



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση σχεδιασμός και υλοποίηση κινητής εφαρμογής (mobile application) και διασύνδεση της με τα πληροφοριακά συστήματα μιας επιχείρησης

ΓΚΙΚΑ ΠΑΝΑΓΙΩΤΑ

ΚΟΥΓΚΟΥΛΗ ΚΩΝΣΤΑΝΤΙΝΑ

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΒΗΣΣΑΡΙΩΝ ΦΕΡΕΝΤΙΝΟΣ

ΠΥΡΓΟΣ | 2018

ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η πτυχιακή εργασία με τίτλο:

**«Ανάλυση σχεδιασμός και υλοποίηση κινητής εφαρμογής
και διασύνδεση της με τα πληροφοριακά συστήματα μιας
επιχείρησης»**

Του φοιτητή του τμήματος ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ του Τ.Ε.Ι Δυτικής Ελλάδας

1. ΓΚΙΚΑ ΠΑΝΑΓΙΩΤΑ
2. ΚΟΥΓΚΟΥΛΗ ΚΩΝΣΤΑΝΤΙΝΑ

παρουσιάστηκε δημόσια και εξετάστηκε στο Τμήμα ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

στις

___/___/___

Ο ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ

Ο ΠΡΟΕΔΡΟΣ ΤΟΥ
ΤΜΗΜΑΤΟΣ

ΒΗΣΣΑΡΙΩΝ ΦΕΡΕΝΤΙΝΟΣ

Δρ. ΙΩΑΝΝΗΣ ΚΟΥΓΙΑΣ

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια στην οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Ακόμα δηλώνω ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία και ότι θα αναλάβω πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχτεί ότι δεν μου ανήκει.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ Ι

ΑΜ

ΥΠΟΓΡΑΦΗ

.....

.....

.....

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ ΙΙ

ΑΜ

ΥΠΟΓΡΑΦΗ

.....

.....

.....

ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να εκφράσουμε τις ευχαριστίες μας στον Επιβλέποντα καθηγητή μας κ. Φερεντίνο Βησσαρίων για τη συνεργασία και την πολύτιμη συμβολή του στην ολοκλήρωση της παρούσας Πτυχιακής Εργασίας.

Ακόμη, ευχαριστούμε όλους τους καθηγητές και τις καθηγήτριες της Σχολής μας για τις γνώσεις που μας προσέφεραν.

Τέλος, ένα μεγάλο ευχαριστώ στις οικογένειες μας για την στήριξη που μας έδωσαν όλα αυτά τα χρόνια των σπουδών μας.

ΠΡΟΛΟΓΟΣ

«Η διαφορετικότητα των πληροφοριακών συστημάτων έγκειται στην ενσωμάτωση όλων των λειτουργιών που χρειάζονται για να δημιουργηθεί ένα ενιαίο, ενοποιημένο σύστημα, παρά μια ομάδα ξεχωριστών εφαρμογών». Alshawi (2004)

Τα πληροφοριακά συστήματα (Li, Z., et. al, 2006) είναι ένα σύνολο διαδικασιών ανθρωπίνου δυναμικού και αυτοματοποιημένων συστημάτων τα οποία έχουν ως στόχο την εκτέλεση των βασικών εργασιών όπως είναι η συλλογή, η εγγραφή, η ανάκτηση, η επεξεργασία, η αποθήκευση και η ανάλυση των εισερχόμενων πληροφοριών. Κύριος στόχος τους είναι η απλούστευση και την αμεσότητα των διεργασιών και της οργάνωσης και διαχείρισης των επιμέρους κλάδων μιας επιχείρησης με σκοπό τη δημιουργία ενός ολοκληρωμένου συστήματος.

