αποκτήσει τις υπόλοιπες που θα χρειαστούν για την παραγωγή. Το ζητούμενο είναι να ακολουθεί μια διαδικασία που θα εξασφαλίσει ότι θα παραχθούν όσο τον δυνατόν πιο ακριβείς προβλέψεις, αξιοποιώντας στο έπακρο όλη την διαθέσιμη ιστορική πληροφορία. Με την εφαρμογή κατάλληλων μεθόδων πρόβλεψης της ζήτησης, μπορεί να επιτευχθεί σημαντική μείωση του επιπέδου των αποθεμάτων ακόμη και με ταυτόχρονη αύξηση του επιπέδου διαθεσιμότητας, να μειώσει το χρόνο παραγωγής και να βελτιώσει τις δυνατότητες των μηχανημάτων.

Συνοψίζοντας, η ικανότητα πρόβλεψης των μελλοντικών γεγονότων με ακρίβεια είναι ένα απαραίτητο μέρος του σημερινού πολύπλοκου διευθυντικού σχεδιασμού και ελέγχου καθώς παρέχει τη δυνατότητα να προγραμματίσει τις απαιτήσεις σε πρώτες ύλες, προσωπικό, παραγωγή, κεφάλαιο αλλά και να βοηθήσει στις αποφάσεις σχετικά με την δημιουργία εγκαταστάσεων, καθώς και βραχυχρόνιων απαιτήσεων για χρηματοδότηση.

3.2 Μέθοδοι Πρόβλεψης

Η πρόβλεψη της ζήτησης είναι μια εκτίμηση της διακύμανσης της ζήτησης στο μέλλον και αποτελεί τη βάση πολλών επιχειρηματικών αποφάσεων. Ακόμα θα μπορούσαμε να πούμε ότι είναι ο συντονισμός και ο έλεγχος όλων των πηγών ώστε να μεγιστοποιηθεί η αποδοτικότητα του παραγωγικού συστήματος και η παράδοση του προϊόντος να πραγματοποιηθεί χωρίς καθυστέρηση. Η ζήτηση για προϊόντα ή υπηρεσίες βασίζεται στα εξής έξι στοιχεία, στη μέση ζήτηση για μία περίοδο, στη τάση, στην εποχικότητα, στη κυκλικότητα, στη τυχαία μεταβατικότητα και η αυτοσυσχέτιση. Η αξιοπιστία της πρόβλεψης εξαρτάται από το χρονικό ορίζοντα της προβολής στο μέλλον. Όσο μεγαλώνει ο χρονικός ορίζοντας της πρόβλεψης τόσο αυξάνεται και το σφάλμα.

Πολύς λόγος γίνεται για την ακρίβεια των μοντέλων προβλέψεων, για τα μεγάλα σφάλματα και την αδυναμία των μοντέλων προβλέψεων να υποδείξουν επερχόμενες αλλαγές που αιφνιδίασαν όλο τον επιχειρηματικό κόσμο. Οι μέθοδοι πρόβλεψης μπορούν να βασιστούν είτε σε μαθηματικά μοντέλα χρησιμοποιώντας διαθέσιμη ποσοτικοποιημένη πληροφορία προηγούμενων περιόδων, είτε σε ποιοτικές μεθόδους βασισμένες στην διοικητική εμπειρία, γνώση και κριτική ικανότητα των στελεχών της, είτε σε συνδυασμό και των δύο. Σε κάθε περίπτωση τα στοιχεία και οι πληροφορίες που χρησιμοποιούνται για τις προβλέψεις θα πρέπει να ανανεώνονται σε συνεχή βάση. Με αυτόν τον τρόπο, εξασφαλίζεται αφενός η αξιοπιστία των προβλέψεων και αφετέρου μειώνονται τα σφάλματα και αυξάνεται η ακρίβεια των προβλέψεων τουλάχιστον βραχυπρόθεσμα.

Ο βασικός παράγοντας που καθορίζει την επιλογή της μεθόδου προβλέψεων είναι το τι απόφαση θα ληφθεί βάσει των προβλέψεων που θα προκύψουν. Εκτός από αυτόν τον παράγοντα η επιλογή της κατάλληλης μεθόδου καθορίζεται από ένα σύνολο παραγόντων όπως η ζητούμενη μορφή της πρόβλεψης, η περίοδος και ορίζοντας πρόβλεψης, επαρκεί δεδομένα, το κόστος της μεθόδου, η επιζητούμενη ακρίβεια, η διαθεσιμότητα εμπειρου προσωπικού, η απλότητα και η ευκολία εφαρμογής.

Ο χρονικός ορίζοντας μιας πρόβλεψης ονομάζεται η απόσταση σε χρόνο ανάμεσα στο σημείο στο οποίο γίνεται η πρόβλεψη και το χρονικό σημείο στο οποίο αυτή αναφέρεται. Ο χρόνος αυτός μπορεί να χρησιμοποιηθεί ως βάση στην ταξινόμηση των προβλέψεων. Οι προβλέψεις μπορούν να ταξινομηθούν ως:

- Βραχυπρόθεσμες, για ενέργειες που επηρεάζουν το παρόν και το άμεσο μέλλον. Οι αποφάσεις αφορούν τρέχουσες λειτουργίες της παραγωγής και μεταφράζονται σε απαιτήσεις σε προσωπικό, υλικό, μηχανήματα. Για αυτές τις προβλέψεις χρησιμοποιούνται συνήθως μέθοδοι προεκβολής.
- Μεσοπρόθεσμες, για ζητήματα που περιλαμβάνουν τον προγραμματισμό του μηνιαίου ύψους της παραγωγής. Που θα χρησιμοποιηθούν μεσοπρόθεσμα σε χρονικό ορίζοντα 6-12 μηνών συνήθως. Γι' αυτές τις προβλέψεις χρησιμοποιούνται συνήθως οι μέθοδοι προεκβολής ή οι αιτιακές μέθοδοι.
- Μακροπρόθεσμες, για ζητήματα που έχουν να κάνουν με τη συνολική δυναμικότητα ενός συστήματος, την τεχνολογία που θα χρησιμοποιηθεί κλπ. σε ορίζοντα ετών. Γι' αυτές τις προβλέψεις χρησιμοποιούνται συνήθως οι ποιοτικές μέθοδοι ή οι αιτιακές μέθοδοι.

Πολλές μέθοδοι έχουν αναπτυχθεί για τη διενέργεια των προβλέψεων και χρησιμοποιούνται για τη λήψη αποφάσεων σε ποικίλες συνθήκες. Η επιλογή της κατάλληλης κάθε φορά μεθόδου, η εγκατάσταση και χρήση της καθώς και η ερμηνεία των αποτελεσμάτων της είναι μερικά από τα προβλήματα που αντιμετωπίζονται στην πρακτική εφαρμογή αυτών των μεθόδων. Ο βαθμός δυσκολίας του εγχειρήματος είναι μεγάλος διότι απαιτεί τον συνδυασμό των ποσοτικών τεχνικών αλλά και πρακτικής εμπειρίας σχετικά με το προϊόν και τον τομέα δραστηριοποίησης της επιχείρησης. Οι μέθοδοι προβλέψεων χωρίζονται σε δύο γενικές κατηγορίες :

- 1) Οι ποιοτικές μέθοδοι ή μέθοδοι κρίσης, οι οποίες βασίζονται στην κρίση δηλαδή σε υποκειμενικές εκτιμήσεις συνήθως ειδικών και στο συνδυασμό ποιοτικών και ποσοτικών στοιχείων .

- 2) Οι ποσοτικές μέθοδοι όπου χρησιμοποιούνται ποσοτικά στοιχεία από το παρελθόν ώστε να προβλεφθεί η τιμή της μεταβλητής στο μέλλον. Εφόσον ο χρονικός ορίζοντας της πρόβλεψης είναι μικρός, αυτές οι μέθοδοι δίνουν αρκετά αξιόπιστα αποτελέσματα.

Στον παρακάτω πίνακα συνοψίζονται οι υπάρχουσες μέθοδοι και μοντέλα πρόβλεψης.

I. Ποιοτικές	Υποκειμενικές και βασισμένες σε εκτιμήσεις και γνώμες
Grass Roots	Η πρόβλεψη πραγματοποιείται από δεδομένα που προέρχονται από εκείνους που έρχονται σε άμεση επαφή με το αντικείμενο της πρόβλεψης, όπως πωλητές στην περίπτωση πρόβλεψης πωλήσεων ενός προϊόντος.
Έρευνα Αγοράς	Συλλογή δεδομένων με διάφορους τρόπους (συνεντεύξεις, ερωτηματολόγια, ...) με στόχο τον έλεγχο συνθηκών της αγοράς. Θεωρείται μακροπρόθεσμη μέθοδος πρόβλεψης κυρίως για νέα προϊόντα.
Συμβούλιο Στελεχών	Ελεύθερη ανταλλαγή απόψεων σε συνελεύσεις. Οι συμμετέχοντες μπορεί να είναι στελέχη, πωλητές και πελάτες.
Ιστορική Αναλογία	Σύνδεση του αντικειμένου πρόβλεψης με ένα παρόμοιο μέγεθος. Χρήσιμη για τον σχεδιασμό νέων προϊόντων όπου η πρόβλεψη πραγματοποιείται σύμφωνα με το ιστορικό ενός παρόμοιου προϊόντος.
Μέθοδος Delphi	Μία ομάδα ειδικών απαντάει σε ένα ερωτηματολόγιο, το οποίο τροποποιείται σύμφωνα με τα αποτελέσματα και αποστέλλεται ξανά με στόχο να δημιουργηθεί μια διαδικασία μάθησης για τα μέλη της ομάδας χωρίς πίεση από προϊστάμενους.

II. Ποσοτικές	Ανάλυση Χρονοσειρών, Αιτιακές Μέθοδοι και Προσομοίωση
Μέθοδοι Προεκβολής ή Ανάλυση Χρονοσειρών	Θεωρεί ότι οι ιστορικές παρατηρήσεις μπορούν να χρησιμοποιηθούν για την πρόβλεψη μελλοντικών παρατηρήσεων
Απλός Κινητός Μέσος	Απλός μέσος όρος των n τελευταίων παρατηρήσεων όπου κάθε σημείο έχει την ίδια βαρύτητα.
Σταθμισμένος Κινητός Μέσος	Ορισμένα σημεία (συνήθως τα πιο πρόσφατα) έχουν μεγαλύτερη βαρύτητα στον υπολογισμό του μέσου όρου.
Εκθετική Εξομάλυνση	Η στάθμιση των δεδομένων μειώνεται εκθετικά σε σχέση με τον χρόνο δίνοντας μεγαλύτερη στάθμιση στις πιο πρόσφατες παρατηρήσεις.
Παλινδρόμηση	Προσαρμογή γραμμικής σχέσης σε συνάρτηση με τον χρόνο. Η μέθοδος των ελάχιστων τετραγώνων είναι η πιο συνηθισμένη τέτοια μέθοδος γραμμικής προσαρμογής.
Box – Jenkins	Περίπλοκη και ακριβής στατιστική μέθοδος που σχετίζει στατιστικά μοντέλα με τα δεδομένα και προσαρμόζει το μοντέλο στην χρονοσειρά με την χρήση Μπειζιανών κατανομών.
Χρονοσειρές Shinskin	Αποτελεσματική μέθοδος ανάλυσης χρονοσειρών σε εποχικότητα, τάση και τυχαιότητα. Απαιτεί δεδομένα τουλάχιστον τριών ετών και είναι πολύ αποτελεσματική στον εντοπισμών σημείων αλλαγής
Εφαρμογή Τάσεων	Προσαρμογή μαθηματικών γραμμών τάσης στα δεδομένα και προεκβολή στο μέλλον.
Αιτιακές Μέθοδοι	Βασίζονται στην υπόθεση ότι η μεταβλητή που πρέπει να προβλεφθεί είναι συνάρτηση άλλων ανεξάρτητων παραγόντων (π.χ. τιμή, διαφήμιση, πορεία των αγορών, την ποιότητα, τους ανταγωνιστές κλπ). Επιδιώκεται να προσδιορισθεί η σχέση ανάμεσα στη μεταβλητή και στους ανεξάρτητους παράγοντες. Επομένως, είναι δυνατή η πρόβλεψη της μελλοντικής τιμής της μεταβλητής εφόσον

	υπάρχουν προβλέψεις για τις μελλοντικές τιμές των ανεξάρτητων παραγόντων
Παλινδρόμηση	Πολυμεταβλητή ανάλυση παλινδρόμησης παρόμοια με εκείνη των χρονοσειρών, που λαμβάνει υπόψη και εξωτερικές μεταβλητές.
Οικονομετρικά Μοντέλα	Εφαρμογή εξαρτημένων εξισώσεων με σκοπό την περιγραφή ενός τομέα της οικονομίας.
Μοντέλα Εισόδου/Εξόδου	Εστιάζεται στις πωλήσεις κάθε βιομηχανίας προς άλλες εταιρείες και Κυβερνήσεις προβλέπει αλλαγές στις πωλήσεις μιας βιομηχανίας σύμφωνα με επικείμενες αλλαγές στις αγορές από άλλες βιομηχανίες.
Leading Indicators	Εντοπισμός προϊόντων ή μεγεθών που έχουν την ίδια πορεία αλλά προηγούνται του μεγέθους πρόβλεψης.

Η επιλογή της σωστής μεθόδου πρόβλεψης ή του σωστού συνδυασμού μεθόδων πρόβλεψης έχει ιδιαίτερη σημασία αφού κάθε σπουδαία απόφαση στις επιχειρήσεις βασίζεται σε μεγάλο βαθμό σε αυτές.

Για να γίνει μία πρόβλεψη πρέπει να ακολουθηθούν τα παρακάτω βήματα:

1) Καθορισμός Προβλήματος

Είναι το πιο δύσκολο μέρος στην διαδικασία πρόβλεψης καθώς θα πρέπει να γίνει απολύτως σαφές πώς θα χρησιμοποιηθούν οι προβλέψεις και ποιοι θα χρησιμοποιήσουν τις προβλέψεις. Ακόμα είναι σκόπιμο να αναλωθεί αρκετός χρόνος στο ποιος θα συλλέξει τα στοιχεία, ποιος θα συντηρεί τις βάσεις δεδομένων και τελικά ποιος θα χρησιμοποιήσει τις προβλέψεις για τον μελλοντικό σχεδιασμό

2) Συλλογή και ανάλυση ιστορικών στοιχείων και πληροφοριών.

Υψηλής ποιότητας δεδομένα θα οδηγήσουν σε ακριβείς προβλέψεις. Απαιτούνται τουλάχιστον δύο είδη πληροφοριών, στατικά δεδομένα και η κρίση, η εμπειρία του προσωπικού που ασχολείται με την συγκεκριμένη αγορά το πρόσφατο χρονικό διάστημα. Τα συλλεφθέντα στοιχεία αναλύονται με στόχο το ξεκαθάρισμα και την επιλογή εκείνων που διαστρεβλώνουν λιγότερο και αποτυπώνουν καλύτερα την παρελθοντική εικόνα της ζήτησης.

3) Προετοιμασία χρονοσειρών και υπολογισμός του σφάλματος πρόβλεψης.

Η ανάλυση μας οδηγεί σε κατάλληλους χειρισμούς αποσύνθεσης ώστε να αποκτήσουμε μία αίσθηση των δεδομένων σχετικά με το εάν υπάρχουν πρότυπα, εάν υπάρχει σημαντική τάση ή εποχικότητα, ή εάν υπάρχουν ασυνήθιστες τιμές. Παράγεται μία προκαταρκτική στατιστική πρόβλεψη της ζήτησης χρησιμοποιώντας την τεχνική πρόβλεψης που ταιριάζει καλύτερα με το προφίλ και τα χαρακτηριστικά της.

4) Επιλογή & Προσαρμογή Μοντέλου

Σε αυτό το στάδιο επιλέγουμε τη μέθοδο πρόβλεψης και καθορίζουμε τις παραμέτρους

5) Χρήση και αποτίμηση του μοντέλου πρόβλεψης

Το τελικό στάδιο αφορά στην παρακολούθηση και τη συνεχή αξιολόγηση των προβλέψεων σε σύγκριση με την πραγματική διακύμανση της ζήτησης. Η αξιοπιστία της πρόβλεψης μπορεί να βελτιωθεί θεαματικά μέσα από την κατανόηση των αιτιών που ενδεχομένως οδήγησαν σε μεγάλα σφάλματα ή λανθασμένες εκτιμήσεις και ακολουθώντας τις απαραίτητες διορθωτικές κινήσεις. Το κατά πόσο το μοντέλο και οι προβλέψεις είναι ικανοποιητικές κρίνονται μόνο από τον χρόνο και τη μέτρηση του σφάλματος.

3.3 Απλός Κινούμενος Μέσος Όρος

Ο αριθμητικός μέσος είναι ένας εκτιμητής που ελαχιστοποιεί το λάθος στην πρόβλεψη της εκτίμησης του μέσου πληθυσμού. Ο μέσος είναι ένας αμερόληπτος εκτιμητής και ο καλύτερος που υπάρχει στις περιπτώσεις που επιθυμούμε να αναλύσουμε τα αποτελέσματα ενός απλού τυχαίου δείγματος. Τα δεδομένα που θα χρησιμοποιηθούν θα πρέπει να κατανέμονται τυχαία και να ακολουθούν σταθερή ανάλυση ως προς το χρόνο. Μεγάλος αριθμός παρατηρήσεων ενός δείγματος εξασφαλίζει την αξιοπιστία του αποτελέσματος. Για την ελαχιστοποίηση της ανακρίβειας μπορούμε να κρατήσουμε τον ίδιο αριθμό παρατηρήσεων στο δείγμα κατά τον υπολογισμό του μέσου. Το αποτέλεσμα θα είναι μία διαδικασία υπολογισμού του μέσου των πιο πρόσφατων σταθερών παρατηρήσεων του δείγματος. Επομένως η μέθοδος τροποποιείται και ονομάζεται κινητός μέσος όρος.

Η τεχνική πρόβλεψης του κινητού μέσου όρου βασίζεται στον μέσο όρο σταθμισμένων παρατηρήσεων του παρελθόντος. Κάθε φορά που μία νέα παρατήρηση γίνεται διαθέσιμη, υπολογίζεται ένας νέος μέσος απορρίπτοντας την παλαιότερη παρατήρηση του δείγματος από τον μέσο και συμπεριλαμβάνοντας τη νεότερη. Επομένως κάθε πρόβλεψη χρησιμοποιεί τον ίδιο αριθμό παρατηρήσεων δείγματος από τη χρονοσειρά και περιλαμβάνει μόνο τις πρόσφατες παρατηρήσεις, αποκλείοντας οποιαδήποτε μη τυχαία τιμή που έχει προκληθεί από κάποιο γεγονός. Συνεπώς είναι απαραίτητο να διευκρινιστεί από την αρχή ο αριθμός των παρατηρήσεων, επιλέγοντας τον αριθμό των περιόδων N , για τις οποίες θα υπολογιστεί ο κινούμενος μέσος. Όσο μεγαλύτερη είναι η τιμή του N προκύπτουν προβλέψεις με μικρότερη μεταβλητότητα οδηγώντας σε μικρότερη ευαισθησία στις μεταβολές της ζήτησης, με αυτό τον τρόπο ελαχιστοποιείται η επίδραση κάποιων ακραίων τύπων. Στις περιπτώσεις που τα δεδομένα είναι πολύ άστατα, προτιμάται μεγάλη τιμή του N . Η ανάλυση των συνθηκών που επικρατούν θα βοηθήσει στην εξαίρεση των τιμών που έχουν προκληθεί εξαιτίας κάποιου μη τυχαίου γεγονότος. Με αυτό τον τρόπο αποκλείουμε την ύπαρξη συστηματικής συμπεριφοράς στην τιμή της πρόβλεψης.

Η μέθοδος του κινητού μέσου όρου είναι μία πολύ εύκολη διαδικασία πρόβλεψης, αφού οι υπολογιστικές απαιτήσεις είναι μικρές. Αυτή η μέθοδος είναι ευέλικτη γιατί οι παρατηρήσεις μπορούν να μεταβληθούν ώστε να ανταποκρίνονται στα πρότυπα των δεδομένων και το αποτέλεσμα δεν θα οδηγήσει σε συστηματικά σφάλματα. Το μοντέλο χρησιμοποιεί τις ιστορικές παρατηρήσεις και τις προσληφθείσες τιμές F_{t+1} , δίνοντας ίση βαρύτητα σε όλες τις προηγούμενες N τιμές της ζήτησης. Αφού έχουμε στη διάθεση μας τις προσληφθείσες τιμές, μπορούμε να τις συγκρίνουμε με τις ήδη γνωστές τιμές και στη συνέχεια να υπολογίσουμε το σφάλμα πρόβλεψης e . Σε αλγεβρική μορφή διατυπώνεται ως εξής:

Η μέση τιμή των τελευταίων N περιόδων είναι :

$$F_t = \frac{D_{t-1} + D_{t-2} + \dots + D_{t-N}}{N}$$

Η πρόβλεψη για την επόμενη περίοδο παράγεται από την παραπάνω σχέση με την προσθήκη στη χρονοσειρά της πιο πρόσφατης τιμής της μεταβλητής και την αφαίρεση από αυτήν της παλαιότερης τιμής. Ο νέος τύπος είναι:

$$F_{t+1} = \frac{F_t + (D_t - D_{t-N})}{N}$$

3.4 Σταθμισμένος Κινητός Μέσος Όρος

Ο Σταθμισμένος Κινητός Μέσος Όρος, όπως και ο Απλός Κινητός Μέσος Όρος, είναι ένας δείκτης της μέσης τιμής των παρατηρήσεων σε κάποια συγκεκριμένη χρονική στιγμή και για ορισμένο διάστημα παρατηρήσεων. Όμως ενώ ο Απλός Κινητός Μέσος Όρος δίνει ίδια στάθμιση σε όλες τις παρατηρήσεις, ο Σταθμισμένος Κινητός Μέσος Όρος δίνει μεγαλύτερη βαρύτητα στις πιο πρόσφατες παρατηρήσεις. Για αυτό το λόγο δίνει τη δυνατότητα γρηγορότερης αντίδρασης σε αλλαγές τιμών (αλλαγή της ζήτησης). Ο πιο κοινός τρόπος ερμηνείας αφορά τη σύγκριση του Κινητού Μέσου Όρου των τιμών του υποκείμενου μέσου με την τρέχουσα τιμή αυτού. Όσο η τρέχουσα τιμή παραμένει πάνω από τον Κινητό Μέσο Όρο.

3.5 Απλή Εκθετική Εξομάλυνση

Η μέθοδος της εκθετικής εξομάλυνσης βασίζεται στην εκθετική μείωση της βαρύτητας που δίνεται στα στοιχεία των προηγούμενων περιόδων. Η βαρύτητα των στοιχείων είναι τόσο μικρότερη όσο παλαιότερα είναι αυτά. Αντίστοιχα, τα πιο πρόσφατα δεδομένα έχουν μεγαλύτερη βαρύτητα από τα παλαιότερα. Αυτή η μέθοδος συνήθως χρησιμοποιείται στο βραχυπρόθεσμο προγραμματισμό, σε περιπτώσεις όπου ο χρονικός ορίζοντας της πρόβλεψης είναι σχετικά μικρός, για πρόβλεψη της ζήτησης μίας ημέρας ή μίας εβδομάδας. Σε αυτή την τεχνική δεν υπάρχουν διαθέσιμες πληροφορίες για την αιτιακή σχέση που συνδέει την προς πρόβλεψη μεταβλητή και τους ανεξάρτητους παράγοντες που την επηρεάζουν. Οι απαιτήσεις σε υπολογιστικό χρόνο και η αποθήκευση δεδομένων για την εφαρμογή τους είναι μικρές, αφού αποθηκεύονται λιγότερα στοιχεία σε σχέση με τη μέθοδο του κινούμενου μέσου και η πρόβλεψη αναπροσαρμόζεται καθώς γίνονται γνωστά τα καινούργια δεδομένα, με την απλή εισαγωγή τους και την εφαρμογή απλών μαθηματικών σχέσεων.

Το πρότυπο της απλής εκθετικής εξομάλυνσης εφαρμόζεται όταν τα δεδομένα δεν έχουν στοιχεία τάσης, εποχικότητας ή κυκλικότητας. Ενώ υπάρχει μία μέση τιμή της μεταβλητής, γύρω από την οποία κυμαίνονται οι τιμές, εξαιτίας τυχαιών παραγόντων. Το πρότυπο χρησιμοποιεί την πρόβλεψη καθώς και την αντίστοιχη πραγματική τιμή της μεταβλητής για την τρέχουσα περίοδο για να προβλέψει την τιμή της μεταβλητής κατά τις επόμενες περιόδους. Στην πραγματικότητα, η εκθετική εξομάλυνση αποτελεί μία εξέλιξη της μεθόδου του κινούμενου μέσου, αφού η μέθοδος εξομάλυνσης ομαλοποιεί τις παρατηρήσεις από το παρελθόν με σκοπό να αντιμετωπίσουν την τυχαιότητα. Η πρόβλεψη γίνεται λαμβάνοντας υπόψη τις τιμές προηγούμενων περιόδων, δίνοντας μεγαλύτερη βαρύτητα στις πιο πρόσφατες τιμές μέσω της σταθεράς εξομάλυνσης a , όπου η τιμή του a κυμαίνεται από 0 έως 1.

Ο μαθηματικός τύπος είναι:

$$F_{t+1} = aD_t + a(1-a)D_{t-1} + a(1-a)^2 D_{t-2} + \dots$$

Απλούστερα ο τύπος γράφεται:

$$F_{t+1} = F_t + a(D_t - F_t)$$

Η παράμετρος a συνήθως παίρνει τιμές μεταξύ 0,10 και 0,30. Κατά αντιστοιχία με την παράμετρο N του κινούμενου μέσου όρου, μεγάλες τιμές του a δίνουν μεγαλύτερη βαρύτητα σε πρόσφατες τιμές και κατά συνέπεια οδηγούν σε μεγαλύτερη ευαισθησία σε μεταβολές της ζήτησης.

3.6 Σφάλμα Πρόβλεψης

Η αξιοπιστία μίας μεθόδου πρόβλεψης μπορεί να εκτιμηθεί βάσει των αποκλίσεων που εμφανίζονται μεταξύ των προβλέψεων που παράγει αυτή και των τιμών της μεταβλητής που διαμορφώνονται στην πραγματικότητα. Για την μέτρηση των αποκλίσεων μπορούν να χρησιμοποιηθούν διαφορετικοί δείκτες.

Σε ένα πλήθος δεδομένων, που αντιστοιχούν σε N περιόδους, για τις οποίες έχουμε διαθέσιμες τις πραγματικές τιμές και τις αντίστοιχες προβλέψεις. Το σφάλμα της πρόβλεψης υπολογίζεται ως η διαφορά της πρόβλεψης από την πραγματική τιμή της ζήτησης στη συγκεκριμένη περίοδο.

Ο μαθηματικός τύπος είναι:

$$e_t = D_t - F_t$$

Για την αξιολόγηση της ακρίβειας των διαφορετικών μεθόδων πρόβλεψης χρησιμοποιούνται σχετικοί δείκτες. Ο πιο αποτελεσματικός και στατιστικά θεμελιωμένος δείκτης για την μέτρηση του σφάλματος θεωρείται το μέσο τετραγωνικό σφάλμα (MSE). Ο τύπος ορίζει το σφάλμα ως το άθροισμα των τετραγώνων των σφαλμάτων της πρόβλεψης, e_t , διαιρεμένο με το μέγεθος του δείγματος.

$$MSE = \frac{\sum (D_t - F_t)^2}{n} = \frac{\sum e_t^2}{n}$$

Στη μέθοδο της εκθετικής εξομάλυνσης χρησιμοποιούνται διαφορετικές τιμές για τη σταθερά της εξομάλυνσης. Η τιμή του δείκτη θα πρέπει να τείνει στο μηδέν ($\alpha < 0,3$) ώστε η μέθοδος να χαρακτηριστεί αξιόπιστη. Σε αντίθετη περίπτωση η υψηλή τιμή του δείκτη ($\alpha > 0,3$) δείχνει ότι οι προβλέψεις αποκλίνουν σημαντικά από τις τιμές της μεταβλητής που διαμορφώνονται στην πράξη και επομένως κάποιο εναλλακτικό μοντέλο πρόβλεψης είναι καταλληλότερο.

4. Αυτόματες Ταμειολογιστικές Μηχανές

Το κεφάλαιο αυτό στηρίζεται σε άρθρα της Εφης Καραγεωργίου, Ηλεκτρονική Τράπεζα: Στροφή από τις τράπεζες στα εναλλακτικά δίκτυα Καλύτερη εξυπηρέτηση για τον καταναλωτή από την Ελληνική Ένωση Τραπεζών, Καρακέρεζης Α, 1999, Συγχρονες τάσεις στρατηγικών στην ηλεκτρονική τραπεζική, Manager Περιοδικό της ΕΕΔΕ, Summary of website research roll results, 2011, από European ATM Security Team (EAST), Financial Institution Atms: The roles they play in the 21st century, The Benefits of Real-Time ATM Management, ATM Software: Trends Analysis for 2007-2008, ATM Software: Trends Analysis for 2009, 2010 ATM Software Trends and Analysis, 2011 ATM Software Trends and Analysis, 2012 ATM Software Trends and Analysis, Reliable Wireless Connections Enable ATM Expansion, Transforming Bank Branches Through Video Banking, Three Ways ATM Processing Can Maximize Interchange Revenues, Protect Your Financial Institution from ATM Fee Reg E Lawsuits καθώς και στα βιβλία των Καρακώστα Κ.Ιωάννη και Τσίρου Π.Σίσσυ – Σπυριδούλα, 2007, Πιστωτική Κάρτα, του Λυμπερόπουλος Κ., 1994, Στρατηγικό τραπεζικό μάρκετινγκ. Οι δικτυακές σελίδες που μας βοήθησαν είναι οι ακόλουθες en.wikipedia.org, www.atmmarketplace.com, www.euretirio.com, www.brinksblog.com, ww.commercialsecuritydevice.com.

Οι Αυτόματες Ταμειολογιστικές Μηχανές (ATM - automated teller machine) είναι μια από τις πιο σημαντικές τεχνολογικές εφευρέσεις του δεύτερου μισού του εικοστού αιώνα, ηλεκτρονικές συσκευές που παρέχουν στους πελάτες ενός χρηματοπιστωτικού ιδρύματος τη δυνατότητα διεκπεραίωσης οικονομικών συναλλαγών σε δημόσιο χώρο χωρίς την ανάγκη παρέμβασης υπαλλήλου ακόμα και όταν τα καταστήματα της τράπεζας είναι κλειστά. Μέσω των ATMς παρέχονται ποικίλες υπηρεσίες όπως είναι η ανάληψη και κατάθεση μετρητών, ενημέρωση υπολοίπου λογαριασμού, μεταφορά ποσού σε λογαριασμό τρίτου, πληρωμή πιστωτικών καρτών ή δανείων.

Τα ATMς παρέχουν 24ωρη on – line εξυπηρέτηση στους πελάτες με δυνατότητα χρήσης τους σε διευρυμένο ωράριο και γεωγραφικής διασποράς των σημείων εξυπηρέτησης και εκτός των καταστημάτων του Οργανισμού. Επομένως το σύστημα διαχείρισης τους παίζει σημαντικό ρόλο στη λειτουργία μίας τράπεζας. Σε αυτό συμβάλλουν οι κατασκευάστριες εταιρείες με την τροφοδότηση της διεθνούς τραπεζικής αγοράς με νέα μοντέλα ATMς που με το κατάλληλο λογισμικό παρουσιάζουν σημαντικά πλεονεκτήματα. Τα νέα μηχανήματα είναι φιλικότερα προς το χρήστη και πραγματοποιούν τις συναλλαγές σε μικρότερο χρόνο παρέχοντας τη δυνατότητα επιλογής μεταξύ δύο γλωσσών, Ελληνικά ή Αγγλικά

Με τη χρήση των ATMς ελαχιστοποιείται ο χρόνος αναμονής και εξυπηρέτησης του πελάτη απελευθερώνεται ανθρώπινο δυναμικό από τις εργασίες απλών συναλλαγών, όπου η τράπεζα το χρησιμοποιεί σε άλλες πιο σύνθετες εφαρμογές, προσφέροντας ποιοτική εξυπηρέτηση στον πελάτη. Επιπλέον, το κόστος των συναλλαγών που εκτελούνται μέσω των ATMς είναι σταθερό και πολύ μικρότερο για τις τράπεζες σε σύγκριση με τις παραδοσιακές συναλλαγές μέσω των ταμείων (tellers). Μια συναλλαγή στο γκισέ κοστίζει στην τράπεζα περίπου 1.5€ (σε εργατώρες, κόστος λειτουργίας καταστήματος), η ίδια συναλλαγή κοστίζει μόλις 0.25€, αν πραγματοποιηθεί μέσω των εναλλακτικών δικτύων. Το ATM μειώνει το κόστος μεταφοράς στους κατόχους καρτών αφού φέρνει τη τράπεζα κοντά στο τόπο διαμονής, εργασίας ή δημοφιλής καταστήματα όπως εμπορικά κέντρα, σούπερ μάρκετ, σιδηροδρομικούς σταθμούς, βελζινάδικα, ταχυδρομεία, πανεπιστήμια, εστιατόρια και μπαρ. Όσον αφορά την προμήθεια του χρηματοοικονομικού οργανισμού για την παροχή υπηρεσιών μέσω ATMς, η τάση που επικρατεί είναι η τράπεζα να τιμολογεί διαφορετικά αυτές που προσφέρει στους πελάτες της, σε σχέση με αυτές που παρέχει σε πελάτες άλλων πιστωτικών ιδρυμάτων.

Τα ATM ανακυκλώνουν το μεγαλύτερο μέρος των τραπεζογραμματίων στις σύγχρονες οικονομίες. Ο συγκεκριμένος κλάδος αποτελεί μία τεράστια παγκόσμια αγορά αξίας περίπου 15 δισεκατομμύρια δολάρια, αφού παρέχει θέσεις εργασίας τόσο σε κατασκευάστριες εταιρείες όσο και σε επιχειρήσεις υποστήριξης & διαχείρισης σχετικά με τα ATM. Έχουν συμβάλει στην επέκταση ωρών λειτουργίας των καταστημάτων πέρα από τα όρια 9-5 και έχουν αυξήσει τις ετήσιες πωλήσεις των εμπόρων λιανικής πώλησης κατά 25% καθώς οι κάτοχοι καρτών αναζητούν ATM ως πηγή μετρητών. Οι άνθρωποι μεταφέρουν λιγότερα μετρητά, δεδομένου ότι μπορούν εύκολα να πραγματοποιήσουν αναλήψεις από το λογαριασμό τους, γεγονός που οδηγεί σε χαμηλότερη μέση απώλεια μετρητών κατά τη διάρκεια ληστειών και κλοπών.

Η ηλεκτρονική μισθοδοσία τόσο στον ιδιωτικό όσο και στο δημόσιο τομέα έχει επηρεαστεί θετικά από τη ύπαρξη των συγκεκριμένων μηχανημάτων αφού η μισθοδοσία των εργαζομένων κατατίθεται απευθείας σε τραπεζικό λογαριασμό όπου το ποσό είναι διαθέσιμο προς ανάληψη από κάθε μέρος όπου υπάρχει ένα ΑΤΜ, παρέχοντας διέξοδο για κοινό όφελος και καλή διαβίωση. Τα ΑΤΜ μπορούν να χρησιμοποιηθούν για εξαγωγή δημόσιες ανακοινώσεις, για παράδειγμα, η αστυνομία στο Ηνωμένο Βασίλειο έχει χρησιμοποιήσει ΑΤΜ για να διαφημίσει την εκστρατείες «μην πίνετε και οδηγείτε». Οι εθνικές, κρατικές και τοπικές υπηρεσίες χρησιμοποιούν ως αποδεικτικά στοιχεία τα αρχεία βίντεο των συναλλαγών για να πιάσουν και να καταδικάσουν εγκληματίες που σχετίζονται με κλοπή ταυτοτήτων και τραπεζικών καρτών με σκοπό την κατάχρηση.

Συνοψίζοντας τα ΑΤΜ έχουν ωφελήσει τις τράπεζες, τους κατόχους καρτών, τους εμπόρους και τις κυβερνήσεις για τέσσερις δεκαετίες. Είναι μια εξαιρετικά δημοφιλές και αξιόπιστη παγκόσμια τεχνολογία που είναι στην πρώτη γραμμή της σύγχρονης λιανικής πώλησης, διότι περίπου το 85% όλων των συναλλαγών σήμερα εξακολουθούν να διεξάγονται σε μετρητά.

4.1 Ιστορική Αναδρομή

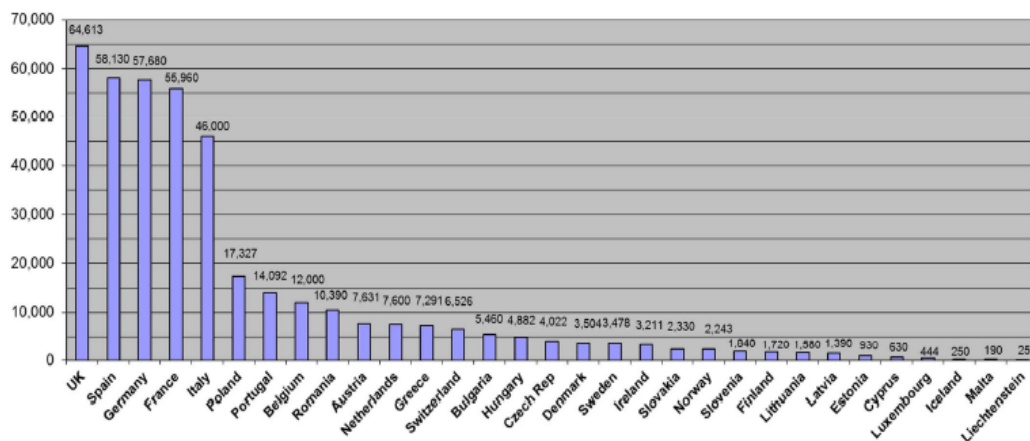
- 1939: Η Τράπεζα αυτοματισμού ξεκίνησε με την εφεύρεση του ATM από τον Luther George Simjian (1905-1997). Η αρχική ιδέα ήταν η δημιουργία μίας «τρύπας» (hole-in-the-wall machine) όπου θα επιτρέπουν στους πελάτες να κάνουν οικονομικές συναλλαγές χωρίς να έρχονται πρόσωπο με πρόσωπο με τους ταμίες. Η ιδέα κατάφερε να πείσει και μετά από έξι μήνες υπήρχε μεγάλη ζήτηση από παίκτες τυχερών παιχνιδιών.
- 1960: Ο προκατόχος ATM εγκαθίσταται από τη Α' Εθνική της Νέας Υόρκης City Bank (σήμερα Citibank) με τη μορφή ενός lobby υποκατάστημα. Η ιδέα είχε ως στόχο να πληρώσουν οι πελάτες τους λογαριασμούς, να πάρουν απόδειξη αλλά να μην έρθουν σε οπτική επαφή με τον ταμεία.
- 1967: Πατέρας των ATM θεωρείται ο John Shepherd- Barron που είχε τη φαινή ιδέα ενός μηχανήματος αυτόματης ανάληψης μετρητών, όταν κλειδώθηκε έξω από την τράπεζα πριν το Σαββατοκύριακο. Στις 27 Ιουνίου του 1967 έξω από το υποκατάστημα της Barclay's Bank της Βρετανίας παρουσιάστηκε η πρώτη μηχανή. Στην αρχή οι ενδιαφερόμενοι προμηθεύονταν ειδικά παραστατικά πληρωμής, τα οποία τοποθετούσαν σε ένα συρτάρι του μηχανήματος, στη συνέχεια εισήγαγαν έναν κωδικό κι από ένα δεύτερο συρτάρι έβγαινε το αντίστοιχο χαρτονόμισμα. Μάλιστα η αρχική ιδέα του Shepherd- Barron προέβλεπε εξαψήφιους κωδικούς, αλλά η απλοποίησή της κρίθηκε αναγκαία γιατί η σύζυγός του Caroline δεν μπορούσε να θυμάται τόσους πολλούς αριθμούς.
- 1968: Ο Wenzel σε συνεργασία με την τράπεζα Barclay δημιούργησε μία μηχανή που μπορούσε να κωδικοποιεί αξίες μετρητών σε πλαστικές κάρτες, τις οποίες οι πελάτες προμηθεύονταν από τους ταμίες των καταστημάτων της. Το πρόβλημα που προέκυψε ήταν ότι το μηχάνημα κρατούσε τις κάρτες, και οι πελάτες πρέπει να αγοράσουν νέες, σε περίπτωση που ήθελαν να κάνουν περισσότερες συναλλαγές.
- 1971: Ο Docute παρήγαγε μία πλήρη μηχανή αυτόματης ανάληψης μετρητών που είχε τη δυνατότητα να εκδίδει τραπεζογραμμάτια διαφορετικού είδους, αυτή η λειτουργία ATM ονομάστηκε "Total Teller".
- 1974: Δόθηκε σε λειτουργία το πρώτο ATM με τη σημερινή μορφή καθώς διέθετε online επικοινωνία με τα συστήματα της τράπεζας.
- 2005: Εμφανίστηκε το πρώτο ATM που λειτουργούσε με Windows.
- 2006: Η Citibank συνάψει συμφωνία για να θέσει μηχανήματα αυτόματης ανάληψης της (ATM) σε περισσότερα από 6.700 καταστήματα ψιλικών στις Ηνωμένες Πολιτείες.
- 2010: Η εταιρεία NCR παρουσίασε το πρώτο ATM όπου παρέχει στο καταναλωτή εξατομικευμένες τραπεζικές μέσω βίντεο κλήσεων. Με αυτό τον τρόπο ο πελάτης εξυπηρετείται από αντιπρόσωπο της τράπεζας με τον οποίο έχει οπτική επαφή.
- 2011: Υπάρχουν σε λειτουργία πάνω από 1.67 εκατομμύρια ATM παγκοσμίως. Από αυτά τα 400.000 βρίσκονται στις Ηνωμένες Πολιτείες Αμερικής, 13.000 στη Νότια Αφρική. Τα μηχανήματα αυτόματης ανάληψης έχουν κερδίσει την εμπιστοσύνη και των πιο δύσπιστων πελατών καθώς έχουν ραγδαία ανάπτυξη αφού κάθε 6 λεπτά μια νέα ATM εγκαθίσταται σε κάποιο σημείο του πλανήτη.
- 2012: Η εταιρεία Rabatit σχεδίασε το πρώτο ATM που θα παρέχει στους καταναλωτές δωροκάρτες και προπληρωμένη κάρτα αντί για μετρητά.

Τέλος ας σημειώσουμε ότι η πλέον απομακρυσμένη μηχανή αυτόματης ανάληψης βρίσκεται στο σταθμό ΜακΜούρντο στο Νότιο Πόλο σε γεωγραφικό πλάτος 77ο 51'N όπου εξυπηρετεί μια μικρή μόνιμη επιστημονική βάση περίπου 200 ατόμων. Τα δύο ATM, που έχουν προσφερθεί από την τράπεζα Wells Fargo, δεν χρεώνουν επιπλέον κόστος για της παροχή χρημάτων. Το μηχάνημα που βρίσκεται ψηλότερα απ' όλα τ' άλλα είναι ένα ATM στη μεθόριο Κίνας-Ινδίας σε υψόμετρο 4.400 μέτρων πάνω από την επιφάνεια της θάλασσας.