Η παρούσα εργασία έχει ως στόχο την εισαγωγή των βασικών τομέων των πληροφοριακών συστημάτων σε μια κινητή εφαρμογή. Η εφαρμογή αυτή έχει όλα τα πλεονεκτήματα ενός πληροφοριακού συστήματος και δίνει στο χρήστη τη δυνατότητα να μπορέσει να ελέγξει τις βασικές διαδικασίες της εταιρίας του μέσα από την κινητή του συσκευή.

ΠΕΡΙΛΗΨΗ

Τα πληροφοριακά συστήματα ERP έχουν εμφανιστεί στην αγορά τα τελευταία χρόνια και έχουν ως στόχο να συγχωνεύσουν τις ανάγκες της επιχείρησης, να μαζέψουν όλα τα απαραίτητα στοιχεία και πληροφορίες τα οποία χρειάζεται εκείνη για τη λειτουργία της και να ομαλοποιήσουν τις πτυχές της, με σκοπό την καλύτερη λειτουργία αυτής. Απώτερος σκοπός των συστημάτων αυτών είναι να μπορέσουν να καταφέρουν να παρέχουν στους υπαλλήλους και τα στελέχη μιας επιχείρησης όλα τα δεδομένα τα οποία είναι απαραίτητα για την εκτέλεση μιας διαδικασίας. Η διαδικασία αυτή μπορεί να αφορά την οργάνωση, τη διοίκηση ή οποιοδήποτε άλλο κομμάτι της επιχείρησης το οποίο θα τη βοηθήσει να ικανοποιήσει το στόχο της.

Στο πλαίσιο της παρούσας εργασίας θα αναπτυχθεί μια εφαρμογή για περιβάλλον Android η οποία έχει ως στόχο να αποτελέσει την εισαγωγή ανάπτυξης εφαρμογών πληροφοριακών συστημάτων. Το κυρίως αντικείμενο της εφαρμογής θα είναι η παροχή των βασικών υπηρεσιών, όπως είναι οι πωλήσεις, η αποθήκη, τα οικονομικά στοιχεία και τα έξοδα της επιχείρησης, στο χρήστη με το κατάλληλο επίπεδο ασφαλείας φυσικά και μην εκθέτοντας την επιχείρηση σε κίνδυνο προσβολής των δεδομένων της.

Λέξεις κλειδιά: Android εφαρμογές, Πληροφοριακά συστήματα, πληροφοριακά συστήματα σε κινητή εφαρμογή.

ABSTRACT

Enterprise Resources Planning Systems – ERP information systems have been on the market in recent years and aim at merging the needs of the business, gathering all the necessary information and information needed for its operation and normalizing its aspects in order to be better operated. The ultimate goal of these systems is to be able to provide employees and business executives with all the data necessary needed in order to carry out a process or make an important decision. This process may involve organization, administration or any other part of the business that will help to achieve its goals.

This thesis presents the development of an Android application that aims to introduce new data into the category of information systems applications. The main purpose of the implementation concerns the integration of basic services such as sales, warehouse, financial data and business expenses into the application, which will be used by the user. This application will have the appropriate security level, of course, in order not to expose the business to the risk of infringing its data or lose these important data.

Keywords: Android application, ERP systems, Android application with ERP system integrated.

ΠΕΡΙΕΧΟΜΕΝΑ

| | |
|---|----|
| ΕΥΧΑΡΙΣΤΙΕΣ | 6 |
| ΠΡΟΛΟΓΟΣ..... | 8 |
| ΠΕΡΙΛΗΨΗ | 10 |
| ABSTRACT | 11 |
| ΠΕΡΙΕΧΟΜΕΝΑ | 13 |
| ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ | 15 |
| ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ..... | 16 |
| 1. ΕΙΣΑΓΩΓΗ | 17 |
| 1.1 Αντικείμενο διπλωματικής εργασίας | 17 |
| 1.2 Σύνοψη διπλωματικής εργασίας..... | 18 |
| 1.3 Ιστορία των πληροφοριακών συστημάτων ERP | 19 |
| 1.4 Χαρακτηριστικά ERP συστημάτων | 20 |
| 2. ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ | 26 |
| 2.1 Λογισμικό και εφαρμογές Android..... | 26 |
| 2.2 Περιβάλλον Android Studio..... | 29 |
| 2.3 Java γλώσσα προγραμματισμού..... | 30 |
| 2.4 Στοιχεία Extensible Markup Language (XML) | 31 |
| 2.4.1 Εμφάνιση (Layout)..... | 31 |
| 2.4.2 Εισαγωγή κουμπιού (Button) | 33 |
| 2.4.3 Εισαγωγή κειμένου (Text View)..... | 35 |
| 2.4.4 Ενέργειες (Activity) | 35 |
| 2.4.5 Android Manifest | 37 |
| 3. ΣΤΟΙΧΕΙΑ ΣΧΕΔΙΑΣΜΟΥ ΕΦΑΡΜΟΓΗΣ..... | 38 |
| ΣΕ ΠΕΡΙΒΑΛΛΟΝ ANDROID..... | 38 |
| 3.1 Απαιτήσεις δημιουργίας εφαρμογής | 38 |
| 3.2 Λειτουργία εφαρμογής | 40 |
| 3.3 Σχεδιασμός εφαρμογής | 41 |
| 3.4 Ανάπτυξη εφαρμογής..... | 44 |

| | |
|--|----|
| 4. ΕΦΑΡΜΟΓΗ..... | 51 |
| 4.1 Αρχική οθόνη | 52 |
| 4.2 Υπηρεσία “Αποθήκη” | 53 |
| 4.3 Υπηρεσία “Υπάλληλοι” | 54 |
| 4.4 Υπηρεσία “Πωλήσεις” | 55 |
| 4.5 Υπηρεσία “Λογιστικά” | 56 |
| 5. ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ | 58 |
| ΑΝΑΦΟΡΕΣ | 59 |

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

| | |
|--|----|
| Πίνακας 1: Πίνακας εκδόσεων λογισμικού Android (JohnHorton, 2015)..... | 28 |
| Πίνακας 2: Παρουσίαση και ανάλυση απαιτήσεων εφαρμογής. | 42 |
| Πίνακας 3: Στοιχεία πινάκων βάσης δεδομένων..... | 42 |