4.2 Δημοσκοπήσεις

Κάθε 6 λεπτά, ένα νέο ATM έχει εγκατασταθεί και το 57% των συναλλαγών από ATM αντιπροσωπεύει τις αναλήψεις μετρητών, το 15% τις καταθέσεις και 11% ερωτήσεις υπόλοιπου και το 17% οποιαδήποτε άλλη επιλογή (π.χ. μεταφορά χρημάτων, πληρωμές παγίων εντολών κλπ). Παγκοσμίως το πόσο των αναλήψεων φτάνει τα 49 δισεκατομμύρια ετησίως. Οι αυτόματες ταμιακές μηχανές συνεχίζουν να είναι δημοφιλείς παγκοσμίως και να κερδίζουν συνεχώς έφοδος σε σύγκριση με τα παραδοσιακά κισσέ των τραπεζικών καταστημάτων. Γεγονός που επιβεβαιώνεται και από τη ραγδαία αύξηση των ATM,

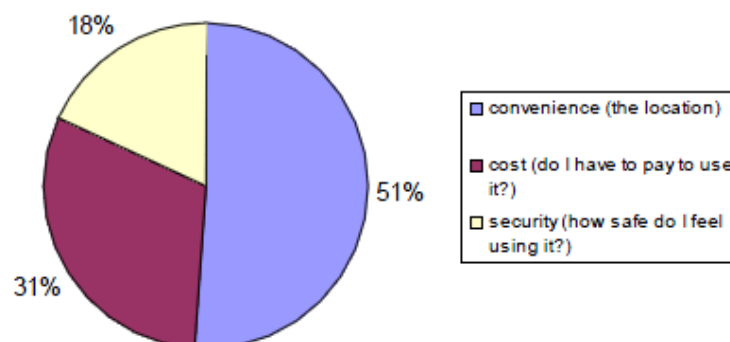
European ATM Numbers (As at 30th June 2011)



συγκεκριμένα σε έρευνα που πραγματοποιήθηκε το 2011 από την European ATM Security, τον μεγαλύτερο διεθνές οργανισμό που ασχολείται με τα ATM (ασφάλεια, αναβάθμιση, ανταγωνισμός) ανέδειξε την Ελλάδα με 7.291 ATM στη 12η θέση ανάμεσα σε 31 χώρες.

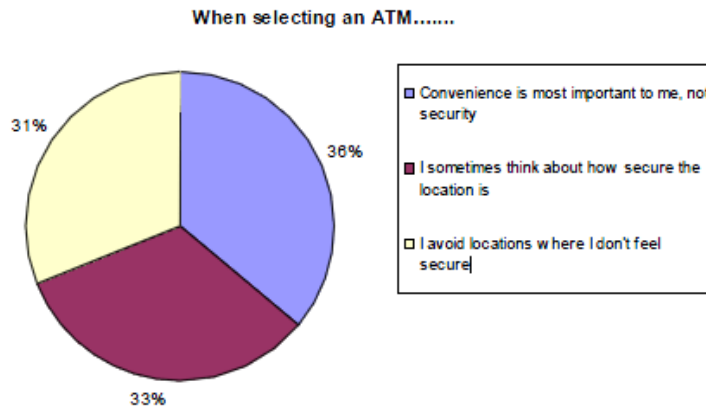
Η ασυνήθιστη για πολλούς εμπιστοσύνη που έχουν κερδίσει τα ATM είναι αποτέλεσμα τις συνεχής και σωστής ενημέρωσης του κοινού σχετικά με τις δυνατότητες και την φερεγγυότητα των συγκεκριμένων μηχανών. Παρακάτω θα δούμε τα αποτελέσματα της ίδιας έρευνας σχετικά με την

When I select an ATM the main factor influencing my choice is.....

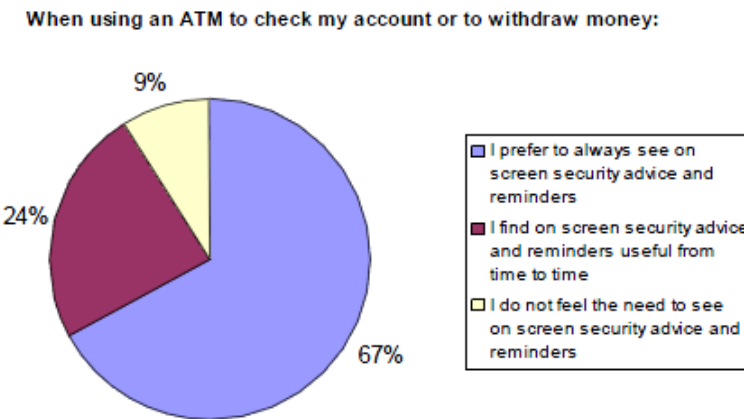


ικανοποίηση και γνώμη των πελατών και εμφανίζει ότι η χρήση των ATM κερδίζει συνεχώς πιστούς υποστηρικτές. Οι χρήστες των ATM επιλέγουν τα μηχανήματα με βάση την τοποθεσία σε συνέχεια το κόστος και στο τέλος για την ασφάλεια που τους παρέχουν. Επομένως όσο περισσότερες μηχανές είναι διαθέσιμες στο κοινό, τόσες περισσότερες επιλογές έχει ο πελάτης.

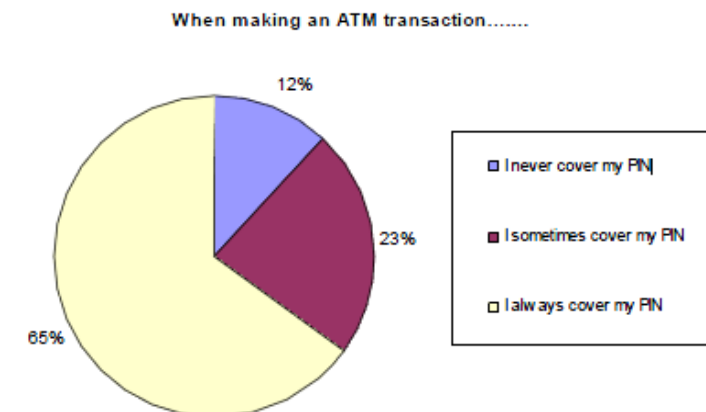
Η δυνατότητα απόκτησης χρημάτων 24 ώρες τη μέρα σε συνδυασμό με το ασφαλές περιβάλλον είναι τα βασικά σημεία που θα πείσουν κάποιον πελάτη να χρησιμοποιήσει τα ATM.



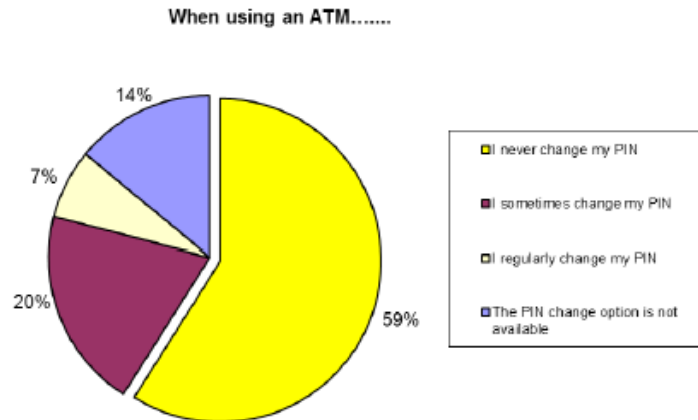
Οι χρήστες είναι εξοικειωμένοι με τις μηχανές και ακολουθούν πιστά τις οδηγίες που εμφανίζονται στην οθόνη.



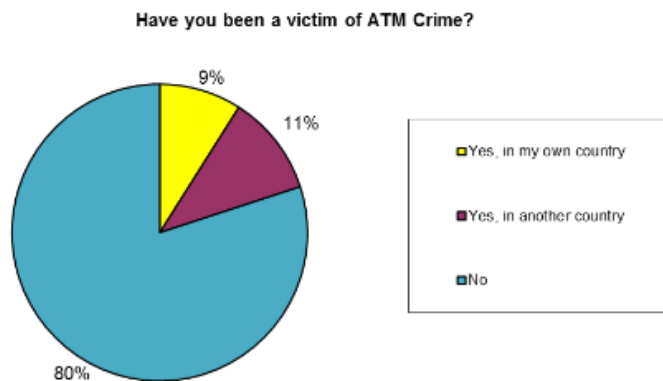
Οι χρήστες των ATM γνωρίζουν τους κινδύνους που παραμονεύουν και είναι προσεκτική σε κάθε συναλλαγή τους καλύπτοντας το PIN της κάρτας τους κατά τη διάρκεια της πληκτρολόγησης.



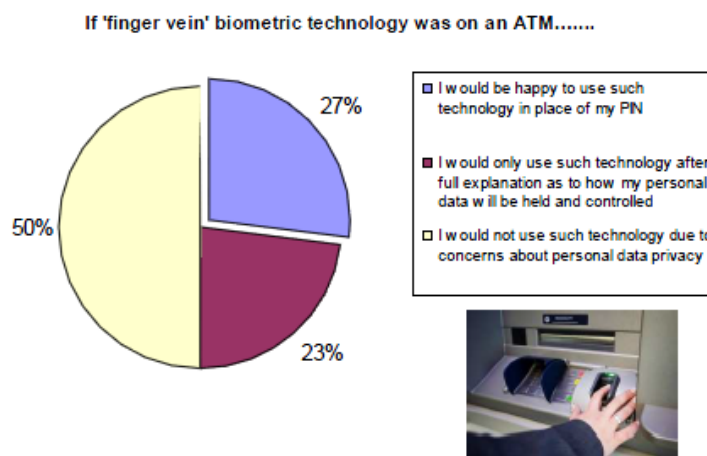
Οι πελάτες σπάνια θα άλλαζαν το PIN της κάρτας τους εάν δε υπήρχε κάποιος σημαντικός λόγος. Αντιμετωπίζουν τα ATM ως ένα τρόπο παραλαβής μετρητών αλλά δε τα εμπιστεύονται ώστε να αλλάξουν οποιαδήποτε προσωπικό τους στοιχείο.



Οι χρήστες φοβούνται την πιθανότητα να παίζουν θύμα ηλεκτρονικών υποκλοπών των προσωπικών τους στοιχείων και για αυτό το λόγο είναι πολύ προσεκτική κατά την εισαγωγή του PIN. Αντίθετα όμως δε θεωρούν μεγάλη τη πιθανότητα να πέσουν θύμα κλοπής από άγνωστο που θα διεκδικήσει το ποσό της ανάληψής τους.



Οι πελάτες εμπιστεύονται τα ATM αλλά θέλουν να έχουν και των έλεγχο της συναλλαγής για αυτό δε θα εμπιστεύονταν μηχανήματα με ανίχνευση δακτυλικών αποτυπωμάτων.



4.3 Δυνατότητες

Οι οικονομικές συναλλαγές καλύπτουν μία μεγάλη ποικιλία συναλλαγών που μπορεί να κάνει ο συναλλασσόμενος και σε ταμείο της τράπεζας. Οι βασικότερες συναλλαγής είναι ανάληψη και κατάθεση μετρητών. Ανάληψη μετρητών μπορεί να πραγματοποιηθεί από τον λογαριασμό ή την πιστωτική κάρτα με βάση τα ημερήσια όρια ανάληψης και με δυνατότητα επιλογής μέχρι και τεσσάρων συνδυασμών χαρτονομισμάτων. Η δυνατότητα επιλογής διαφέρει από τράπεζα σε τράπεζα αλλά στην πλειοψηφία τα ATM περιέχουν τραπεζογραμμάτια των 10€,20€ και 50€ ενώ πολύ σπάνια συναντάμε ATM που να έχουν διαθέσιμα χαρτονομίσματα αξίας 100€ . Κατάθεση μετρητών μπορεί να πραγματοποιηθεί σε λογαριασμούς του χρήστη με ή χωρίς χρήση του προσωπικού κωδικού (PIN) και σε αρκετές τράπεζες με ή χωρίς τη χρήση κάρτας, αρκεί να γνωρίζει τον αριθμό του τραπεζικού λογαριασμού ή τον αριθμό της πιστωτικής κάρτας.

Άλλες διατραπεζικές συναλλαγές αφορούν πληρωμή δανείων και καρτών (της ίδιας ή άλλης τράπεζας), κατάθεση μετρητών με σκοπό την αυτόματη εξόφληση λογαριασμού τρίτων οργανισμών τηλεπικοινωνίας και ενέργειας (OTE, Tellas, Wind, Vodafone, HOL, Forthnet, Multichoice-Nova και ΔΕΗ) καθώς και οφειλές προς το δημόσιο (ΙΚΑ, ΤΕΒΕ, ΦΠΑ, Φόρος εισοδήματος, ενιαίο τέλος ακίνητης περιουσίας). Οι μεταφορές κεφαλαίων εντός τράπεζας εκτελούνται σε πραγματικό χρόνο (on line) και αφορούν είτε σε λογαριασμό του ίδιου είτε σε λογαριασμό τρίτου όπου χρειάζεται να τον πληκτρολογήσει είτε να γνωρίζει τον IBAN αριθμό. Για όλες τις μεταφορές σε τρίτους ισχύουν τα ημερήσια όρια. Οι συναλλαγές υλοποιούνται στα πλαίσια διατραπεζικών συστημάτων κυρίως της ΔΙΑΣ ΑΕ και του ΕΡΜΗΣ, επιβαρύνοντας τον πελάτη με έξοδα συναλλαγής από 0,47€ έως 2.47€ ανάλογα με το ποσό της συναλλαγής. Μετά την ολοκλήρωση της συναλλαγής ο χρήστης μπορεί να εκτυπώσει τη σχετική απόδειξη, η οποία θεωρείται νόμιμο παραστατικό της συναλλαγής.

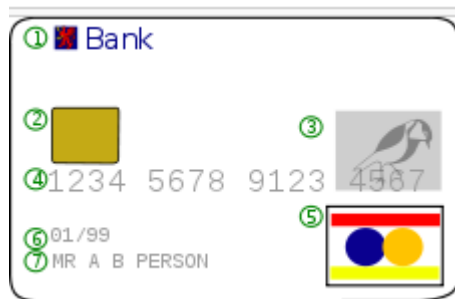
Δεδομένου ότι η ψηφιακή τεχνολογία γίνεται όλο και πιο διαδεδομένες στην αγορά, ATM μπορεί να χρησιμοποιηθεί ως κάτι περισσότερο από ένα κανάλι διανομής μετρητών. Το ATM δίνει τη δυνατότητα στο πελάτη του πιστωτικού ιδρύματος να πραγματοποιήσει απλές συναλλαγές όπως ερώτηση υπολοίπου τραπεζικού λογαριασμού και η δυνατότητα έκδοσης mini-statement (10 τελευταίες κινήσεις είτε για λογαριασμό είτε για πιστωτική κάρτα), αλλαγή του προσωπικού κωδικού (PIN) της κάρτας, ενεργοποίηση ή φόρτιση προπληρωμένης πιστωτικής κάρτας, υπόλοιπο του τελευταίου λογαριασμού, πιστωτικό όριο, διαθέσιμο υπόλοιπο, συνολικό υπόλοιπο, διαθέσιμο ποσό ανάληψης, ελάχιστη καταβολή και ημερομηνία πληρωμής. Ο πελάτης μπορεί να δει λεπτομέρειες σχετικά με τα επενδυτικά του προϊόντα όπως αξίες, αριθμός μεριδίων, τιμή μεριδίου.

Τα ATM συνδέονται σε ένα διεθνές σύστημα αλληλένδετων δικτύων που δίνουν τη δυνατό σε τουρίστες και σε ταξιδιώτες στο εξωτερικό να πραγματοποιήσουν συναλλαγές σε ξένο νόμισμα με εγχώρια τραπεζική κάρτα. Παγκοσμίως εκατομμύρια μετανάστες μπορούν να πραγματοποιήσουν διασυνοριακά εμβάσματα παρέχοντας μία επιπλέον πηγή πρόσθετου εισοδήματος στις οικογένειες τους που βρίσκονται πίσω στη πατρίδα.

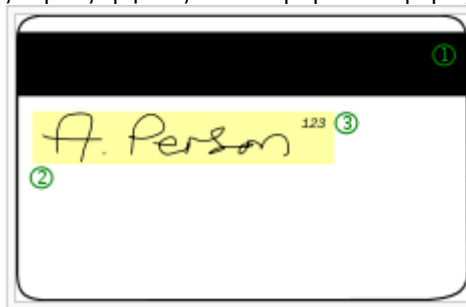
4.4 Κάρτες αυτόματης συναλλαγής

Οι κάρτες αυτόματης συναλλαγής ή χρεωστικές κάρτες χρησιμοποιούνται ως μέσω πραγματοποίησης συναλλαγών σε Αυτόματες Ταμειακές Μηχανή (ΑΤΜ). Τα χρηματοπιστωτικά ιδρύματα εκδίδουν τις συγκεκριμένες κάρτες, δίνοντας τη δυνατότητα στους πελάτες τους να έχουν πρόσβαση σε ΑΤΜ και σε τερματικά καρτών εμπορικών καταστημάτων. Σε αντίθεση με στις πιστωτικές κάρτες, οι αγορές με μία κάρτα ΑΤΜ απαιτεί έλεγχο ταυτότητας μέσω του προσωπικού αριθμού αναγνώρισης (PIN).

Η κάρτα αυτόματης συναλλαγής μοιάζει πολύ εξωτερικά με τις πιστωτικές κάρτες καθώς έχουν το ίδιο μέγεθος δηλαδή 85,60 × 53,98 χιλιοστά. Εσωτερικά όμως παρατηρούνται σημαντικές διαφορές καθώς η συγκεκριμένη κάρτα ενσωματώνει ένα μικροεπεξεργαστή οποίος είναι συνδεδεμένος με έναν υπολογιστή (τραπεζικό σύστημα) επιτρέποντας τη πρόσβαση στα στοιχεία του καρτούχου μετά την ταυτοποίηση των στοιχείων του. Με αυτό τον τρόπο διασφαλίζεται η χρήση της κάρτας από μη νόμιμο χρήστη. Στη πιστωτική κάρτα τα δεδομένα του καρτούχου είναι αποθηκευμένα στη μαγνητική ταινία που ενσωματώνεται στο πλαστικό και είναι εύκολο να παραλλαχθούν ή να διαγραφούν.



1. Λογότυπο της εκδοτήρια τράπεζας
2. Chip: Οι πληροφορίες που αποθηκεύονται στο chip (μικροεπεξεργαστή) προστατεύονται με κρυπτογράφηση. Οι πληροφορίες αυτές, συνδυάζονται με την υπογραφή του κατόχου της κάρτας ή με τον αριθμό PIN για να είναι ακόμα πιο ασφαλείς οι συναλλαγές.
3. Τρισδιάστατο ολόγραμμα που αλλάζει χρώμα κατά την περιστροφή της κάρτας κάτω από το φως διαφορετικό για κάθε είδος κάρτας. Στη Visa απεικονίζεται το περιστέρι, στη Mastercard η υδρόγειος. Τα τέσσερα τελευταία ψηφία του αριθμού της κάρτας πρέπει να είναι τυπωμένα στο πλαίσιο του ολογράμματος.
4. Τυπωμένος αριθμός λογαριασμού. Οι τυπωμένοι αριθμοί δεν ξεχωρίζουν με την αφή από την επιφάνεια της κάρτας. Οι αριθμοί καρτών περιλαμβάνουν 16 ψηφία, ξεκινούν με ένα 4 και είναι ταξινομημένοι σε τέσσερις ομάδες των τεσσάρων αριθμών.
5. Λογότυπο του είδους της κάρτας
6. Ημερομηνία λήξης. Πάνω στη κάρτα πρέπει να αναγράφεται πάντα ο μήνας και το έτος λήξης της. Η ημερομηνία προσδιορίζει το χρονικό περιθώριο μέσα στο οποίο μπορεί να χρησιμοποιηθεί η συγκεκριμένη κάρτα. Η περίοδος χρήσης λήγει την τελευταία ημέρα του αναγραφόμενου μήνα.
7. Το όνομα του κατόχου της κάρτας εμφανίζεται στο μπροστινό μέρος της κάρτας.



1. Μια παραδοσιακή μαγνητική ταινία εμφανίζεται στο πίσω μέρος της κάρτας.

2. Πλαίσιο υπογραφής. Στον ειδικό χώρο για την υπογραφή θα πρέπει πάντα να υπάρχει η υπογραφή του πελάτη καθώς χρησιμεύει στην ταυτοποίηση της υπογραφής του κατόχου με αυτή στην απόδειξη της κάρτας. Το μήκος του πλαισίου υπογραφής ποικίλει ανάλογα με τον τύπο της κάρτας.
3. Ο κωδικός ασφαλείας είναι ένας αριθμός που αναγράφεται στο πίσω μέρος μίας χρεωστικής κάρτας και χρησιμοποιείται για την επαλήθευση της ταυτότητας του πελάτη και την ελαχιστοποίηση του κινδύνου απάτης στις ηλεκτρονικές συναλλαγές ή στα ΑΤΜ. Στις ηλεκτρονικές συναλλαγές, ο πελάτης πρέπει να συμπληρώσει τον κωδικό αριθμό ασφαλείας (CVV) μετά την εισαγωγή όλων των στοιχείων της κάρτας, κι επειδή δεν αποτελεί μέρος του αριθμού της κάρτας, πρέπει να τον καταχωρεί εκ νέου κάθε φορά. Ακόμα και στην περίπτωση που ένας υπολογιστής έχει αποθηκευμένα και συμπληρώνει αυτόματα τα στοιχεία της πιστωτικής κάρτας, ο αριθμός CVV πρέπει να εισάγεται κάθε φορά. Ο κωδικός αριθμός ασφαλείας έχει διάφορες ονομασίες ανάλογα με τον εκδότη. Στην περίπτωση της Visa ονομάζεται "Card Verification Value" (CVV), στην περίπτωση της MasterCard ονομάζεται "Card Validation Code" (CVC) και στην περίπτωση της American Express ονομάζεται "Card Identification Number" (CID)

4.5 Παρουσίαση γενικών στοιχείων και παραμέτρων

Κατόπιν έρευνας που πραγματοποιήθηκε σε διαφορετικά χρηματοπιστωτικά ιδρύματα καταλήξαμε σε κάποια συμπεράσματα σχετικά με τρόπο διαχείρισης και επεξεργασίας των Αυτόματων Ταμειακών Μηχανών. Οι τράπεζες ενδιαφέρονται για την καθημερινή πρόβλεψη του συνολικού ποσού αναλήψεων και αντίστοιχα καταθέσεων ώστε να προγραμματίζουν έγκαιρα τις τροφοδοσίες (παραλαβή των χρημάτων που βρίσκονται στο ATM και τοποθέτηση νέου επαρκούς χρηματικού πόσου). Οι τροφοδοσίες μπορούν να πραγματοποιούνται καθημερινά, εκτός σαββατοκύριακου, και όσο γίνεται νωρίς το πρωί, επιθυμητή ώρα μέχρι τις 9π.μ. Στα πληροφοριακά συστήματα των τραπεζών έχουν οριστεί τα σαββατοκύριακα και οι αργίες ως μη διαθέσιμες ημερομηνίες, επομένως όλο τα στοιχεία που παρουσιάζουν αναφέρουν αποτελέσματα ή ενέργειες για τις πέντε εργάσιμες ημέρες της εβδομάδας. Κάθε συναλλαγή χαρακτηρίζεται από δύο ημερομηνίες, την ημερομηνία που πραγματοποιήθηκε η συναλλαγή και την ημερομηνία εγγραφής, δηλαδή ημερομηνία ενημέρωσης του λογαριασμού (πίστωση ή χρέωση). Οι συναλλαγές που πραγματοποιήθηκαν κατά τη διάρκεια του σαββατοκύριακου εμφανίζονται με την ημερομηνία της επόμενης εργάσιμης ημέρας. Τα πληροφορικά συστήματα των τραπεζών παράγουν δεδομένα με ανάλυση σε πέντε και όχι επτά ημέρες.

Στην παρούσα εργασία έχουμε χρησιμοποιήσει δεδομένα συναλλαγών, ποσά και αριθμό συναλλαγών, για τρία μηχανήματα ένα στην Αθήνα, ένα στη Θεσσαλονίκη και ένα στη Λάρισα. Τα τρία μηχανήματα χρησιμοποιούνται για πραγματοποίηση αναλήψεων και καταθέσεων. Τα αρχεία περιέχουν συγκεντρωτικές πληροφορίες ανά ημερομηνία για το χρονικό διάστημα ενός έτους. Οι συναλλαγές που έχουν πραγματοποιηθεί κατά τη διάρκεια σαββατοκύριακου ή επίσημων αργιών έχουν προστεθεί και εμφανίζονται στις συναλλαγές της πρώτης εργάσιμης μέρας που ακολουθεί. Αναλύοντας τα δεδομένα καλούμαστε να προβλέψουμε την καλύτερη ημερομηνία πραγματοποίησης τροφοδοσίας ώστε το μηχάνημα να περιέχουν επαρκές υπόλοιπα για τους πελάτες που θέλουν να πραγματοποιήσουν αναλήψεις και ταυτόχρονα επαρκές χώρος στο μηχάνημα ώστε να δέχεται τις καταθέσεις (φάκελο ή μετρητά). Οι δύο μεγαλύτερες εταιρίες Brink's Hellas και G4S που δραστηριοποιούνται στις Υπηρεσίες Ασφάλειας (μεταφοράς αξιών) και παρέχουν ολοκληρωμένες λύσεις στον τομέα των μηχανημάτων αυτόματων ταμιακών συναλλαγών ATM, προτείνουν στα συνεργαζόμενα πιστωτικά ιδρύματα ανώτερο όριο καταθέσεων 50.000€ και ανώτερο όριο τροφοδοσίας 140.000€.

Στη εργασία θα χρησιμοποιήσουμε το όριο των 50.000€ για καταθέσεις και 100.000€ για τροφοδοσία μηχανημάτων. Επιπλέον οι συναλλαγές που πραγματοποιούνται τις Κυριακές είναι ελάχιστες σε σύγκριση με αυτές που πραγματοποιούνται το Σάββατο και τη Δευτέρα. Οι πελάτες δεν χρησιμοποιούν τα ATM τις Κυριακές καθώς η ανάγκη για μετρητά είναι περιορισμένη αφού τα εμπορικά καταστήματα είναι κλειστά. Η Κυριακή θεωρείται μία μέρα ξεκούρασης και ελάχιστα άτομα θα επισκεφτούν ένα ATM για να καταθέσουν χρήματα με σκοπό την πληρωμή κάρτας ή δανείου. Η χρήση του ATM περιορίζεται με μικρές αναλήψεις από άτομα που χρειάζονται άμεσα χρήματα, συνήθως για να καλύψουν τα έξοδα ψυχαγωγίας. Επομένως από τις τράπεζες η Κυριακή αντιμετωπίζεται ως ουδέτερη μέρα που δεν θα επηρεάσει αισθητά τα υπόλοιπα των μηχανημάτων, ATM. Για αυτό το λόγο στην εργασία θα θεωρήσουμε ότι τη Κυριακή δεν πραγματοποιούνται συναλλαγές στα ATM. Όπως όλα τα μηχανήματα έτσι και οι αυτόματοι ταμιακοί πολίτες μπορούν να τεθούν ανά πάσα στιγμή εκτός λειτουργίας είτε από τεχνικούς λόγους είτε από κακή χρήση. Οι πιο συνηθισμένοι λόγοι είναι να μπλοκάρουν στους ιμάντες χαρτονόμισμα κακής ποιότητας (φθαρμένα ή κομμένα) ή να μπλοκάρει η κάρτα συναλλαγής αλλά και απρόβλεπτοι παράγοντες όπως μία πολύωρη διακοπή ρεύματος θέτει το μηχάνημα εκτός λειτουργίας. Σε αυτές τις περιπτώσεις αυτομάτως ο αριθμός των συναλλαγών μειώνεται αισθητά όχι όμως γιατί δεν υπάρχει ζήτηση από τους πελάτες αλλά το μηχάνημα αδυνατεί να τους εξυπηρετήσει. Σε αντίθετη περίπτωση μία βλάβη άλλου κοντινού μηχανήματος (ίδιας ή διαφορετικής τράπεζας) ή μία ομιλία κυβερνητικού αντιπρόσωπου που θα αναφέρεται σε πιθανή στάση μισθών, αυξάνει των αριθμό των συναλλαγών. Η χρήση του συγκεκριμένου ATM είναι λύση ανάγκης ή ενέργεια πανικού, αλλά σε καμία περίπτωση δεν μπορεί να θεωρηθεί αντιπροσωπευτική χρήση του μηχανήματος. Οι προαναφερθέντες ακραίες περιπτώσεις θα πρέπει να αποκλείονται από το δείγμα για να μην οδηγηθούμε σε λάθος αποτελέσματα.

4.6 Παρουσία του τρόπου ανάλυσης των δεδομένων

Αρχικά θα αναφέρουμε και θα αναλύσουμε τον τρόπο με τον οποίο θα επεξεργαστούμε τα δεδομένα ώστε να επιλέξουμε την πιο αξιόπιστη μέθοδο υπολογισμού την πρόβλεψη την οποία θα χρησιμοποιούμε στην εφαρμογή του προγράμματος μας. Ως δεδομένα έχουμε δύο αρχείο σε μορφή txt, ένα για τις αναλήψεις και ένα για τις καταθέσεις. Κάθε αρχείο περιέχει τις τέσσερις βασικές πληροφορίες που χρειαζόμαστε για την ανάλυση μας, δηλαδή την ημερομηνία εγγραφής των συναλλαγών στα συστήματα του πιστωτικού ιδρύματος (εργάσιμες ημερομηνίες) , το κωδικό του μηχανήματος, τον συνολικό ποσό σε ευρώ των αναλήψεων ή καταθέσεων αντίστοιχα και τον συνολικό αριθμό των συναλλαγών.

Για τον υπολογισμό του απλού κινούμενου μέσου όρου θα χρησιμοποιήσουμε ως δείγμα τα δεδομένα για τις τελευταίες 18εργάσιμες μέρες αφαιρώντας κάθε φορά τις δύο ακραίες τιμές, τι μέγιστη και την ελάχιστη. Με αυτό τον τρόπο υπολογίζουμε τον μέσο όρο των 16 πιο πρόσφατων τιμών και ταυτόχρονα το δείγμα μας δεν επηρεάζεται από τυχαία γεγονότα (π.χ. βλάβη μηχανήματος).

Για τον υπολογισμό του σταθμισμένου κινητού μέσου όρου θα χρησιμοποιήσουμε ως δείγμα τα δεδομένα για τις τελευταίες 18εργάσιμες μέρες. Το δείγμα θα το χωρίσουμε σε τρία ίσα μέρη σχηματίζοντας υποσύνολα με 6 τιμές το καθένα. Από κάθε υποσύνολο θα αφαιρέσουμε τις δύο ακραίες τιμές, θα υπολογίσουμε το μέσο όρο και τελικά θα δώσουμε μεγαλύτερη βαρύτητα στις πιο πρόσφατες παρατηρήσεις. Οι συντελεστές που θα χρησιμοποιούμε είναι 0.7, 0.2 και 0.1 σε αντιστοιχία από τις πιο πρόσφατες στις παλαιότερες τιμές. Με αυτό τον τρόπο περιορίσουμε στο ελάχιστο την επιρροή των ακραίων τιμών.

Για τον υπολογισμό της πρόβλεψης με τη χρήση της μεθόδου της εκθετικής εξομάλυνσης θα χρησιμοποιήσουμε δείγμα από τις 12 πιο πρόσφατες τιμές. Καθώς η συγκεκριμένη μέθοδος βασίζεται στην εκθετική μείωση της βαρύτητας των στοιχείων με σκοπό την αντιμετώπιση της τυχαιότητας δεν υπάρχει λόγος να αφαιρέσουμε τις ακραίες τιμές. Επομένως μειώνουμε το δείγμα μας από 18 σε 12 τιμές αλλά ταυτόχρονα το δείγμα θα πρέπει να περιέχει παρατηρήσεις με τα ίδια χαρακτηριστικά. Για αυτό το λόγο εντοπίζουμε τις εργάσιμες ημέρες του χρόνου που είναι Δευτέρες και διαιρούμε τις τιμές δια του δύο (συναλλαγές Σαββάτου & Δευτέρας). Με αυτό τον τρόπο έχουμε δημιουργήσει ένα ομοιογενές δείγμα για όλες τις τιμές των προηγούμενων περιόδων. Η τιμές του συντελεστή της σταθεράς εξομάλυνσης που θα χρησιμοποιούμε είναι 0.6, 0.3 και 0.1.

Με όλο τους προαναφερθέν τρόπους έχουμε υπολογίσει τη πρόβλεψη των αναλήψεων ή καταθέσεων της επόμενη εργάσιμη ημέρα και μόνο για μία ημέρα. Επομένως κάθε Πέμπτη που καλούμαστε να προβλέψουμε τις τροφοδοσίες της Παρασκευής θα πρέπει να ελέγξουμε την χρηματική επάρκεια του μηχανήματος για τις συναλλαγές που θεωρούμε ότι θα γίνουν τη Πέμπτη, τη Παρασκευή και το Σάββατο (η Κυριακή θεωρείται ημέρα χωρίς συναλλαγές). Άρα το ποσό της πρόβλεψης θα το πολλαπλασιάσουμε επί τρία. Αντίστοιχα τη Παρασκευή για να αποφασίσουμε για τις τροφοδοσίες της Δευτέρας θα πολλαπλασιάσουμε το ποσό της πρόβλεψης επί δύο καθώς θα πρέπει να υπολογίσουμε τις συναλλαγές της Παρασκευής και του Σαββάτου. Γενικεύοντας τον παραπάνω τρόπο κάθε φορά που ακολουθεί μη εργάσιμη ημέρα, θα πρέπει να πολλαπλασιάσουμε το ποσό των συναλλαγών επί τον αριθμό των αργιών ώστε να αποφασίσουμε αν χρειάζεται ή όχι τροφοδοσία.

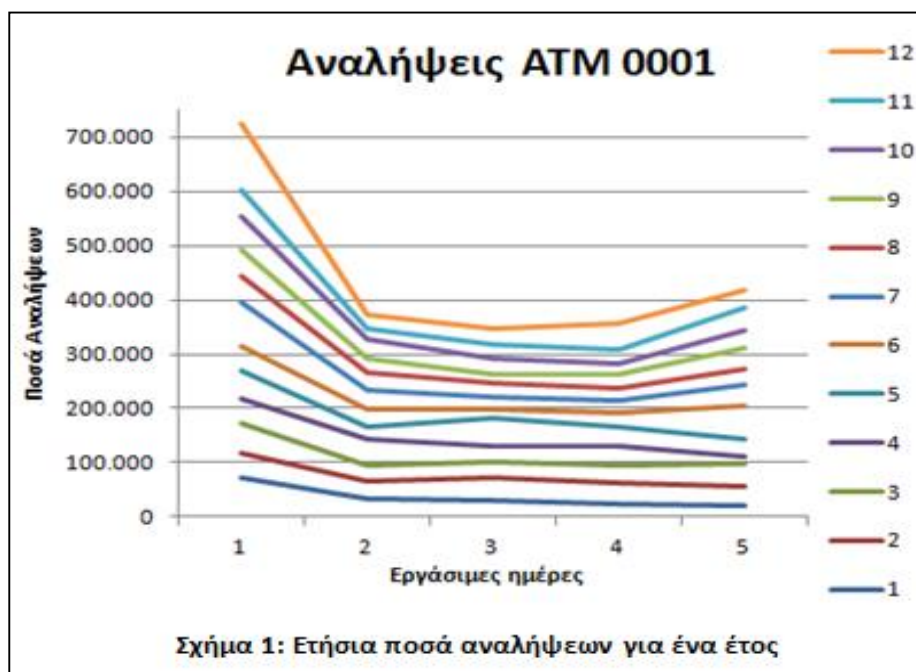
Εφόσον έχουμε ολοκλήρωση τη διαδικασία της επιθυμητής ημέρας τροφοδοσίας με τρεις διαφορετικούς τρόπους και με τρεις διαφορετικές τιμές συντελεστή εξομάλυνσης, καλούμαστε να τους συγκρίνουμε για τα καταλήξουμε στον πιο αξιόπιστο. Ο υπολογισμός του σφάλματος θα πραγματοποιηθεί με τη σύγκριση του προβλεπόμενου με το πραγματικό ποσό συναλλαγών (καθώς τα πραγματικά στοιχεία είναι στη διάθεση μας). Σε αντιστοιχία τη πρόβλεψη της Πέμπτης δε θα τη συγκρίνουμε με το ποσό των συναλλαγών εκείνης της μέρας αλλά με το σύνολο ποσό αναλήψεων ή καταθέσεων για την περίοδο από Πέμπτη μέχρι Σάββατο. Με τον ίδιο τρόπο τη πρόβλεψη της Παρασκευής δε θα τη συγκρίνουμε με το ποσό των συναλλαγών εκείνης της μέρας αλλά με το σύνολο ποσό αναλήψεων ή καταθέσεων που ολοκληρώθηκαν τη Παρασκευή και το Σάββατο.

4.6.1 Αυτόματη Ταμιακή Μηχανή στη περιοχή της Αθήνας

Το πρώτο μηχάνημα ATM δραστηριοποιείται στη περιοχή της Αθήνας (το οποίο θα αναφέρεται με τον κωδικό 0001) και χρησιμοποιείται για αναλήψεις και καταθέσεις. Στο σχήμα 1.1 παρουσιάζεται τα μηνιαία ποσά αναλήψεων χωρισμένα ανά μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή) για ένα έτος 2012. Όπως είναι φανερό οι συναλλαγές δεν παρουσιάζουν διακυμάνσεις από μήνα σε μήνα, άρα μιλάμε για στατική ζήτηση. Οι συναλλαγές την ημέρα Δευτέρα είναι ιδιαίτερα υψηλές καθώς περιέχουν και τις αναλήψεις που πραγματοποιήθηκαν το σαββατοκύριακο που προηγήθηκε. Από Τρίτη έως Πέμπτη οι αναλήψεις παρουσιάζουν στατικότητα ενώ αυξάνονται τη Παρασκευή.

Πίνακας 1: Δεδομένα αναλήψεων 12 μηνών για το 0001

Εργάσιμες Μήνες	Ημέρες 1	2	3	4	5	Γενικό άθροισμα
1	71.210	34.760	31.910	24.410	21.320	183.610
2	47.160	32.550	40.390	37.170	35.390	192.660
3	65.960	26.610	31.650	37.120	43.070	204.410
4	42.170	46.190	31.130	33.980	15.810	169.280
5	54.110	25.510	49.700	38.180	30.940	198.440
6	43.300	32.030	17.780	25.110	60.950	179.170
7	81.670	33.990	22.330	20.780	37.450	196.220
8	48.000	33.350	24.910	22.890	29.640	158.790
9	49.830	25.860	16.150	28.010	40.170	160.020
10	59.900	33.590	29.660	17.400	32.130	172.680
11	48.010	19.930	26.020	25.650	43.060	162.670
12	122.980	24.790	29.800	49.890	31.460	258.920



Στη συνέχεια θα υπολογίσουμε την πρόβλεψη των αναλήψεων για τις εργάσιμες ημέρες του Μαρτίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων. Οι υπολογισμοί έχουν γίνει με βάση τις προδιαγραφές που αναφέρθηκαν παραπάνω (Παρουσία του τρόπου ανάλυσης των δεδομένων).

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 2: Πρόβλεψη των αναλήψεων για το μήνα Μάρτιο με τις τρεις διαφορετικές μεθόδους

Ημερ/γία	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
01/03/2012	28.213	26.498	16.794	24.953	8.602
02/03/2012	18.645	17.629	11.230	16.687	8.565
05/03/2012	8.836	8.672	5.785	8.684	9.180
06/03/2012	9.195	8.910	5.912	8.525	8.598
07/03/2012	9.068	8.875	5.859	8.209	7.987
08/03/2012	27.259	26.570	18.088	25.190	8.535
09/03/2012	17.463	17.206	11.990	15.985	7.686
12/03/2012	8.262	8.544	6.176	8.252	8.420
13/03/2012	8.747	9.671	6.078	8.108	8.111
14/03/2012	9.049	8.620	6.049	8.311	8.608
15/03/2012	27.118	25.892	17.939	24.203	8.009
16/03/2012	18.650	17.565	12.066	16.804	8.796
19/03/2012	9.233	8.895	6.072	8.455	8.732
20/03/2012	9.814	10.135	6.206	8.832	9.391
21/03/2012	9.660	9.011	5.606	7.016	5.502
22/03/2012	28.519	25.876	17.050	22.461	7.421
23/03/2012	18.640	16.344	10.804	13.425	5.974
26/03/2012	9.199	7.934	5.409	7.059	7.184
27/03/2012	9.575	8.918	5.371	7.022	7.094
28/03/2012	9.490	8.067	5.300	7.032	7.146
29/03/2012	27.879	24.187	15.572	20.466	6.722
30/03/2012	17.425	15.208	10.264	13.754	6.967
Γενικό άθροισμα	339.939	319.223	211.620	289.432	173.229

Καθώς είναι δύσκολο να εξάγουμε συμπεράσματα σχετικά με ποια μέθοδος κατάφερε να πλησιάσει πιο κοντά στη πραγματική ζήτηση για μετρητά συνεχίζουμε στον υπολογισμό των σφαλμάτων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των αναλήψεων ανάλογα με την ημέρα τροφοδοσίας.

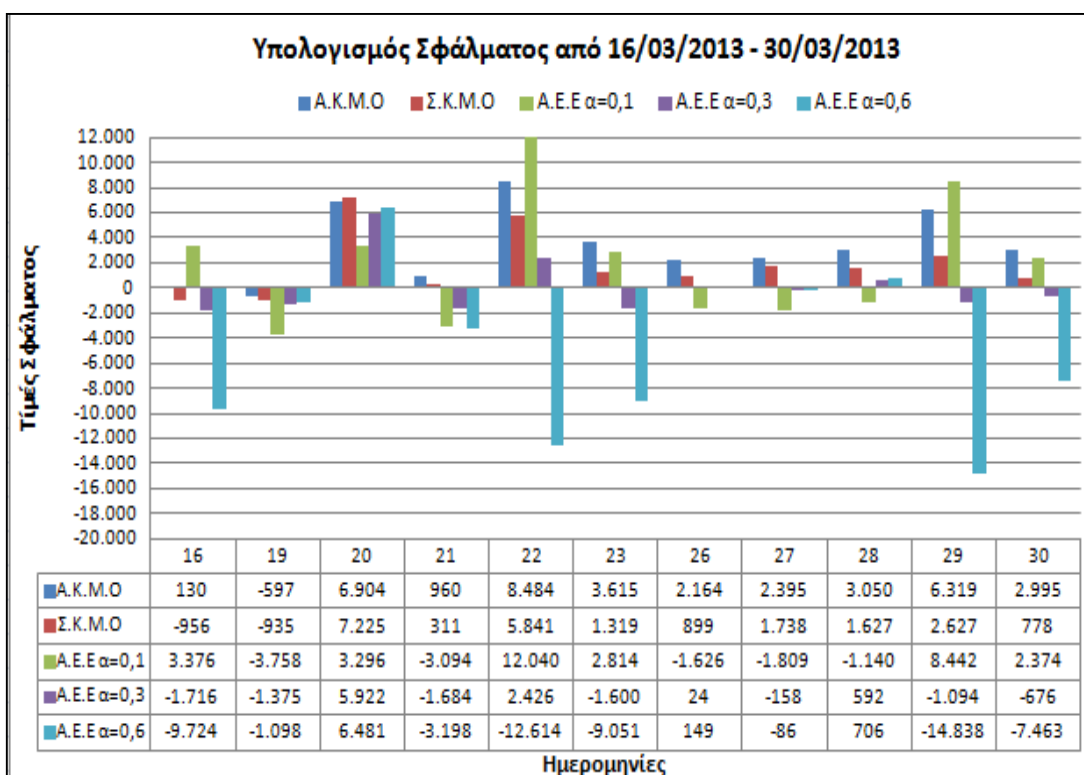
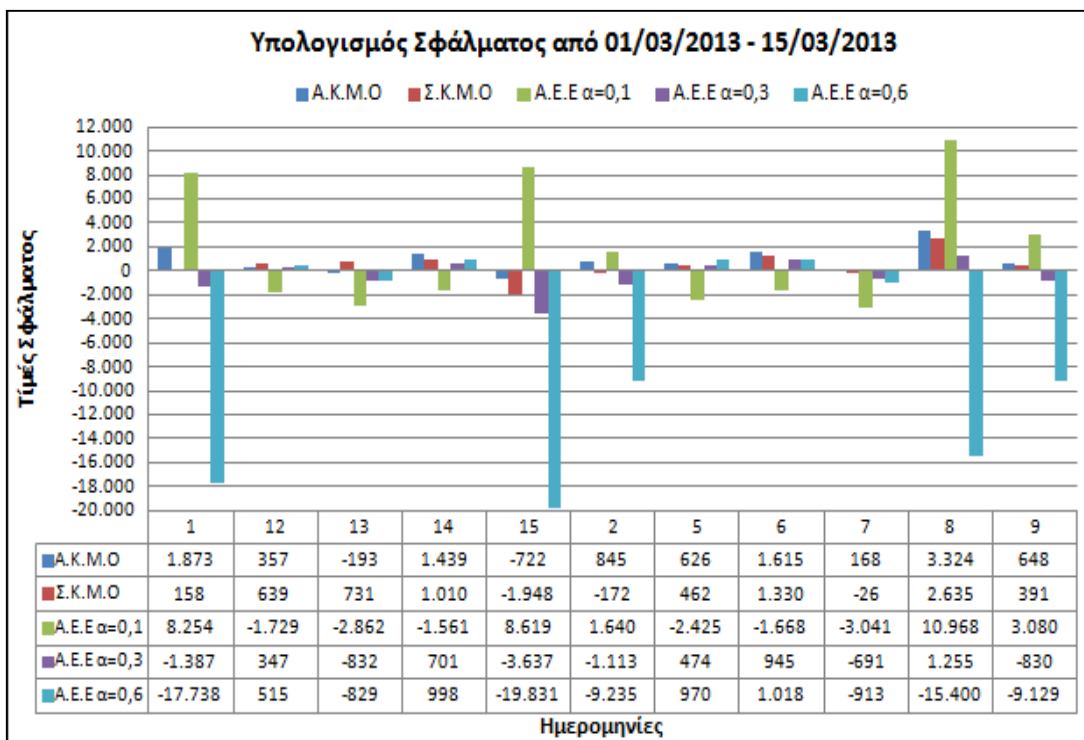
- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές αναλήψεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές αναλήψεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές αναλήψεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές αναλήψεις της Τετάρτης
- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευής σύγκριση με τις πραγματικές αναλήψεις της Πέμπτης, Παρασκευής & Σάββατο.

Ημερ/νία	Α.Κ.Μ.Ο	Σ.Κ.Μ.Ο	Α.Ε.Ε α=0,1	Α.Ε.Ε α=0,3	Α.Ε.Ε α=0,6
01/03/2013	1.873	158	8.254	-1.387	-17.738
02/03/2013	845	-172	1.640	-1.113	-9.235
05/03/2013	626	462	-2.425	474	970
06/03/2013	1.615	1.330	-1.668	945	1.018
07/03/2013	168	-26	-3.041	-691	-913
08/03/2013	3.324	2.635	10.968	1.255	-15.400
09/03/2013	648	391	3.080	-830	-9.129
12/03/2013	357	639	-1.729	347	515
13/03/2013	-193	731	-2.862	-832	-829
14/03/2013	1.439	1.010	-1.561	701	998
15/03/2013	-722	-1.948	8.619	-3.637	-19.831
16/03/2013	130	-956	3.376	-1.716	-9.724
19/03/2013	-597	-935	-3.758	-1.375	-1.098
20/03/2013	6.904	7.225	3.296	5.922	6.481
21/03/2013	960	311	-3.094	-1.684	-3.198
22/03/2013	8.484	5.841	12.040	2.426	-12.614
23/03/2013	3.615	1.319	2.814	-1.600	-9.051
26/03/2013	2.164	899	-1.626	24	149
27/03/2013	2.395	1.738	-1.809	-158	-86
28/03/2013	3.050	1.627	-1.140	592	706
29/03/2013	6.319	2.627	8.442	-1.094	-14.838
30/03/2013	2.995	778	2.374	-676	-7.463

Πίνακας 3: Υπολογισμός Σφαλμάτων

Αναλύοντας τα παραπάνω στοιχεία απορρίπτουμε τη μέθοδο της εκθετικής εξομάλυνσης καθώς οι προβλέψεις παρουσιάζουν μεγάλες αποκλίσεις. Στην περίπτωση όπου το $a=0.6$ εντοπίσουμε προβλέψεις όπου σε σύγκριση με την πραγματικότητα απέχουν πάνω από 10.000 μονάδες. Στην περίπτωση όπου το $a=0.3$ οι αποκλίσεις μειώνονται αλλά κυριαρχούν οι αρνητικές τιμές, δηλαδή τα ποσά των προβλέψεων είναι μικρότερα από τις πραγματικές αναλήψεις. Σε αυτή τη περίπτωση ο κίνδυνος το ATM να μείνει χωρίς χρήματα είναι άμεσος και για αυτό το λόγο απορρίπτουμε και αυτή τη μέθοδο. Στην περίπτωση όπου το $a=0.1$ εντοπίζουμε απότομες αντιθέσεις καθώς τη μία μέρα η πρόβλεψη απέχει 10.968 μονάδες και την επόμενη μόλις 3.080, γεγονός που παρατηρήθηκε δύο φορές στη διάρκεια ενός μήνα, επομένως και αυτή μέθοδος δεν μπορεί να χαρακτηριστεί ως αξιόπιστη.

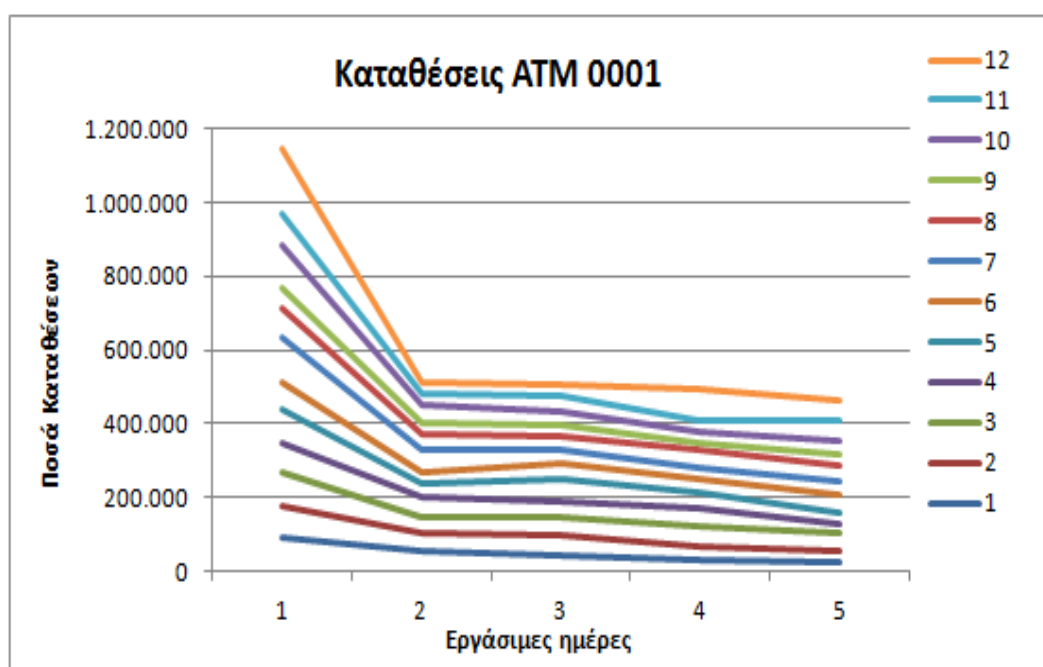
Ανάμεσα στον απλός κινούμενος μέσος όρος και στο σταθμισμένο κινούμενος μέσος όρος παρατηρούμε ότι τις μικρότερες αποκλίσεις εμφανίζει η μέθοδο του σταθμισμένου κινούμενου μέσου. Ακόμα οι τιμές των σφαλμάτων είναι θετικές επομένως οι τιμές των προβλέψεων είναι κατά κύριο λόγο υπερτιμημένες, δηλαδή η πρόβλεψη είναι μεγαλύτερη από την πραγματικότητα, περιορίζοντας έτσι την πιθανότητα το ATM να τεθεί εκτός λειτουργίας λόγω έλλειψης μετρητών. Σε περίπτωση λάθος πρόβλεψης λόγω τυχαίου γεγονότος (π.χ. βλάβης), η διαφορά δεν μεταφέρεται στις επόμενες προβλέψεις, γεγονός που εμφανίζεται στις 22/03, σφάλμα πρόβλεψης 5.811 μονάδων και την επόμενη ημέρα μόλις 1.319 μονάδες. Τέλος θα πρέπει να τονίσουμε ότι η απόκλιση για 5.000 (5%) δεν επηρεάζει δραματικά την απόφαση μας για τροφοδοσία. Τα παρακάτω διαγράμματα απεικονίζουν τα παραπάνω συμπεράσματα.



Με τον ίδιο τρόπο θα αναλύσουμε και τις καταθέσεις για το ATM 0001. Στον πίνακα 4 εμφανίζονται οι καταθέσεις για ένα έτος 2012. Στο σχήμα 4 παρουσιάζεται τα μηνιαία ποσά καταθέσεων χωρισμένα ανά μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή). Όπως είναι φανερό οι συναλλαγές δεν παρουσιάζουν διακυμάνσεις από μήνα σε μήνα, άρα μιλάμε για στατική ζήτηση. Οι συναλλαγές την ημέρα Δευτέρα είναι ιδιαίτερα υψηλές καθώς περιέχουν και τις αναλήψεις που πραγματοποιήθηκαν το σαββατοκύριακο που προηγήθηκε. Από Τρίτη έως Πέμπτη οι αναλήψεις παρουσιάζουν στατικότητα ενώ αυξάνονται τη Παρασκευή.

Πίνακας 4: Δεδομένα καταθέσεων 12 μηνών για το 0001

Εργάσιμες Μήνες	Ημέρες					
	1	2	3	4	5	Γενικό άθροισμα
1	90.950	54.610	42.185	29.330	25.895	242.970
2	87.510	48.310	55.890	35.850	31.855	259.415
3	88.100	46.320	45.520	56.855	46.135	282.930
4	78.525	49.880	46.755	48.530	25.195	248.885
5	95.810	39.520	58.995	44.740	31.245	270.310
6	69.355	32.440	43.115	33.985	48.800	227.695
7	122.310	58.835	38.805	32.295	35.945	288.190
8	78.330	40.790	32.975	44.820	43.655	240.570
9	59.390	29.030	30.720	21.065	28.655	168.860
10	114.740	49.390	39.305	28.875	35.100	267.410
11	83.865	31.130	40.845	31.070	53.165	240.075
12	178.225	29.660	33.520	88.915	56.220	386.540



Σχήμα 4: Ετήσια ποσά καταθέσεων για ένα έτος

Στη συνέχεια θα υπολογίσουμε την πρόβλεψη των καταθέσεων για τις εργάσιμες ημέρες του Μαρτίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων. Οι υπολογισμοί έχουν γίνει με βάση τις προδιαγραφές που αναφέρθηκαν παραπάνω (Παρουσία του τρόπου ανάλυσης των δεδομένων).

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 5: Πρόβλεψη των καταθέσεων για το μήνα Μάρτιο με τις τρεις διαφορετικές μεθόδους

Ημερ/νία	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
01/03/2012	36.930	30.314	21.275	28.887	9.647
02/03/2012	24.329	18.726	13.918	18.820	9.295
05/03/2012	10.650	8.706	6.491	8.308	7.267
06/03/2012	11.478	9.773	6.648	9.007	9.370
07/03/2012	11.029	10.380	6.887	9.591	10.378
08/03/2012	33.443	30.147	21.450	30.463	11.093
09/03/2012	22.690	21.146	14.206	20.652	10.998
12/03/2012	10.404	10.014	6.890	9.662	9.364
13/03/2012	11.048	12.162	7.175	10.213	10.698
14/03/2012	11.242	10.907	7.458	10.916	11.893
15/03/2012	34.615	33.892	23.245	33.566	11.897
16/03/2012	23.249	21.589	14.801	20.717	9.940
19/03/2012	10.798	10.582	7.429	10.204	9.949
20/03/2012	11.517	13.022	7.633	10.713	11.196
21/03/2012	11.850	11.690	7.888	11.033	11.594
22/03/2012	35.788	33.193	23.626	32.755	11.118
23/03/2012	25.029	23.652	16.606	24.359	13.612
26/03/2012	12.127	11.582	8.072	11.256	11.001
27/03/2012	12.832	13.245	7.922	10.732	10.198
28/03/2012	12.936	12.446	7.968	10.694	10.511
29/03/2012	39.365	36.317	23.908	32.438	10.954
30/03/2012	25.169	24.332	16.218	22.803	12.152
Γενικό άθροισμα	438.516	407.814	277.714	387.788	234.124

Επεξεργαζόμαστε τα παραπάνω στοιχεία και προσπαθούμε να εξάγουμε συμπεράσματα σχετικά με ποια την μεθόδου το αποτέλεσμα πλησιάζει περισσότερο στις πραγματικές καταθέσεις. Για το σκοπό αυτό προχωράμε στον υπολογισμό των σφαλμάτων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των καταθέσεων ανάλογα με την ημέρα τροφοδοσίας.

- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές καταθέσεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές καταθέσεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές καταθέσεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές καταθέσεις της Τετάρτης

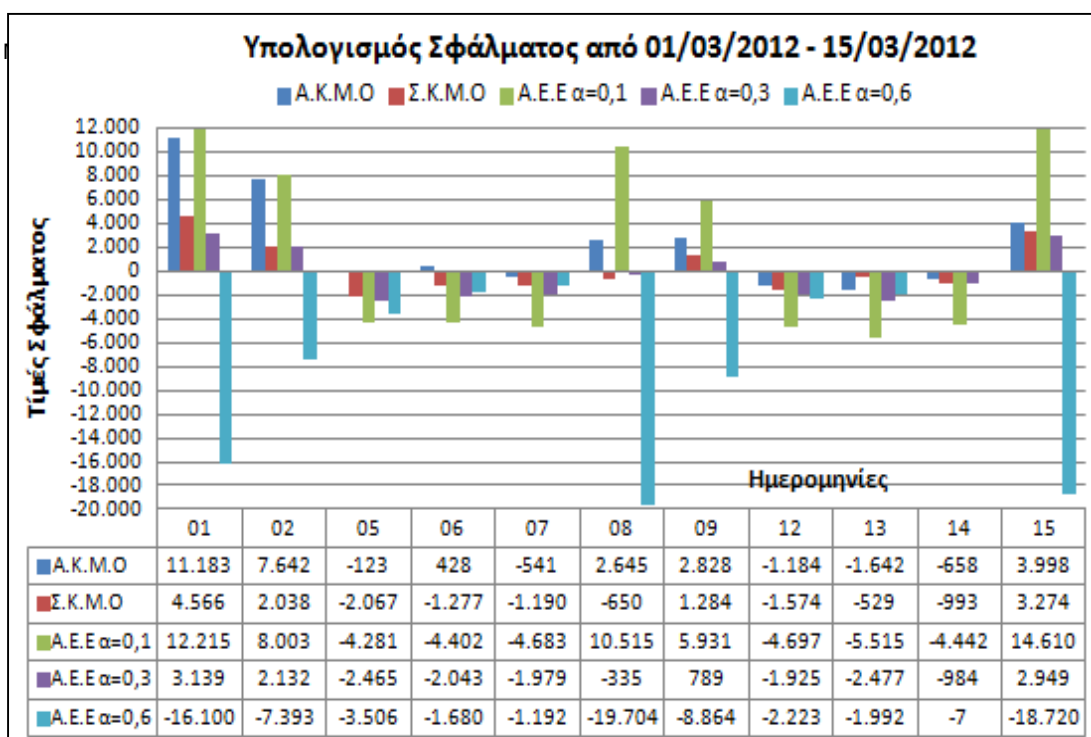
Πίνακας 6: Υπολογισμός Σφαλμάτων

Ημερ/νία	A.K.M.O	Σ.K.M.O	A.E.E $\alpha=0,1$	A.E.E $\alpha=0,3$	A.E.E $\alpha=0,6$
01/03/2012	11.183	4.566	12.215	3.139	-16.100
02/03/2012	7.642	2.038	8.003	2.132	-7.393
05/03/2012	-123	-2.067	-4.281	-2.465	-3.506
06/03/2012	428	-1.277	-4.402	-2.043	-1.680
07/03/2012	-541	-1.190	-4.683	-1.979	-1.192
08/03/2012	2.645	-650	10.515	-335	-19.704
09/03/2012	2.828	1.284	5.931	789	-8.864
12/03/2012	-1.184	-1.574	-4.697	-1.925	-2.223
13/03/2012	-1.642	-529	-5.515	-2.477	-1.992
14/03/2012	-658	-993	-4.442	-984	-7
15/03/2012	3.998	3.274	14.610	2.949	-18.720
16/03/2012	1.267	-394	4.846	-1.266	-12.043
19/03/2012	-1.229	-1.446	-4.599	-1.823	-2.079
20/03/2012	-343	1.162	-4.227	-1.147	-664
21/03/2012	1.050	890	-2.912	233	794
22/03/2012	1.591	-1.004	8.351	-1.443	-23.080
23/03/2012	6.106	4.729	7.346	5.436	-5.311
26/03/2012	2.464	1.919	-1.590	1.593	1.338
27/03/2012	2.112	2.525	-2.798	12	-522
28/03/2012	1.686	1.196	-3.282	-556	-739
29/03/2012	2.682	-366	10.958	-4.244	-25.728
30/03/2012	1.436	600	3.488	-930	-11.581
Γενικό άθροισμα	43.396	12.694	38.834	-7.332	-160.996

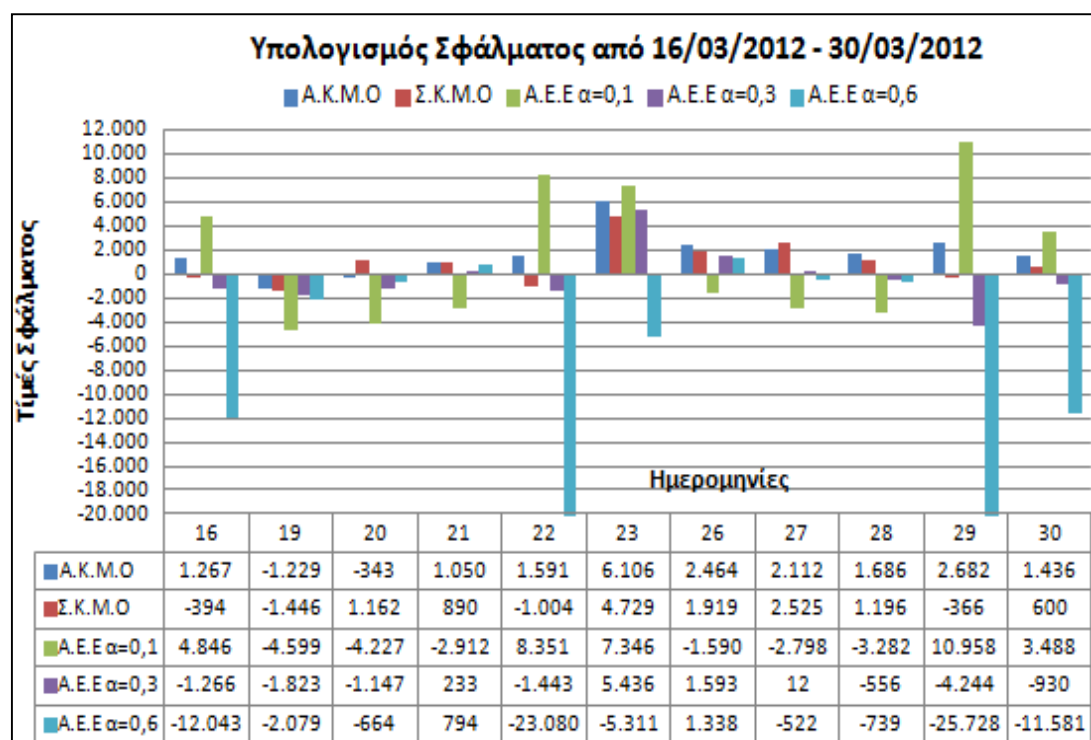
- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευή σύγκριση με τις πραγματικές καταθέσεις της Πέμπτη, Παρασκευή & Σάββατο.

Παρατηρώντας τις τιμές των σφαλμάτων καταλήγουμε στο συμπέρασμα ότι η μέθοδος της εκθετικής εξομάλυνσης δεν είναι κατάλληλη για την πρόβλεψη των καταθέσεων. Στην περίπτωση όπου το $\alpha=0.6$ εμφανίζονται προβλέψεις που απέχουν από την πραγματικότητα κατά μέσω όρο 7.500 μονάδες και πολύ συχνά φτάνουν τις 20.000 μονάδες. Στην περίπτωση όπου το $\alpha=0.3$ οι αποκλίσεις είναι αισθητά μικρότερες αλλά κυριαρχούν οι αρνητικές τιμές, δηλαδή τα ποσά των προβλέψεων είναι μικρότερα από τις πραγματικές καταθέσεις. Σε αυτή τη περίπτωση ο κίνδυνος το ATM να ξεπεράσει το ποσό των καταθέσεων είναι άμεσος και για αυτό το λόγο απορρίπτουμε και αυτή τη μέθοδο. Στην περίπτωση όπου το $\alpha=0.1$ εντοπίζουμε απότομες αντιθέσεις καθώς τη μία μέρα η πρόβλεψη απέχει 10.515 μονάδες και την επόμενη μόλις 5.931, γεγονός που παρατηρήθηκε τρεις φορές στη διάρκεια ενός μήνα, επομένως και αυτή μέθοδος δεν μπορεί να χαρακτηριστεί ως αξιόπιστη.

Συγκρίνοντας τις δύο τελευταίες μεθόδους απλό κινούμενο μέσο όρο και στο σταθμισμένο κινούμενος μέσος όρος παρατηρούμε ότι τις μικρότερες αποκλίσεις εμφανίζει η μέθοδο του σταθμισμένου κινούμενου μέσου. Ακόμα οι τιμές των σφαλμάτων είναι θετικές επομένως οι τιμές των προβλέψεων είναι κατά κύριο λόγο υπερτιμημένες, δηλαδή η πρόβλεψη είναι μεγαλύτερη από την πραγματικότητα, περιορίζοντας έτσι την πιθανότητα το ATM να τεθεί εκτός λειτουργίας λόγω αυξημένων μετρητών. Σε περίπτωση λάθος πρόβλεψης λόγω τυχαίου γεγονότος (π.χ. βλάβης), η διαφορά δεν μεταφέρεται στις επόμενες προβλέψεις, γεγονός που εμφανίζεται στις 15/03, σφάλμα πρόβλεψης 3.274 μονάδων και την επόμενη ημέρα μόλις -394 μονάδες. Τέλος με τις τρεις διαφορετικές μεθόδους αναλύσαμε τα δεδομένα των αναλήψεων και τον καταθέσεων και οδηγηθήκαμε στο ίδιο συμπέρασμα, ο πιο αξιόπιστος τρόπος πρόβλεψης είναι ο σταθμισμένο κινούμενος μέσος όρος Τα παρακάτω διαγράμματα απεικονίζουν τα παραπάνω συμπεράσματα.



οφία

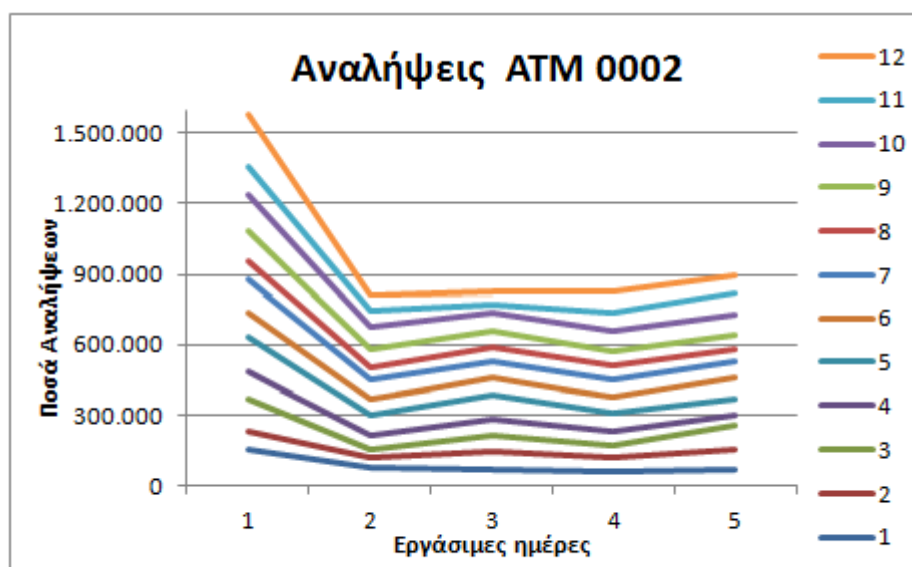


4.6.2 Αυτόματη Ταμιακή Μηχανή στη περιοχή της Θεσσαλονίκης

Το δεύτερο μηχάνημα ATM δραστηριοποιείται στη περιοχή της Θεσσαλονίκης για την πραγματοποίηση αναλήψεων και καταθέσεων (το οποίο θα αναφέρεται με τον κωδικό 0002). Για την ανάλυση των δεδομένων θα εργαστούμε με τον ίδιο τρόπο, δηλαδή θα χωρίσουμε τις συναλλαγές ανά μέρα και μόνο στις πέντε εργάσιμες μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή) για ένα έτος 2012. Σε σύγκριση με το 0001 μηχάνημα η ζήτηση είναι μεγαλύτερη, σχεδόν διπλάσια αλλά τα δεδομένα παρουσιάζουν σταθερή τάση ανά μήνα χωρίς διακυμάνσεις. Τις Δευτέρες οι συναλλαγές εμφανίζονται ιδιαίτερα αυξημένες καθώς περιέχουν και τις συναλλαγές του σαββατοκύριακου, τις Τρίτες οι τιμές μειώνονται και κυμαίνονται σχεδόν σταθερές μέχρι τη Παρασκευή όπου και εμφανίζεται μία μικρή αύξηση.

Πίνακας 7: Δεδομένα αναλήψεων 12 μηνών για το 0002

Εργάσιμες Μήνες	Ημέρες 1	2	3	4	5	Γενικό άθροισμα
1	150.200	79.900	67.000	59.200	67.500	423.800
2	78.000	42.800	80.800	58.200	90.400	350.200
3	139.200	34.700	68.800	50.900	100.700	394.300
4	119.100	60.200	66.500	59.400	41.600	346.800
5	147.200	85.000	102.700	79.900	69.400	484.200
6	97.500	64.600	74.400	66.900	94.900	398.300
7	145.000	89.000	68.900	76.300	65.200	444.400
8	84.200	51.700	56.600	63.600	49.800	305.900
9	120.200	71.400	73.400	61.500	61.700	388.200
10	157.000	93.000	72.800	82.100	82.000	486.900
11	120.100	72.900	34.700	78.300	96.700	402.700
12	222.400	64.000	61.700	88.300	73.700	510.100



Σχήμα 7: Ετήσια ποσά αναλήψεων για ένα έτος

Στη συνέχεια θα υπολογίσουμε την πρόβλεψη των αναλήψεων για τις εργάσιμες ημέρες του Οκτωβρίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων.

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 8: Πρόβλεψη των αναλήψεων για το μήνα Οκτώβριο με τις τρεις διαφορετικές μεθόδους

Ημερ/νία	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
01/10/2012	18.569	16.790	11.833	16.157	16.020
02/10/2012	19.125	18.833	11.613	15.431	14.778
03/10/2012	19.206	17.568	11.631	15.597	15.631
04/10/2012	57.825	51.458	35.014	47.321	16.092
05/10/2012	37.375	34.430	23.985	33.819	18.317
08/10/2012	18.650	17.495	12.115	17.195	18.187
09/10/2012	19.288	19.400	11.868	16.372	16.094
11/10/2012	58.388	54.990	36.220	50.658	17.418
12/10/2012	39.225	36.815	24.386	34.301	17.767
15/10/2012	19.069	18.773	12.722	18.596	20.427
16/10/2012	19.913	20.560	12.489	17.578	17.440
17/10/2012	20.025	20.810	12.701	17.969	18.436
18/10/2012	60.094	58.035	38.363	53.738	18.174
19/10/2012	40.950	39.915	26.790	38.642	20.950
22/10/2012	19.813	20.148	13.628	19.547	20.560
23/10/2012	20.700	21.873	13.462	18.595	18.184
24/10/2012	20.819	20.458	13.537	18.874	19.153
25/10/2012	63.038	61.635	40.715	56.511	19.061
26/10/2012	42.688	41.410	27.899	39.150	20.524
29/10/2012	20.938	20.445	14.178	20.046	21.050
30/10/2012	21.988	22.858	14.031	19.298	19.100
31/10/2012	22.088	21.018	13.961	19.296	19.400
Γενικό άθροισμα	679.769	655.713	433.139	604.692	402.761

Για την εξαγωγή αποτελεσμάτων προχωράμε στον υπολογισμό των σφαλμάτων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των αναλήψεων ανάλογα με την ημέρα τροφοδοσίας.

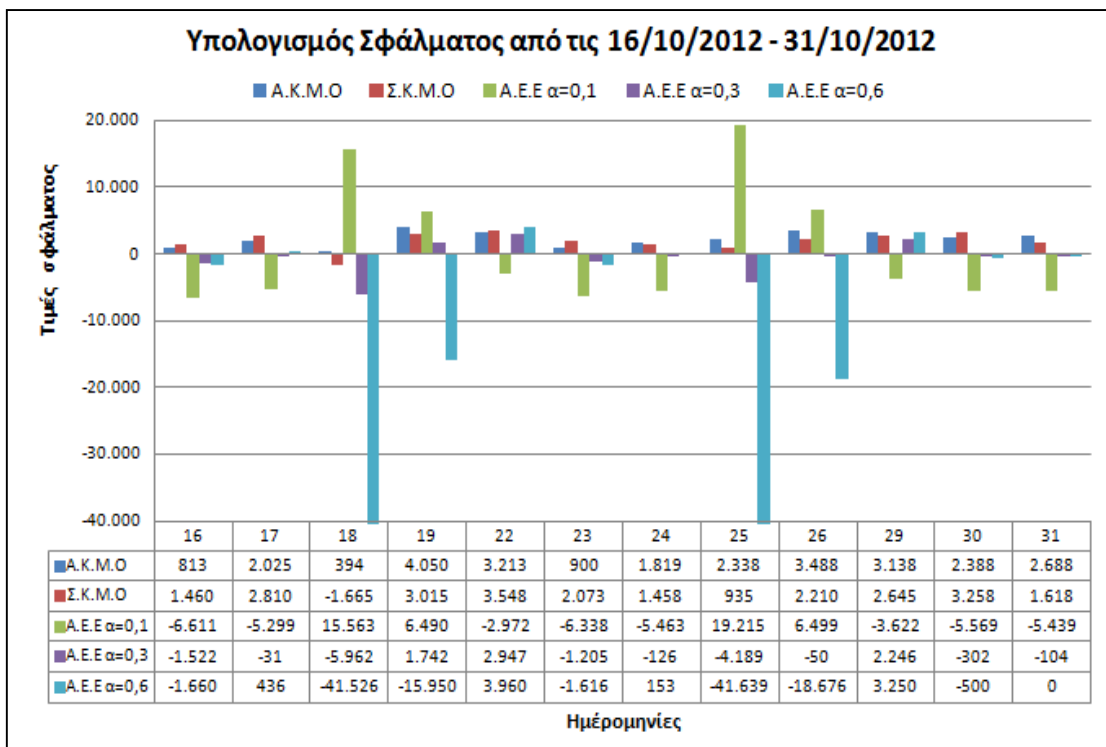
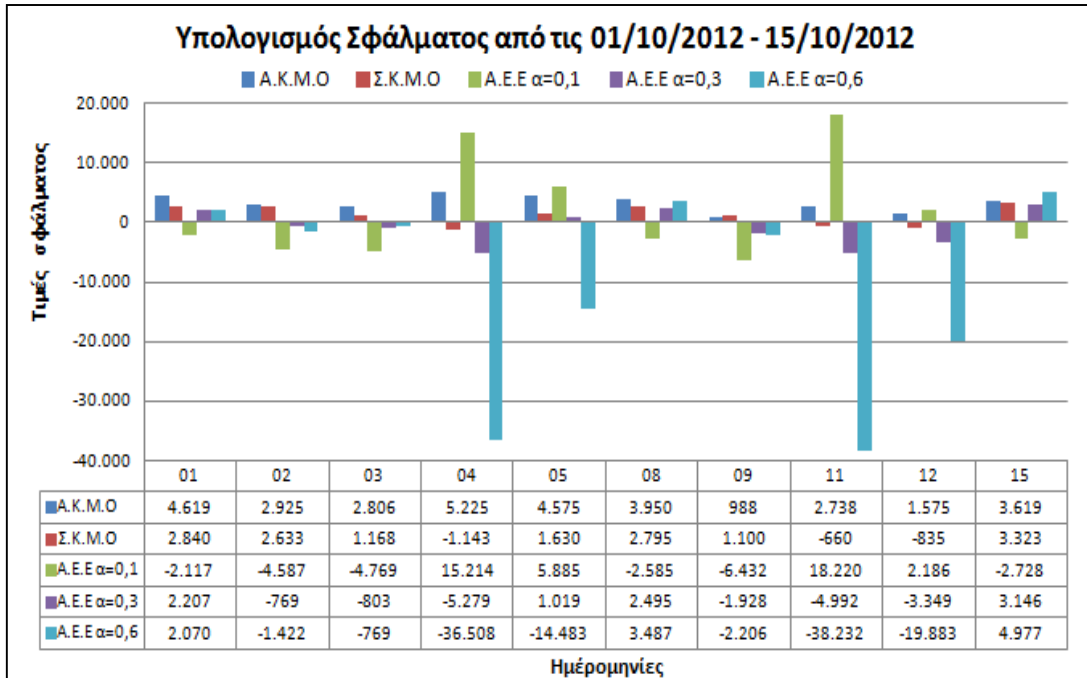
- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές αναλήψεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές αναλήψεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές αναλήψεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές αναλήψεις της Τετάρτης
- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευής σύγκριση με τις πραγματικές αναλήψεις της Πέμπτης, Παρασκευής & Σάββατο.

Πίνακας 9: Υπολογισμός σφαλμάτων για το μήνα Οκτώβριο με τις τρεις διαφορετικές μεθόδους

Ημερ/νία	Α.Κ.Μ.Ο	Σ.Κ.Μ.Ο	Α.Ε.Ε α=0,1	Α.Ε.Ε α=0,3	Α.Ε.Ε α=0,6
01/10/2012	4.619	2.840	-2.117	2.207	2.070
02/10/2012	2.925	2.633	-4.587	-769	-1.422
03/10/2012	2.806	1.168	-4.769	-803	-769
04/10/2012	5.225	-1.143	15.214	-5.279	-36.508
05/10/2012	4.575	1.630	5.885	1.019	-14.483
08/10/2012	3.950	2.795	-2.585	2.495	3.487
09/10/2012	988	1.100	-6.432	-1.928	-2.206
11/10/2012	2.738	-660	18.220	-4.992	-38.232
12/10/2012	1.575	-835	2.186	-3.349	-19.883
15/10/2012	3.619	3.323	-2.728	3.146	4.977
16/10/2012	813	1.460	-6.611	-1.522	-1.660
17/10/2012	2.025	2.810	-5.299	-31	436
18/10/2012	394	-1.665	15.563	-5.962	-41.526
19/10/2012	4.050	3.015	6.490	1.742	-15.950
22/10/2012	3.213	3.548	-2.972	2.947	3.960
23/10/2012	900	2.073	-6.338	-1.205	-1.616
24/10/2012	1.819	1.458	-5.463	-126	153
25/10/2012	2.338	935	19.215	-4.189	-41.639
26/10/2012	3.488	2.210	6.499	-50	-18.676
29/10/2012	3.138	2.645	-3.622	2.246	3.250
30/10/2012	2.388	3.258	-5.569	-302	-500
31/10/2012	2.688	1.618	-5.439	-104	0

Παρατηρώντας τα δεδομένα οι τιμές των σφαλμάτων είναι υψηλότερες από τις αντίστοιχες τιμές για το ATM 0001 που εξετάσαμε στην αρχή. Παρόλα αυτά απορρίπτουμε και αυτή τη φορά τη μέθοδο της εκθετικής εξομάλυνσης καθώς οι προβλέψεις παρουσιάζουν μεγάλες αποκλίσεις. Στην περίπτωση όπου το $\alpha=0.6$ εντοπίσουμε προβλέψεις όπου σε σύγκριση με την πραγματικότητα απέχουν πάνω από 40.000 μονάδες, σφάλμα με ιδιαίτερο ρίσκο. Στην περίπτωση όπου το $\alpha=0.3$ οι αποκλίσεις μειώνονται φτάνοντας τις 5.000 μονάδες αλλά κυριαρχούν οι αρνητικές τιμές, δηλαδή τα ποσά των προβλέψεων είναι μικρότερα από τις πραγματικές αναλήψεις. Σε αυτή τη περίπτωση ο κίνδυνος το ATM να μείνει χωρίς χρήματα είναι άμεσος και για αυτό το λόγο απορρίπτουμε και αυτή τη μέθοδο. Στην περίπτωση όπου το $\alpha=0.1$ εντοπίζουμε απότομες αντιθέσεις καθώς τη μία μέρα η πρόβλεψη απέχει 18.220 μονάδες και την επόμενη μόλις 2.186, γεγονός που παρατηρήθηκε δύο φορές στη διάρκεια ενός μήνα, επομένως και αυτή μέθοδος δεν μπορεί να χαρακτηριστεί ως αξιόπιστη.

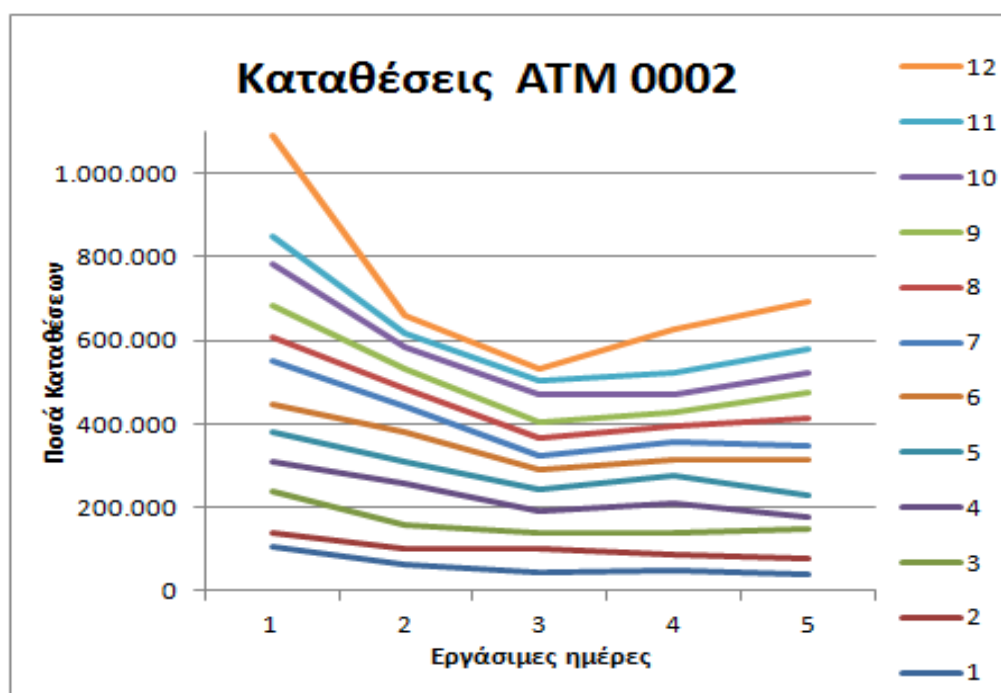
Στις υπόλοιπες δύο μεθόδους, στον απλός κινούμενος μέσος όρος και στο σταθμισμένο κινούμενος μέσος όρος παρατηρούμε ότι τις μικρότερες αποκλίσεις εμφανίζει η μέθοδο του σταθμισμένου κινούμενου μέσου. Συνεχίζουμε με τη γραφική αναπαράσταση των τιμών των σφαλμάτων χωρίζοντας το μήνα σε δύο διαστήματα



Με τον ίδιο τρόπο θα αναλύσουμε και τις καταθέσεις για το ATM 0002. Στον πίνακα 10 εμφανίζονται οι καταθέσεις για ένα έτος 2012. Στο σχήμα 10 παρουσιάζεται τα μηνιαία ποσά καταθέσεων χωρισμένα ανά μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή). Η ζήτηση μπορεί να χαρακτηριστεί ως στατική καθώς οι συναλλαγές δεν παρουσιάζουν διακυμάνσεις από μήνα σε μήνα εκτός από το Δεκέμβριο όπου οι καταθέσεις είναι γενικά αυξημένες. Οι συναλλαγές την ημέρα Δευτέρα είναι ιδιαίτερα υψηλές καθώς περιέχουν και τις συναλλαγές που πραγματοποιήθηκαν το σαββατοκύριακο που προηγήθηκε. Από Τρίτη τα ποσά των καταθέσεων μειώνονται μέχρι τη Πέμπτη όπου αρχίζουν πάλι να αυξάνονται.

Πίνακας 10: Δεδομένα καταθέσεων 12 μηνών για το ATM 0002

Μήνες	1	2	3	4	5	Γενικό άθροισμα
1	105.200	63.700	44.700	48.400	41.500	303.500
2	34.000	37.300	54.600	39.600	35.300	200.800
3	98.700	59.400	40.200	51.200	73.700	323.200
4	70.900	96.700	50.800	73.200	25.400	317.000
5	74.200	50.500	55.300	65.400	53.500	298.900
6	64.700	71.900	45.600	34.600	87.500	304.300
7	103.500	63.900	33.800	44.500	31.300	277.000
8	57.200	42.100	43.100	37.100	66.700	246.200
9	75.400	48.700	34.400	35.900	59.200	253.600
10	129.180	66.360	54.340	45.400	55.400	350.680
11	70.000	32.300	32.100	52.200	60.200	246.800
12	237.400	42.600	30.500	102.100	110.900	523.500
Γενικό άθροισμα	1.120.380	675.460	519.440	629.600	700.600	3.645.480



Σχήμα 10: Ετήσια ποσά καταθέσεων για ένα έτος

Υπολογίσουμε την πρόβλεψη των καταθέσεων για τις εργάσιμες ημέρες του Οκτωβρίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων. Οι υπολογισμοί έχουν γίνει με βάση τις προδιαγραφές που αναφέρθηκαν παραπάνω (Παρουσία του τρόπου ανάλυσης των δεδομένων).