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

| | |
|--|----|
| Εικόνα 1: Συστατικά συστήματος ERP..... | 21 |
| Εικόνα 2: Λειτουργίες επιχείρησης τις οποίες διαχειρίζεται ένα ERP σύστημα. | 22 |
| Εικόνα 3: Στατιστικά στοιχεία για το λογισμικό Android.(NiallMcCarthy, 2014) | 28 |
| Εικόνα 4: Απεικόνιση πλατφόρμας Android Studio. | 30 |
| Εικόνα 5: Διαβάθμιση στοιχείων εμφάνισης. | 32 |
| Εικόνα 6: Διαβάθμιση Layoutστο Android.(Griffiths, D., &Griffiths, D., 2017)..... | 33 |
| Εικόνα 7: Εικόνα κουμπιού σε μια εφαρμογή. | 34 |
| Εικόνα 8: Εμφάνιση Text view σε μια εφαρμογή..... | 35 |
| Εικόνα 9: Διάγραμμα μεθόδων στο Activity.(Griffiths, D., &Griffiths, D., 2017)..... | 36 |
| Εικόνα 10: Βήματα σχεδιασμού εφαρμογής. | 38 |
| Εικόνα 11: Παρουσίαση τελικών πινάκων βάσης δεδομένων. | 42 |
| Εικόνα 12: Παρουσίαση συνάρτησης onCreate..... | 43 |
| Εικόνα 13: Εισαγωγή πεδίων - τιμών σε κάθε πίνακα. | 43 |
| Εικόνα 14: Χρήση συναρτήσεων get για τη λήψη στοιχείων από τους πίνακες και την παρουσίαση των δεδομένων. Εκτελούνται MySQLερωτήματα. | 44 |
| Εικόνα 15: Αρχική οθόνη εφαρμογής παρουσιασμένη μέσα από το πρόγραμμα AndroidStudio. | 45 |
| Εικόνα 16: Στοιχεία αρχικής οθόνης για την είσοδο του χρήστη. | 45 |
| Εικόνα 17: Στοιχεία χρηστών και ο τρόπος δόμησής τους στη βάση δεδομένων. | 46 |
| Εικόνα 18: Διαδικασία ταυτοποίησης στοιχείων της εφαρμογής και εμφάνιση των κατηγοριών εφόσον γίνει επιτυχημένη είσοδος χρήστη. | 50 |
| Εικόνα 19: Παρουσίαση οθόνης η οποία χρησιμοποιείται σε όλες τις κατηγορίες για την παρουσίαση των στοιχείων της εκάστοτε κατηγορίας..... | 48 |
| Εικόνα 20: Κώδικας που χρησιμοποιείται για την παρουσίαση των στοιχείων τα οποία λαμβάνονται από τη βάση δεδομένων. | 48 |
| Εικόνα 21: Αρχείο το οποίο φέρει τις πληροφορίες της υπηρεσίας Αποθήκη..... | 49 |
| Εικόνα 22: Εισαγωγή στοιχείων από την υπηρεσία Αποθήκη στη βάση δεδομένων. Τα στοιχεία αφορούν τα προϊόντα τα οποία βρίσκονται στην αποθήκη της επιχείρησης..... | 50 |
| Εικόνα 23: Εισαγωγή στοιχείων για την είσοδο του χρήστη..... | 52 |
| Εικόνα 24: Επιλογές από υπηρεσίες που δίνονται στην εφαρμογή. | 53 |
| Εικόνα 25: Υπηρεσία Αποθήκη. | 54 |
| Εικόνα 26: Υπηρεσία Υπάλληλοι. | 55 |
| Εικόνα 27: Υπηρεσία Πωλήσεις. | 56 |
| Εικόνα 28: Υπηρεσία Λογιστικά..... | 60 |

1. ΕΙΣΑΓΩΓΗ

Κάθε επιχείρηση αναπτύσσει ή χρησιμοποιεί τα ήδη υπάρχοντα πληροφοριακά συστήματα του εμπορίου. Ένα πληροφοριακό σύστημα είναι η σύνδεση της επιχείρησης με τις διάφορες λειτουργίες της και παράλληλα υποστηρίζει τον απομακρυσμένο έλεγχο αυτής προκειμένου να μπορέσει ο χρήστης να ελέγξει τις λειτουργίες της και να μπορέσει εύκολα να διαχειριστεί την επιχείρηση, αλλά και επιμέρους στοιχεία αυτής.

Το παρόν κεφάλαιο παρουσιάζει τους στόχους αλλά και το αντικείμενο εργασίας της παρούσας εφαρμογής, καθώς και τους τρόπους ανάπτυξης και τα μέσα τα οποία χρησιμοποιήθηκαν προκειμένου να αναπτυχθεί μια εφαρμογή η οποία θα ικανοποιεί όλες τις απαιτήσεις οι οποίες έχουν τεθεί. Απαραίτητο στοιχείο το οποίο παρουσιάζεται στο παρόν κεφάλαιο είναι η κατανόηση των ERP πληροφοριακών συστημάτων, της δομής και της λειτουργίας τους. Ο αναγνώστης στο κεφάλαιο αυτό θα πρέπει να κατανοήσει τις βασικές έννοιες λειτουργίας των συστημάτων αυτών, προκειμένου να μπορέσει να καταλάβει το αντικείμενο και την ανάπτυξη της εφαρμογής που παρουσιάζονται στην υπόλοιπη εργασία.