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 11: Προβλέψεις των καταθέσεων για το μήνα Οκτώβριο με τις τρεις διαφορετικές μεθόδους

Ημερ/νία	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
01/10/2012	12.444	12.150	7.374	10.154	11.146
02/10/2012	12.806	12.558	7.738	10.976	12.252
03/10/2012	12.716	11.846	7.872	11.397	12.431
04/10/2012	38.738	35.663	24.040	34.828	12.322
05/10/2012	23.950	22.820	15.633	21.509	10.269
08/10/2012	11.963	12.085	8.143	11.457	12.027
09/10/2012	13.194	14.463	8.399	11.715	12.251
11/10/2012	40.890	38.757	24.796	36.878	13.288
12/10/2012	27.248	25.456	16.810	24.455	12.635
15/10/2012	13.161	12.782	8.619	12.594	13.214
16/10/2012	13.974	14.587	8.825	12.268	12.216
17/10/2012	14.361	14.924	9.061	12.661	13.106
18/10/2012	44.443	41.873	27.313	38.347	13.192
19/10/2012	28.004	26.211	17.954	24.687	12.177
22/10/2012	13.808	13.424	9.215	13.058	13.811
23/10/2012	14.757	15.190	9.352	13.305	13.954
24/10/2012	14.907	13.705	9.296	12.667	12.362
25/10/2012	45.643	41.576	28.126	38.873	13.236
26/10/2012	30.366	27.150	18.735	25.718	12.974
29/10/2012	14.416	13.601	9.406	13.053	13.410
30/10/2012	15.331	15.879	9.481	13.167	13.524
31/10/2012	15.493	14.223	9.632	13.609	14.307
Γενικό άθροισμα	472.611	450.921	295.820	417.378	280.104

Για την εξαγωγή αποτελεσμάτων προχωράμε στον υπολογισμό των σφαλμάτων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των καταθέσεων ανάλογα με την ημέρα τροφοδοσίας.

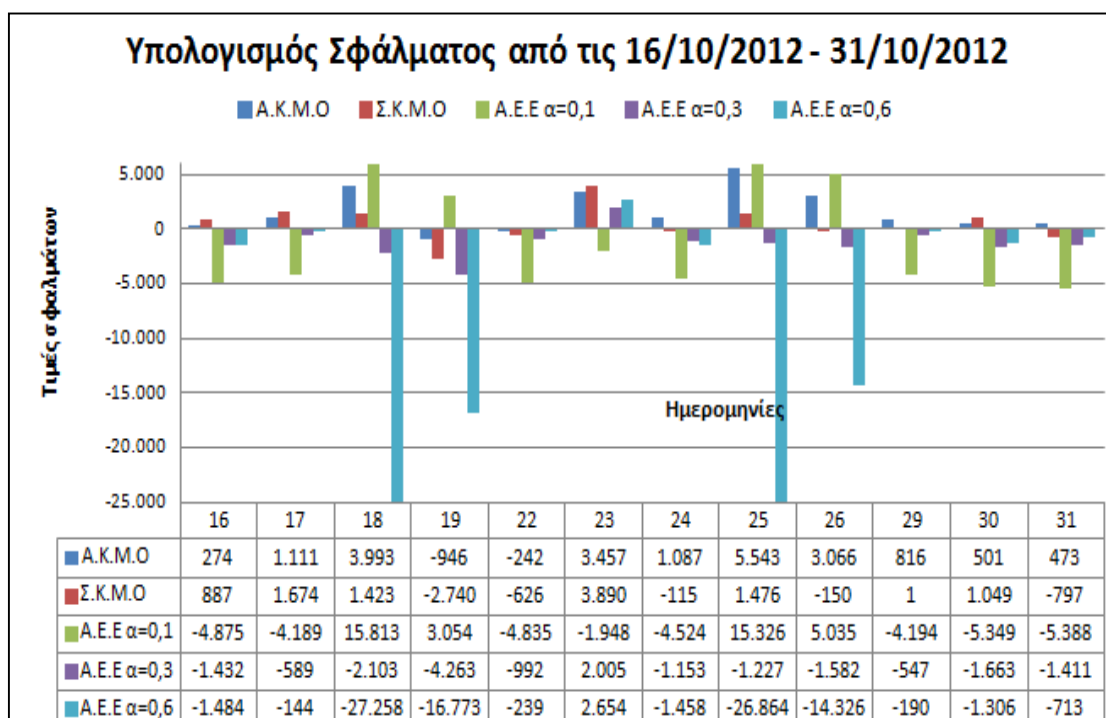
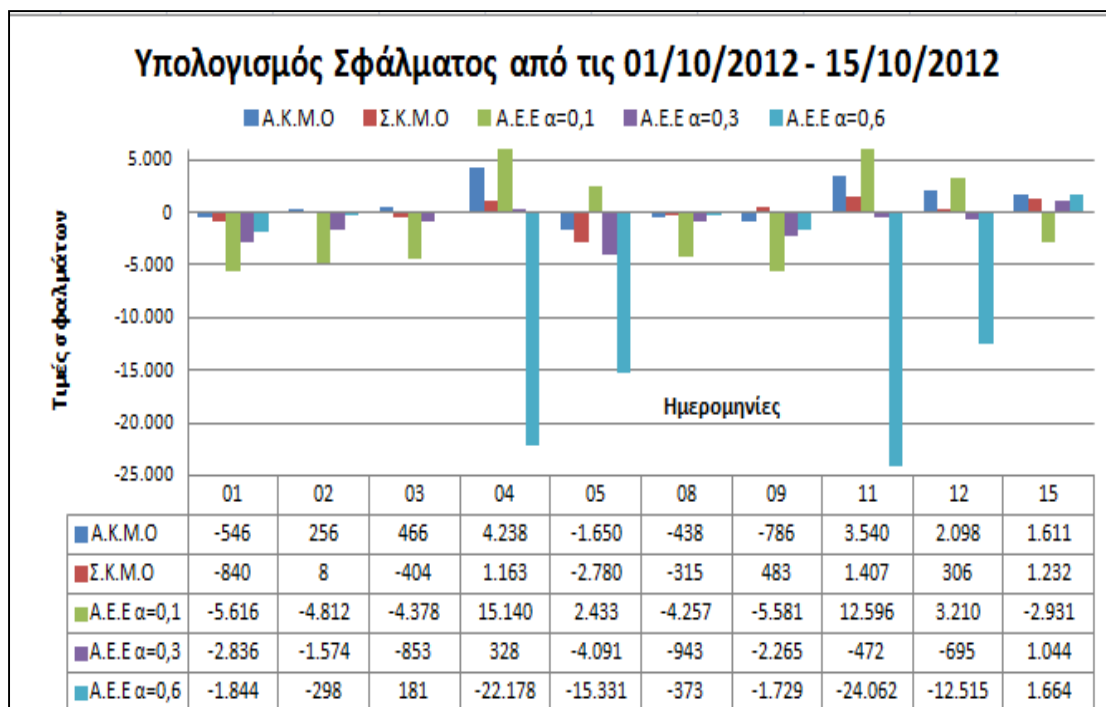
- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές καταθέσεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές καταθέσεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές καταθέσεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές καταθέσεις της Τετάρτης
- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευή σύγκριση με τις πραγματικές καταθέσεις της Πέμπτης, Παρασκευής & Σάββατο.

Πίνακας 12: Υπολογισμός Σφαλμάτων

Ημερ/νία	Α.Κ.Μ.Ο	Σ.Κ.Μ.Ο	Α.Ε.Ε α=0,1	Α.Ε.Ε α=0,3	Α.Ε.Ε α=0,6
01/10/2012	-546	-840	-5.616	-2.836	-1.844
02/10/2012	256	8	-4.812	-1.574	-298
03/10/2012	466	-404	-4.378	-853	181
04/10/2012	4.238	1.163	15.140	328	-22.178
05/10/2012	-1.650	-2.780	2.433	-4.091	-15.331
08/10/2012	-438	-315	-4.257	-943	-373
09/10/2012	-786	483	-5.581	-2.265	-1.729
11/10/2012	3.540	1.407	12.596	-472	-24.062
12/10/2012	2.098	306	3.210	-695	-12.515
15/10/2012	1.611	1.232	-2.931	1.044	1.664
16/10/2012	274	887	-4.875	-1.432	-1.484
17/10/2012	1.111	1.674	-4.189	-589	-144
18/10/2012	3.993	1.423	15.813	-2.103	-27.258
19/10/2012	-946	-2.740	3.054	-4.263	-16.773
22/10/2012	-242	-626	-4.835	-992	-239
23/10/2012	3.457	3.890	-1.948	2.005	2.654
24/10/2012	1.087	-115	-4.524	-1.153	-1.458
25/10/2012	5.543	1.476	15.326	-1.227	-26.864
26/10/2012	3.066	-150	5.035	-1.582	-14.326
29/10/2012	816	1	-4.194	-547	-190
30/10/2012	501	1.049	-5.349	-1.663	-1.306
31/10/2012	473	-797	-5.388	-1.411	-713
Γενικό άθροισμα	27.921	6.231	9.730	-27.312	-164.586

Παρατηρώντας τις τιμές των σφαλμάτων καταλήγουμε στο συμπέρασμα ότι η μέθοδος της εκθετικής εξομάλυνσης δεν είναι κατάλληλη για την πρόβλεψη των καταθέσεων. Στην περίπτωση όπου το $a=0.6$ εμφανίζονται προβλέψεις που απέχουν από την πραγματικότητα κατά μέσω όρο 7.500 μονάδες και πολύ συχνά φτάνουν τις 25.000 μονάδες. Στην περίπτωση όπου το $a=0.3$ οι αποκλίσεις είναι αισθητά μικρότερες αλλά κυριαρχούν οι αρνητικές τιμές, δηλαδή τα ποσά των προβλέψεων είναι μικρότερα από τις πραγματικές καταθέσεις. Σε αυτή τη περίπτωση ο κίνδυνος το ATM να ξεπεράσει το ποσό των καταθέσεων είναι άμεσος και για αυτό το λόγο απορρίπτουμε και αυτή τη μέθοδο. Στην περίπτωση όπου το $a=0.1$ εντοπίζουμε απότομες αντιθέσεις καθώς τη μία μέρα η πρόβλεψη απέχει 15.140 μονάδες και την επόμενη μόλις 2.433, γεγονός που παρατηρήθηκε τρεις φορές στη διάρκεια ενός μήνα, επομένως και αυτή μέθοδος δεν μπορεί να χαρακτηριστεί ως αξιόπιστη.

Ο πιο αξιόπιστος τρόπος πρόβλεψης είναι ο σταθμισμένο κινούμενος μέσος όρος καθώς εντοπίζονται οι μικρότερες αποκλίσεις σε σχέση με τη μέθοδο του απλού κινητού μέσου. Διαγραμματική απεικόνιση των τιμών.

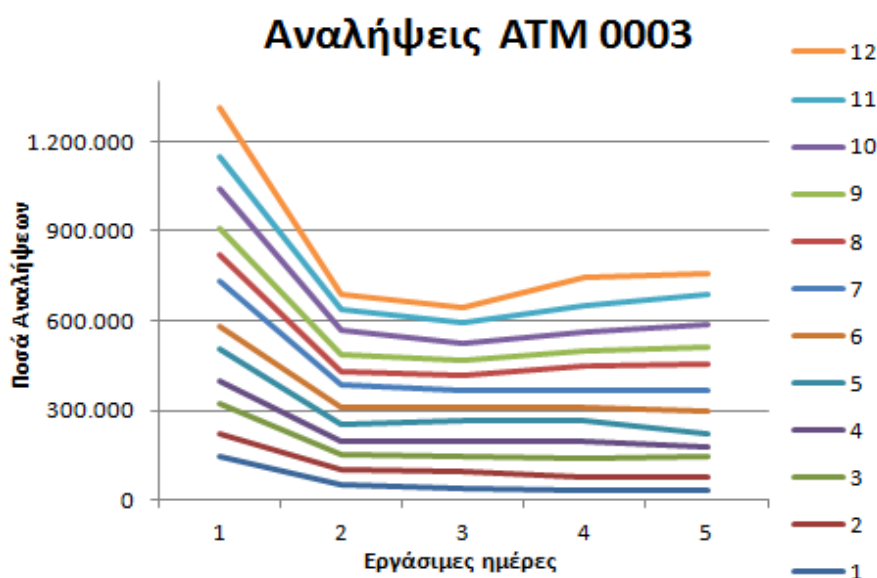


4.6.3 Αυτόματη Ταμιακή Μηχανή στη περιοχή της Λάρισας

Το τρίτο μηχάνημα ATM δραστηριοποιείται στη περιοχή της Λάρισας για την πραγματοποίηση αναλήψεων και καταθέσεων (το οποίο θα αναφέρεται με τον κωδικό 0003). Για την ανάλυση των δεδομένων θα εργαστούμε με τον ίδιο τρόπο, δηλαδή θα χωρίσουμε τις συναλλαγές ανά μέρα και μόνο στις πέντε εργάσιμες μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή) για ένα έτος 2012. Τις Δευτέρες οι συναλλαγές εμφανίζονται ιδιαίτερα αυξημένες καθώς περιέχουν και τις συναλλαγές του σαββατοκύριακου, τις Τρίτες οι τιμές μειώνονται και κυμαίνονται σχεδόν σταθερές μέχρι τη Παρασκευή όπου και εμφανίζεται μία μικρή αύξηση

Πίνακας 13: Δεδομένα αναλήψεων 12 μηνών για το ATM 0003

Εργάσιμες Μήνες	Ημέρα					Γενικό άθροισμα
	1	2	3	4	5	
1	146.000	49.400	34.900	33.300	33.200	296.800
2	74.000	49.850	58.400	42.000	45.500	269.750
3	101.800	50.700	51.700	65.100	67.600	336.900
4	75.100	48.290	49.200	55.710	27.400	255.700
5	106.900	54.520	72.480	66.600	50.100	350.600
6	79.200	54.420	45.800	47.700	73.280	300.400
7	148.400	75.420	51.330	58.480	68.180	401.810
8	88.600	47.600	54.140	79.220	89.000	358.560
9	89.000	54.200	50.220	52.800	57.600	303.820
10	130.800	86.500	55.700	63.500	77.400	413.900
11	107.400	64.800	69.720	88.490	101.980	432.390
12	167.600	55.620	51.800	90.120	67.200	432.340



Στη συνέχεια θα υπολογίσουμε την πρόβλεψη των αναλήψεων για τις εργάσιμες ημέρες του Ιουλίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων.

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 14: Προβλέψεις των αναλήψεων για το μήνα Ιούλιο με τις τρεις διαφορετικές μεθόδους

Ημερ/νίες	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
02/07/2012	14.663	13.905	9.335	13.047	13.617
03/07/2012	15.669	15.980	9.655	13.739	14.747
04/07/2012	15.925	14.930	9.806	14.347	15.498
05/07/2012	47.738	44.376	29.718	43.272	15.079
06/07/2012	30.400	29.009	20.019	28.789	14.732
09/07/2012	15.518	15.242	10.560	15.696	17.233
10/07/2012	16.375	17.283	10.490	14.744	14.489
11/07/2012	16.338	16.025	9.693	12.213	9.695
12/07/2012	47.025	42.329	26.172	28.668	5.996
13/07/2012	29.510	27.322	17.628	21.410	10.546
16/07/2012	14.930	13.804	9.190	12.072	13.458
17/07/2012	15.936	15.187	9.241	12.601	13.813
18/07/2012	16.331	14.221	9.557	13.628	15.251
19/07/2012	49.389	46.952	29.350	43.006	15.820
20/07/2012	31.614	30.224	19.606	28.636	15.028
23/07/2012	16.051	15.666	10.195	15.331	16.751
24/07/2012	17.181	17.955	10.170	14.884	15.124
25/07/2012	17.331	16.564	10.399	15.680	16.730
26/07/2012	52.611	49.448	32.045	47.891	16.772
27/07/2012	33.549	33.289	22.040	31.835	16.249
30/07/2012	16.773	16.467	11.416	15.922	16.087
31/07/2012	17.604	18.388	11.686	16.474	17.205
Γενικό άθροισμα	548.458	524.561	337.972	473.883	319.920

Υπολογίζουμε το σφάλμα που προκύπτει με τη χρήση των τριών διαφορετικών μεθόδων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των αναλήψεων ανάλογα με την ημέρα τροφοδοσίας.

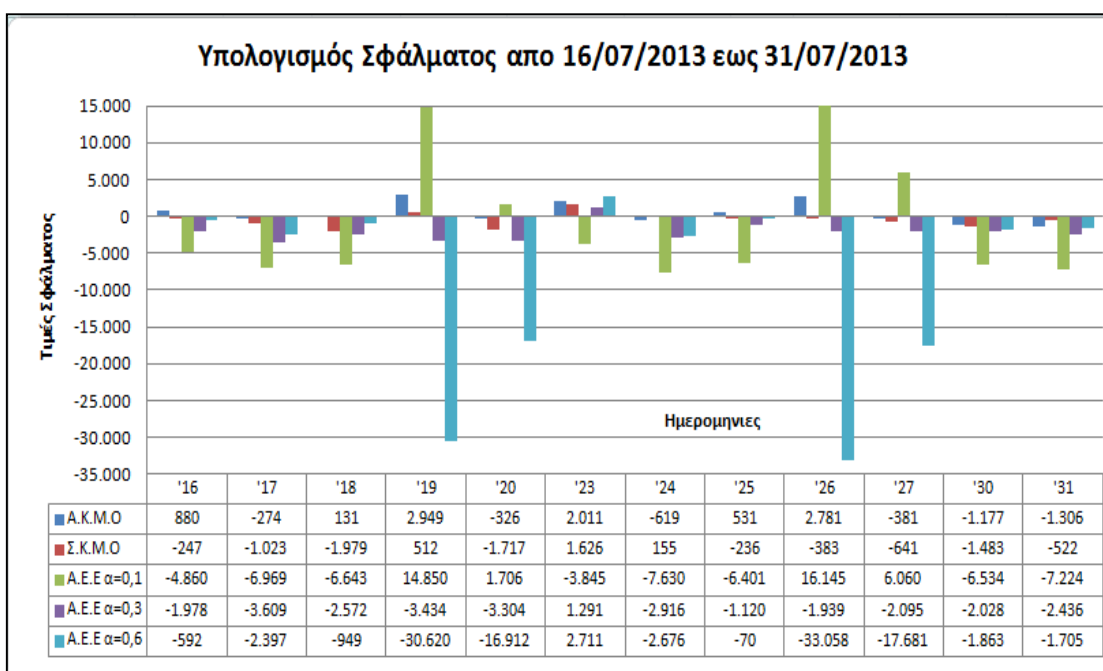
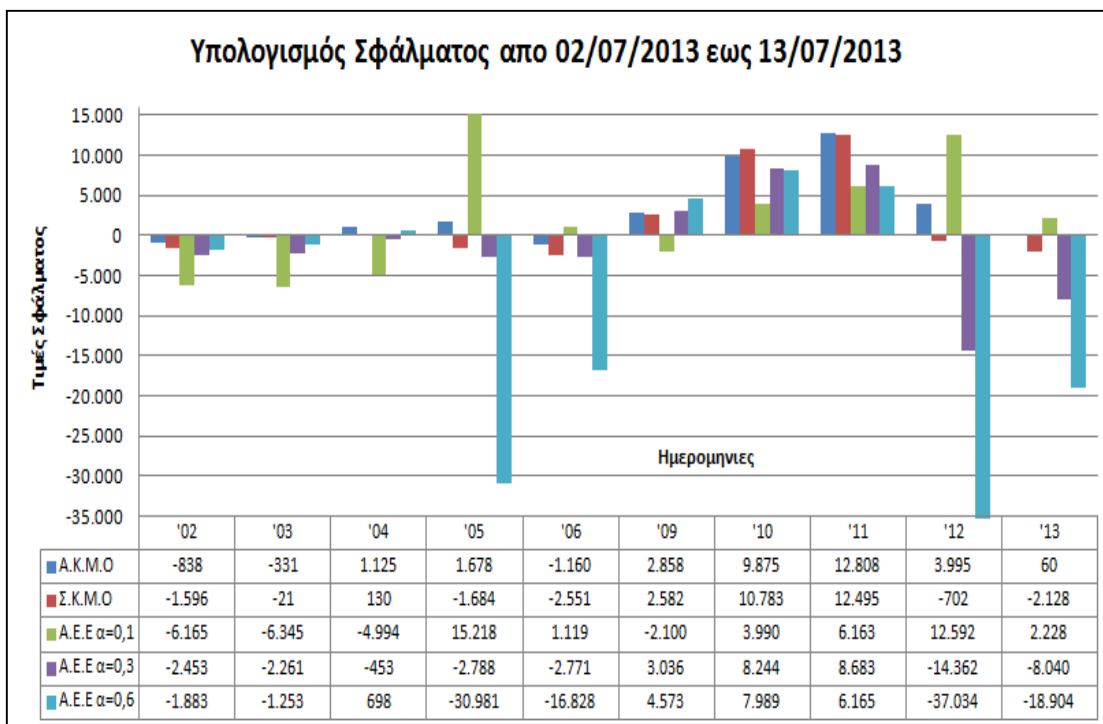
- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές αναλήψεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές αναλήψεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές αναλήψεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές αναλήψεις της Τετάρτης
- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευής σύγκριση με τις πραγματικές αναλήψεις της Πέμπτης, Παρασκευής & Σάββατο.

Πίνακας 15: Υπολογισμός σφαλμάτων

Ημερ/νία	Α.Κ.Μ.Ο	Σ.Κ.Μ.Ο	Α.Ε.Ε α=0,1	Α.Ε.Ε α=0,3	Α.Ε.Ε α=0,6
02/07/2012	-838	-1.596	-6.165	-2.453	-1.883
03/07/2012	-331	-21	-6.345	-2.261	-1.253
04/07/2012	1.125	130	-4.994	-453	698
05/07/2012	1.678	-1.684	15.218	-2.788	-30.981
06/07/2012	-1.160	-2.551	1.119	-2.771	-16.828
09/07/2012	2.858	2.582	-2.100	3.036	4.573
10/07/2012	9.875	10.783	3.990	8.244	7.989
11/07/2012	12.808	12.495	6.163	8.683	6.165
12/07/2012	3.995	-702	12.592	-14.362	-37.034
13/07/2012	60	-2.128	2.228	-8.040	-18.904
16/07/2012	880	-247	-4.860	-1.978	-592
17/07/2012	-274	-1.023	-6.969	-3.609	-2.397
18/07/2012	131	-1.979	-6.643	-2.572	-949
19/07/2012	2.949	512	14.850	-3.434	-30.620
20/07/2012	-326	-1.717	1.706	-3.304	-16.912
23/07/2012	2.011	1.626	-3.845	1.291	2.711
24/07/2012	-619	155	-7.630	-2.916	-2.676
25/07/2012	531	-236	-6.401	-1.120	-70
26/07/2012	2.781	-383	16.145	-1.939	-33.058
27/07/2012	-381	-641	6.060	-2.095	-17.681
30/07/2012	-1.177	-1.483	-6.534	-2.028	-1.863
31/07/2012	-1.306	-522	-7.224	-2.436	-1.705

Απορρίπτουμε και αυτή τη φορά τη μέθοδος της εκθετικής εξομάλυνσης καθώς οι προβλέψεις παρουσιάζουν μεγάλες αποκλίσεις. Στην περίπτωση όπου το $\alpha=0.6$ εντοπίσουμε προβλέψεις όπου σε σύγκριση με την πραγματικότητα απέχουν πάνω από 30.000 μονάδες. Στην περίπτωση όπου το $\alpha=0.3$ οι αποκλίσεις μειώνονται φτάνοντας έως και τις 14.000 μονάδες αλλά κυριαρχούν οι αρνητικές τιμές, δηλαδή τα ποσά των προβλέψεων είναι μικρότερα από τις πραγματικές αναλήψεις. Σε αυτή τη περίπτωση ο κίνδυνος το ATM να μείνει χωρίς χρήματα είναι άμεσος και για αυτό το λόγο απορρίπτουμε και αυτή τη μέθοδο. Στην περίπτωση όπου το $\alpha=0.1$ εντοπίζουμε απότομες αντιθέσεις καθώς τη μία μέρα η πρόβλεψη απέχει 15.218 μονάδες και την επόμενη μόλις 1.119, γεγονός που παρατηρήθηκε τέσσερις φορές στη διάρκεια ενός μήνα, επομένως και αυτή μέθοδος δεν μπορεί να χαρακτηριστεί ως αξιόπιστη.

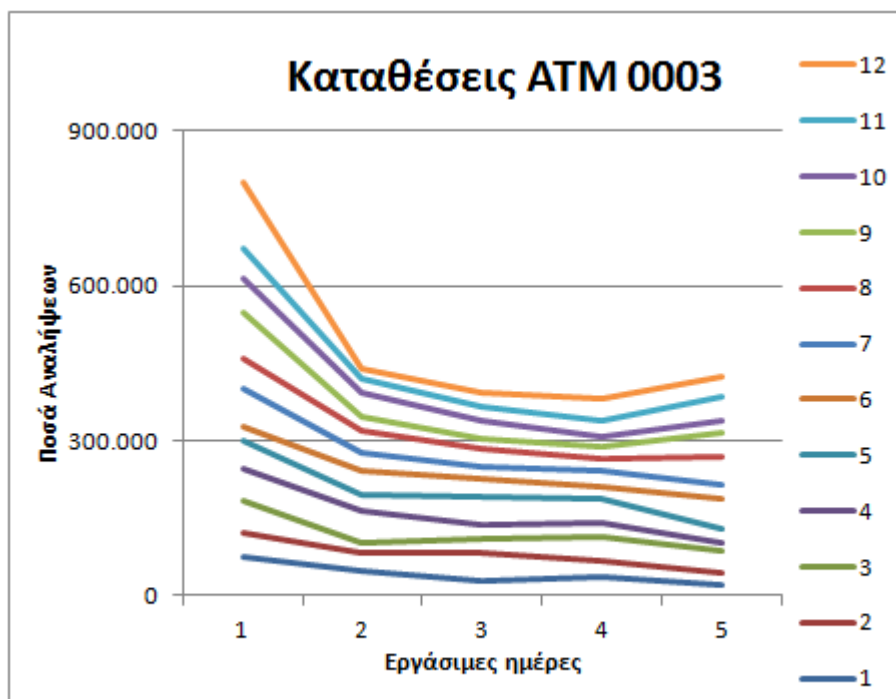
Επομένως από τις υπόλοιπες δύο μεθόδους, στον απλός κινούμενος μέσος όρος και στο σταθμισμένο κινούμενος μέσος όρος παρατηρούμε ότι τις μικρότερες αποκλίσεις εμφανίζει η μέθοδο του σταθμισμένου κινούμενου μέσου. Τις ημέρες 10/07 και 11/07 το μηχάνημα αντιμετώπιζε τεχνικό πρόβλημα και βρισκόταν αρκετές ώρες εκτός λειτουργίας με αποτέλεσμα ο αριθμός των συναλλαγών να είναι μειωμένος. Το παραπάνω γεγονός επηρέασε την πρόβλεψη των συγκεκριμένων ημερών (σφάλμα 10.783 μονάδες και 12.495 μονάδες αντίστοιχα) χωρίς όμως να επηρεάσει αρνητικά τις προβλέψεις των επόμενων ημερών. Ένα τυχαίο γεγονός επηρεάζει τις αντίστοιχες προβλέψεις χωρίς να μεταφέρεται το λάθος και στις επόμενες, αρά η συγκεκριμένη μέθοδος διατηρεί την αξιοπιστία της. Συνεχίζουμε με τη γραφική αναπαράσταση των τιμών των σφαλμάτων χωρίζοντας το μήνα σε δύο διαστήματα



Με τον ίδιο τρόπο θα αναλύσουμε και τις καταθέσεις για το ATM 0003. Στον πίνακα 16 εμφανίζονται οι καταθέσεις για ένα έτος 2012. Στο σχήμα 16 παρουσιάζεται τα μηνιαία ποσά καταθέσεων χωρισμένα ανά μέρα (1: Δευτέρα, 2: Τρίτη, 3: Τετάρτη, 4: Πέμπτη, 5: Παρασκευή). Η ζήτηση μπορεί να χαρακτηριστεί ως στατική καθώς οι συναλλαγές δεν παρουσιάζουν διακυμάνσεις από μήνα σε μήνα εκτός από το Δεκέμβριο όπου οι καταθέσεις είναι γενικά αυξημένες. Οι συναλλαγές την ημέρα Δευτέρα είναι ιδιαίτερα υψηλές καθώς περιέχουν και τις συναλλαγές που πραγματοποιήθηκαν το σαββατοκύριακο που προηγήθηκε. Από Τρίτη τα ποσά των καταθέσεων μειώνονται μέχρι τη Πέμπτη όπου αρχίζουν πάλι να αυξάνονται.

Πίνακας 16: Δεδομένα καταθέσεων 12 μηνών για το ATM 0003

Μήνες	1	2	3	4	5	Γενικό άθροισμα
1	76.400	49.600	29.500	34.900	21.300	211.700
2	45.200	31.700	53.300	32.900	21.300	184.400
3	61.800	21.000	26.300	45.300	44.900	199.300
4	61.000	63.000	29.000	30.000	16.600	199.600
5	56.000	28.500	51.700	45.900	26.400	208.500
6	24.800	49.700	38.300	22.500	57.100	192.400
7	75.700	33.520	21.520	30.000	27.300	188.040
8	59.600	42.400	34.100	23.600	55.700	215.400
9	86.300	25.900	19.100	24.500	43.100	198.900
10	67.600	48.100	35.900	19.700	24.600	195.900
11	58.000	27.800	26.200	31.300	46.400	189.700
12	128.700	19.900	28.500	42.200	40.800	260.100
Γενικό άθροισμα	801.100	441.120	393.420	382.800	425.500	2.443.940



Σχήμα 16: Ετήσια ποσά καταθέσεων για το έτος 2012

Στη συνέχεια θα υπολογίσουμε την πρόβλεψη των καταθέσεων για τις εργάσιμες ημέρες του Μαρτίου και με τις τρεις διαφορετικές μέθοδο υπολογισμού των προβλέψεων. Οι υπολογισμοί έχουν γίνει με βάση τις προδιαγραφές που αναφέρθηκαν παραπάνω (Παρουσία του τρόπου ανάλυσης των δεδομένων).

- **A.K.M.O:** Απλός κινούμενος μέσος όρος
- **Σ.Κ.Μ.Ο:** Σταθμισμένος κινούμενος μέσος όρος με συντελεστές 0.7,0.2 και 0.1 από τις πιο πρόσφατες στις παλαιότερες τιμές
- **A.E.E:** Απλή εκθετικής εξομάλυνσης με συντελεστή σταθεράς εξομάλυνσης α 0.1,0.3 και 0,6

Πίνακας 17: Προβλέψεις των καταθέσεων για το μήνα Ιούλιο με τις τρεις διαφορετικές μεθόδους

Ημερ/νία	A.K.M.O	Σ.Κ.Μ.Ο	A.E.E. $\alpha=0,1$	A.E.E. $\alpha=0,3$	A.E.E $\alpha=0,6$
02/07/2012	8.550	8.070	5.665	8.749	11.286
03/07/2012	8.403	8.149	5.314	8.003	8.399
04/07/2012	8.359	8.324	5.327	7.936	8.100
05/07/2012	25.382	25.664	16.276	23.894	8.112
06/07/2012	17.559	17.760	11.412	16.928	9.065
09/07/2012	8.586	9.095	5.919	8.551	8.906
10/07/2012	9.098	9.345	6.002	8.645	8.947
11/07/2012	8.298	8.484	5.061	6.036	3.651
12/07/2012	23.286	21.344	13.558	12.659	1.460
13/07/2012	15.699	14.229	9.390	10.543	5.264
16/07/2012	7.581	6.507	4.652	5.493	5.766
17/07/2012	8.206	7.455	4.650	5.825	6.326
18/07/2012	8.162	6.485	4.501	6.178	6.850
19/07/2012	23.229	21.712	13.345	18.113	6.220
20/07/2012	15.949	13.962	8.801	12.107	6.208
23/07/2012	8.043	6.778	4.381	6.154	6.383
24/07/2012	8.556	7.935	4.404	6.472	6.963
25/07/2012	8.631	7.330	4.565	7.044	7.885
26/07/2012	24.636	21.271	13.845	21.521	7.714
27/07/2012	15.593	14.531	9.560	13.822	6.866
30/07/2012	7.671	7.091	4.892	6.608	6.286
31/07/2012	8.208	8.453	5.018	7.098	7.524
Γενικό άθροισμα	277.682	259.970	166.538	228.379	154.181

Με σκοπό την εξαγωγή συμπερασμάτων σχετικά με ποιας μεθόδου το αποτέλεσμα πλησιάζει περισσότερο στις πραγματικές καταθέσεις, προχωράμε στον υπολογισμό των σφαλμάτων. Όπως αναφέραμε και παραπάνω θα υπολογίσουμε την πρόβλεψη με τα πραγματικά ποσά των καταθέσεων ανάλογα με την ημέρα τροφοδοσίας.

- Προτεινόμενη ημέρα τροφοδοσίας Δευτέρα σύγκριση με τις πραγματικές καταθέσεις της Παρασκευής και του Σαββάτου.
- Προτεινόμενη ημέρα τροφοδοσίας Τρίτη σύγκριση με τις πραγματικές καταθέσεις της Δευτέρας
- Προτεινόμενη ημέρα τροφοδοσίας Τετάρτη σύγκριση με τις πραγματικές καταθέσεις της Τρίτης
- Προτεινόμενη ημέρα τροφοδοσίας Πέμπτη σύγκριση με τις πραγματικές καταθέσεις της Τετάρτης

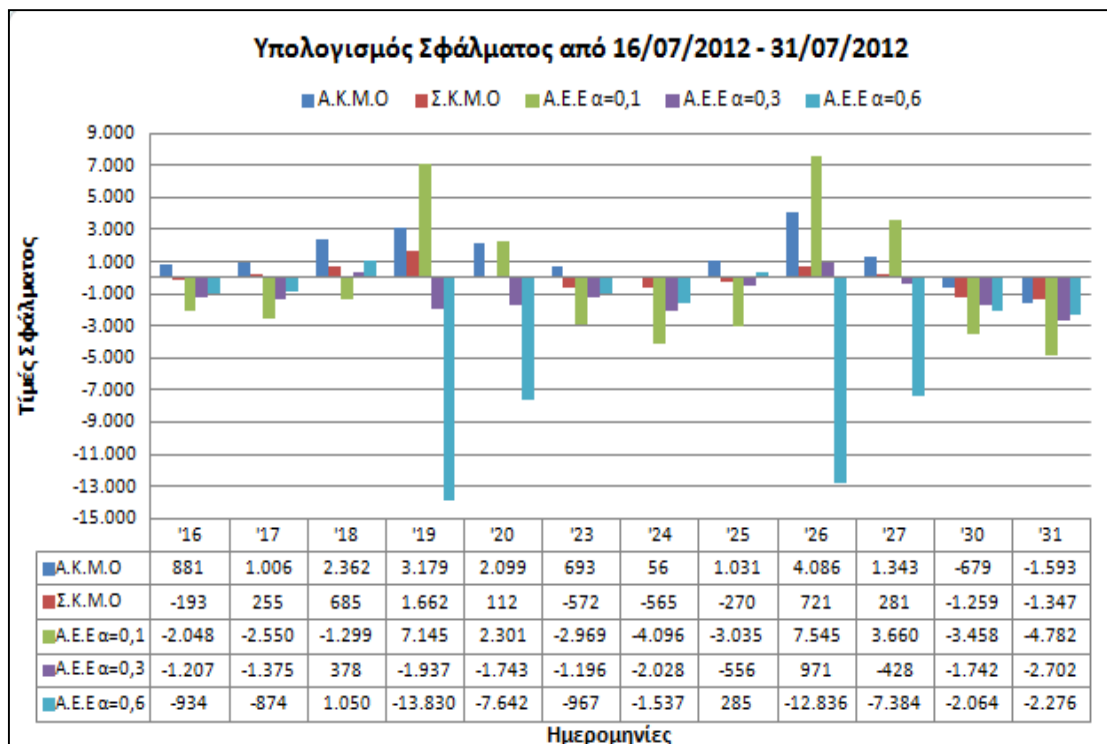
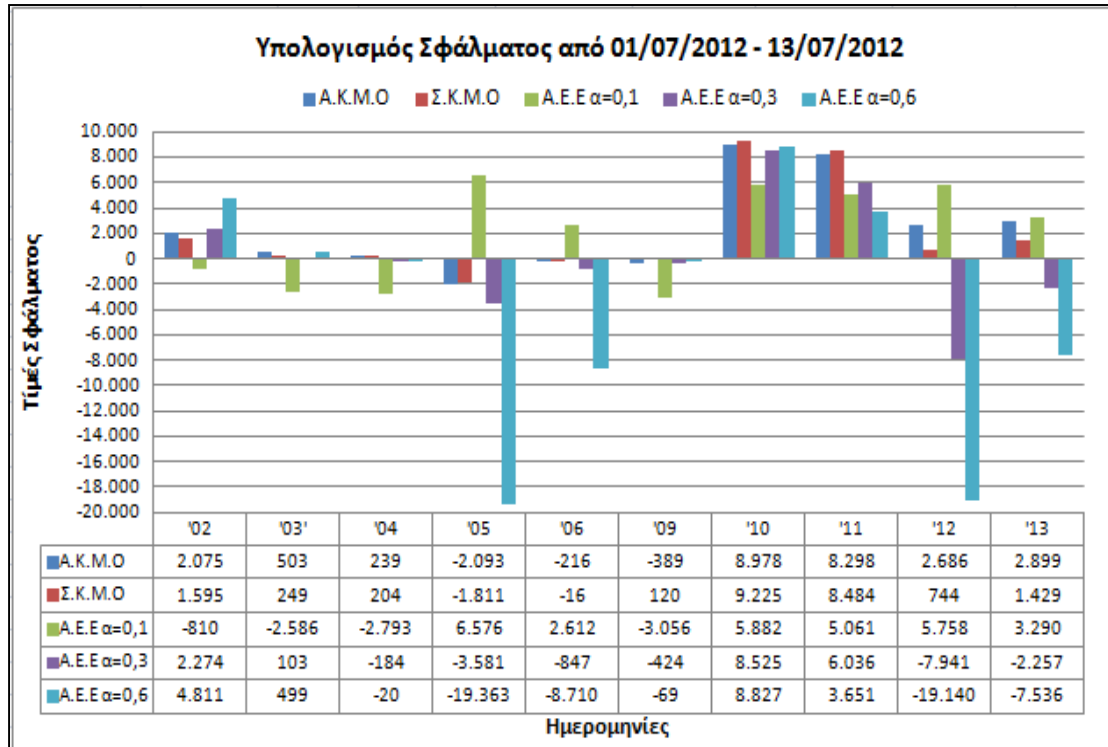
Πίνακας 18: Υπολογισμός Σφαλμάτων

Ημερ/νία	Α.Κ.Μ.Ο	Σ.Κ.Μ.Ο	Α.Ε.Ε α=0,1	Α.Ε.Ε α=0,3	Α.Ε.Ε α=0,6
02/07/2012	2.075	1.595	-810	2.274	4.811
03/07/2012	503	249	-2.586	103	499
04/07/2012	239	204	-2.793	-184	-20
05/07/2012	-2.093	-1.811	6.576	-3.581	-19.363
06/07/2012	-216	-16	2.612	-847	-8.710
09/07/2012	-389	120	-3.056	-424	-69
10/07/2012	8.978	9.225	5.882	8.525	8.827
11/07/2012	8.298	8.484	5.061	6.036	3.651
12/07/2012	2.686	744	5.758	-7.941	-19.140
13/07/2012	2.899	1.429	3.290	-2.257	-7.536
16/07/2012	881	-193	-2.048	-1.207	-934
17/07/2012	1.006	255	-2.550	-1.375	-874
18/07/2012	2.362	685	-1.299	378	1.050
19/07/2012	3.179	1.662	7.145	-1.937	-13.830
20/07/2012	2.099	112	2.301	-1.743	-7.642
23/07/2012	693	-572	-2.969	-1.196	-967
24/07/2012	56	-565	-4.096	-2.028	-1.537
25/07/2012	1.031	-270	-3.035	-556	285
26/07/2012	4.086	721	7.545	971	-12.836
27/07/2012	1.343	281	3.660	-428	-7.384
30/07/2012	-679	-1.259	-3.458	-1.742	-2.064
31/07/2012	-1.593	-1.347	-4.782	-2.702	-2.276
Γενικό άθροισμα	37.442	19.730	16.348	-11.861	-86.059

- Προτεινόμενη ημέρα τροφοδοσίας Παρασκευή σύγκριση με τις πραγματικές καταθέσεις της Πέμπτη, Παρασκευή & Σάββατο.

Η μέθοδος της εκθετικής εξομάλυνσης απορρίπτεται αφού οι προβλέψεις παρουσιάζουν μεγάλες αποκλίσεις. Στην περίπτωση όπου το $\alpha=0.6$ εντοπίσουμε προβλέψεις όπου σε σύγκριση με την πραγματικότητα απέχουν πάνω από 19.000 μονάδες. Στην περίπτωση όπου το $\alpha=0.3$ και $\alpha=0.1$ οι αποκλίσεις μειώνονται φτάνοντας έως και τις 8.500 και 7.000 μονάδες αντίστοιχα, αλλά κυριαρχούν οι αρνητικές τιμές. Με τους παραπάνω τρόπους η πρόβλεψη είναι υψηλότερη από τις πραγματικές καταθέσεις, άρα η πιθανότητα να οδηγηθούμε σε τροφοδοσία εξαιτίας πιθανών υψηλών καταθέσεων είναι υπαρκτή.

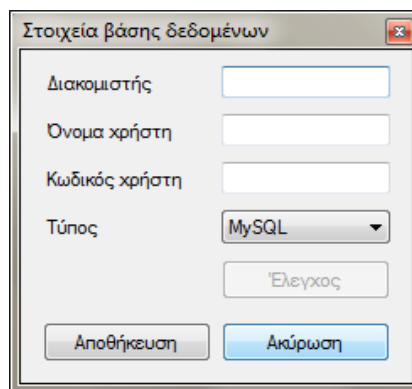
Τέλος ανάμεσα στον απλός κινούμενος μέσος όρος και στο σταθμισμένο κινούμενος μέσος όρος παρατηρούμε ότι τις μικρότερες αποκλίσεις εμφανίζει η μέθοδο του σταθμισμένου κινούμενου μέσου. Τις ημέρες 10/07 και 11/07 το μηχάνημα αντιμετώπιζε τεχνικό πρόβλημα και βρισκόταν αρκετές ώρες εκτός λειτουργίας με αποτέλεσμα ο αριθμός των συναλλαγών να είναι μειωμένος. Το παραπάνω γεγονός επηρέασε την πρόβλεψη των συγκεκριμένων ημερών (σφάλμα 9.225 μονάδες και 8.454 μονάδες αντίστοιχα) χωρίς όμως να επηρεάσει αρνητικά τις προβλέψεις των επόμενων ημερών. Ένα τυχαίο γεγονός επηρεάζει τις αντίστοιχες προβλέψεις χωρίς να μεταφέρεται το λάθος και στις επόμενες, άρα η συγκεκριμένη μέθοδος διατηρεί την αξιοπιστία της. Συνεχίζουμε με τη γραφική αναπαράσταση των τιμών των σφαλμάτων χωρίζοντας το μήνα σε δύο διαστήματα



5. Ανάπτυξη της εφαρμογής

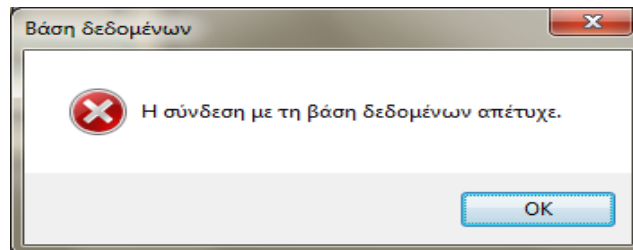
5.1 Γενική Περιγραφή εφαρμογής

Τη πρώτη φορά χρήσης της εφαρμογής θα πρέπει να καταχωρήσουμε τα στοιχεία σύνδεσης με τη βάση (Διακομιστής: localhost, Όνομα χρήστη: developer, Κωδικός χρήστη: developer και Τύπος:MySQL). Η εφαρμογή είναι ανεξάρτητη από τη βάση δεδομένων με την οποία συνδέεται επομένως μπορεί να χρησιμοποιηθεί είτε με διαφορετικές βάσεις είτε με διαφορετικού τύπους (Access, MySQL, Oracle). Με αυτό τον τρόπο έχουμε δημιουργήσει μία εφαρμογή που μπορεί να προσαρμοστεί γρήγορα και εύκολα σε πολλά τμήματα ενός πιστωτικού ιδρύματος με διαφορετικό τρόπο οργάνωσης δεδομένων. Στη συνέχεια γίνεται έλεγχος των στοιχείων σύνδεσης, εάν είναι λανθασμένα ενημερώνει το χρήστη ότι η σύνδεση με τη βάση απέτυχε και τον οδηγεί στην αρχική καρτέλα ώστε να επαναλάβει την καταχώρηση.