1.1 Αντικείμενο διπλωματικής εργασίας

Το 2009 οι Laudon και Price (2009) έδωσαν στα Συστήματα Διαχείρισης Επιχειρηματικών Πόρων (ERP – Enterprise Resource Planning) έναν ορισμό ο οποίος τα καθορίζει ως «ένα πληροφοριακό σύστημα το οποίο σχετίζεται με τις διαδικασίες ολόκληρης της επιχείρησης, και συνδέει όλες αυτές τις διαδικασίες με τους επιχειρηματικούς στόχους των εταιριών, ενώνοντας παράλληλα όλα τα επιμέρους τμήματα των επιχειρήσεων μέσα από διάφορες λειτουργικές διεργασίες. Μέσω της συγχώνευσης αυτής επιτυγχάνεται η μια πιο άμεση, ακριβής και έγκαιρη μετάδοση των απαραίτητων πληροφοριών στο εσωτερικό της επιχείρησης. Οι πληροφορίες αυτές αφορούν οποιοδήποτε στοιχείο το οποίο συνδέεται είτε με κάποιο προϊόν, είτε με κάποιο πελάτη ή διαδικασία της επιχείρησης.»

Ο παραπάνω ορισμός ωστόσο δεν είναι μοναδικός, καθώς αρκετοί επιστήμονες με το πέρασμα των χρόνων και την ανάπτυξη των συστημάτων αυτών προσπάθησαν να διευρύνουν τον ορισμό και να ορίσουν περισσότερες πτυχές, ώστε να παρουσιάζεται καλύτερα η λειτουργικότητα των συστημάτων αυτών και η καθοριστική συμβολή της στην επιχείρηση. Τα πληροφοριακά συστήματα καλύπτουν ένα μεγάλο μέρος των επιχειρησιακών αναγκών στη σημερινή πραγματικότητα, με στόχο τη μελλοντική εξ' ολοκλήρου λειτουργία και παρουσίαση αυτής μέσα από το περιβάλλον της κάθε εφαρμογής. Στόχος των πληροφοριακών συστημάτων είναι η διοικητική και γενική λειτουργία και διαχείριση της επιχείρησης μέσα από ένα πληροφοριακό σύστημα, το οποίο θα συλλέγει πληροφορίες, θα τις

επεξεργάζεται και θα ενημερώνει το σύστημα και τους χρήστες ανάλογα με αυτές, προκειμένου να πάρουν πιο δομημένες αποφάσεις.

Τα τελευταία χρόνια η ανάπτυξη των πληροφοριακών συστημάτων είναι ραγδαία, καθώς οι επιχειρήσεις αναπτύσσονται όλο και πιο γρήγορα για λόγους ανταγωνισμού και ανάπτυξης. Επιπλέον, οι ρυθμοί ζωής και τεχνολογικής ανάπτυξης είναι πιο μεγάλοι, με αποτέλεσμα να αυξάνονται και οι ανάγκες και οι απαιτήσεις των χρηστών. Το κενό το οποίο υπάρχει αυτή τη στιγμή στην αγορά αφορά την απομακρυσμένη χρήση των πληροφοριακών συστημάτων ή την επιμέρους κάλυψη των αναγκών των χρηστών και συνάμα την απομακρυσμένη λειτουργία και εκτέλεση κάποιων βασικών διεργασιών.

Το κενό αυτό καλείται να καλύψει η παρούσα εργασία μέσα από την ανάπτυξη μιας εφαρμογής για έξυπνες συσκευές, όπου θα μπορεί ο κάθε χρήστης και ελέγξει τα χαρακτηριστικά της αποθήκης του, του λογιστηρίου της επιχείρησης και γενικότερα τις οικονομικές απολαβές και στοιχεία τα οποία είναι απαραίτητα για την ευρύτερη λειτουργία της επιχείρησης. Οι παραπάνω τομείς προέκυψαν από την εξέταση των συστημάτων ERP και των απαιτήσεων που έχουν οι εταιρίες από τα συστήματα αυτά.