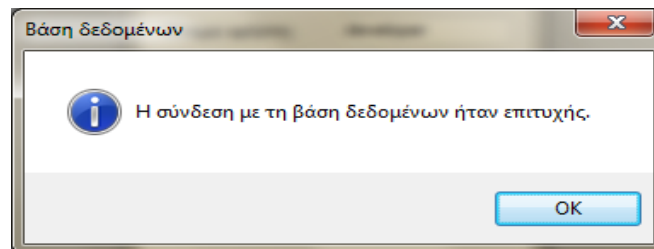
Εικόνα 1: Εισαγωγή στοιχείων βάσης δεδομένων

Στη συνέχεια γίνεται έλεγχος των στοιχείων σύνδεσης, εάν είναι λανθασμένα ενημερώνει το χρήστη ότι η σύνδεση με τη βάση απέτυχε και τον οδηγεί στην αρχική καρτέλα ώστε να επαναλάβει την καταχώρηση.

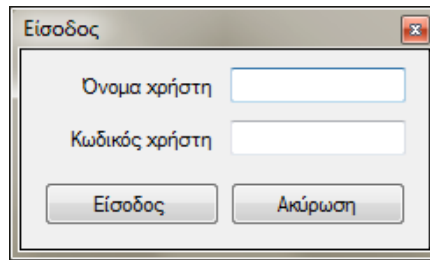


Εικόνα 2: Ανεπιτυχής σύνδεση με τη βάση δεδομένων

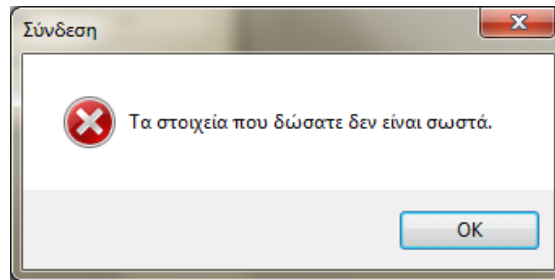
Όταν ο χρήστης καταχωρίσει τα σωστά δεδομένα η σύνδεση ολοκληρώνεται με επιτυχία και μετά η εφαρμογή ζητάει τα προσωπικά στοιχεία εισαγωγής του χρήστη.



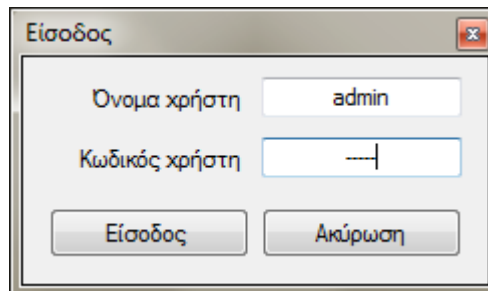
Εικόνα 3: Επιτυχής σύνδεση με τη βάση δεδομένων

**Εικόνα 4: Εισαγωγή στοιχείων**

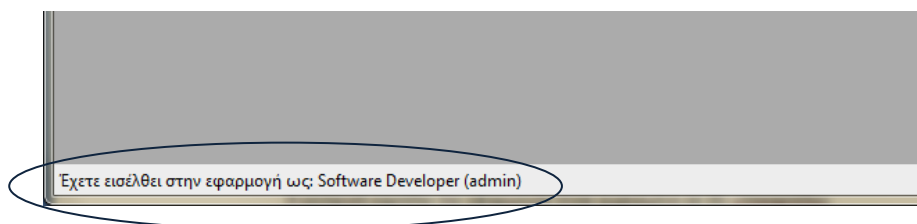
Γίνεται έλεγχος εγκυρότητας των στοιχείων που καταχωρήθηκαν.

**Εικόνα 5: Λανθασμένα στοιχεία εισαγωγής**

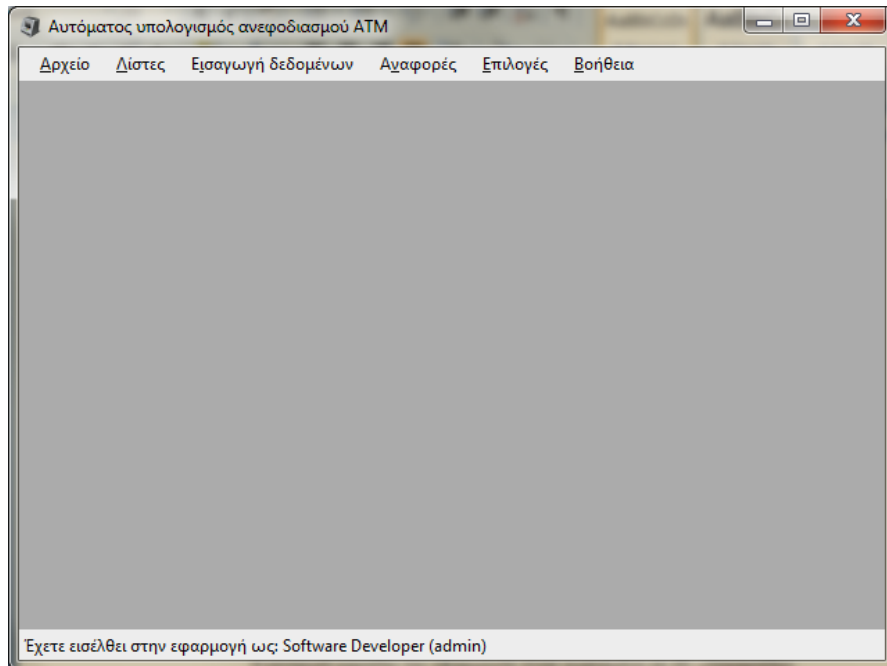
Στην περίπτωση που τα στοιχεία που καταχωρήθηκαν δεν είναι σωστά, η εφαρμογή δίνει τη δυνατότητα στο χρήστη να εισάγει ξανά τα προσωπικά του στοιχεία.

**Εικόνα 6: Δεύτερη προσπάθεια εισαγωγής στοιχείων**

Όταν τα στοιχεία εισαγωγής, όνομα και κωδικός χρήστη, είναι σωστά η εφαρμογή ανοίγει και στη μπάρα πληροφοριών εμφανίζεται το όνομα του χρήστη που έχει συνδεθεί.

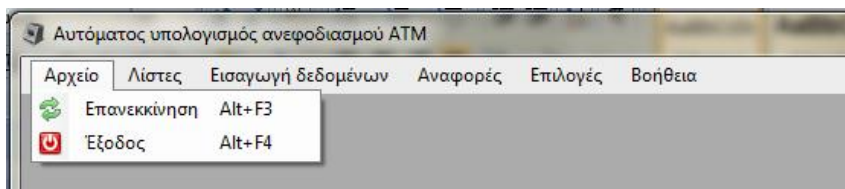
**Εικόνα 7: Επιτυχής εισαγωγή στην εφαρμογή**

Η κεντρική καρτέλα της εφαρμογής είναι χωρισμένη σε έξι υποκαρτέλες.



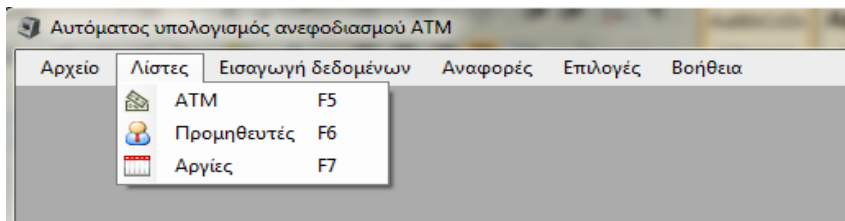
Εικόνα 8: Κεντρική Φόρμα

Από τη καρτέλα «Αρχείο» μπορούμε να πραγματοποιήσουμε επανεκκίνηση του προγράμματος ή έξοδο από αυτό. Κάθε επιλογή μπορεί να ενεργοποιηθεί με τη χρήση του πληκτρολογίου, επιλέγοντας Alt+F3 για επανεκκίνηση ή Alt+F4 για έξοδο.



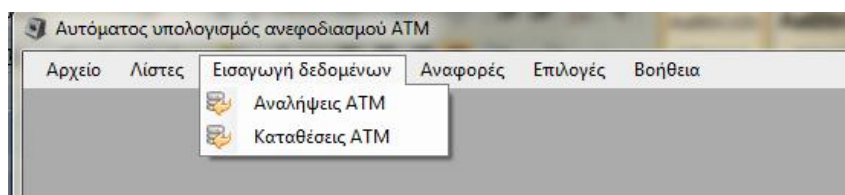
Εικόνα 9: Καρτέλα Αρχείο

Η καρτέλα «Λίστες» περιέχει πληροφορίες σχετικά με τις αυτόματες ταμειακές μηχανές που χειρίζεται η συγκεκριμένη εφαρμογή, τους προμηθευτές και τις αργίες, επίσημες και τοπικές. Οι προαναφερθέντες πίνακες μπορούν να ενεργοποιηθούν και από το πληκτρολόγιο, επιλέγοντας F5, F6 και F7.



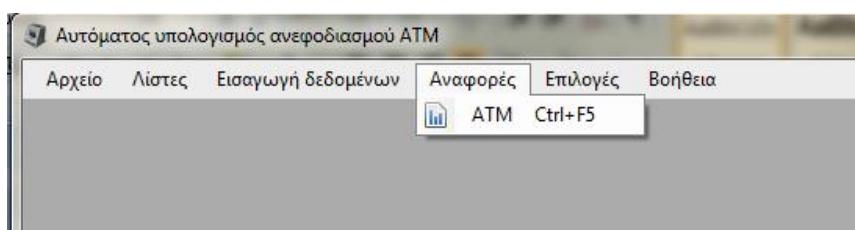
Εικόνα 10: Καρτέλα Λίστες

Η καρτέλα «Εισαγωγή Δεδομένων» δίνει τη δυνατότητα εισαγωγής των δεδομένων, δηλαδή τον αριθμό των συναλλαγές (αναλήψεις & καταθέσεις) ανά μέρα ανά ATM, στο σύστημα ώστε να ενημερωθεί η αντίστοιχη βάση.



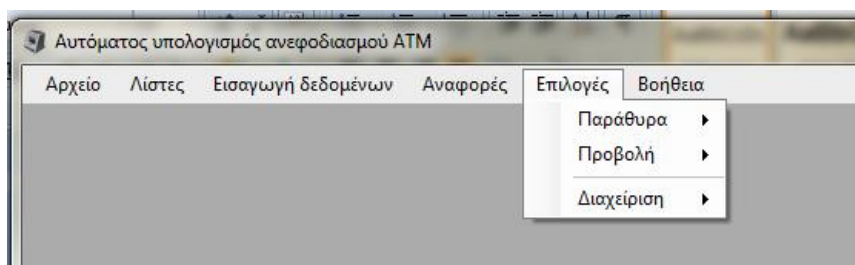
Εικόνα 11: Καρτέλα Εισαγωγή Δεδομένων

Η καρτέλα «Αναφορές» παράγει τα επιθυμητά αποτελέσματα της εφαρμογής. Οι αναφορές ενεργοποιούνται και από το πληκτρολόγιο επιλέγοντας Ctrl+F5.



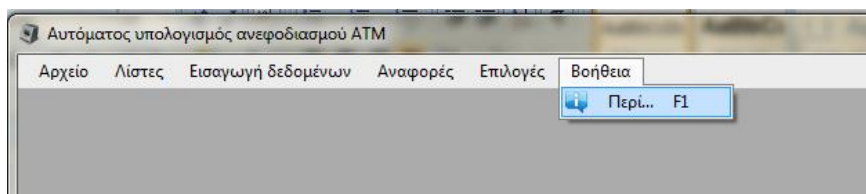
Εικόνα 12: Καρτέλα Αναφορές

Η καρτέλα «Επιλογές» δίνει τη δυνατότητα στο χρήστη να αλλάξει τη διάταξη των παραθύρων, να ενεργοποιήσει ή απενεργοποιήσει τη μπάρα πληροφοριών και έχει άμεση πρόσβαση στη καρτέλα των χρηστών καθώς και στη βάση δεδομένων.



Εικόνα 13: Καρτέλα Επιλογές

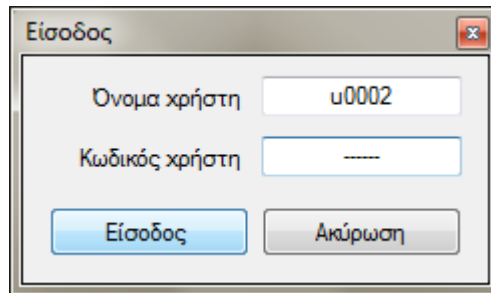
Τέλος η καρτέλα «Βοήθεια» περιέχει γενικές βοηθητικές πληροφορίες στους χρήστες και μπορεί να ενεργοποιηθεί από το πληκτρολόγιο με το F1



Εικόνα 14: Καρτέλα Βοήθεια

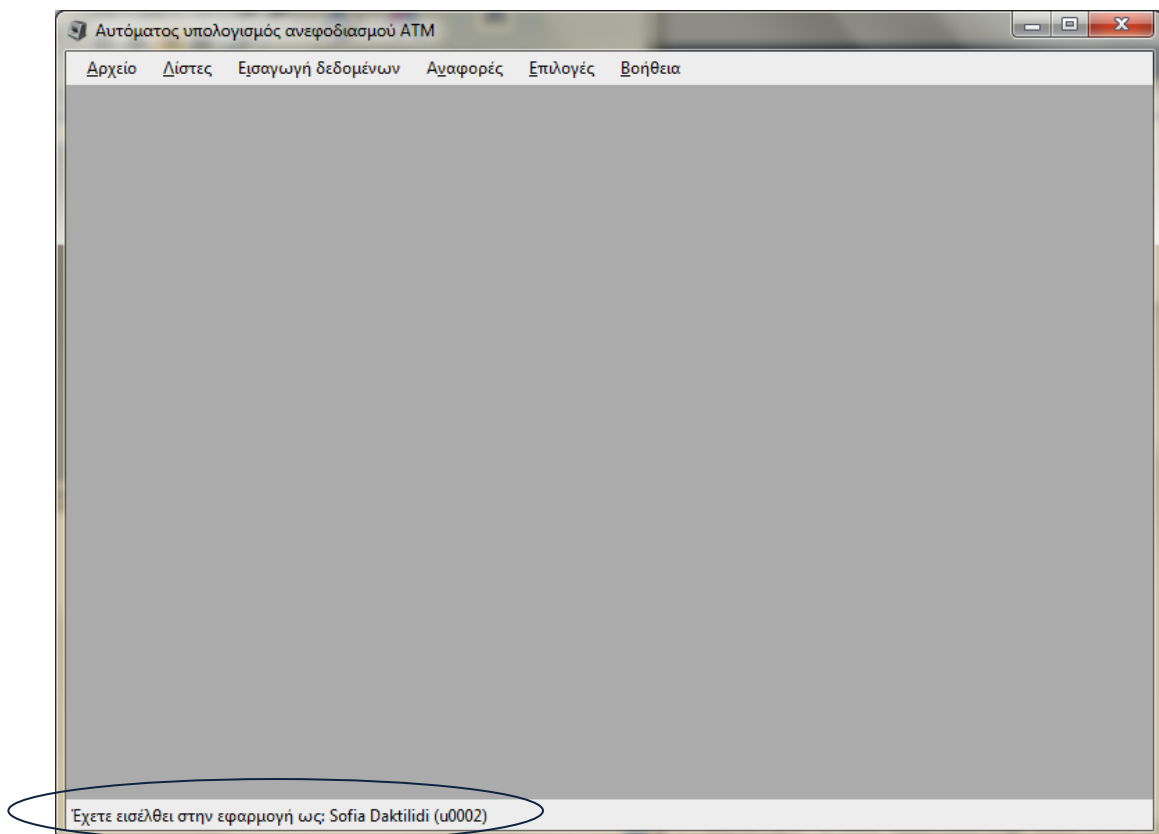
5.2 Αναλυτική περιγραφή κάθε καρτέλας

Από τη καρτέλα «Αρχείο» ενεργοποιούμε την επιλογή επανεκκίνηση. Πραγματοποιείται αυτόματα έξοδος και επανεκκίνηση του προγράμματος. Η νέα εισαγωγή στην εφαρμογή μπορεί να γίνει από τον ίδιο ή από άλλο χρήστη, αφού καταχωρηθούν τα σωστά στοιχεία. Για να κλείσουμε το πρόγραμμα επιλέγουμε έξοδος.



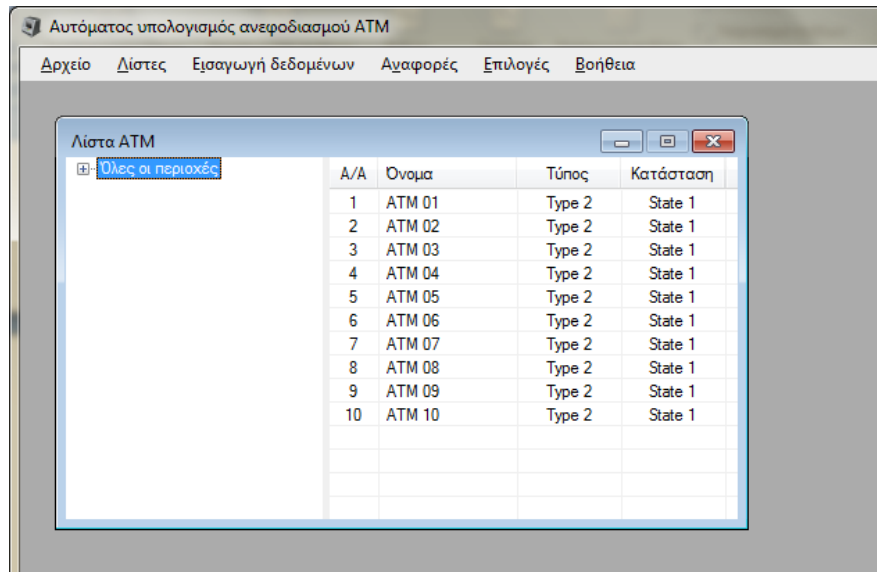
Εικόνα 15: Εισαγωγή νέου χρήστη

Στη μπάρα πληροφοριών εμφανίζονται τα στοιχεία του χρήστη (όνομα, επώνυμο, όνομα χρήστη) που συνδέθηκε στην εφαρμογή μετά την επανεκκίνηση.



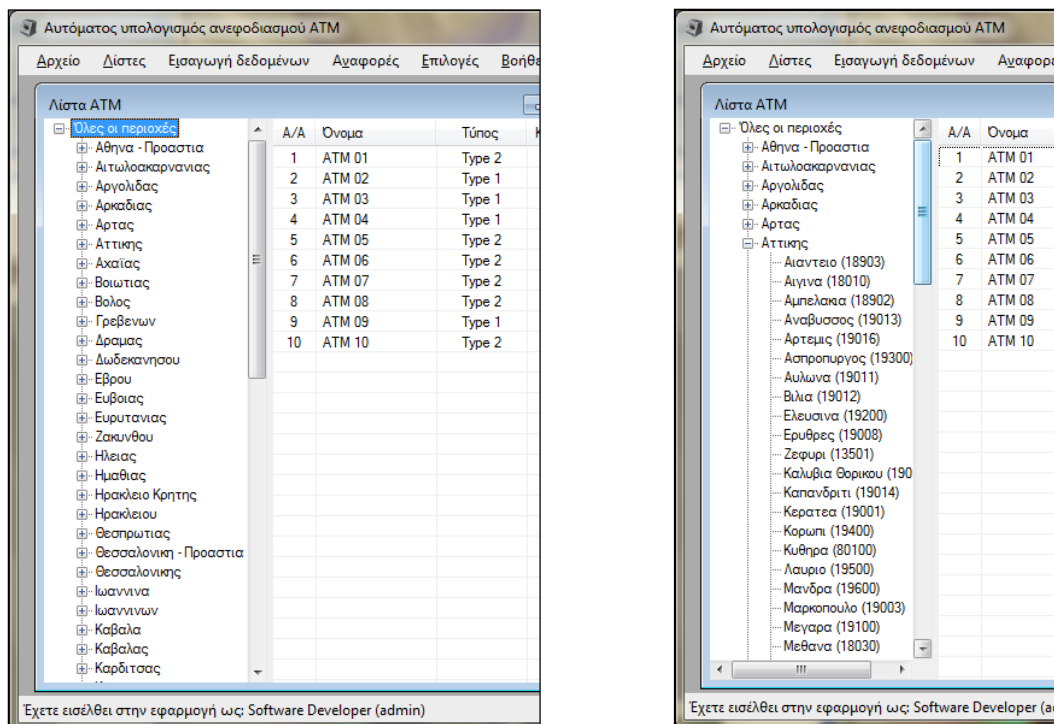
Εικόνα 16: Κεντρική Φόρμα

Από τη καρτέλα «Λίστες» επιλέγουμε το πίνακα ATM. Στο κέντρο της φόρμας εμφανίζονται όλα τα ATM που είναι καταχωρημένα στην εφαρμογή μας με τις δύο βασικές τους ιδιότητες τον τύπο και την κατάσταση.



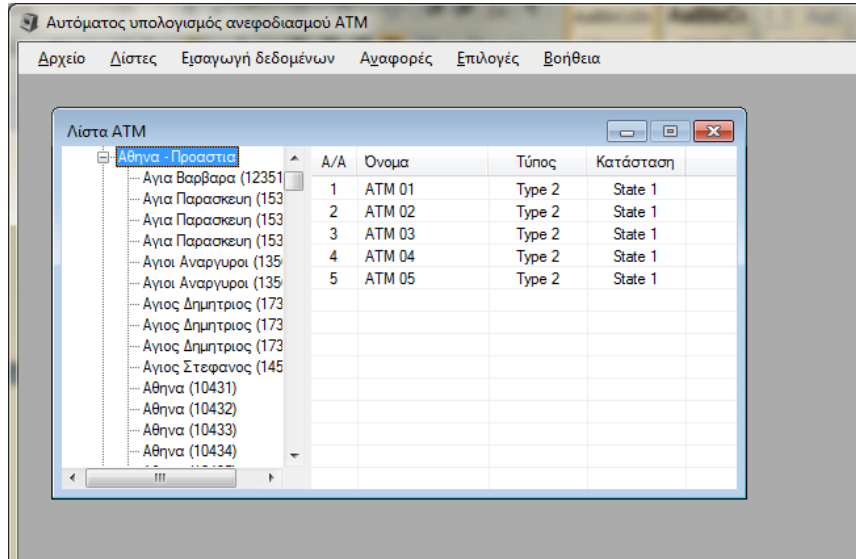
Εικόνα 17: Λίστα με τα ATM που είναι καταχωρημένα

Στο αριστερό μέρος εμφανίζονται όλες οι περιοχές της Ελλάδας ανά νομό. Κάνοντας κλικ πάνω σε κάποιο νομό εμφανίζονται οι περιοχές με τους αντίστοιχους ταχυδρομικούς κώδικες.



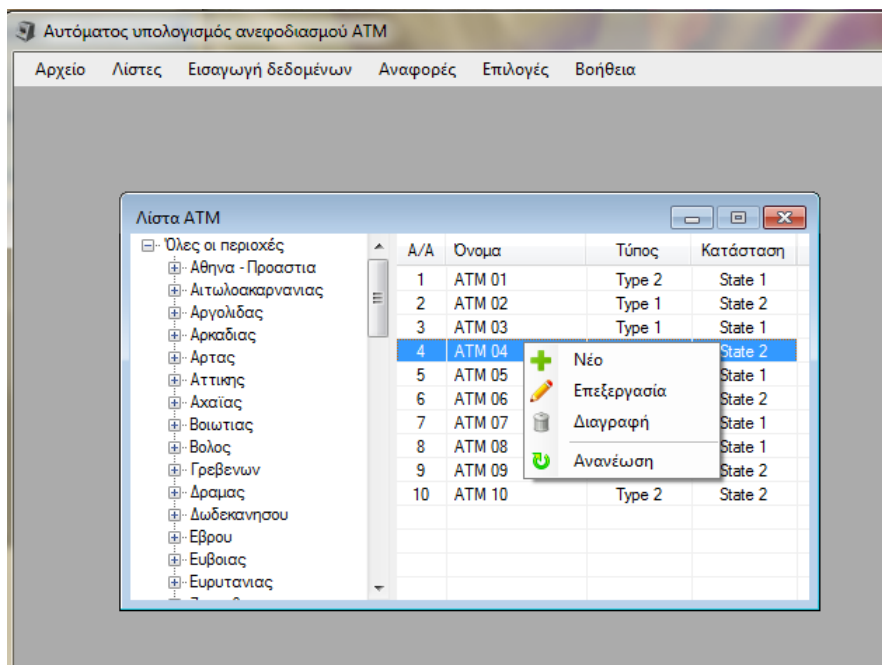
Εικόνα 18: Εμφάνιση όλων των πιθανών τοποθεσιών ATM στην Ελλάδα ανά νομό και στη συνέχεια ανά περιοχή με τον αντίστοιχο T.K.

Επιλέγοντας μία περιοχή με διπλό κλικ εμφανίζονται μόνο τα μηχανήματα που είναι τοποθετημένα στην επιλεγμένη περιοχή, με αυτό τον τρόπο έχουμε γεωγραφική ομαδοποίηση των μηχανημάτων.



Εικόνα 19: Εμφάνιση μόνο των ATM που βρίσκονται στα Προάστια της Αθήνας

Επιλέγοντας δεξί κλικ πάνω σε κάποιο ATM μπορούμε να επεξεργαστούμε τα δεδομένα του, να το διαγράψουμε, να πραγματοποιήσουμε απλή ανανέωση των στοιχείων ή να καταχωρήσουμε κάποιο νέο μηχάνημα. Την καρτέλα κάθε ATM μπορούμε να την καλέσουμε και με διπλό κλικ πάνω στην αντίστοιχη καταχώρηση.



Εικόνα 20: Με δεξί κλικ πάνω στις εγγραφές δίνονται οι απεικονιζόμενες επιλογές

Η καρτέλα κάθε ATM μας ενημερώνει για το όνομα του μηχανήματος, τη διεύθυνση του, τη περιοχή στην οποία είναι τοποθετημένο όπου την επιλέγουμε από τη λίστα, τον τύπο του (Type 1 ή Type 2), την κατάσταση του (State 1 ή State 2), το ελάχιστο όριο αναλήψεων δηλαδή τα λιγότερα χρήματα που μπορεί να έχει το ATM διαθέσιμα για αναλήψεις, το μέγιστο όριο καταθέσεων και το ποσό της τροφοδοσίας. Στη συγκεκριμένη καρτέλα μπορούμε να πραγματοποιήσουμε καθαρισμό των στοιχείων, αποθήκευση ή ακύρωση αλλαγών που πραγματοποιήσαμε.

Το κάθε μηχάνημα χαρακτηρίζεται είτε ως τύπος Type 1 όπου υποστηρίζει μόνο αναλήψεις είτε ως τύπος Type 2 όπου πραγματοποιεί αναλήψεις και καταθέσεις. Στη περίπτωση που το ATM είναι τύπου Type 1 τα ποσά των καταθέσεων θα είναι πάντα μηδέν.

The screenshot shows a form titled "Προσθήκη ATM" with the following fields and values:

A/A	1			
Όνομα	ATM 01	Κατάσταση	State 1	
Διεύθυνση	Address 01	Ελ. όριο αναλ.	15000	
Περιοχή	Αθήνα - Προαστ.	Μέγ. όριο κατάθ.	5000	
Τύπος	Type 2	Ποσό τροφοδο.	100000	

Buttons at the bottom: Καθαρισμός, Αποθήκευση, Ακύρωση

Εικόνα 21:Καρτέλα στοιχείων ATM

Κάθε μηχάνημα μπορεί να είναι σε λειτουργία δηλαδή κατάσταση ATM είναι State1 ή προσωρινά εκτός λειτουργίας δηλαδή κατάσταση ATM είναι State2 λόγω κάποιος πιθανής τεχνικής βλάβης

The screenshot shows the same form as in Figure 21, but with the "Τύπος" dropdown menu open, showing the following options:

- Type 1
- Type 2

Buttons at the bottom: Καθαρισμός, Αποθήκευση, Ακύρωση

Εικόνα 22: Η εφαρμογή υποστηρίζει δύο τύπους ATM

The screenshot shows the same form as in Figure 21, but with a list of available areas displayed in the "Περιοχή" field. The list includes:

- Αθήνα - Προαστια (10439)
- Αθήνα - Προαστια (10440)
- Αθήνα - Προαστια (10441)
- Αθήνα - Προαστια (10442)
- Αθήνα - Προαστια (10443)
- Αθήνα - Προαστια (10444)
- Αθήνα - Προαστια (10445)
- Αθήνα - Προαστια (10446)
- Αθήνα - Προαστια (10447)
- Αθήνα - Προαστια (10551)
- Αθήνα - Προαστια (10552)
- Αθήνα - Προαστια (10553)

Buttons at the bottom: Καθαρισμός, Αποθήκευση, Ακύρωση

Εικόνα 23:Η περιοχή κάθε ATM πρέπει να επιλεγεί από την υπάρχουσα λίστα

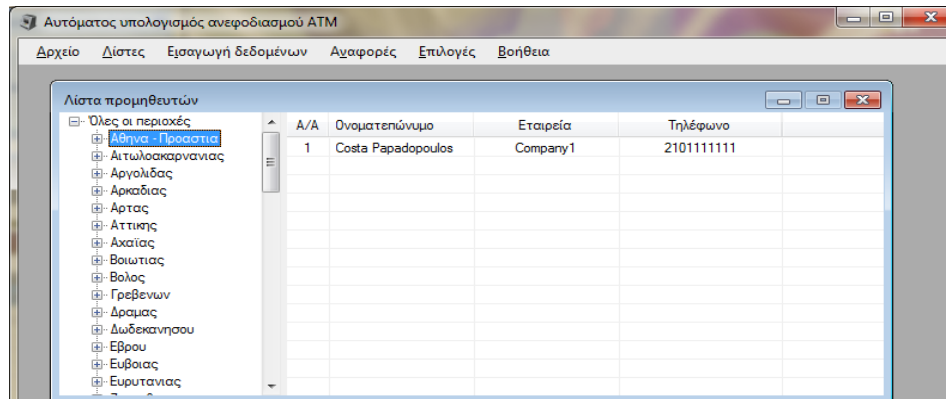
Εικόνα 22:Καθε ATM μπορεί να είναι ενεργό ή όχι

Από τη καρτέλα «Λίστες» επιλέγουμε το πίνακα Λίστα προμηθευτών. Στο κέντρο της φόρμας εμφανίζονται όλοι οι συνεργάτες του τμήματος που είναι καταχωρημένοι στην εφαρμογή μας με τις δύο βασικές τους ιδιότητες την εταιρεία στην οποία εργάζονται και το τηλεφωνικό αριθμό τους. Στο αριστερό μέρος εμφανίζονται όλες οι περιοχές της Ελλάδας ανά νομό. Κάνοντας κλικ πάνω σε κάποιο νομό εμφανίζονται οι περιοχές με τους αντίστοιχους ταχυδρομικούς κώδικες.

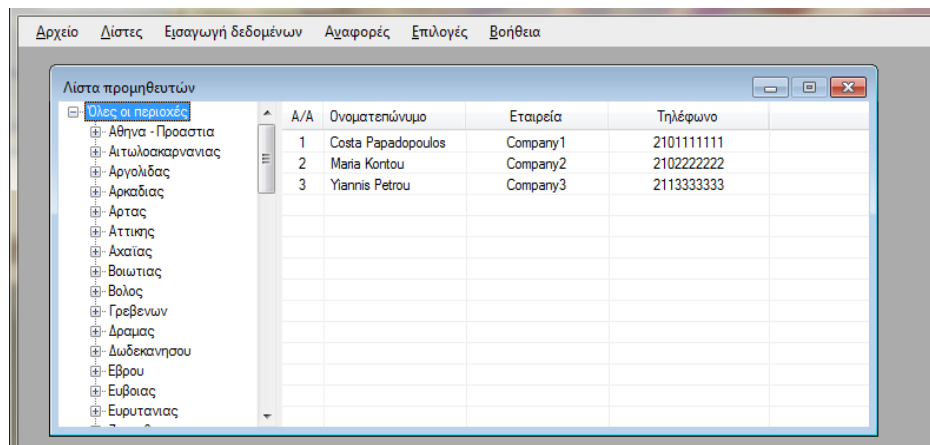
A/A	Όνοματεπώνυμο	Εταιρεία	Τηλέφωνο
1	Costa Papadopoulos	Company1	2101111111
2	Maria Kontou	Company2	2102222222
3	Yannis Petrou	Company3	2113333333

Εικόνα 23:Λίστα προμηθευτών

Επιλέγοντας κάποιο νομό, στη κεντρική φόρμα εμφανίζονται μόνο οι συνεργάτες της συγκεκριμένης περιοχής ενώ για να δούμε τη συγκεντρωτική λίστα θα πρέπει να επιλέξουμε «Όλες τις περιοχές» από την αριστερή στήλη.

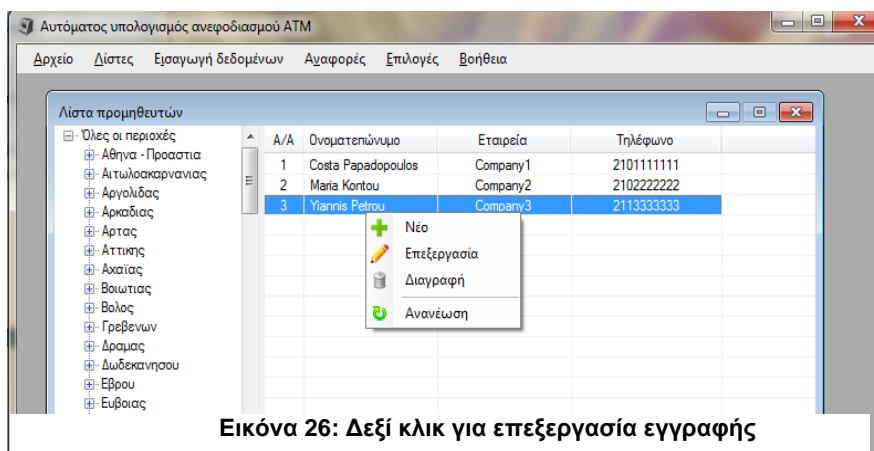


Εικόνα 24: Συνεργάτες που εργάζονται στα προάστια της Αθήνας



Εικόνα 25: Συνεργάτες από όλες τις περιοχές της Ελλάδας

Επιλέγοντας δεξί κλικ πάνω σε κάποια εγγραφή μπορούμε να επεξεργαστούμε τα δεδομένα της, να τη διαγράψουμε, να πραγματοποιήσουμε απλή ανανέωση των στοιχείων ή να καταχωρήσουμε κάποιο νέο συνεργάτη.

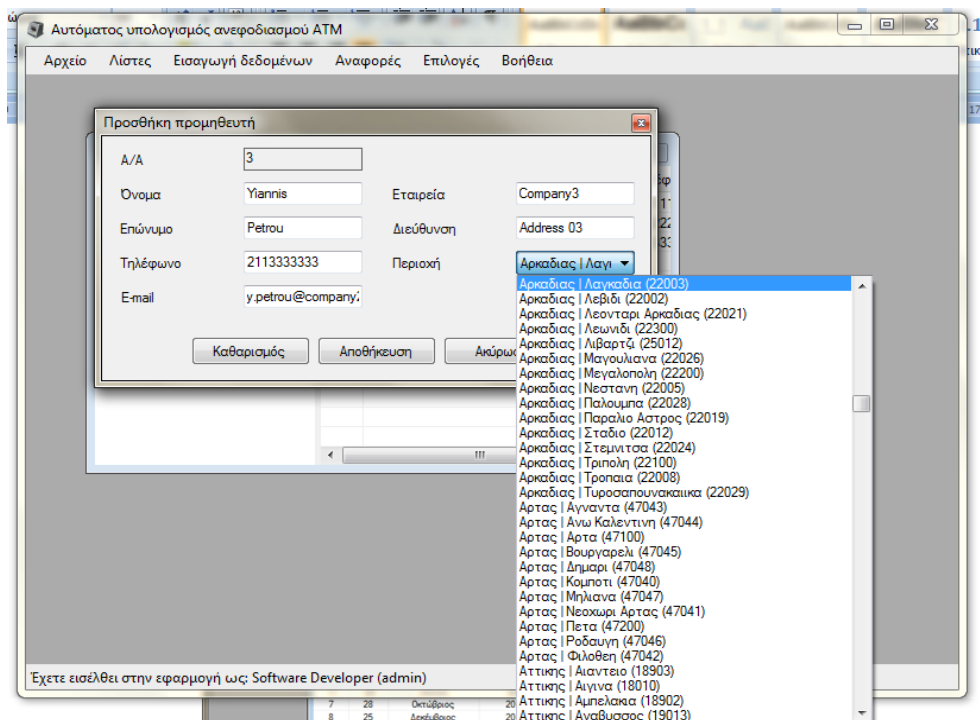


Εικόνα 26: Δεξί κλικ για επεξεργασία εγγραφής

Την καρτέλα κάθε προμηθευτή μπορούμε να την ενεργοποιήσουμε με διπλό κλικ πάνω στην αντίστοιχη καταχώρηση. Η καρτέλα κάθε προμηθευτή μας ενημερώνει για το ονοματεπώνυμο του συνεργάτη μας, το τηλέφωνο του, το email του, την εταιρεία στην οποία εργάζεται, τη διεύθυνση και τη περιοχή του γραφείου του. Στη συγκεκριμένη καρτέλα μπορούμε να πραγματοποιήσουμε καθαρισμό των στοιχείων, αποθήκευση ή ακύρωση αλλαγών που πραγματοποιήσαμε.

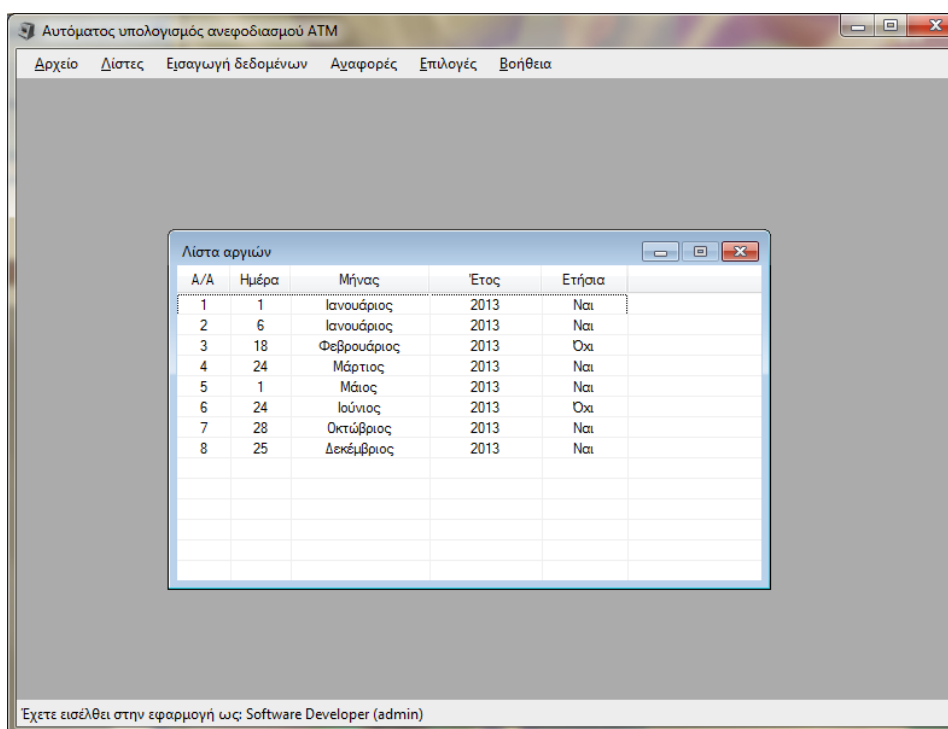
Εικόνα 27:Καρτέλα προμηθευτών

Την περιοχή την επιλέγουμε από τη λίστα που εμφανίζεται και αναφέρει το νομό, τη περιοχή και τον αντίστοιχο ταχυδρομικό κώδικα. Με αυτό τον τρόπο έχουμε πετύχει τη σωστή γεωγραφική ομαδοποίηση των συνεργατών καθώς αποκλείσαμε το ενδεχόμενο της διαφορετικής πληκτρολόγησης της ίδιας περιοχής ή την καταχώρηση λάθους ταχυδρομικού κώδικα.



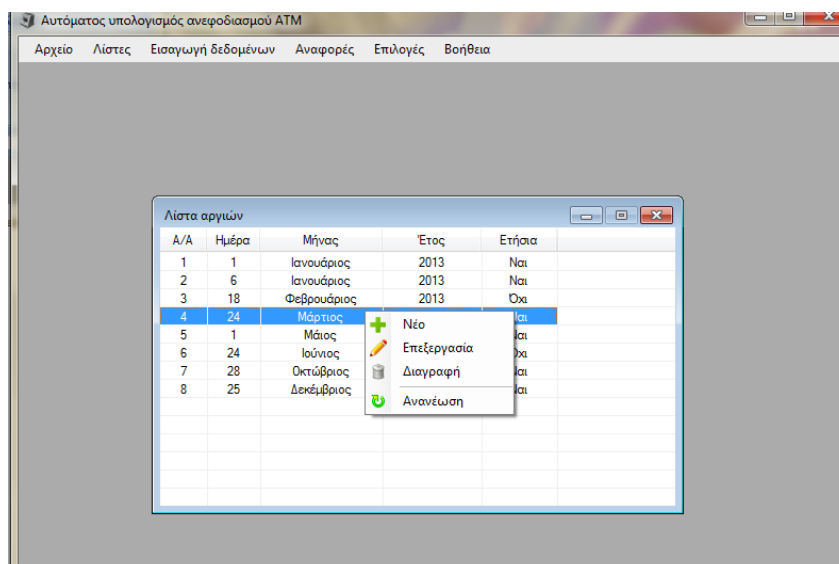
Εικόνα 28:Επιλογή περιοχή από τη προεπιλεγμένη λίστα

Από τη καρτέλα «Λίστες» επιλέγουμε το πίνακα λίστα αργιών. Στο κέντρο της φόρμας εμφανίζονται όλες οι αργίες (ημέρα και μήνας), το έτος στο οποίο αναφέρονται καθώς και αν είναι ετήσιες ή όχι.



Εικόνα 29: Καρτέλα αργιών

Ακολουθώντας την ίδια λογική επεξεργασίας με δεξί κλικ πάνω σε κάποια εγγραφή μπορούμε να πραγματοποιήσουμε επεξεργασία, διαγραφή, ανανέωση ή καταχώρηση νέας αργίας.



Εικόνα 30: Δεξί κλικ για επεξεργασία εγγραφής

Στη καρτέλα με τις αργίες διαλέγουμε την ημέρα, το μήνα, το έτος καθώς και αν η συγκεκριμένη αργία επαναλαμβάνεται τα επόμενα ετη με τα ίδια χαρακτηριστικά (μέρα και μήνα). Με αυτό το τροπο χωρίζουμε τις επίσημες (π.χ. 28^η Οκτωμβρίου) από τις κινητες ή τοπικές αργίες (π.χ. Πάσχα). Στη καρτέλα μπορούμε να πραγματοποιήσουμε καθαρισμό των στοιχείων, αποθήκευση ή ακύρωση αλλαγών που πραγματοποιήσαμε.

Εικόνα 31: Καρτέλα αργιών

Εικόνα 33: Επιλογή ημέρας

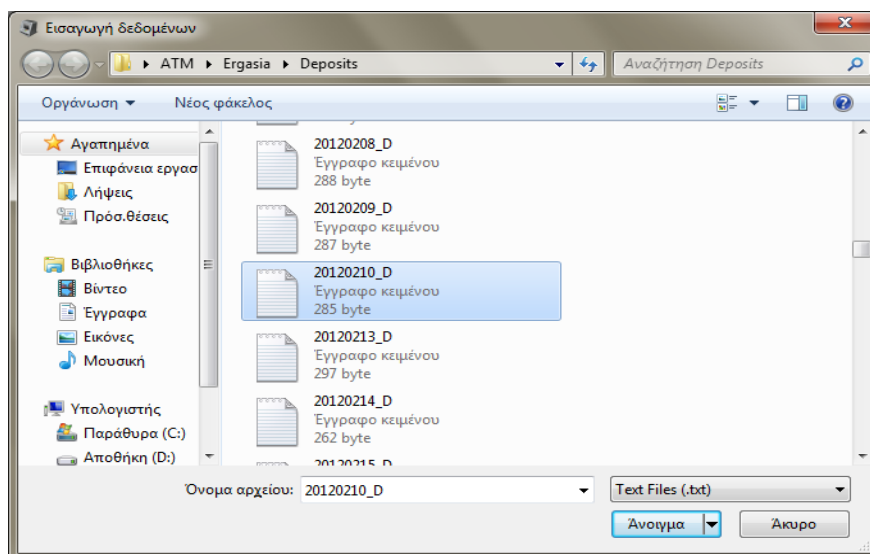
Εικόνα 32: Επιλογή μήνα

Εικόνα 34: Επιλογή έτους

Εικόνα 35: Χαρακτηρισμός ετήσιας

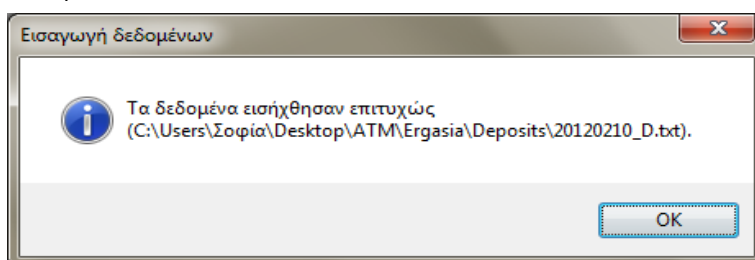
Από την καρτέλα «Εισαγωγή Δεδομένων» μπορούμε καθημερινά να ενημερώσουμε η βάση μας σχετικά με τις συναλλαγές (ποσό και αριθμό συναλλαγών) που πραγματοποιήθηκαν την προηγούμενη εργάσιμη μέρα. Επιλέγοντας καταθέσεις ανοίγει ένας καθοδηγητής όπου μας βοηθάει να επιλέξουμε το αρχείο των καταθέσεων από τη θέση στην οποία το έχουμε σώσει και στο τέλος επιλέγουμε άνοιγμα. Με αυτό τον τρόπο μπορούμε να εντοπίσουμε το αρχείο σε όποιο σημείο (φάκελο) το έχουμε σώσει, χωρίς να έχουμε προκαθορισμένη τοποθεσία αναφοράς. Με αυτό τον

τρόπο κάθε χρήστης μπορεί να σώζει τα αρχεία σε διαφορετικό φάκελο χωρίς να επηρεάζει την εφαρμογή.



Εικόνα 36: Εντοπισμός αρχείων με τα δεδομένα των συναλλαγών (καταθέσεις)

Όταν τα δεδομένα προστεθούν επιτυχώς στη βάση μας, η εφαρμογή μας ενημερώνει με σχετικό μήνυμα στο οποίο αναφέρεται η τοποθεσία και το όνομα του αρχείου από το οποίο χρησιμοποίησε τα δεδομένα.



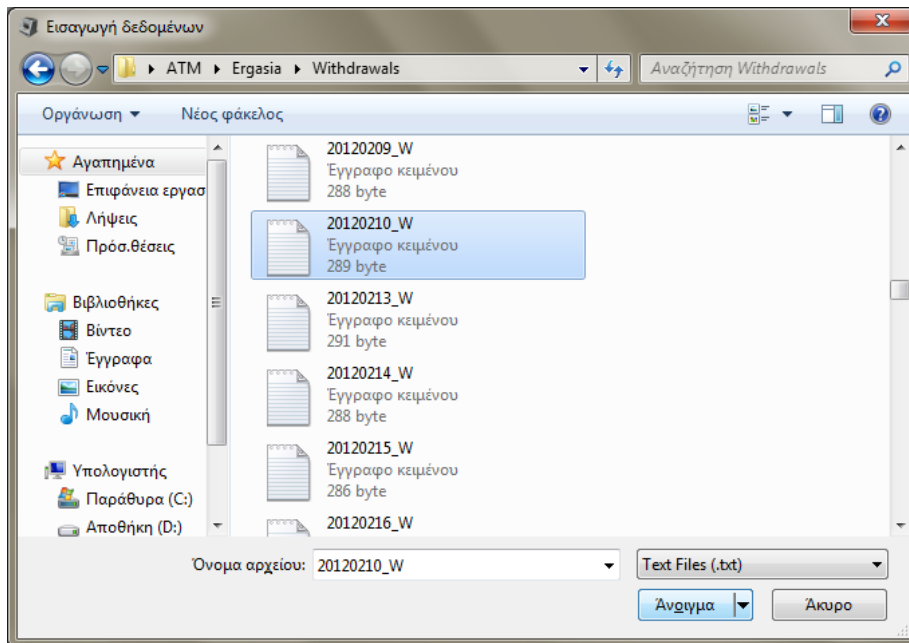
Εικόνα 37: Μήνυμα επιτυχούς καταχώρησης των δεδομένων

Το αρχείο με τα δεδομένα των συναλλαγών έχει προκαθορισμένη μορφή καθώς παράγεται από σύστημα του χρηματοπιστωτικού ιδρύματος. Καθημερινά δημιουργείται ένα αρχείο τύπου txt για τις αναλήψεις και ένα για τις καταθέσεις, όπου περιέχει την ημερομηνία, τον αριθμό του ATM, το συνολικό ποσό των συναλλαγών και το συνολικό αριθμό των συναλλαγών.

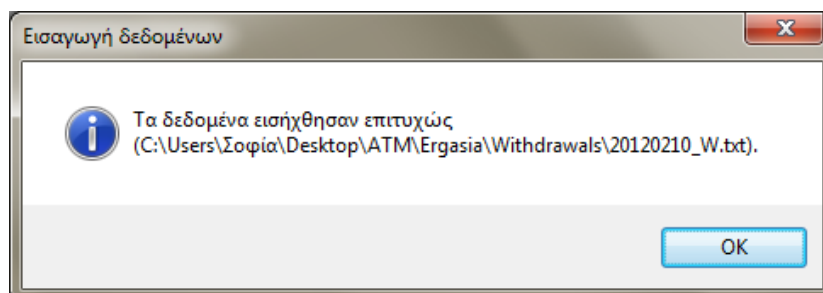
Αρχείο	Επεξεργασία	Μορφή	Προβολή	Βοήθεια
02/01/2012	:	1	:	19330 : 66
02/01/2012	:	2	:	19300 : 80
02/01/2012	:	3	:	16400 : 59
02/01/2012	:	4	:	7800 : 30
02/01/2012	:	5	:	20400 : 70
02/01/2012	:	6	:	35100 : 97
02/01/2012	:	7	:	18500 : 62
02/01/2012	:	8	:	30600 : 111
02/01/2012	:	9	:	40400 : 165
02/01/2012	:	10	:	10900 : 52

Εικόνα 38: Πρότυπο αρχείο συναλλαγών (καταθέσεις)

Με τον ίδιο τρόπο εντοπίσουμε και εισάγουμε τα δεδομένα για τις αναλήψεις στη βάση μας, επιλέγοντας αυτή τη φορά «Αναλήψεις ATM». Το όνομα κάθε αρχείου αποτελείται από την ημερομηνία των συναλλαγών και “W” για αναλήψεις και “D” για καταθέσεις.



Εικόνα 39:Εντοπισμός αρχείων με τα δεδομένα των συναλλαγών (αναλήψεων)



Εικόνα 41: Μήνυμα επιτυχούς καταχώρησης των δεδομένων

Αρχείο	Επεξεργασία	Μορφή	Προβολή	Βοήθεια
02/01/2012	:	1	:	19330 : 66
02/01/2012	:	2	:	20.100 : 63
02/01/2012	:	3	:	32500 : 100
02/01/2012	:	4	:	42300 : 120
02/01/2012	:	5	:	34800 : 125
02/01/2012	:	6	:	22100 : 95
02/01/2012	:	7	:	13000 : 50
02/01/2012	:	8	:	4000 : 19
02/01/2012	:	9	:	8700 : 26
02/01/2012	:	10	:	13000 : 40

Εικόνα 42: Πρότυπο αρχείο συναλλαγών (αναλήψεων)

Στην καρτέλα «Αναφορές» εμφανίζονται τα μηχανήματα ATM όπου η εφαρμογή προτείνει τροφοδοσία για την αυριανή μέρα. Ο χρήστης έχει στη διάθεση του αρκετές πληροφορίες ώστε να πάρει την τελική απόφαση για την πραγματοποίηση ή όχι της τροφοδοσίας. Εμφανίζεται το όνομα του μηχανήματος, αθροιστικά οι αναλήψεις που έχουν πραγματοποιηθεί από την τελευταία τροφοδοσία μέχρι τώρα (στοιχεία που προκύπτουν από τα αρχεία των αναλήψεων), το ποσό του τελευταίου ανεφοδιασμού (ποσό το οποίο αναφέρεται στη καρτέλα του κάθε ATM), αθροιστικά το ποσό των καταθέσεων από την τελευταία τροφοδοσία μέχρι τώρα (στοιχεία που προκύπτουν από τα αρχεία των καταθέσεων), το επιθυμητό ελάχιστο όριο καταθέσεων (ποσό το οποίο αναφέρεται στη καρτέλα κάθε ATM) και στο τέλος η πρόβλεψη των αναλήψεων και των καταθέσεων για την επιθυμητή περίοδο, τρέχουσα μέρα ή μέρες (πρόβλεψη σαββατοκύριακου ή προσεχών αργιών).

A/A	ATM	Αναλήψεις	Ποσό ανεφ.	Καταθέσεις	Όριο κατάθ.	Πρόβλεψη αν.	Πρόβλεψη κατ.
1	ATM 01	25600	100000	12800	5000	9955	10857
2	ATM 02	63000	120000	33200	45000	19602	12762
4	ATM 04	181600	200000	64000	50000	15034	6912
5	ATM 05	60900	90000	49400	40000	17912	20697
6	ATM 06	82000	110000	45300	55000	21562	14580
8	ATM 08	23100	100000	44000	50000	3894	12747

Εικόνα 40: Αναφορά προτεινόμενων τροφοδοσιών

Καθημερινά εκτός από τα αρχεία σχετικά με τα ποσά των αναλήψεων και των καταθέσεων που πραγματοποιήθηκαν την προηγούμενη εργάσιμη μέρα ή περίοδο, λαμβάνουμε από τους συνεργάτες του τμήματος μία επιβεβαίωση για το εάν οι τροφοδοσίες που προγραμματίσαμε την προηγούμενη εργάσιμη μέρα, ολοκληρώθηκαν με επιτυχία σήμερα ώστε να τις εξαιρέσουμε από τη λίστα και στη συνέχεια να δώσουμε νέες εντολές τροφοδοσίας με ημερομηνία εκτέλεσης την επόμενη εργάσιμη μέρα. Σε περίπτωση που κάποιο μηχάνημα δεν τροφοδοτήθηκε (π.χ. πραγματοποίηση πορείας στη περιοχή) απλά δεν το επιλέγουμε στην αναφορά και στέλνουμε νέα εντολή ανεφοδιασμού, γνωρίζοντας όπως την κατάσταση στην οποία βρίσκεται αυτή τη στιγμή (ποσά αναλήψεων και καταθέσεων). Επιπλέον αν κρίνουμε αναγκαία τη επείγουσα πραγματοποίηση τροφοδοσίας ενός ATM την τρέχουσα ημέρα και όχι την επόμενη, απλά στέλνουμε εντολή έκτακτης τροφοδοσίας και αφού επιβεβαιώσουμε την ολοκλήρωση της ενημερώνουμε την εφαρμογή. Με αυτό τον τρόπο η εφαρμογή δεν επηρεάζεται από τις ακυρωμένες ή τις έκτακτες τροφοδοσίες.

Μέσω ενός παραδείγματος θα εξηγήσουμε τον τρόπο λειτουργίας του προγράμματος. Υποθέσουμε ότι εμφανίζεται ο παρακάτω πίνακας όπου η εφαρμογή μας προτείνει για τροφοδοσία τα ATM με αριθμό 1,2,3,4,5,6,8,9 και 10. Την προηγούμενη εργάσιμη μέρα είχαμε δώσει εντολή τροφοδοσία για τα ATM με αριθμό 4,5 και 6 και σήμερα λάβαμε επιβεβαίωση από τους συνεργάτες μας ότι οι προγραμματισμένες τροφοδοσίες ολοκληρώθηκαν με επιτυχία.

Αυτόματος υπολογισμός ανεφοδιασμού ATM

Αρχείο Λίστες Εισαγωγή δεδομένων Αναφορές Επιλογές Βοήθεια

Αναφορές

A/A	ATM	Αναλήψεις	Ποσό ανεφ.	Καταθέσεις	Όριο κατάθ.	Πρόβλεψη αν.	Πρόβλεψη κατ.
<input type="checkbox"/>	1 ATM 01	0	100000	0	5000	11444	11701
<input type="checkbox"/>	2 ATM 02	78600	120000	51250	45000	16530	13090
<input type="checkbox"/>	3 ATM 03	87800	100000	33900	45000	14219	8349
<input type="checkbox"/>	4 ATM 04	227300	200000	80100	50000	18487	7870
<input type="checkbox"/>	5 ATM 05	110900	90000	85300	40000	20595	24754
<input type="checkbox"/>	6 ATM 06	99200	110000	88800	55000	18182	19250
<input type="checkbox"/>	8 ATM 08	26200	100000	85100	50000	3292	17762
<input type="checkbox"/>	9 ATM 09	42600	90000	63300	54000	5149	13677
<input type="checkbox"/>	10 ATM 10	33500	140000	40100	45000	5684	10402

Εικόνα 41: Προτεινόμενα ATM για τροφοδοσία

Επομένως προχωράμε στην ενημέρωση της βάσης μας, μαρκάροντας τα συγκεκριμένα ATM με αριθμό 4,5 και 6 που έχουν τροφοδοτηθεί και με δεξί κλικ επιλέγουμε τροφοδοσία και στη συνέχεια ανανέωση.

Αυτόματος υπολογισμός ανεφοδιασμού ATM

Αρχείο Λίστες Εισαγωγή δεδομένων Αναφορές Επιλογές Βοήθεια

Αναφορές

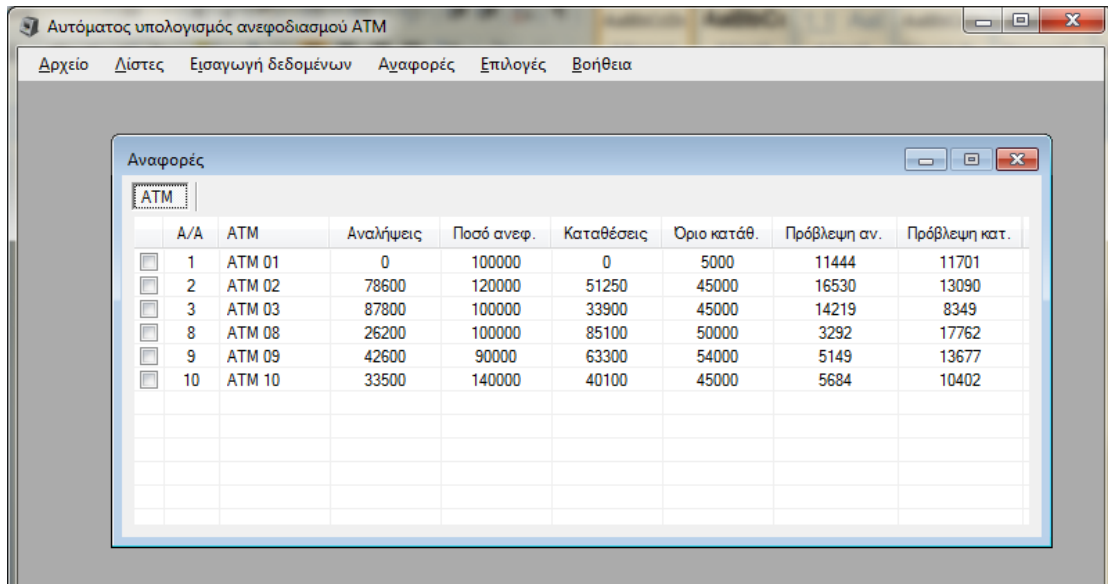
A/A	ATM	Αναλήψεις	Ποσό ανεφ.	Καταθέσεις	Όριο κατάθ.	Πρόβλεψη αν.	Πρόβλεψη κατ.
<input checked="" type="checkbox"/>	1 ATM 01	53100	100000	32420	5000	11444	11701
<input type="checkbox"/>	2 ATM 02			51250	45000	16530	13090
<input type="checkbox"/>	3 ATM 03			33900	45000	14219	8349
<input type="checkbox"/>	4 ATM 04			80100	50000	18487	7870

Αναφορές

- Ανεφοδιασμός
- Ανανέωση

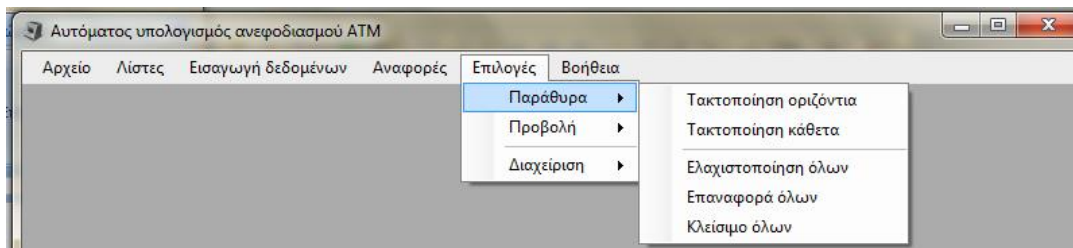
Εικόνα 42: Ενημέρωση βάσης

Ο νέος πίνακας που εμφανίζεται δεν περιέχει τα συγκεκριμένα ATM καθώς δεν χρειάζεται να προχωρήσουμε σε νέα τροφοδοσία και την επόμενη μέρα. Στην αναφορά παραμένουν μόνο εκείνα τα ATM για τα οποία δεν έχει πραγματοποιηθεί τροφοδοσίας αλλά με βάση τους υπολογισμούς της εφαρμογής χρειάζονται. Σε αυτό το σημείο ο χρήστης έχει στη διάθεση του τα μηχανήματα με τις απαραίτητες πληροφορίες και μπορεί να προχωρήσει στην απόφαση για ποια ή για όλα τα ATM θα προχωρήσει σε τροφοδοσία την επομένη εργάσιμη μέρα. Τα προτεινόμενα ATM είναι 1,2,3,8,9 και 10 για τα οποία όμως δε θα προχωρήσουμε σε καμία αλλαγή της κατάστασης τους μέχρι να λάβουμε επιβεβαίωση της ολοκλήρωσης της τροφοδοσίας από το αντίστοιχο τμήμα, πληροφορία την οποία αναμένουμε την επομένη εργάσιμη μέρα και μέχρι τότε τα προαναφερθέντα μηχανήματα θα εμφανίζονται σε κατάσταση «προτεινόμενα» για ανεφοδιασμό. Την επομένη μέρα περιμένουμε στη μία συγκεντρωτική αναφορά από την εφαρμογή στην οποία θα περιέχονται και τα προαναφερθέντα μηχανήματα συν τα καινούργια με πλάνο τροφοδοσίας την μεθεπόμενη μέρα.

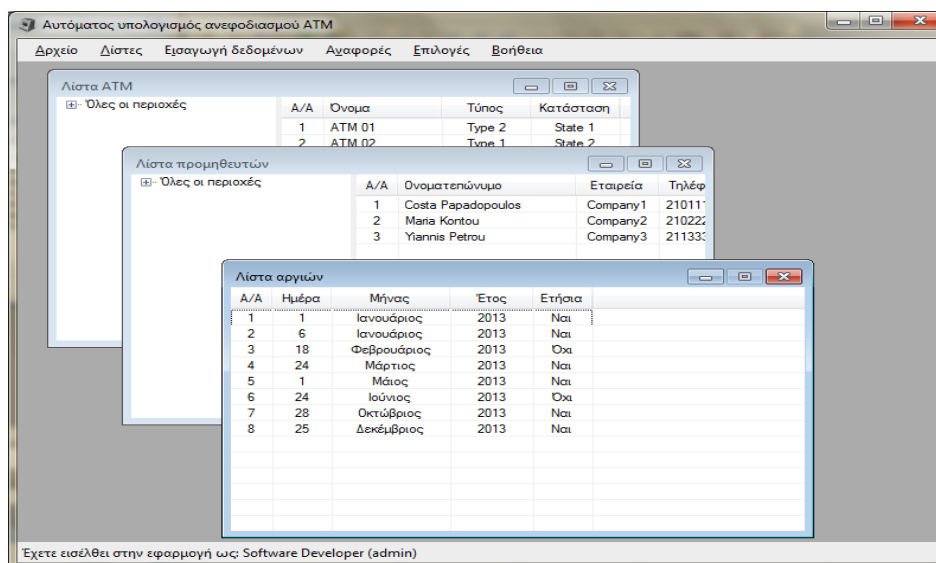


Εικόνα 43: Αναφορά μετά την ενημέρωση της εφαρμογής

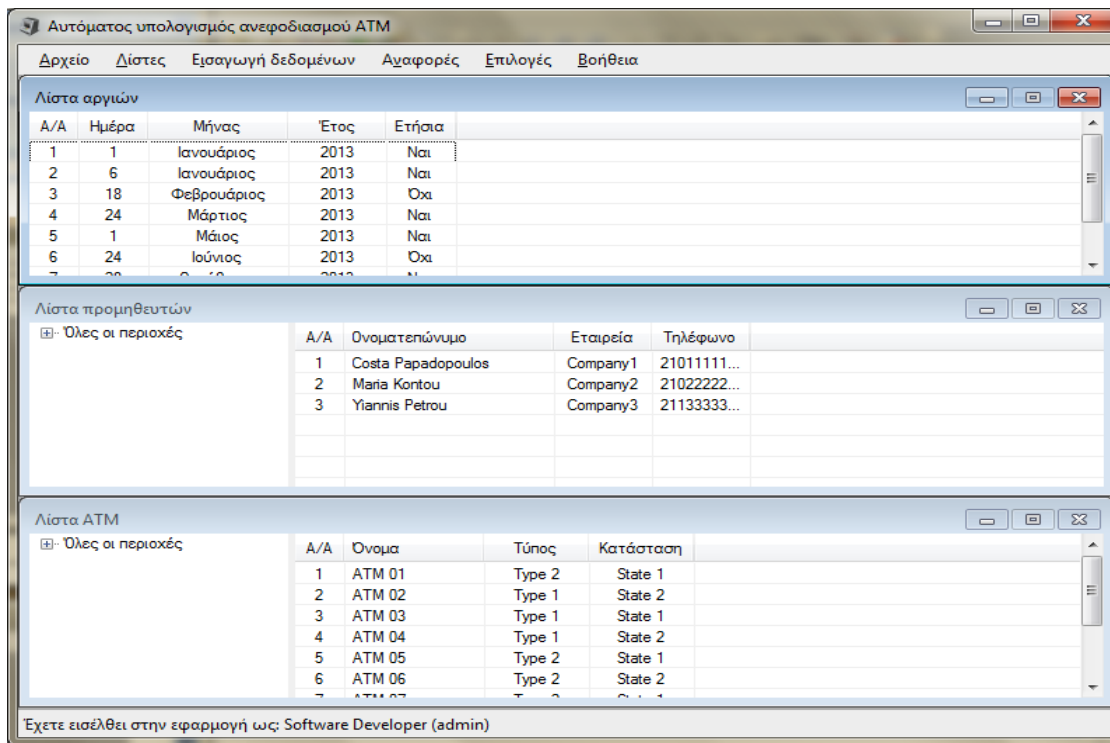
Στη καρτέλα «επιλογές» ο χρήστης έχει τη δυνατότητα να αλλάξει τον τρόπο εμφάνισης των ενεργών παραθύρων. Η εμφάνιση τους μπορεί να γίνει είτε οριζόντια είτε κάθετα, επιπλέον μπορεί να πραγματοποιήσει μαζική ελαχιστοποίηση ή επαναφορά ή κλείσιμο των παραθύρων.



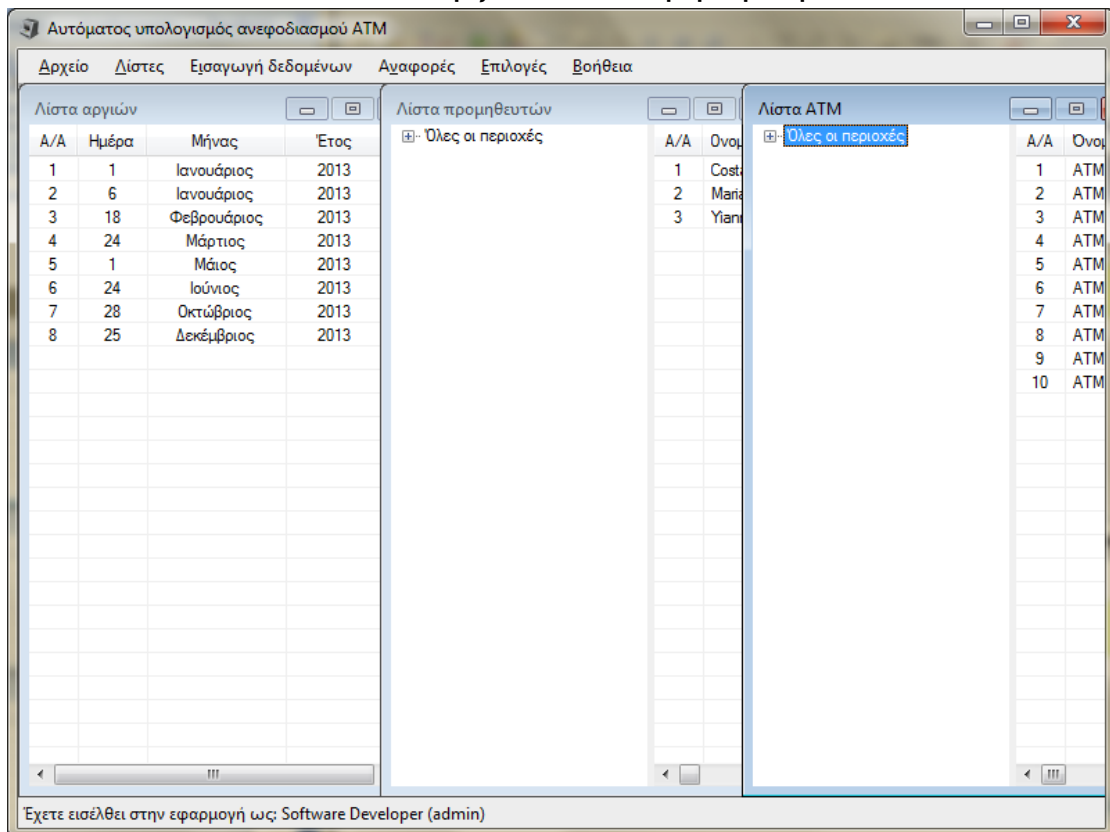
Εικόνα 44: Διαχείριση παραθύρων



Εικόνα 45: Ενεργά παράθυρα



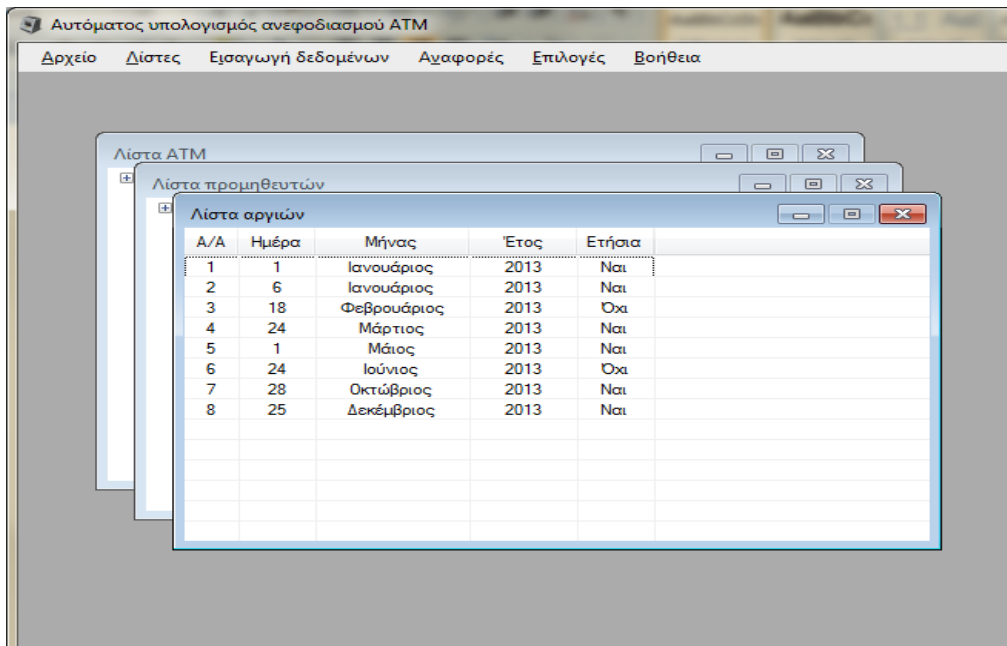
Εικόνα 46: Οριζόντια τακτοποίηση παραθύρων



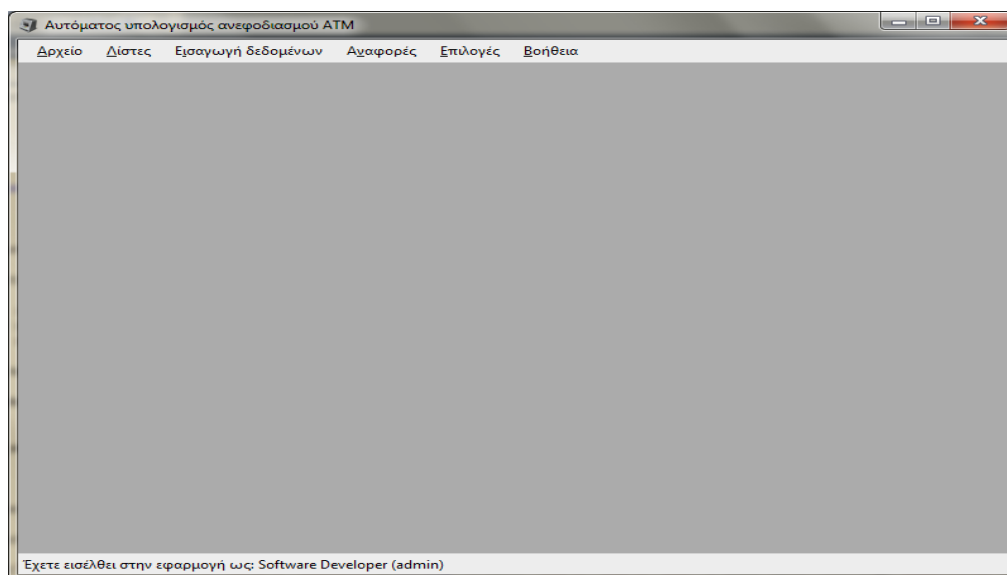
Εικόνα 47: Κάθετη τακτοποίηση παραθύρων



Εικόνα 50: Ελαχιστοποίηση ενεργών παραθύρων

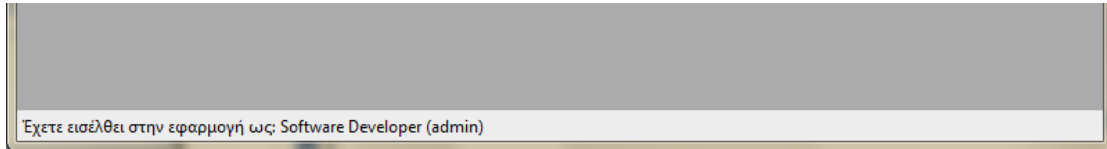


Εικόνα 48: Επαναφορά ενεργών παραθύρων

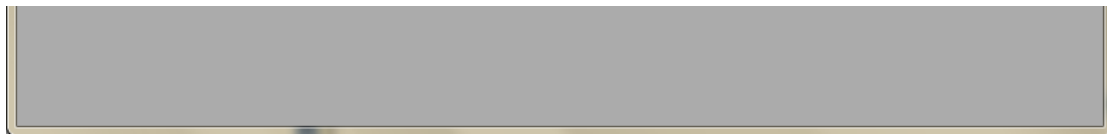


Εικόνα 49: Μαζικό κλείσιμο ενεργών παραθύρων

Από τη επιλογή «Προβολή» η εφαρμογή δίνει στο χρήστη τη δυνατότητα αν ενεργοποιήσει ή να απενεργοποιήσει τη μπάρα πληροφοριών.

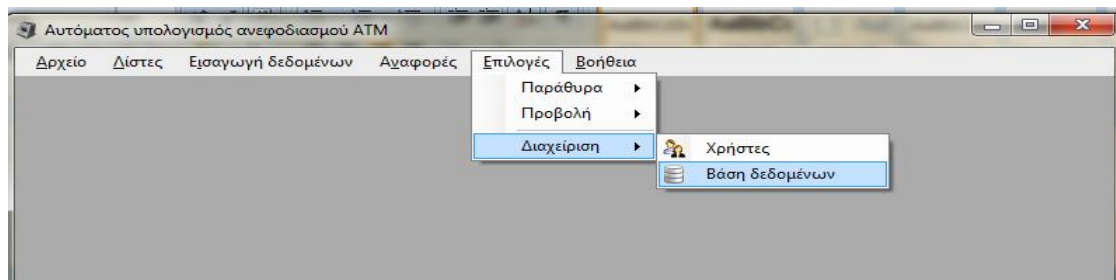


Εικόνα 51: Ενεργοποιημένη μπάρα πληροφοριών



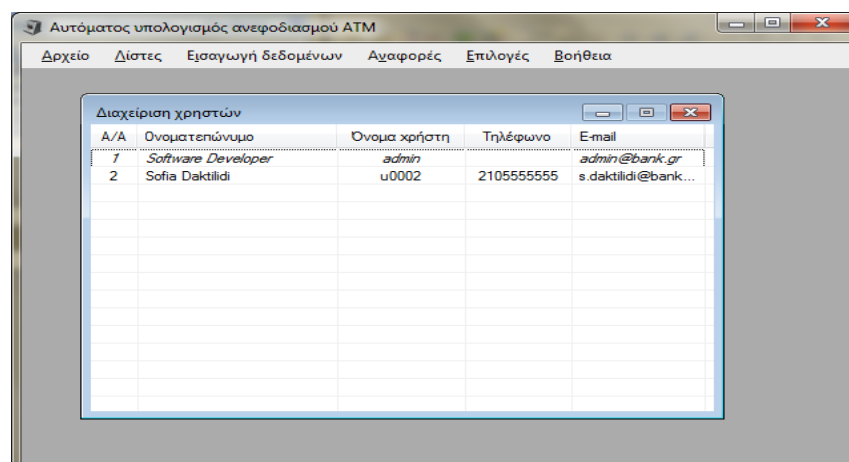
Εικόνα 52: Απενεργοποιημένη μπάρα πληροφοριών

Από τη καρτέλα Επιλογές, Διαχειριστής μπορούμε να ενημερωθούμε για τους χρήστες της και λεπτομέρειες σχετικά με τη βάση δεδομένων που χρησιμοποιεί η εφαρμογή.



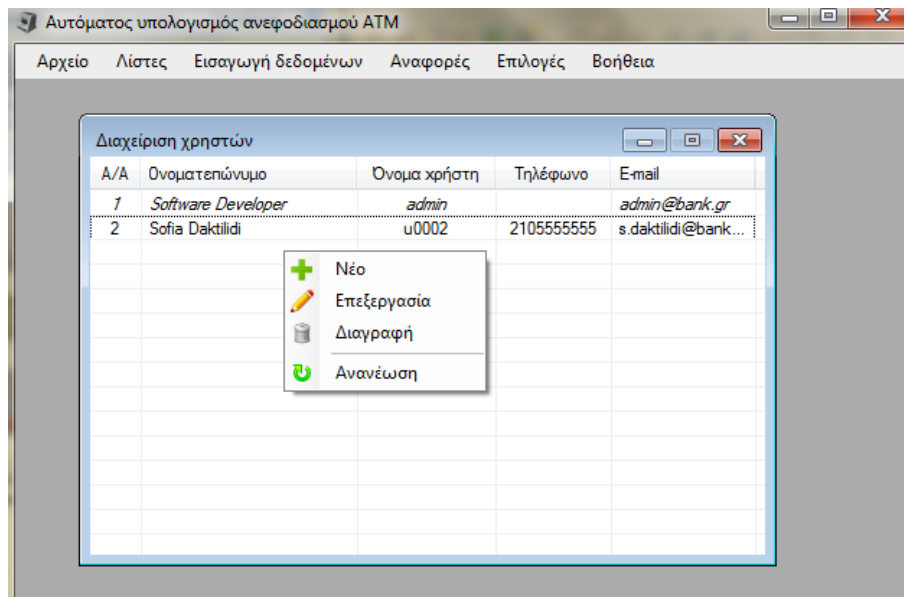
Εικόνα 53: Δυνατότητες διαχείρισης

Επιλέγοντας «Χρήστες» εμφανίζεται μια συγκεντρωτική καρτέλα των χρηστών του συστήματος. Με πλάγια γράμματα εμφανίζεται ο χρήστης που εκείνη τη στιγμή χρησιμοποιεί την εφαρμογή. Για κάθε χρήστη εμφανίζονται το ονοματεπώνυμο, το όνομα εισαγωγής, το τηλέφωνο του και το προσωπικό του email.



Εικόνα 54: Ο χρήστης admin είναι συνδεδεμένος με την εφαρμογή

Με δεξί κλικ πάνω σε κάθε εγγραφή μπορούμε να επιλέξουμε επεξεργασία, διαγραφή ή ανανέωση στοιχείων καθώς και να δημιουργήσουμε ένας νέο χρήστη.

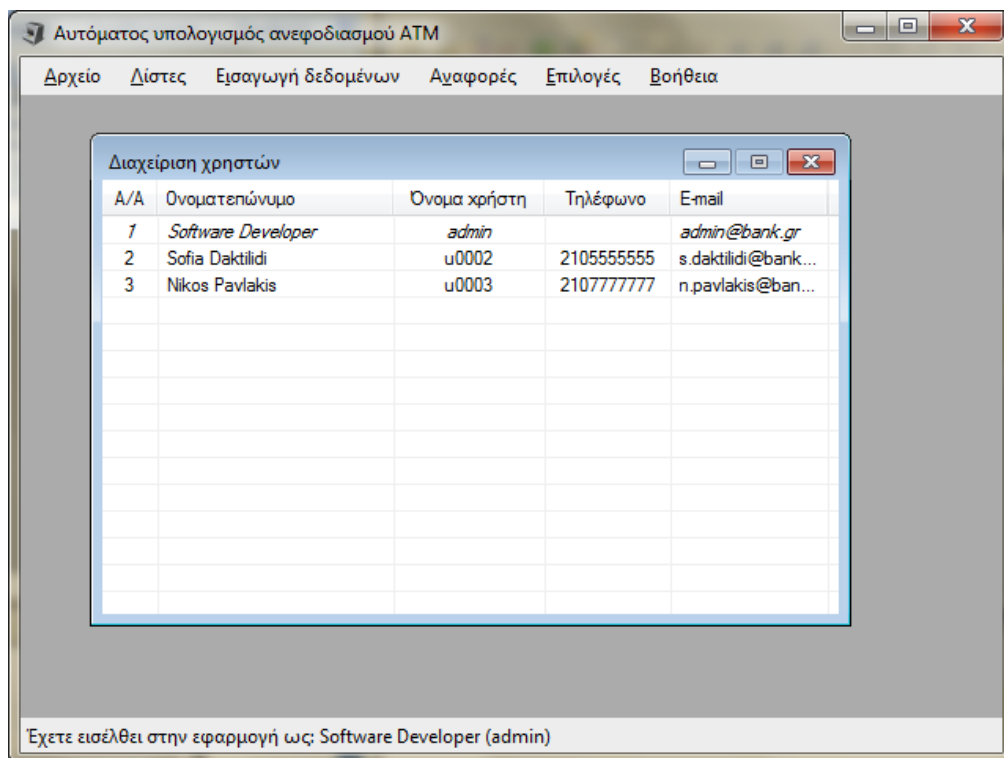


Εικόνα 55: Διαχείριση χρηστών

Για την δημιουργία νέου χρήστη ανοίγει η παρακάτω φόρμα προσθήκης χρήστη όπου θα πρέπει να συμπληρώσουμε τα ακόλουθα στοιχεία, όνομα, επώνυμο, τηλέφωνο, email και κωδικό πρόσβασης. Το όνομα του χρήστη δίνεται αυτόματα από την εφαρμογή ώστε να αποφευχθούν διπλοεγγραφές. Το όνομα χρήστη είναι κωδικοποιημένο και αποτελείται από το γράμμα u(user) και τρία ψηφία όπου δηλώνουν τη θέση του στη λίστα των χρηστών.

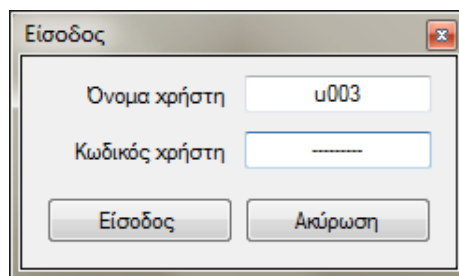
Εικόνα 56: Καταχώρηση νέου χρήστη

Μέχρι αυτή τη στιγμή είχαμε δύο χρήστες (admin & u0002) καταχωρημένους στην εφαρμογή, τώρα δημιουργήσαμε και ένα τρίτο χρήστη με το όνομα Νίκος Παβλάκης, άρα ως όνομα χρήστη θα πρέπει να είχε το u003.