Προκειμένου να προφυλαχθούν τα στοιχεία της επιχείρησης, πρέπει να υπάρχει υψηλή ασφάλεια στη λειτουργία της εφαρμογής. Στην αρχική οθόνη της εφαρμογής ο χρήστης θα πρέπει να εισάγει τα στοιχεία του τα οποία θα ταυτοποιηθούν προκειμένου να έχει πρόσβαση στις διαδικασίες και τα στοιχεία της επιχείρησής του. Η βάση δεδομένων η οποία θα χρησιμοποιηθεί για την ταυτοποίηση και την εισαγωγή των στοιχείων θα είναι τοπική, ώστε να μπορεί να επικοινωνεί ο χρήστης με αυτή οπουδήποτε και αν βρίσκεται και να μην απαιτείται η συνεχής σύνδεσή του με το διαδίκτυο.

1.2 Σύνοψη διπλωματικής εργασίας

Το κείμενο το οποίο συνοδεύει και αναλύει την ανάπτυξη της εφαρμογής της εργασίας αυτής, χωρίζεται ομοιόμορφα σε πέντε κεφάλαια. Κάθε κεφάλαιο αναλύει ένα διαφορετικό σημαντικό στοιχείο ανάπτυξης της εφαρμογής, και έχει διαφορετικό στόχο καθώς εξηγεί λεπτομερώς τις διαδικασίες και τις τεχνικές οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της παρούσας εργασίας.

Το παρόν κεφάλαιο αποτελεί εισαγωγή στο αντικείμενο, τους σκοπούς και στόχους της εργασίας. Η εισαγωγή αυτή είναι απαραίτητη καθώς πρέπει ο αναγνώστης να κατανοήσει το θέμα της εργασίας και να μπορέσει να ακολουθήσει την ανάλυση και την παρουσίαση της εφαρμογής.

Το δεύτερο κεφάλαιο αρχικά αναλύει γενικά τους τρόπους και τα μέσα τα οποία χρησιμοποιούνται για την ανάπτυξη μιας εφαρμογής σε περιβάλλον Android. Επιπλέον, παρουσιάζονται χαρακτηριστικά του κώδικα τα οποία έχουν χρησιμοποιηθεί για την ανάπτυξη της εφαρμογής, όπως είναι τα δομικά στοιχεία αυτής. Τα στοιχεία αυτά είναι απαραίτητα για να στηρίξουν την ανάπτυξη της εφαρμογής και να κατανοήσει ο αναγνώστης τις βασικές αρχές ανάπτυξης αυτής.

Το τρίτο κεφάλαιο παρουσιάζει λεπτομερώς το σχεδιασμό της εφαρμογής. Στο σχεδιασμό της εφαρμογής συγκαταλέγεται η ανάλυση των απαιτήσεων, η αρχιτεκτονική της

εφαρμογής και ο τελικός σχεδιασμός της, στοιχεία τα οποία είναι απαραίτητα προκειμένου να υλοποιηθεί μια ολοκληρωμένη εφαρμογή που να παράγει σωστά αποτελέσματα.

Εφόσον έχει παρουσιαστεί ο σχεδιασμός και η λογική η οποία θα ακολουθηθεί για την ανάπτυξη της εφαρμογής, στο επόμενο (τέταρτο) κεφάλαιο παρουσιάζεται η λειτουργία και η λειτουργικότητα της εφαρμογής. Επίσης, μέσα από εικόνες, παρουσιάζεται η υλοποίηση της εφαρμογής.

Τέλος, το πέμπτο κεφάλαιο αναγράφει τα συμπεράσματα τα οποία προέκυψαν από την ανάλυση όλων των στοιχείων που έχουν παρουσιαστεί στην υπόλοιπη εργασία και καταγράφει επίσης τα συμπεράσματα τα οποία προκύπτουν από τη μελέτη της εργασίας.