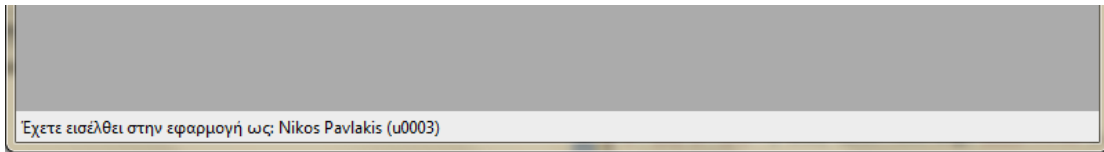
Εικόνα 57: Εμφάνιση νέου χρήστη στη λίστα

Πραγματοποιούμε επανεκκίνηση της εφαρμογής και ο νέος χρήστης προσπαθεί εισαγωγή στην εφαρμογή.



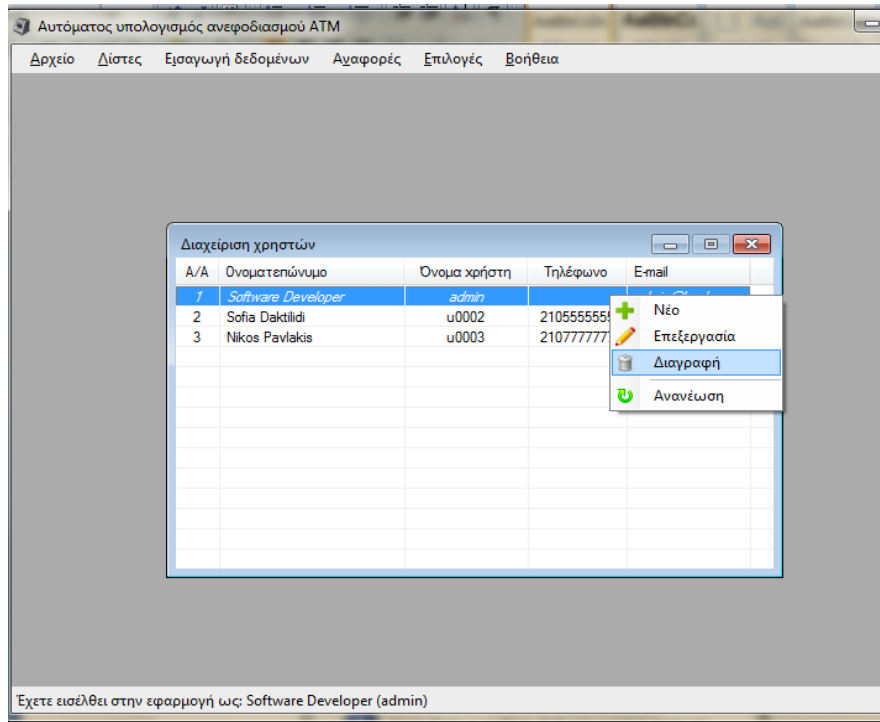
Εικόνα 58: Νέος χρήστης

Ο νέος χρήστης εισέλθει στη εφαρμογή με επιτυχία.

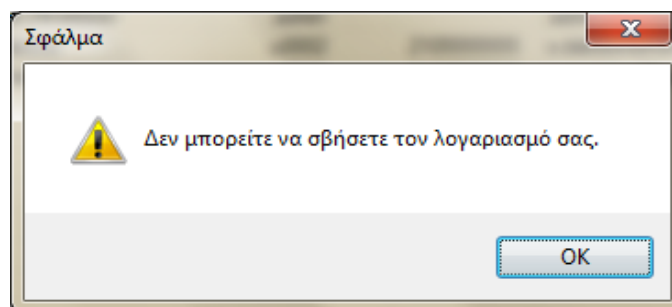


Εικόνα 59: Επιτυχής εισαγωγή νέου χρήστη

Κάθε χρήστης μπορεί να διαχειριστεί τα στοιχεία της καρτέλας του και των υπολοίπων αλλά δε μπορεί να διαγράψει τον εαυτό του από τη λίστα.

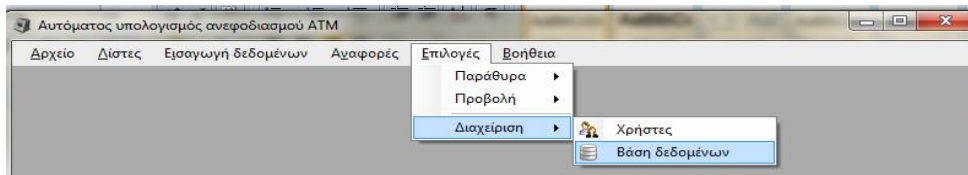


Εικόνα 60: Προσπάθεια διαγραφής ενεργού χρήστη

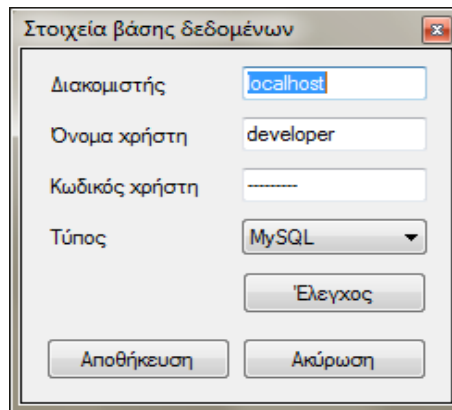


Εικόνα 61: Αποτυχία διαγραφής ενεργού

Από τη καρτέλα Επιλογές, Διαχειριστής μπορούμε δούμε βασικές πληροφορίες της βάσης που χρησιμοποιεί η εφαρμογή, όπως το όνομα του διακομιστή, το όνομα χρήστη, τον κωδικό και το τύπο της βάσης.

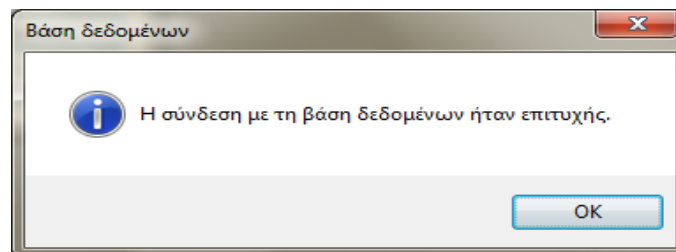


Εικόνα 62: Διαχείριση βάσης δεδομένων



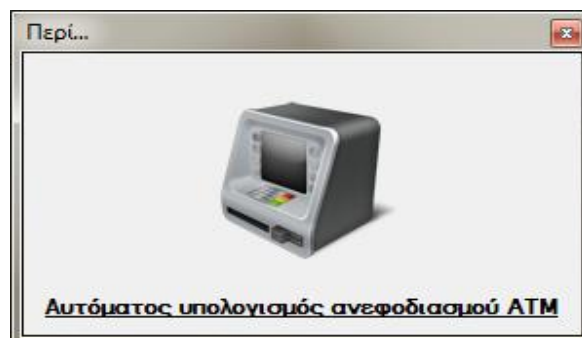
Εικόνα 63: Στοιχεία βάσης δεδομένων

Επιπλέον μπορούμε να ελέγξουμε τη σύνδεση της εφαρμογής με τη βάση ή να πραγματοποιήσουμε αλλαγές τις οποίες μπορούμε στη συνέχεια να ακυρώσουμε ή να αποθηκεύσουμε.



Εικόνα 64: Έλεγχος με τη βάση δεδομένων

Τέλος επιλέγοντας βοήθεια το ενημερωνόμαστε για το σκοπό της εφαρμογής.



Εικόνα 65: Επιλογή Βοήθεια

5.3 Περιγραφή της βάσης δεδομένων

Για την εφαρμογή μας έχουμε δημιουργήσει μία βάση δεδομένων με τη χρήση της MySQL με διεύθυνση διακομιστή localhost και έκδοση 5.5.20-log, με έκδοση πρωτοκόλλου 10 και χαρακτήρες τύπου UTF-8 Unicode ώστε η εφαρμογή μας να υποστηρίζει ελληνικούς χαρακτήρες. Η βάση μας ονομάζεται cs_atm_acard και αποτελείται από δέκα πίνακες atm_atms, atm_atm_states, atm_atm_types, atm_deposits, atm_holidays atm_locations atm_refills atm_users atm_vendors atm_withdrawals. Τα πεδία όλων των πινάκων ακολουθούν την ίδια λογική, τα αλφαριθμητικά πεδία έχουν οριστεί ως varchar(255) και utf8_general_ci για να υποστηρίζουν ελληνική γραμματοσειρά και τα αριθμητικά ως int(10) (το μέγεθος της μεταβλητής διαφέρει ανάλογα με τη τιμή που αναμένουμε να καταχωρηθεί)

Ο πίνακας atm_atms αποτελείται από τα παρακάτω 12 πεδία και περιέχει τα αναγνωριστικά και βοηθητικά στοιχεία των μηχανημάτων.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
2	name	varchar(255)	utf8_general_ci		Όχι		
3	address	varchar(255)	utf8_general_ci		Όχι		
4	location	int(10)		UNSIGNED	Όχι	Καμία	
5	type	int(10)		UNSIGNED	Όχι	Καμία	
6	state	int(10)		UNSIGNED	Όχι	Καμία	
7	withdraw_limit	int(10)			Όχι	Καμία	
8	deposit_limit	int(10)			Όχι	Καμία	
9	refill_ammount	int(10)			Όχι	Καμία	
10	sum_withdrawals	int(10)			Όχι	Καμία	
11	sum_deposits	int(10)			Όχι	Καμία	
12	refill_count	int(11)			Όχι	0	

Id:	αυξανόμενος αριθμός που μετράει τα ATM και αποτελεί το κλειδί του πίνακα
Name:	ονομασία του ATM
Address:	ακριβής διεύθυνση όπου είναι τοποθετημένο
Location:	περιοχή
Type:	τύπος (δέχεται αναλήψεις και καταθέσεις ή μόνο αναλήψεις)
State:	κατάσταση λειτουργίας (σε λειτουργία ή εκτός λειτουργίας)
Withdraw_limit:	ελάχιστο επιτρεπτό όριο χρημάτων που προορίζονται για αναλήψεις
Deposit_limit:	μέγιστο επιτρεπτό όριο χρημάτων που προέρχονται από καταθέσεις
Sum_withdrawals:	προσθέτει καθημερινά τα ποσά των αναλήψεων μέχρι να πραγματοποιηθεί τροφοδοσία
Sum_deposit:	προσθέτει καθημερινά τα ποσά των καταθέσεων μέχρι να πραγματοποιηθεί τροφοδοσία
Refill_count:	μετρητής που αυξάνεται κάθε φορά που δίνουμε εντολή τροφοδοσίας

Ο πίνακας atm_atm_states αποτελείται από δύο πεδία και περιέχει τις πιθανές καταστάσεις των μηχανημάτων.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 state	varchar(32)	utf8_general_ci		Όχι		

Id: αυξανόμενος αριθμός που μετράει τις καταστάσεις των ATM και αποτελεί το κλειδί του πίνακα

State: περιέχει τις πιθανές καταστάσεις των ATM, State 1 σε λειτουργία και State 2 εκτός λειτουργίας

Ο πίνακας atm_atm_types αποτελείται από δύο πεδία και περιέχει τους τύπους των μηχανημάτων.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 type	varchar(32)	utf8_general_ci		Όχι		

Id: αυξανόμενος αριθμός που μετράει τους τύπους των ATM και αποτελεί το κλειδί του πίνακα

Type: περιέχει τους τύπους των ATM σε type 1 μηχάνημα στο οποίο μπορούν να πραγματοποιηθούν μόνο αναλήψεις και type 2 μηχάνημα στο οποίο μπορούν να πραγματοποιηθούν αναλήψεις και καταθέσεις.

Ο πίνακας atm_deposits αποτελείται από πέντε πεδία και τα στοιχεία προέρχονται από τα αρχεία που παράγονται από τα συστήματα του πιστωτικού ιδρύματος.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 atm	int(10)		UNSIGNED	Όχι	Καμία	
<input type="checkbox"/>	3 date	date			Όχι	Καμία	
<input type="checkbox"/>	4 ammount	int(10)		UNSIGNED	Όχι	Καμία	
<input type="checkbox"/>	5 transactions	int(10)		UNSIGNED	Όχι	Καμία	

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Atm: αριθμός ATM που πραγματοποιήθηκαν οι καταθέσεις

Date: ημερομηνία συναγωγών

Amount: συνολικό ποσό καταθέσεων

Transactions: συνολικός αριθμός καταθέσεων

Ο πίνακας atm_holidays αποτελείται από τρία πεδία

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 date	date			Όχι	Καμία	
<input type="checkbox"/>	3 recurring	int(10)		UNSIGNED	Όχι	Καμία	

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Data: καταχώρηση επισμών ή τοπικών αργιών

Recurring: με επαναλαμβανόμενη ετήσια συχνότητα

Ο πίνακας atm_locations αποτελείται από τέσσερα πεδία και περιέχει όλες τις γεωγραφικές περιχές.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 prefecture	varchar(255)	utf8_general_ci		Όχι		
<input type="checkbox"/>	3 area	varchar(255)	utf8_general_ci		Όχι		
<input type="checkbox"/>	4 postcode	varchar(12)	utf8_general_ci		Όχι		

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Prefecture: νομός

Area: περιοχές

Postcode: ταχυδρομικός κώδικας

Ο πίνακας atm_refills αποτελείται από τέσσερα πεδία και περιέχει γενικές πληροφορίες σχετικά με τις τροφοδοσίες των μηχανημάτων.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
<input type="checkbox"/>	2 atm	int(10)		UNSIGNED	Όχι	Καμία	
<input type="checkbox"/>	3 date	date			Όχι	Καμία	
<input type="checkbox"/>	4 ammount	int(10)		UNSIGNED	Όχι	Καμία	

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Atm: κωδός μηχανήματος ATM

Date: ημερομηνία τροφοδοσίας

Ammount: ποσό τροφοδοσίας

Ο πίνακας `atm_users` αποτελείται από επτά πεδία και περιέχει γενικές πληροφορίες σχετικά με τους χρήστες της εφαρμογής.

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
2	name	varchar(255)	utf8_general_ci		Όχι		
3	surname	varchar(255)	utf8_general_ci		Όχι		
4	telephone	varchar(32)	utf8_general_ci		Όχι		
5	email	varchar(255)	utf8_general_ci		Όχι		
6	username	varchar(8)	utf8_general_ci		Όχι		
7	password	varchar(64)	utf8_general_ci		Όχι		

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Name: όνομα χρήστη

Surname: επώνυμο χρήστη

Telephone: τηλέφωνο χρήστη

Email: ηλεκτρονική διεύθυνση

Username: όνομα χρήστη για την εισαγωγή του στην εφαρμογή

Password: κωδικός πρόσβασης στην εφαρμογή

Ο πίνακας `atm_vendors` αποτελείται από οκτώ πεδία και περιέχει γενικές πληροφορίες σχετικά με τους συνεργάτες του πιστωτικού ιδρύματος

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
2	name	varchar(255)	utf8_general_ci		Όχι		
3	surname	varchar(255)	utf8_general_ci		Όχι		
4	telephone	varchar(32)	utf8_general_ci		Όχι		
5	email	varchar(255)	utf8_general_ci		Όχι		
6	company	varchar(255)	utf8_general_ci		Όχι		
7	address	varchar(255)	utf8_general_ci		Όχι		
8	location	int(10)		UNSIGNED	Όχι	Καμία	

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Name: όνομα συνεργάτη

Surname: επώνυμο συνεργάτη

Telephone: τηλέφωνο συνεργάτη

Email: ηλεκτρονική διεύθυνση συνεργάτη

Company: ονομασία εταιρείας

Address: διεύθυνση εταιρείας

Location: περιοχή εταιρείας

Ο πίνακας atm_withdrawals αποτελείται από πέντε πεδία και τα στοιχεία προέρχονται από τα αρχεία που παράγονται από τα συστήματα του πιστωτικού ιδρύματος

#	Στήλη	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	id	int(10)		UNSIGNED	Όχι	Καμία	AUTO_INCREMENT
2	atm	int(10)		UNSIGNED	Όχι	Καμία	
3	date	date			Όχι	Καμία	
4	ammount	int(10)		UNSIGNED	Όχι	Καμία	
5	transactions	int(10)		UNSIGNED	Όχι	Καμία	

Id: αυξανόμενος αριθμός που μετράει της καταχωρήσεις

Atm: αριθμός ATM που πραγματοποιήθηκαν οι αναλήψεις

Date: ημερομηνία συναλλαγών

Amount: συνολικό ποσό αναλήψεων

Transactions: συνολικός αριθμός αναλήψεων

5.4 Βασικοί κώδικες της εφαρμογής

Για τη δημιουργία της πλατφόρμας στηριχτήκαμε στα βιβλία του Νεβράντζας Βάιος – Γερμανός, 2011, Μελέτης Εργαστηρίου C#, του Παναγιωτόπουλου Ιωάννη – Χρήστου, Διαδικτυακός Προγραμματισμός: C#, του John Sharp, Visual C# 2008 βήμα βήμα, Dubois, Paul, Hinz, Stefan και Pedersen, Carsten, 2006, Ο επίσημος οδηγός MySQL 5 καθώς επίσης και στις ακόλουθες ιστοσελίδες στο διαδίκτυο www.microsoft.com/sqlserver, en.wikipedia.org/wiki/Microsoft_SQL_Server, www.connectionstrings.com/sql-server-2008, csharp.net-informations.com/data-providers/csharp-sql-serverconnection, msdn.microsoft.com/el-gr/library, www.microsoft.com/visualstudio, msdn.microsoft.com.

5.4.1 Επεξήγηση του κώδικά που βρίσκεται στο Controls\ControlATM.cs

PopulateFields

```
Dictionary<int, string> locations = this._model.LoadLocations();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλες οι τοποθεσίες.

```
Dictionary<int, string> types = this._model.LoadATMTypes();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλοι οι τύποι των ATM.

```
Dictionary<int, string> states = this._model.LoadATMStates();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλες οι καταστάσεις των ATM.

```
this._form.p_cmbLocation.Items.Clear();
```

Καθαρίζει τη λίστα με τις περιοχές από τυχόν τιμές.

```
foreach (KeyValuePair<int, string> location in locations)
```

Για κάθε ζευγάρι τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή locations, θα γίνουν τα ακόλουθα:

```
this._form.p_cmbLocation.Items.Add(new ComboBoxItem(location.Value, location.Key));
```

Προσθέτει μια εγγραφή στη λίστα με τιμή το όνομα της τοποθεσίας και αναγνωριστικό το ID της τοποθεσίας από τη βάση δεδομένων.

```
this._form.p_cmbType.Items.Clear();
```

Καθαρίζει τη λίστα με τους τύπους από τυχόν τιμές.

```
foreach (KeyValuePair<int, string> type in types)
```

Για κάθε ζευγάρι τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή *types*, θα γίνουν τα ακόλουθα

```
this._form.p_cmbType.Items.Add(new ComboBoxItem(type.Value, type.Key));
```

Προσθέτει μια εγγραφή στη λίστα με τιμή το όνομα του τύπου και αναγνωριστικό το ID του τύπου από τη βάση δεδομένων

```
this._form.p_cmbState.Items.Clear();
```

Καθαρίζει τη λίστα με τις καταστάσεις από τυχόν τιμές

```
foreach (KeyValuePair<int, string> state in states)
```

Για κάθε ζευγάρι τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή *states*, θα γίνουν τα ακόλουθα

```
this._form.p_cmbState.Items.Add(new ComboBoxItem(state.Value, state.Key));
```

Προσθέτει μια εγγραφή στη λίστα με τιμή το όνομα την κατατάση και αναγνωριστικό το ID της κατάστασης από τη βάση δεδομένων

```
this.ClearFields();
```

Η function αυτή καθαρίζει όλα τα πεδία από τυχόν τιμές. Στα textboxes θέτει το κείμενό τους ίσο με το κενό ("") και στις λίστες επιλέγει την πρώτη εγγραφή.

ClearFields

Σε κάθε καρτέλα μπορούμε να καθαρίσουμε τα δεδομένα πριν προχωρήσουμε σε οριστική αποθήκευση τους.

Τα πεδία στα οποία έχουμε καταχωρήσει δεδομένα τα αδειάζουμε (*string.Empty*) ενώ σε αυτά που έχουμε επιλέξει τιμές από τις προτεινόμενες τις αντιπαθήσουμε με το 0.

```
public void ClearFields()
```

```
this._form.p_txtName.Text = string.Empty;
```

```
this._form.p_txtAddress.Text = string.Empty;
```

```
this._form.p_cmbLocation.SelectedIndex = 0;
```

```
this._form.p_cmbType.SelectedIndex = 0;
```

```
this._form.p_cmbState.SelectedIndex = 0;
```

```
this._form.p_txtWithdrawLimit.Text = string.Empty;
```

```
this._form.p_txtDepositLimit.Text = string.Empty;
```

```
this._form.p_txtRefillAmmount.Text = string.Empty;
```

LoadItem

Καταχώρηση στοιχείων στη καρτέλα *itemID*. Για κάθε ATM θα πρέπει να εισάγουμε πληροφορίες *id*, *name*, *address*, *withdraw_limit*, *deposit_limit*, *refill_ammount*. Ενώ για τα δεδομένα *Location*, *Type* και *State* δεν εισάγουμε αλλά επιλέγουμε τιμές από τις προκαθορισμένες (*drop-down list*)

```
public void PopulateFields(int itemID)
```

```
ATM item = this._model.LoadItem(itemID);
```

```

this.PopulateFields();
this._form.p_txtID.Text = item.id.ToString();
this._form.p_txtName.Text = item.name;
this._form.p_txtAddress.Text = item.address;
this._form.p_cmbLocation.SelectedIndex=Helper.GetIndexFromValue(this._form.p_cmbLocation,
item.location);
this._form.p_cmbType.SelectedIndex=Helper.GetIndexFromValue(this._form.p_cmbType, tem.type);
this._form.p_cmbState.SelectedIndex=Helper.GetIndexFromValue(this._form.p_cmbState,
item.state);
this._form.p_txtWithdrawLimit.Text = item.withdraw_limit.ToString();
this._form.p_txtDepositLimit.Text = item.deposit_limit.ToString();
this._form.p_txtRefillAmmount.Text = item.refill_ammount.ToString();

```

Saveltem

Ελέγχουμε την αποθήκευση των αποτελεσμάτων στο πίνακα των ATM.

```
public void Saveltem()
```

```
bool result;
```

επιστρέφουμε το αποτέλεσμα τύπου bool

```
Dictionary<string, string> data = new Dictionary<string, string>();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλα τα χαρακτηριστικά των ATM.

```
data["name"] = this._form.p_txtName.Text;
```

Πληκτρολογούμε το ονομα του ATM στο αντίστοιχο text

```
data["address"] = this._form.p_txtAddress.Text;
```

Πληκτρολογούμε τη διεύθυνση του ATM στο αντίστοιχο text

```
data["location"] = ((ComboBoxItem)this._form.p_cmbLocation.SelectedItem).value.ToString();
```

Επιλέγουμε την τοποθεσία από τη λίστα, τυπος string

```
data["type"] = ((ComboBoxItem)this._form.p_cmbType.SelectedItem).value.ToString();
```

Επιλέγουμε τον τύπο από τη λίστα, τυπος string

```
data["state"] = ((ComboBoxItem)this._form.p_cmbState.SelectedItem).value.ToString();
```

Επιλέγουμε την κατάσταση από τη λίστα, τυπος string

```
data["withdraw_limit"] = this._form.p_txtWithdrawLimit.Text;
```

Πληκτρολογούμε το ελάχιστο ποσο των αναλήψεων στο αντίστοιχο text

```
data["deposit_limit"] = this._form.p_txtDepositLimit.Text;
```

Πληκτρολογούμε το ελάχιστο ποσο των καταθέσεων στο αντίστοιχο text

```
data["refill_ammount"] = this._form.p_txtRefillAmmount.Text;
```

Πληκτρολογούμε το ποσό του ανεφοδιασμού στο αντίστοιχο text

Ελέγχουμε αν έχουν καταχωρηθεί στοιχεία στη καρτέλα πριν προχωρήσουμε στην αποθήκευση.

```
if (this._form.p_txtID.Text.Trim() != "")
```

Γίνεται ελεγχον αν η φόρμα έχει συμπληρωθεί ή υπάρχουν κενά text

```
result = this._model.UpdateItem(data, int.Parse(this._form.p_txtID.Text.Trim()));
```

Επιστέφει το αποτέλεσμα του ελεγχου

```
result = this._model.Saveltem(data);
```

Εάν τα δεδομένα είναι πλήρες, αποθηκεύονται

```
if (result == true)
```

Αν το αποτέλεσμα επιστρέφει τιμή true

```
this.CloseForm();
```

κλείνει η φόρμα επικοινωνίας

```
MessageBox.Show("Παρουσιάστηκε ένα σφάλμα κατά την αποθήκευση του ATM.", "Σφάλμα",
MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);
```

Εάν η καρτέλα είναι κενή επιστρέφει μήνυμα λάθους (Παρουσιάστηκε ένα σφάλμα κατά την αποθήκευση του ATM) μέσω MessageBox.

5.4.2 Επεξήγηση του κώδικά που βρίσκεται στο Controls\ControlATMs.cs

LoadLocations

```
Dictionary<int, string> locations = this._model.LoadLocations();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλες οι τοποθεσίες.

```
Dictionary<string, TreeNode> nodes = new Dictionary<string, TreeNode>();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί ένα αλφαριθμητικό σε ένα TreeNode. Το TreeNode είναι κάθε μια καταχώρηση στο «δέντρο» με τις τοποθεσίες. Σε αυτή τη μεταβλητή θα αποθηκευτούν όλα τα Nodes του «δέντρου» με τις τοποθεσίες.

```
this._form.p_trvLocations.Nodes.Clear();
```

Καθαρίζει το «δέντρο» από τυχόν Nodes.

```
this._form.p_trvLocations.Nodes.Add("All", "Όλες οι περιοχές");
```

Προσθέτει ένα Node με όνομα «Όλες οι περιοχές» και αναγνωριστικό «All». Αυτό το Node βρίσκεται πάνω πάνω στο «δέντρο» και είναι ο γονέα όλων των υπόλοιπων Nodes.

```
foreach (KeyValuePair<int, string> location in locations)
```

Για κάθε ζευγάρι τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή locations, θα γίνουν τα ακόλουθα:

```
string[] parts = location.Value.Split('|');
```

Σπάμε το αλφαριθμητικό σε Χ κομμάτια, στα σημεία που υπάρχει ο χαρακτήρας "|". Κάθε κομμάτι αποθηκεύεται στον πίνακα parts.

```
string prefecture = parts[0].Trim();
```

Το πρώτο κομμάτι του πίνακα parts αντιστοιχεί στο νομό της κάθε περιοχής. Το αποθηκεύουμε στη μεταβλητή prefecture, αφού φροντίσουμε να αφαιρέσουμε τυχόν κενούς χαρακτήρες από το αλφαριθμητικό (αυτό κάνει το Trim()).

```
string area = parts[1].Trim();
```

Το δεύτερο κομμάτι του πίνακα parts αντιστοιχεί στην τοποθεσία και τον ταχυδρομικό κώδικα. Το αποθηκεύουμε στο μεταβλητή area, αφού φροντίσουμε να αφαιρέσουμε τυχόν κενούς χαρακτήρες από το αλφαριθμητικό (αυτό κάνει το Trim()).

```
if (this._form.p_trvLocations.Nodes["All"].Nodes[prefecture] != null)
```

Ελέγχουμε εάν υπάρχει ένα Node με αναγνωριστικό ίσο με την τιμή της μεταβλητής prefecture.

```
this._form.p_trvLocations.Nodes["All"].Nodes[prefecture].Nodes.Add(location.Key.ToString(), area);
```

Εάν υπάρχει, φτιάχνουμε ένα Node με αναγνωριστικό τον ακέραιο και τιμή το αλφαριθμητικό από το ζευγάρι τιμών της περιοχής. Συνεπώς, σαν αναγνωριστικό θα έχει την τιμή του ID της εγγραφής από τη βάση, και σαν τιμή το όνομα της περιοχής μαζί με τον ταχυδρομικό κώδικα. Το Node αυτό, θα είναι παιδί του Node του αντιστοιχού νομού.

```
this._form.p_trvLocations.Nodes["All"].Nodes.Add(prefecture, prefecture);
```

Εάν δεν υπάρχει, φτιάχνουμε ένα Node με αναγνωριστικό και τιμή το όνομα του νομού (μεταβλητή prefecture). Οι νομοί, καθώς δεν έχουν δικό τους πίνακα στη βάση δεδομένων, δεν έχουν κάποιο μοναδικό ID. Για αυτό το λόγο το αναγνωριστικό του Node είναι ίσο με το όνομα του νομού.

```
this._form.p_trvLocations.Nodes["All"].Nodes[prefecture].Nodes.Add(location.Key.ToString(), area);
```

Στη συνέχεια, φτιάχνουμε ένα Node με αναγνωριστικό τον ακέραιο και τιμή το αλφαριθμητικό από το ζευγάρι τιμών της περιοχής. Συνεπώς, σαν αναγνωριστικό θα έχει την τιμή του ID της εγγραφής από τη βάση, και σαν τιμή το όνομα της περιοχής μαζί με τον ταχυδρομικό κώδικα. Το Node αυτό, θα είναι παιδί του Node του αντίστοιχου νομού.

Μόλις τελειώσουν τα ζευγάρια τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή locations, θα έχει συμπληρωθεί επιτυχώς το “δέντρο” με τις περιοχές.

5.4.3 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Database.cs

TestConnection

```
public void TestConnection()
```

Ελέγχονται τα δεδομένα που εισάγονται σχετικά με τα χαρακτηριστικά της βάσης δεδομένων.

```
if (this._model.TestConnection(this._form.p_txtServer.Text, this._form.p_txtUsername.Text, this._form.p_txtPassword.Text))
```

Ο χρήστης πληκτρολογεί τα ζητούμε στοιχεία

```
MessageBox.Show("Η σύνδεση με τη βάση δεδομένων ήταν επιτυχής.", "Βάση δεδομένων", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);
```

Ελέγχονται με τα προκαθορισμένα στοιχεία της βάσης και αν είναι τα ίδια εμφανίζεται το μήνυμα της έγκυρης πρόσβασης.

```
else
```

```
MessageBox.Show("Η σύνδεση με τη βάση δεδομένων απέτυχε.", "Βάση δεδομένων", MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);
```

Ελέγχονται με τα προκαθορισμένα στοιχεία της βάσης και αν είναι τα διαφορετικά εμφανίζεται το μήνυμα της ανεπιτυχής πρόσβασης.

SaveSettings()

```
public void SaveSettings()
```

Αποθηκευση των πληροφοριών

```
if (this._model.TestConnection(this._form.p_txtServer.Text, this._form.p_txtUsername.Text, this._form.p_txtPassword.Text))
```

Ελέγχουμε τα στοιχεία της σύνδεσης Server, Username & Password που πληκτρολογήσαμε είναι και τα σωστά

```
StreamWriter file = new StreamWriter(this._XML);
```

Μεταφέρουμε τα πληκτρολογημένα δεδομένα σε μία νέα μεταβλητή file

```
String XML =
```

```
string.Format("<database>\r\n<server>{0}</server>\r\n<username>{1}</username>\r\n<password>{2}</password>\r\n</database>", this._form.p_txtServer.Text, this._form.p_txtUsername.Text, this._form.p_txtPassword.Text);
```

Αλλάζουμε την εμφάνιση/διάταξη των στοιχείων ώστε να αναγνωρίζονται ως XML

```
file.WriteLine(XML);
```

```
file.Close();
```

Κλείνει η ενεργεί φόρμα επικοινωνίας

```

MessageBox.Show("Τα στοιχεία αποθηκεύτηκαν επιτυχώς.", "Βάση δεδομένων",
MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);
Ελέγχονται με τα προκαθορισμένα στοιχεία της βάσης και αν είναι τα ίδια εμφανίζεται το μήνυμα
της έγκυρης πρόσβασης.
if (Session.user.id == 0)
Αφού συνδεθήκαμε για πρώτη φορά με τη βάση ο user παίρνει αυτόματα το id μηδέν.
MessageBox.Show("Θα γίνει επανεκκίνηση της εφαρμογής τώρα.", "Επανεκκίνηση",
MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);
Γίνεται επανεκκίνηση του προγράμματος και η εφαρμογή ενημερώνει το χρήστη για τη
συγκεκριμένη λειτουργία
Application.Restart();
Επανεκκίνηση της εφαρμογής
this.CloseForm();
Κλείνει η ενεργεί φόρμα επικοινωνίας
else
MessageBox.Show("Η σύνδεση με τη βάση δεδομένων απέτυχε.", "Βάση δεδομένων",
MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);
Ελέγχονται με τα προκαθορισμένα στοιχεία της βάσης και αν είναι τα διαφορετικά εμφανίζεται το
μήνυμα της ανεπιτυχής πρόσβασης.

```

5.4.4 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Holiday.cs

PopulateField

Το ακόλουθο κομμάτι του κώδικα χωρίζεται σε τέσσερα πεδία ημέρα, μήνα, έτος και το πεδίο στο οποίο δηλώνουμε ένα η αργία είναι επαναλαμβανόμενη ή όχι. Για κάθε ένα από τα πεδία, αρχικά σβήνει τυχών προηγούμενες εγγραφές και στη συνέχεια το γεμίζει με τις καινούγιες τιμές.

```

public void PopulateFields()
this._form.p_cmbDay.Items.Clear();
Ημέρα
for (int i = 1; i < 32; i++)
this._form.p_cmbDay.Items.Add(new ComboBoxItem(i.ToString("00"), i));
this._form.p_cmbMonth.Items.Clear();
Μήνας
for (int i = 1; i < this._months.Length; i++)
this._form.p_cmbMonth.Items.Add(new ComboBoxItem(this._months[i], i));
this._form.p_cmbYear.Items.Clear();
Έτος
for (int i = 2013; i < 2016; i++)
this._form.p_cmbYear.Items.Add(new ComboBoxItem(i.ToString(), i));
this._form.p_cmbRecurring.Items.Clear();
Επαναλαμβανόμενη ή όχι αργία
for (int i = 0; i < this._recurring.Length; i++)
this._form.p_cmbRecurring.Items.Add(new ComboBoxItem(this._recurring[i], i));
this.ClearFields();
this._populated = true;

```


5.4.5 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Holidays.cs

ShowHolidaysContextMenuStrip

Το κομμάτι που παραθέτουμε είναι υπεύθυνο για την εμφάνιση του μενού όταν πατάμε το δεξί κλικ του ποντικίου. Ο ίδιος κώδικας υπάρχει και σε όλες τις άλλες φόρμες με τις λίστες.

```
public void ShowHolidaysContextMenuStrip(MouseEventArgs e)
```

Όταν κάνουμε κλικ με το ποντίκι μέσα στην λίστα, ελέγχει εάν το κουμπί που επιλέξαμε ήταν το δεξί (e.Button == MouseButtons.Right). Εάν ήταν αυτό, τότε κάνει τους ακόλουθους ελέγχους:

```
if (e.Button == MouseButtons.Right)
```

```
switch (this._form.p_lstHolidays.SelectedItems.Count)
```

Εάν δεν υπάρχουν επιλεγμένες εγγραφές (this._form.p_lstHolidays.SelectedItems.Count == 0) τότε απενεργοποιεί τις επιλογές «Επεξεργασία» και «Διαγραφή».

```
case 0:
```

```
this._form.p_cmiHolidaysEdit.Enabled = false;
```

```
this._form.p_cmiHolidaysDelete.Enabled = false;
```

```
break;
```

Εάν υπάρχει μόνο μία επιλεγμένη εγγραφή, τότε ενεργοποιεί τις επιλογές «Επεξεργασία» και «Διαγραφή».

```
case 1:
```

```
this._form.p_cmiHolidaysEdit.Enabled = true;
```

```
this._form.p_cmiHolidaysDelete.Enabled = true;
```

```
break;
```

Εάν υπάρχουν πάνω από μια επιλεγμένες εγγραφές, τότε απενεργοποιεί την επιλογή «Επεξεργασία» και ενεργοποιεί την επιλογή «Διαγραφή».

```
default:
```

```
this._form.p_cmiHolidaysEdit.Enabled = false;
```

```
this._form.p_cmiHolidaysDelete.Enabled = true;
```

```
break;
```

```
this._form.p_cmsHolidays.Show(Cursor.Position);
```

5.4.6 Επεξήγηση του κώδικά που βρίσκεται στο Controls>Login.cs

VerifyCredential

```
public void VerifyCredentials(string username, string password)
```

Ζητείται επιβεβαίωση στοιχείων username και password

```
User user = this._model.VerifyCredentials(username, password);
```

Δημιουργούμε μία νέα μεταβλητή user όπου αποθηκεύονται τα πληκτρολογημένα στοιχεία που έδωσε ο χρήστης

```
if (user != null)
```

Σε περίπτωση που ο χρήστης δεν άφησε κενά πεδία

```
Session.user = user;
```

Ελεγχουμε τα δεδομένα που δόθηκαν να τα υπάρχουν

```
this._loggedIn = true;
```

Καληται η function loggedIn και αν είναι true

```
this.CloseForm();
```

κλείνει η ενεργεί φόρμα

else

this._loggedIn = false;

Αν function loggedIn και αν είναι false

*MessageBox.Show("Τα στοιχεία που δώσατε δεν είναι σωστά.", "Σύνδεση",
MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);*

εμφανίζεται σχετικό μήνυμα λανθασμένων στοιχείων.

5.4.7 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Main.cs

ImportATMWithdrawals

Ο ακόλουθος κώδικας είναι υπεύθυνος να διαβάσει το αρχείο με τις εγγραφές και να το περνάει στη βάση δεδομένων.

public void ImportATMWithdrawals()

DialogResult result = this._form.p_ofdATMData.ShowDialog(this._form);

Αρχικά ανοίγει το παράθυρο όπου επιλέγεις το αρχείο που θέλεις. Εάν πατήσεις το κουπί "OK", τότε ελέγχει εάν έχεις επιλέξει κάποιο/κάποια αρχείο/αρχεία.

if (result == DialogResult.OK)

Για κάθε ένα από τα επιλεγμένα αρχεία (είτε είναι 1 είτε περισσότερα) κάνει τα ακόλουθα:

Ανοίγει το αρχείο για διάβασμα και το διαβάζει γραμμή-γραμμή.

Έχουμε συμφωνήσει οι στήλες να χωρίζονται με το σύμβολο « ; », οπότε χωρίζουμε την κάθε γραμμή στα σημεία που υπάρχει το σύμβολο « ; » (*line.Split(new char[] { ';' });*).

Στη συνέχεια αποθηκεύουμε την τιμή της κάθε στήλης στην αντίστοιχη μεταβλητή του «λεξικού». Φροντίζουμε να αφαιρέσουμε τυχόν κενά από την αρχή και το τέλος της τιμής. Το « \t » αντιπροσωπεύει το TAB.

foreach (string file in this._form.p_ofdATMData.FileNames)

string line = string.Empty;

StreamReader reader = new StreamReader(string.Format(@"{0}", file));

bool save = false;

while ((line = reader.ReadLine()) != null)

string[] parts = line.Split(new char[] { ';' });

Dictionary<string, object> data = new Dictionary<string, object>();

DateTime date = Convert.ToDateTime(parts[0].Trim(new char[] { ' ', '\t' }));

data["atm"] = int.Parse(parts[1].Trim(new char[] { ' ', '\t' }));

data["date"] = string.Format("{0}-{1}-{2}", date.Year, date.Month, date.Day);

data["ammount"] = int.Parse(parts[2].Trim(new char[] { ' ', '\t' }).Replace(".", ""));

data["transactions"] = int.Parse(parts[3].Trim(new char[] { ' ', '\t' }));

Στη συνέχεια αποθηκεύουμε στη βάση.

save = this._model.SaveWithdrawal(data);

reader.Close();

Εάν η αποθήκευση των τιμών ήταν επιτυχής, τότε παρουσιάζεται στον χρήστη το μήνυμα ότι όλα πήγαν καλά. Αλλιώς εμφανίζεται ένα μήνυμα που τον ενημερώνει πως υπήρξε κάποιο σφάλμα.

if (save == true)

```

    MessageBox.Show(string.Format("Τα δεδομένα εισήχθησαν επιτυχώς ({0}).", file), "Εισαγωγή
    δεδομένων", MessageBoxButtons.OK, MessageBoxIcon.Information,
    MessageBoxDefaultButton.Button1);

```

```

Else

```

```

    MessageBox.Show(string.Format("Παρουσιάστηκε ένα σφάλμα κατά την εισαγωγή των δεδομένων
    ({0}).", file), "Εισαγωγή δεδομένων", MessageBoxButtons.OK, MessageBoxIcon.Error,
    MessageBoxDefaultButton.Button1);

```

5.4.8 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Reports.cs

LoadItems

Ο παρακάτω κώδικας διαβάζει όλα τα ATM από τη βάση δεδομένων (χωρίς κάποιον περιορισμό, πχ/ περιοχή) και στη συνέχεια τα προσθέτει στην λίστα προς εμφάνιση.

```

public void LoadItems()

```

```

    List<ATM> items = this._model.LoadItems();

```

```

    this._form.p_lstATMs.Items.Clear();

```

```

    foreach (ATM item in items)

```

Η τιμή της κάθε στήλης καθορίζεται από τη σειρά των μεταβλητών στη γραμμή. Όλες οι λίστες έχουν μια κρυφή στήλη, την πρώτη, για αυτό και η πρώτη τιμή είναι το κενό "".

```

    this._form.p_lstATMs.Items.Add(new ListViewItem(new string[] { "", item.id.ToString(), item.name,
    item.sum_withdrawals.ToString(), item.refill_ammount.ToString(), item.sum_deposits.ToString(),
    item.deposit_limit.ToString(), item.forecast_withdrawals.ToString(),
    item.forecast_deposits.ToString() }));

```

RefillItems

Ο παρακάτω κώδικας υπολογίζει τον τον ανεφοδιασμό για κάθε ATM. Ελέγχει ένα από την λίστα με τα ATM προς ανεφοδιασμό υπάρχουν επιλεγμένες εγγραφές και εάν υπάρχουν, τότε για κάθε μία από αυτές κάνει τον ανεφοδιασμό.

```

public void RefillItems()

```

```

    if (this._form.p_lstATMs.CheckedItems.Count > 0)

```

```

    foreach (ListViewItem item in this._form.p_lstATMs.CheckedItems)

```

Επειδή η πρώτη στήλη της λίστας είναι το checkbox, θέλουμε να πάρουμε την τιμή της στήλης 2 (item.SubItems[1].Text – το μέτρημα αρχίζει από το 0, οπότε το 1 είναι η δεύτερη στήλη) η οποία αντιστοιχεί στο ID του εκάστοτε ATM.

```

    if (!this._model.RefillItem(int.Parse(item.SubItems[1].Text)))

```

Εάν κάτι δεν πάει καλά ειδοποιείται ο χρήστης. Μόλις ολοκληρωθεί ο ανεφοδιασμός των ATM, διαβάζουμε και υπολογίζουμε εκ νέου ποια ATM θέλουν ανεφοδιασμό και τα εμφανίζουμε (εάν υπάρχουν κάποια).

```

    MessageBox.Show(string.Format("Παρουσιάστηκε ένα σφάλμα κατά τον ανεφοδιασμό του ATM
    με A/A \"{0}\"", item.SubItems[1].Text), "Σφάλμα", MessageBoxButtons.OK, MessageBoxIcon.Error,
    MessageBoxDefaultButton.Button1);

```

```

    this.LoadItems();

```

5.4.9 Επεξήγηση του κώδικά που βρίσκεται στο Controls\User.cs

LoadItem

Καταχώρηση στοιχείων στη καρτέλα itemID. Για κάθε user θα πρέπει να εισάγουμε πληροφορίες id, name, Surname, Telephone, Email, Username και Password

```
public void PopulateFields(int itemID)
User user = this._model.LoadItem(itemID);
this.PopulateFields();
this._form.p_txtID.Text = user.id.ToString();
this._form.p_txtName.Text = user.name;
this._form.p_txtSurname.Text = user.surname;
this._form.p_txtTelephone.Text = user.telephone;
this._form.p_txtEmail.Text = user.email;
this._form.p_txtUsername.Text = user.username;
this._form.p_txtPassword.Text = user.password;
```

ClearFields

Σε κάθε καρτέλα μπορούμε να καθαρίσουμε τα δεδομένα πριν προχωρήσουμε σε οριστική αποθήκευση τους. Τα πεδία στα οποία έχουμε καταχωρήσει δεδομένα τα αδειάζουμε (string.Empty)

```
public void ClearFields()
this._form.p_txtName.Text = string.Empty;
this._form.p_txtSurname.Text = string.Empty;
this._form.p_txtTelephone.Text = string.Empty;
this._form.p_txtEmail.Text = string.Empty;
this._form.p_txtUsername.Text = string.Empty;
this._form.p_txtPassword.Text = string.Empty;
```

Saveltem

Ελέγχουμε την αποθήκευση των αποτελεσμάτων στο πίνακα των user.

```
public void Saveltem()
bool result;
επιστρέφουμε το αποτέλεσμα τύπου bool
Dictionary<string, string> data = new Dictionary<string, string>();
Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλα τα χαρακτηριστικά των user.
data["name"] = this._form.p_txtName.Text;
Πληκτρολογούμε το ονομα του user στο αντίστοιχο text
data["surname"] = this._form.p_txtSurname.Text;
Πληκτρολογούμε το επώνυμο του user στο αντίστοιχο text
data["telephone"] = this._form.p_txtTelephone.Text;
Επιλέγουμε το τηλέφωνο του user, τυπος string
data["email"] = this._form.p_txtEmail.Text
Επιλέγουμε το email του user, τυπος string
```

```

data["password"] = this._form.p_txtPassword.Text;
Επιλέγουμε το password του user, τυπος string
Ελέγχουμε αν έχουν καταχωρηθεί στοιχεία στη καρτέλα πριν προχωρήσουμε στην αποθήκευση.
if (this._form.p_txtID.Text.Trim() != "")
Γίνεται ελεγχον αν η φόρμα έχει συμπληρωθεί ή υπάρχουν κενά text
result = this._model.UpdateItem(data, int.Parse(this._form.p_txtID.Text.Trim()));
Επιστέφει το αποτέλεσμα του ελεγχου
result = this._model.SaveItem(data);
Εάν τα δεδομένα είναι πλήρες, αποθηκεύονται
if (result == true)
Αν το αποτέλεσμα επιστρέφει τιμή true
this.CloseForm();
κλείνει η φόρμα επικοινωνίας
MessageBox.Show("Παρουσιάστηκε ένα σφάλμα κατά την αποθήκευση του χρήστη.", "Σφάλμα",
MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);
Εάν η καρτέλα είναι κενή επιστρέφει μήνυμα λάθους (Παρουσιάστηκε ένα σφάλμα κατά την
αποθήκευση του user) μέσω MessageBox.

```

5.4.10 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Users.cs

LoadItems

Αυτός ο κώδικας είναι υπεύθυνος για την εμφάνιση των συνεργατών που ανήκουν σε μια περιοχή ή ένα νομό.

```
public void LoadItems(string search)
```

Η μεταβλητή "search" αντιστοιχεί στην τιμή της επιλεγμένης περιοχής από την αριστερή στήλη με τις περιοχές. Όπως έχουν ξαναπει, οι περιοχές έχουν ένα συγκεκριμένο ID ενώ οι νομοί δεν έχουν.

```
this._search = search;
```

```
List<Vendor> items = new List<Vendor>();
```

```
int locationID;
```

Προσπαθούμε λοιπόν να δούμε εάν η τιμή της μεταβλητής "search" είναι ακέραιος αριθμός. Εάν είναι τότε ψάχνουμε συνεργάτες που ανήκουν σε μια περιοχή με βάση το ID της περιοχής. Εάν δεν είναι, τότε ψάχνουμε συνεργάτες που ανήκουν σε ένα νομό με βάση το όνομα του νομού (βλέπε τον κώδικα του "LoadItems(string prefecture)").

```
if (int.TryParse(search, out locationID))
```

```
items = this._model.LoadItems(locationID);
```

```
else
```

```
if (search == "All")
```

```
items = this._model.LoadItems();
```

```
else
```

```
items = this._model.LoadItems(search);
```

Έπειτα καθαρίζουμε την λίστα από τυχόν εγγραφές και εμφανίζουμε τα νέα αποτελέσματα.

```
this._form.p_lstVendors.Items.Clear();
```

```
foreach (Vendor item in items)
```

```
string name = string.Format("{0} {1}", item.name, item.surname);
```

```
this._form.p_lstVendors.Items.Add(new ListViewItem(new string[] { item.id.ToString(),
item.id.ToString(), name, item.company, item.telephone }));
```

5.4.11 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Vendor.cs

PopulateFields

```
Dictionary<int, string> locations = this._model.LoadLocations();
```

Φτιάχνει ένα «λεξικό» όπου αντιστοιχεί έναν ακέραιο σε ένα αλφαριθμητικό. Σε αυτή τη μεταβλητή αποθηκεύονται όλες οι τοποθεσίες.

```
this._form.p_cmbLocation.Items.Clear();
```

Καθαρίζει τη λίστα με τις περιοχές από τυχόν τιμές.

```
foreach (KeyValuePair<int, string> location in locations)
```

Για κάθε ζευγάρι τιμών (ακέραιος, αλφαριθμητικό) από τη μεταβλητή locations, θα γίνουν τα ακόλουθα:

```
this._form.p_cmbLocation.Items.Add(new ComboBoxItem(location.Value, location.Key));
```

Προσθέτει μια εγγραφή στη λίστα με τιμή το όνομα της τοποθεσίας και αναγνωριστικό το ID της τοποθεσίας από τη βάση δεδομένων.

```
this.ClearFields();
```

Η function αυτή καθαρίζει όλα τα πεδία από τυχόν τιμές. Στα textboxes θέτει το κείμενό τους ίσο με το κενό ("") και στις λίστες επιλέγει την πρώτη εγγραφή

5.4.12 Επεξήγηση του κώδικά που βρίσκεται στο Controls\Vendor.cs

DeleteItem

Ο ίδιος κώδικας υπάρχει σε όλες τις φόρμες με τις λίστες. Ελέγχει εάν υπάρχουν επιλεγμένες εγγραφές στην λίστα και εάν υπάρχουν, προσπαθεί να τις σβήσει μια-μια.

Το «κείμενο» (Text) κάθε εγγραφής αντιστοιχεί στο ID της στη βάση δεδομένων. Εάν υπάρξει κάποιο πρόβλημα με μια διαγραφή, θα εμφανιστεί ένα μήνυμα που ενημερώνει τον χρήστη ότι η διαγραφή απέτυχε.

```
public void DeleteItem()
```

```
if (this._form.p_lstVendors.SelectedItems.Count > 0)
```

```
foreach (ListViewItem item in this._form.p_lstVendors.SelectedItems)
```

```
if (!this._model.DeleteItems(int.Parse(item.Text)))
```

```
MessageBox.Show(string.Format("Παρουσιάστηκε ένα σφάλμα κατά τη διαγραφή του προμηθευτή με A/A \"{0}\"", item.Text), "Σφάλμα", MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1);
```

```
this.LoadItems();
```

5.4.13 Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelATM.cs

SaveItem.cs

```
public bool SaveItem(Dictionary<string, string> data)
```

Για τη function αυτή θα πρέπει να δωθεί ένα όρισμα, ένα «λεξικό» όπου αντιστοιχεί ένα αλφαριθμητικό σε ένα αλφαριθμητικό.

```
List<string> into = new List<string>();
```

Φτιάχνει μια λίστα από αλφαριθμητικά. Σε αυτή τη μεταβλητή θα αποθηκευτούν το ονόματα των στηλών.

```
List<string> values = new List<string>();
```

Φτιάχνει μια λίστα από αλφαριθμητικά. Σε αυτή τη μεταβλητή θα αποθηκευτούν οι τιμές για τις αντίστοιχες στήλες.

```
foreach (KeyValuePair<string, string> pair in data)
```

Για κάθε ένα ζευγάρι τιμών (αλφαριθμητικό, αλφαριθμητικό) της μεταβλητής *data*, θα γίνουν τα ακόλουθα:

```
into.Add(string.Format("{0}", pair.Key));
```

Στην λίστα “into” θα προστεθεί το αναγνωριστικό του ζευγαριού του «λεξικού» (το όνομα της στήλης).

```
values.Add(string.Format("{0}", pair.Value));
```

Στην λίστα “values” θα προστεθεί η τιμή του ζευγαριού του «λεξικού» (η τιμή της στήλης).

Το Query για εγγραφή δεδομένων στη βάση δεδομένων πρέπει να είναι της μορφής:

```
INSERT INTO <όνομα πίνακα> (όνομα στήλης 1, όνομα στήλης 2, ...) VALUES (τιμή στήλης 1, τιμή στήλης 2, ...)
```

Προς το παρών, έχουμε τα ονόματα των στηλών και τις αντίστοιχες τιμές τους στις αντίστοιχες λίστες. Για να πετύχουμε το παραπάνω αποτέλεσμα:

```
string.Join(" ", into.ToArray())
```

Δημιουργούμε ένα αλφαριθμητικό το οποίο θα περιέχει όλες τις τιμές της λίστας με τα ονόματα των στηλών, χωρισμένα μεταξύ τους με «,» (κόμμα). Επειδή δεν μπορούμε να «ενώσουμε» κατευθείαν τις τιμές μιας λίστας, την μετατρέπουμε πρώτα σε πίνακα (.ToArray()). Κάνουμε το ίδιο και για τις τιμές των στηλών.

UpdateAtms.cs

```
public bool UpdateItem(Dictionary<string, string> data, int itemID)
```

Για τη function αυτή θα πρέπει να δωθούν δυο ορίσματα, ένα «λεξικό» όπου αντιστοιχεί ένα αλφαριθμητικό σε ένα αλφαριθμητικό και ένας ακέραιος, ο οποίος αντιστοιχεί στο ID της εγγραφής που θέλουμε να αλλάξουμε.

```
List<string> pairs = new List<string>();
```

Φτιάχνει μια λίστα από αλφαριθμητικά. Σε αυτή τη μεταβλητή θα αποθηκευτούν τα ζευγάρια στήλη-τιμή που θέλουμε να αλλάξουν.

```
foreach (KeyValuePair<string, string> pair in data)
```

Για κάθε ένα ζευγάρι τιμών (αλφαριθμητικό, αλφαριθμητικό) της μεταβλητής *data*, θα γίνουν τα ακόλουθα:

```
pairs.Add(string.Format("{0}` = '{1}'", pair.Key, pair.Value));
```

Στην λίστα “pairs” θα προστεθεί το ζευγάρι όνομα στήλης-τιμή στήλης, με το μορφή “όνομα στήλης = τιμή στήλης”.

Το Query για αλλαγή δεδομένων στη βάση δεδομένων πρέπει να είναι της μορφής:

```
UPDATE <όνομα πίνακα> SET όνομα στήλης 1 = τιμή στήλης 1, ..., WHERE ...
```

Προς το παρών, έχουμε τα ζευγάρια όνομα στήλης-τιμή στήλης στην λίστα “pairs”. Για να πετύχουμε το παραπάνω αποτέλεσμα:

```
string.Join(" ", pairs.ToArray())
```

Δημιουργούμε ένα αλφαριθμητικό το οποίο θα περιέχει όλες τις τιμές της λίστας με τα ζευγάρια, χωρισμένα μεταξύ τους με «,» (κόμμα). Επειδή δεν μπορούμε να «ενώσουμε» κατευθείαν τις τιμές μιας λίστας, την μετατρέπουμε πρώτα σε πίνακα (.ToArray()).

5.4.14 Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelATMs.cs

DeleteItems.cs

```
public bool DeleteItems(int itemID)
```

Για τη function αυτή θα πρέπει να δωθεί ένα όρισμα, το ID της επιθυμητής εγγραφής που θέλουμε να σβήσουμε. Το όρισμα πρέπει να είναι ακέραιος.

```
return this._sql.Delete(this._table, itemID);
```

Θα επιστρέψει την τιμή του αποτελέσματος της function “Delete” από τον μεσάζοντα (η class που βρίσκεται μεταξύ του μοντέλου και της βάσης δεδομένων).

Εάν η διαγραφή της εγγραφής ήταν επιτυχημένη, η function θα επιστρέψει «true» αλλιώς θα επιστρέψει «false».

Η τιμή της μεταβλητής “this._table” αντιστοιχεί στο όνομα του πίνακα της βάσης δεδομένων από τον οποίο θέλουμε να σβήσουμε μια εγγραφή.

LoadItems.cs

```
public List<ATM> LoadItems(int locationID)
```

Για τη function αυτή θα πρέπει να δωθεί ένα όρισμα, το ID της επιθυμητής τοποθεσίας. Το όρισμα πρέπει να είναι ακέραιος.

```
return this._sql.LoadATMs(string.Format("WHERE a.`location` = {0}", locationID));
```

Θα επιστρέψει την τιμή του αποτελέσματος της function “LoadATMs” από τον μεσάζοντα (η class που βρίσκεται μεταξύ του μοντέλου και της βάσης δεδομένων).

Εάν υπάρχουν ATM στη συγκεκριμένη τοποθεσία στη βάση δεδομένων, η function θα επιστρέψει μια μεταβλητή τύπου “Λίστας<ATM>” με τα στοιχεία των ATM που βρέθηκαν. Κάθε εγγραφή της λίστας αντιστοιχεί σε ένα ATM (μεταβλητή τύπου “ATM”).

Εάν δεν υπάρχουν ATM στη συγκεκριμένη τοποθεσία στη βάση δεδομένων, η function θα επιστρέψει μια κενή “Λίστα<ATM>”.

```
"WHERE a.`location` = {0}", locationID
```

Φτιάχνουμε ένα κομμάτι από το Query που θα εκτελεστεί και ψάχνουμε μόνο για τα ATM που βρίσκονται στην περιοχή με ID ίσο με αυτό της τιμής της μεταβλητής locationID. Η τιμή της στήλης “location” στον πίνακα των “ATM”, αντιστοιχεί στη τιμή της στήλης “ID” του πίνακα “Locations”.

5.4.15 Επεξήγηση του κώδικά που βρίσκεται στο Model\ Database.cs

TestConnection.cs

```
public bool TestConnection(string server, string username, string password)
```

Για τη function αυτή θα πρέπει να δωθούν τρία ορίσματα, η διεύθυνση του διακομιστή της βάσης δεδομένων, το όνομα του χρήστη και ο κωδικός του χρήστη που έχουν πρόσβαση στη βάση δεδομένων. Και τα τρία θα πρέπει να είναι αλφαριθμητικά.

```
return this._sql.TestConnection(server, username, password);
```

Θα επιστρέψει την τιμή του αποτελέσματος της function “TestConnection” από τον μεσάζοντα (η class που βρίσκεται μεταξύ του μοντέλου και της βάσης δεδομένων). Εάν υπάρχουν τα στοιχεία είναι σωστά και η σύνδεση με τη βάση δεδομένων είναι επιτυχημένη, η function θα επιστρέψει «true» αλλιώς θα επιστρέψει «false».

5.4.16 Επεξήγηση του κώδικά που βρίσκεται στο Model\ Holidays.cs

ModelHolidays

Εδώ απλά υπάρχουν οι function που καλούνται για να σβήσεις μια αργία και για να διαβάσεις τις αργίες από τη βάση.

```
class ModelHolidays
```

```
private MySQL _sql = new MySQL();
```

```
private string _table = "atm_holidays";
```

```
public bool DeleteItems(int itemID)
```

```
return this._sql.Delete(this._table, itemID);
```



```
public List<Holiday> LoadItems()
return this._sql.LoadHolidays();
```

5.4.17 Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelLogin.cs

Login.cs

```
public User VerifyCredentials(string username, string password)
```

Για τη function αυτή, θα πρέπει να δοθούν δυο ορίσματα, το όνομα του χρήστη και ο κωδικός του. Και τα δυο πρέπει να είναι αλφαριθμητικά.

```
return this._sql.LoadUser(username, password);
```

Θα επιστρέψει την τιμή του αποτελέσματος της function “LoadUser” από τον μεσάζοντα (η class που βρίσκεται μεταξύ του μοντέλου και της βάσης δεδομένων). Εάν υπάρχουν τα στοιχεία του χρήστη στη βάση δεδομένων, η function θα επιστρέψει μια μεταβλητή τύπου “User” με τα στοιχεία του χρήστη. Εάν δεν υπάρχουν τα στοιχεία του χρήστη στη βάση δεδομένων, η function θα επιστρέψει μια μεταβλητή τύπου “User”, αλλά δεν θα έχει κανένα στοιχείο του χρήστη (θα είναι κενή). Με βάση τα παραπάνω, μπορούμε να ελέγξουμε εάν τα στοιχεία που έδωσε ο χρήστης είναι σωστά, εάν το User.id έχει κάποια τιμή εκτός του 0 (μηδέν).

5.4.18 Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelMain.cs

sum_withdrawals.cs

```
public bool SaveWithdrawal(Dictionary<string, object> data)
```

Ισχύει ό,τι και στη function LoadItems.cs στο atm_atms.cs. Η μόνη διαφορά είναι η εξής:

Επειδή θέλουμε κατά την αποθήκευση μιας ανάληψης να ενημερώνεται αυτόματα και το αντίστοιχο ATM, κάνουμε τα παρακάτω:

```
if (this._sql.Insert("atm_withdrawals", string.Join(", ", into.ToArray()), string.Join(", ", values.ToArray())))
```

Εάν η αποθήκευση της ανάληψης ήταν επιτυχημένη, τότε:

```
return this._sql.Update("atm_atms", string.Format("`sum_withdrawals` = `sum_withdrawals` + {0}", values[2].ToString().Trim(new char[] { '\ ' })), int.Parse(values[0].Trim(new char[] { '\ ' })));
```

Ανανέωσε την τιμή της στήλης “sum_withdrawals” του ATM με ID «τάδε». Η νέα τιμή θα ισούται με την παλιά τιμή συν την τιμή της ανάληψης (sum_withdrawals = sum_withdrawals + XXX).

Επειδή τα δεδομένα βρίσκονται μέσα σε « ’ », φροντίζουμε να τα αφαιρέσουμε χρησιμοποιώντας την εντολή Trim(). Η εντολή Trim() μπορεί να αφαιρέσει κάποιους χαρακτήρες από την αρχή και το τέλος ενός αλφαριθμητικού. Οι χαρακτήρες αυτοί δίνονται σαν όρισμα στην εντολή. Στην προκειμένη περίπτωση, θέλουμε να αφαιρέσουμε τα “ ’ ”, οπότε δίνουμε σαν όρισμα το “ \ ”. Επειδή οι χαρακτήρες δηλώνονται με μονό “αυτάκι”, δεν μπορούμε να έχουμε ως όρισμα το “ ” (τρια “αυτάκια στη σειρά), για αυτό χρησιμοποιούμε το “ \ ”.

5.4.19 Επεξήγηση του κώδικά που βρίσκεται στο Model\ ModelReports.cs

Για τον υπολογισμό της πρόβλεψης των συναλλαγών την επόμενη εργάσιμη ημέρα χρησιμοποιούμε τη μέθοδος του σταθμισμένου κινητού μέσου όρου με δείγμα τα δεδομένα των τελευταίων 18εργάσιμων ημερών. Στη συνέχεια χωρίζουμε το δείγμα σε τρία ίσα μέρη σχηματίζοντας υποσύνολα με 6 τιμές το καθένα. Από κάθε υποσύνολο αφαιρέσουμε τις δύο ακραίες τιμές και υπολογίσουμε το μέσο όρο δίνοντας μεγαλύτερη βαρύτητα στις πιο πρόσφατες παρατηρήσεις. Οι συντελεστές που θα χρησιμοποιούμε είναι 0.7, 0.2 και 0.1 σε αντιστοιχία από τις πιο πρόσφατες στις παλαιότερες τιμές. Ακολουθεί επεξήγηση του κώδικά που χρησιμοποιήσαμε και βρίσκεται στο Model\ ModelReports.cs

Εντοπίζουμε για κάθε ATM ξεχωριστά (WHERE atm = {0}) τις τιμές των αναλήψεων από τον πίνακα atm_withdrawals και τις ταξινομούμε κατά φθίνουσα σειρά με σκοπό να ξεχωρίσουμε τις 18 πιο πρόσφατες τιμές.

```
SELECT `date` FROM `atm_withdrawals` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
AS t ORDER by `date`)
```

Εντοπίζουμε την μικρότερη ημερομηνία αναλήψεων από το υποσύνολο που έχουμε δημιουργήσει

```
SELECT MIN(`date`)
```

Συγκεντρώνουμε τις αντίστοιχες τιμές των αναλήψεων από τον πίνακα atm_withdrawals που έχουν ημερομηνία μεγαλύτερη ή ίση με τη μικρότερη ημερομηνία του υποσύνολου

```
SELECT `ammount` FROM atm_withdrawals AS w WHERE w.`atm` = {0} AND w.`date` >= (
```

Ταξινομούμε το δείγμα μας κατά αύξουσα σειρά με βάση την ημερομηνία και επιλέγουμε τις 6 πρώτες τιμές

```
ORDER BY `date` ASC LIMIT 6)
```

Από το δείγμα αφαιρούμε τη μέγιστη και την ελάχιστη τιμή , διαιρούμε δια τέσσερα για να βρούμε το μέσο όρο και τελικά πολλαπλασιάζουμε με το συντελεστή 0.1 Το αποτέλεσμα το ονομάζουμε ως wt01

```
as wt01
```

Για να εντοπίσουμε τις έξι επόμενες τιμές από το υποσύνολο έχουμε δημιουργήσει, εργαζόμεστε με τον ίδιο τρόπο αλλά στο τελικό δείγμα ξεχωρίζουμε τις έξι τιμές αρχίζοντας να μετράμε από την έκτη και μετά. Τον μέσο όρο τον πολλαπλασιάζουμε με το συντελεστή 0.2.

```
SELECT ((SUM(`ammount`) - MIN(`ammount`) - MAX(`ammount`))/4) * 0.2 *{1} AS su FROM (
```

```
SELECT `ammount` FROM atm_withdrawals AS w WHERE w.`atm` = {0} AND w.`date` >= (
```

```
SELECT MIN(`date`) FROM (
```

```
SELECT `date` FROM `atm_withdrawals` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
```

```
AS t ORDER by `date`)
```

```
ORDER BY `date` ASC LIMIT 6, 6)
```

```
as wt02
```

Για να εντοπίσουμε τις έξι πιο πρόσφατες τιμές από το υποσύνολο έχουμε δημιουργήσει, εργαζόμεστε με παρόμοιο τρόπο, αλλάζοντας τον τρόπο ταξινόμησης. Αφού έχουμε ξεχωρίσει τις τιμές των αναλήψεων από τον πίνακα atm_withdrawals, τις ταξινομούμε με φθίνουσα σειρά ώστε να συλλέγουμε τις έξι πρώτες. Τον μέσο όρο τον πολλαπλασιάζουμε με το συντελεστή 0.7

```
SELECT ((SUM(`ammount`) - MIN(`ammount`) - MAX(`ammount`))/4) * 0.7 *{1} AS su FROM (
```

```
SELECT `ammount` FROM atm_withdrawals AS w WHERE w.`atm` = {0} AND w.`date` >= (
```

```
SELECT MIN(`date`) FROM (
```

```
SELECT `date` FROM `atm_withdrawals` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
```

```
AS t ORDER by `date`)
```

```
ORDER BY `date` DESC LIMIT 6)
```

```
as wt03
```

Με τον ίδιο τρόπο εργαζόμεστε ώστε να παράγουμε την πρόβλεψη των καταθέσεων χρησιμοποιώντας τον πίνακα atm_deposits.

Συλλογή δεδομένων για τις παλιότερες έξι (από το δείγμα των δεκαοκτώ τιμών) και τον πολλαπλασιασμό του μέσου όρου με το 0.1.

```
SELECT ((SUM(`ammount`) - MIN(`ammount`) - MAX(`ammount`))/4) * 0.1 *{1} AS su FROM (
```

```
SELECT `ammount` FROM atm_deposits AS w WHERE w.`atm` = {0} AND w.`date` >= (
```

```
SELECT MIN(`date`) FROM (
```

```
SELECT `date` FROM `atm_deposits` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
```

```
AS t ORDER by `date`)
ORDER BY `date` ASC LIMIT 6)
as dt01
```

Συλλογή δεδομένων για τις παλιότερες έξι επόμενων (από το δείγμα των δεκαοκτώ τιμών) και τον πολλαπλασιασμό του μέσου όρου με το 0.2.

```
SELECT ((SUM(`ammount`) - MIN(`ammount`) - MAX(`ammount`))/4) * 0.2 *{1} AS su FROM (
SELECT `ammount` FROM atm_deposits AS w WHERE w.`atm` = {0} AND w.`date` >= (
SELECT MIN(`date`) FROM (
SELECT `date` FROM `atm_deposits` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
AS t ORDER by `date`)
ORDER BY `date` ASC LIMIT 6, 6)
```

```
as dt02
```

Συλλογή δεδομένων για τις παλιότερες έξι πρόσφατων (από το δείγμα των δεκαοκτώ τιμών) και τον πολλαπλασιασμό του μέσου όρου με το 0.7.

```
SELECT ((SUM(`ammount`) - MIN(`ammount`) - MAX(`ammount`))/4) * 0.7 *{1} AS su FROM (
SELECT `ammount` FROM atm_deposits AS w WHERE w.`atm` = {0} AND w.`date` >= (
SELECT MIN(`date`) FROM (
SELECT `date` FROM `atm_deposits` WHERE atm = {0} ORDER BY `date` DESC LIMIT 18)
AS t ORDER by `date`)
ORDER BY `date` DESC LIMIT 6)
```

```
as dt03
```

Η εφαρμογή για μα προτίνει ένα ATM για τροφοδοσία ή όχι θα πρέπει να εξετάσει την επαρκεία των χρημάτων στις κασσετες των αναλήψεων και ταυτόχρονα την επάρκεια του χώρου στο κουτί των καταθέσεων.

Από το ποσό της αρχικής τροφοδοσίας αφαιρεί το άθροισμα των αναλήψεων έως και την χθεσινή εργάσιμη μέρα και στη συνέχεια αφαιρεί τα τρία ποσά που έχουν υπολογιστεί νωρίτερα όπου άθροισμα τους μας δίνουν την πρόβλεψη της μέρας.

Στο άθροισμα των καταθέσεων έως και την χθεσινή εργάσιμη μέρα προσθέτει τα τρία ποσά που έχουν υπολογιστεί νωρίτερα όπου το άθροισμα τους μας δίνουν την πρόβλεψη της μέρας.

```
WHERE a.`id` = {0} HAVING (
a.`refill_ammount` - a.`sum_withdrawals` - `forecast_withdrawals01` - `forecast_withdrawals02` -
`forecast_withdrawals03`) < a.`withdraw_limit`
OR (
a.`sum_deposits` + `forecast_deposits01` + `forecast_deposits02` + `forecast_deposits03`) >
a.`deposit_limit`"
, atm.id)
```

Όταν η εφαρμογή καλείται να δώσει πρόβλεψη καταθέσεων και αναλήψεων εξετάζει εάν η επόμενη εργάσιμη μέρα είναι Σάββατο ή Κυριακή ή Αργία ώστε να πολλαπλασιάσει ανάλογα και την τιμή της πρόβλεψης. Ακολουθεί επεξήγηση του κώδικά που χρησιμοποιήσαμε και βρίσκεται στο Model\ModelReports.cs

Δημιουργούμε μία λίστα (μεταβλητού μεγέθους) για το αντικείμενο ATM, όπου φορτώνουμε όλα τα ATM που περιέχονται στη βάση μας. Στη συνέχεια δημιουργούμε μία νέα κενή λίστα όπου θα αποθηκευτούν τα ATM τα οποία θα χρειάζονται τροφοδοσία.

```
public List<ATM> LoadItems()
{
    List<ATM> atms = this._sql.LoadATMs();
```

```
List<ATM> ret = new List<ATM>();
```

Στη σημερινή μέρα προσθέτουμε άλλη μία και δίνουμε την τιμή 1 σε μία μεταβλητή με το όνομα *multi*.

```
DateTime day = DateTime.Now.AddDays(1);
```

```
int multi = 1;
```

Εάν η σημερινή μέρα είναι Πέμπτη (4) η μεταβλητή *multi* παίρνει τη τιμή 3, ενώ εάν η σημερινή μέρα είναι Παρασκευή (5) η μεταβλητή *multi* παίρνει τη τιμή 2 και στη σημερινή μέρα προσθέτουμε άλλες τρεις ημέρες με σκοπό να περάσουμε από τη Παρασκευή στη Δευτέρα.

```
if ((int)DateTime.Now.DayOfWeek == 4)
```

```
{
```

```
    multi = 3;
```

```
}
```

```
else if ((int)DateTime.Now.DayOfWeek == 5)
```

```
{
```

```
    multi = 2;
```

```
    day = DateTime.Now.AddDays(3);
```

```
}
```

Σε περίπτωση που η αυριανή μέρα είναι αργία τότε θα συμπεριλαμβάνεται στο πίνακα των αργιών *atm_holidays*. Σε αυτή τη περίπτωση ενεργοποιείται το *while* όπου εξετάσει πόσες φορές εμφανίζεται μία ημερομηνία στο πίνακα των αργιών (>0 τότε η ζητούμενη μέρα είναι αργία) και αυξάνει τη μεταβλητή *multi* κατά 1.

```
while (this._sql.Count("atm_holidays", string.Format("WHERE `date` = \"{0}-{1}-{2}\"", day.Year, day.Month, day.Day)) > 0)
```

```
{
```

```
    multi = multi + 1;
```

Για να συνεχίσει τον έλεγχο της ύπαρξης αργίας και την μεθεπόμενη μέρα αλλά ταυτόχρονα να προσπεράσει τα σαββατοκύριακα εργάζεται με τον ίδιο τρόπο που αναφέραμε και παραπάνω.

```
f ((int)day.DayOfWeek == 5)
```

```
{
```

```
    day = day.AddDays(3);
```

```
}
```

```
else
```

```
{
```

```
    day = day.AddDays(1);
```

```
}
```

5.4.20 Επεξήγηση του κώδικά που βρίσκεται στο Model\ModelVendors.cs

LoadItems(string prefecture)

Ψάχνει να βρει τους συνεργάτες (ομοίως και με τα ATM) που ανήκουν σε μια συγκεκριμένη περιοχή.

```
LEFT JOIN `atm_locations` AS l ON v.`location` = l.`id` WHERE l.`prefecture` LIKE \"{0}\"", prefecture)
```

Το παραπάνω query ενώνει δύο πίνακες από τη βάση δεδομένων. Αυτή των συνεργατών και αυτή των περιοχών. Κάθε συνεργάτης έχει μια στήλη στη βάση δεδομένων που αντιστοιχεί στην περιοχή που ανήκει. Σύμφωνα με αυτή την τιμή και με το ID της κάθε περιοχής γίνεται η ένωση των πινάκων. Έπειτα το query ψάχνει να βρει ποιοι κατασκευαστές ανήκουν στην περιοχή "prefecture".

Το παραπάνω γίνεται ώστε να μπορείς να δεις τους συνεργάτες που ανήκουν σε ένα νομό και όχι ακριβώς σε ποια περιοχή (πχ/ όλους τους κατασκευαστές στην Αθήνα). Μιας και οι νομοί δεν έχουν δικό τους πίνακα στη βάση δεδομένων, δεν έχουν και κάποιο συγκεκριμένο ID. Οπότε αναγκαστικά τους ψάχνουμε με το όνομά τους.

```
public List<Vendor> LoadItems(string prefecture)
return this._sql.LoadVendors(string.Format("LEFT JOIN `atm_locations` AS l ON v.`location` = l.`id`
WHERE l.`prefecture` LIKE \"{0}\"", prefecture));
```

6. Συμπέρασμα

Στα πλαίσια της παρούσας πτυχιακής εργασίας πραγματοποιήθηκε καταγραφή και ανάλυση των συναλλαγών που πραγματοποιούνται σε αυτόματες ταμιακές μηχανές (ATM), της μεθοδολογίας για την πρόβλεψη ζήτησης χρημάτων.

Στα προηγούμενα κεφάλια εν συντομία έγινε μία εισαγωγή γύρω από τη διαχείριση χρήματος σε αυτόματες ταμιακές μηχανές με δυνατότητα καταθέσεων. Αναφέρθηκαν μία συνοπτική ιστορική αναδρομή της ποιάς των ATM ώστε να κατανοήσουμε τη σημαντικότητα τους στο πέρασμα το χρόνων. Δώσαμε ιδιαίτερη βάση στις δυνατότητες των μηχανημάτων και μέσω πρόσφατων δημοσκοπήσεων παρουσιάσαμε την ικανοποίηση των καταναλωτών και την πρόθεση τους να αντικαταστήσουν τους παραδοσιακούς τρόπους συναλλαγών (κισσέ τράπεζας) με τη χρήση ATM BNA. Έγινε αναφορά στα πληροφοριακά συστήματα, στις διαδικασίες ανάπτυξης τους καθώς και ανάλυση της αξίας των βάσεων δεδομένων, αναφέρθηκαν τρόποι ασφάλειας, τεχνικές και περιορισμοί που εάν ακολουθηθούν σε οποιαδήποτε εφαρμογή, αυτομάτων την θέτουν ικανή και αξιόπιστη ώστε να διαχειρίζεται μεγάλο όγκο δεδομένων.

Εκτιμώντας με ακρίβεια τη ζήτηση για προϊόντα (τραπεζογραμμάτια) στο βραχυπρόθεσμο και μεσοπρόθεσμο μέλλον, μία τράπεζα μπορεί να υπολογίσει όλες τις παραμέτρους που απαιτούνται για τη σωστή διαχείριση των αποθεμάτων της, καθώς η υπερβολική ύπαρξη χρημάτων όσο και η έλλειψη στοιχίζει σε κάθε πιστωτικό ίδρυμα. Είναι ένα δύσκολο εγχείρημα που απαιτεί το συνδυασμό ποσοτικών τεχνικών αλλά και πρακτικής εμπειρίας σχετικά με το προϊόν. Εφόσον εφαρμοσθούν οι κατάλληλες μέθοδοι πρόβλεψης ζήτησης, για τον υπολογισμό των παραμέτρων χρησιμοποιηθούν τα σωστά δεδομένα, μπορεί να επιτευχθεί σημαντική μείωση του επιπέδου των αποθεμάτων ακόμη και με ταυτόχρονη αύξηση του επιπέδου διαθεσιμότητας.

Με την εκπόνηση της παρούσας εργασίας έγιναν σημαντικές προσπάθειες για την επίλυση των προβλημάτων στο κομμάτι της πρόβλεψης ζήτησης κατάλληλης ημέρας ανεφοδιασμού μηχανημάτων αυτόματης ανάληψης και κατάθεσης. Αναλύθηκαν και παρουσιάστηκαν οι μέθοδοι ζήτησης αλλά και τρία παραδείγματα από διαφορετικές περιοχές λειτουργίας ATM με την αντίστοιχη προβλεπόμενη ζήτηση χρημάτων μέσω αναλήψεων και ζήτηση διαθέσιμου χώρου στο χρηματοκιβώτιο των ATM για πραγματοποίηση καταθέσεων.

Τέλος υλοποιήθηκε και παρουσιάστηκε μια εφαρμογή διαχείρισης χρήματος με σκοπό τον εντοπισμό της επιθυμητής ημέρας τροφοδοσίας. Η εφαρμογή βασίστηκε στη μέθοδο ζήτησης με τη χρήση του σταθμισμένου κινητού μέσου καθώς αυτή αναδείχτηκε ως ο πιο αξιόπιστος τρόπος πρόβλεψης μετά τον υπολογισμό του σφάλματος των προτεινόμενων μεθόδων. Στην εφαρμογή χρησιμοποιηθεί η τεχνική του αντικειμενοστραφούς προγραμματισμού (object oriented programming) και συγκεκριμένα η γλώσσα προγραμματισμού C# καθώς και η χρήση του προγράμματος Microsoft Visual Studio 2010 και με την αρχιτεκτονική ανάπτυξης που προσφέρει το μοντέλο MCV (Model View Control) ενώ για την επικοινωνία του χρήστη με τη βάση δεδομένων θα χρησιμοποιηθεί το σύστημα MySql.

7. Βιβλιογραφία

Βιβλία

- 1) Ramakrishnan R., Gehrke J.(2002), “Συστήματα Διαχείρισης Βάσεων Δεδομένων”, Εκδόσεις Τζιόλα
- 2) Δημήτρης Φωλίνας,2006, Ολοκληρωμένα πληροφοριακά συστήματα διαχείρισης επιχειρηματικών πόρων, Εκδόσεις Ανικούλα.
- 3) Ι. Ε. Σαμουηλίδης, Μ. Ανδρουλάκης, Α. Αραμπατζή, Δ. Ασκούνης, Ν. Κοσματόπουλος, 2000, Συστήματα Διοίκησης και Πληροφοριών, έκδοση Εθνικού Μετσόβιου Πολυτεχνείου
- 4) Jeffrey Jarrett, 2002, Μέθοδοι προβλέψεων για οικονομικές – Επιχειρηματικές αποφάσεις, Εκδόσεις Gutenberg.
- 5) Ζαχαροπούλου Χρυσούλα, 2003, Στατιστική μέθοδοι – εφαρμογές, Εκδόσεις Σοφία.
- 6) Κωνσταντίνος Ρόντος και Ευστράτιος Παπάνης,2006, Στατιστική Έρευνα-Μέθοδοι και Εφαρμογές, Εκδόσεις Ι.Σιδέρης
- 7) Καρακώστα Κ.Ιωάννη και Τσίρου Π.Σίσσυ – Σπθριδούλα, 2007, Πιστωτική Κάρτα, Εκδόσεις Νομική Βιβλιοθήκη ΑΕΒΕ
- 8) Λυμπερόπουλος Κ.,1994, Στρατηγικό τραπεζικό μάρκετινγκ, Εκδόσεις Interbooks
- 9) Νεβράντζας Βάιος – Γερμανός,2011, Μελέτης Εργαστηρίου C#, ΤΕΙ Λάρισας
- 10) Τμήμα Τεχνολογίας Πληροφορικής & Τηλεπικοινωνιών, ΤΕΙ Λάρισας
- 11) Παναγιωτόπουλου Ιωάννη – Χρήστου, Διαδικτυακός Προγραμματισμός: C#, εκδόσεις Σταμούλης
- 12) John Sharp, Visual C# 2008 βήμα βήμα, εκδόσεις Κλειδάριθμος
- 13) Dubois, Paul, Hinz, Stefan και Pedersen, Carsten, 2006, Ο επίσημος οδηγός MySQL 5, εκδόσεις Γκιούρδας Μ
- 14) Elmasri & Navathe,2000, Θεμελιώσεις Αρχές Συστημάτων Βάσεων Δεδομένων, Εκδόσεις ΔΙΑΥΛΟΣ)
- 15) Χρήστος Σκουρλάς,2001, Σχεσιακές Βάσεις Δεδομένων, Εκδόσεις Νέων Τεχνολογιών
- 16) C.J. Date, Εισαγωγή στα Συστήματα Βάσεων Δεδομένων, 1996, Εκδόσεις Κλειδάριθμος)

Διαλέξεις

- 1) Δριτσάκης Νικόλαος και Πετράκης Ανδρέας για το μάθημα Στατιστικές Μέθοδοι με Εντατική Χρήση Η/Υ του Πανεπιστημίου Μακεδονίας
- 2) Μονάδα Προβλέψεων & Στρατηγικής του Εθνικού Μετσόβιου Πολυτεχνείου
- 3) Διαχείριση και πρόβλεψη ζήτηση, για το μάθημα Διοικηση Παραγωγής και Συστημάτων Υπηρεσιών του Εθνικού Μετσόβιου Πολυτεχνείου
- 4) Operations Management, ICBS Business College

Άρθρα

- 1) Γρηγόρη Κιοσέογλου και Ανδρέα Δημητρίου, Ανάλυση προβλέψεων: Στατιστική Μεθοδολογία για την αξιολόγηση μοντέλων πρόβλεψης με ποιοτικά δεδομένα στις κοινωνικές επιστήμες, Τμήμα Φιλοσοφίας, Παιδαγωγικής και Ψυχολογίας Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
- 2) Εφης Καραγεωγίου, Ηλεκτρονική Τράπεζα: Στροφή από τις τράπεζες στα εναλλακτικά δίκτυα Καλύτερη εξυπηρέτηση για τον καταναλωτή από την Ελληνική Ένωση Τραπεζών,
- 3) Καρακέρεζης Α, 1999, Συγχρονες τάσεις στρατηγικών στην ηλεκτρονική τραπεζική, Manager Περιοδικό της ΕΕΔΕ
- 4) Summary of website research roll results, 2011, από European ATM Security Team (EAST)
- 5) Financial Institution Atms: The roles they play in the 21st century από ATM marketplace.com
- 6) The Benefits of Real-Time ATM Management από ATM marketplace.com
- 7) ATM Software: Trends Analysis for 2007-2008 από ATM marketplace.com
- 8) ATM Software: Trends Analysis for 2009 από ATM marketplace.com
- 9) 2010 ATM Software Trends and Analysis από ATM marketplace.com
- 10) 2011 ATM Software Trends and Analysis από ATM marketplace.com

- 11) 2012 ATM Software Trends and Analysis από ATM marketplace.com
- 12) Reliable Wireless Connections Enable ATM Expansion από ATM marketplace.com
- 13) Transforming Bank Branches Through Video Banking από ATM marketplace.com
- 14) Three Ways ATM Processing Can Maximize Interchange Revenues από ATM marketplace.com
- 15) The Solution - Pay Your Bills Quickly and Efficiently! από ATM marketplace.com
- 16) The Solution - Tickets! Tickets! Get Your Tickets! από ATM marketplace.com
- 17) Protect Your Financial Institution from ATM Fee Reg E Lawsuits από GetBranded.com

Διαδικτυακές ιστοσελίδες

- 1) <http://www.markets.com/el/education/technical-analysis>
- 2) <http://gr.investing.com/education>
- 3) http://library.tee.gr/digital/kma/kma_m1274.pdf (σημειώσεις στη Διαχείριση Αποθεμάτων)
- 4) <http://en.wikipedia.org>
- 5) <http://www.ops.gr/Ergorama>
- 6) <http://www.mou.gr>
- 7) <http://www2.e-yliko.gr>
- 8) <http://infolab.cs.unipi.gr/pre-eclass/courses>
- 9) <http://dide.flo.sch.gr/Plinet>
- 10) <http://www.dblab.upatras.gr>
- 11) <http://www.atmmarketplace.com>
- 12) <http://www.euretirio.com>
- 13) <http://www.brinksblog.com>
- 14) <http://www.commercialsecuritydevices.com/el>
- 15) http://en.wikipedia.org/wiki/Microsoft_SQL_Server
- 16) <http://www.connectionstrings.com/sql-server-2008>
- 17) <http://csharp.net-informations.com/data-providers/csharp-sql-serverconnection>
- 18) <http://msdn.microsoft.com>
- 19) <http://msdn.microsoft.com/el-gr/library>
- 20) <http://www.microsoft.com/visualstudio>
- 21) <http://msdn.microsoft.com/vstudio>
- 22) <http://www.microsoft.com/sqlserver>