1.3 Ιστορία των πληροφοριακών συστημάτων ERP

Τα πληροφοριακά συστήματα ERP μπορεί να αναπτύχθηκαν ραγδαία και να εξαπλώθηκαν σχεδόν σε όλες τις χώρες τις τελευταίες δεκαετίες, ωστόσο η εμφάνισή τους ξεκίνησε πολύ νωρίτερα. Τα συστήματα αυτά πρωτοεμφανίστηκαν τη δεκαετία του 1960, όπου οι επιχειρήσεις ξεκίνησαν να παρουσιάζουν αυξημένη ανάπτυξη και να ενδιαφέρονται για τη μηχανογράφηση των λειτουργιών τους και την καταγραφή των διαδικασιών και των στοιχείων τους σε αρχεία ώστε να μπορούν να ανατρέξουν σε αυτά οποιαδήποτε στιγμή επιθυμούν. Η ανάπτυξη και η ζήτηση αυτή των χρηστών αρχικά οδήγησαν στην ανάπτυξη ξεχωριστών τεχνολογικών εφαρμογών που αφορούσαν το λογιστήριο και τη μισθοδοσία των υπαλλήλων μιας επιχείρησης, καθώς επίσης και τον έλεγχο των αποθεμάτων, προκειμένου οι βασικές λειτουργίες της εταιρίας να μπορούν να ενημερώνονται και να συντονίζονται.

Τα πρώτα πληροφοριακά συστήματα τα οποία αναπτύχθηκαν ήταν τα συστήματα σχεδιασμού απαιτήσεων υλικών (Material Requirements Planning - MRP) τα οποία ξεκίνησαν να εισάγονται σε όλες τις επιχειρήσεις με σκοπό την κάλυψη των αναγκών τους και αποτέλεσαν τα μελλοντικά συστήματα μηχανογράφησης των επιχειρήσεων αυτών.

Τότε ο στόχος των εν λόγω συστημάτων ήταν η υλοποίηση μιας ολοκληρωμένης λύσης για κάθε επιχείρηση η οποία θα είχε όσο το δυνατόν περισσότερα στοιχεία συγκεντρωμένα τα οποία θα αφορούσαν την ανάπτυξη της επιχείρησης αυτής και θα τη βοηθούσαν στη συγκέντρωση των διαδικασιών της. Η αυξημένη ανάπτυξη των απαιτήσεων αυτών οδήγησε στη δημιουργία των συστημάτων διαχείρισης επιχειρησιακών πόρων. Παράλληλα με την εξέλιξη της τεχνολογίας, χρησιμοποιώντας μόνο έναν υπολογιστή και καταγράφοντας όλες τις απαραίτητες πληροφορίες που αφορούσαν την αποθήκη, τους πελάτες αλλά και κάθε απαραίτητη διαδικασία, μπορούσε η επιχείρηση να έχει συγκεντρωμένα όλα τα απαραίτητα στοιχεία για τη λειτουργία και την επέκτασή της.

Το 1980 εμφανίστηκε ανανεωμένο το παλιό σύστημα MRP με ονομασία MRPII, η επέκταση του οποίου συνέδεε μεταξύ τους διεργασίες που αφορούσαν τον προγραμματισμό παραγωγής, τον έλεγχο παραγωγής, την κοστολόγηση, και τις προμήθειες. Στα νέα συστήματα τα οποία αναπτύχθηκαν από τη δεκαετία αυτή και μετά, χρησιμοποιήθηκαν βάσεις δεδομένων (databases) για την αποθήκευση πληροφοριών. Επιπλέον, τα νέα συστήματα προσπαθούν να ενώσουν βασικές επιχειρηματικές διαδικασίες και κυρίως την οικονομική διαχείριση και την παραγωγή μιας επιχείρησης, με απώτερο σκοπό την καλύτερη

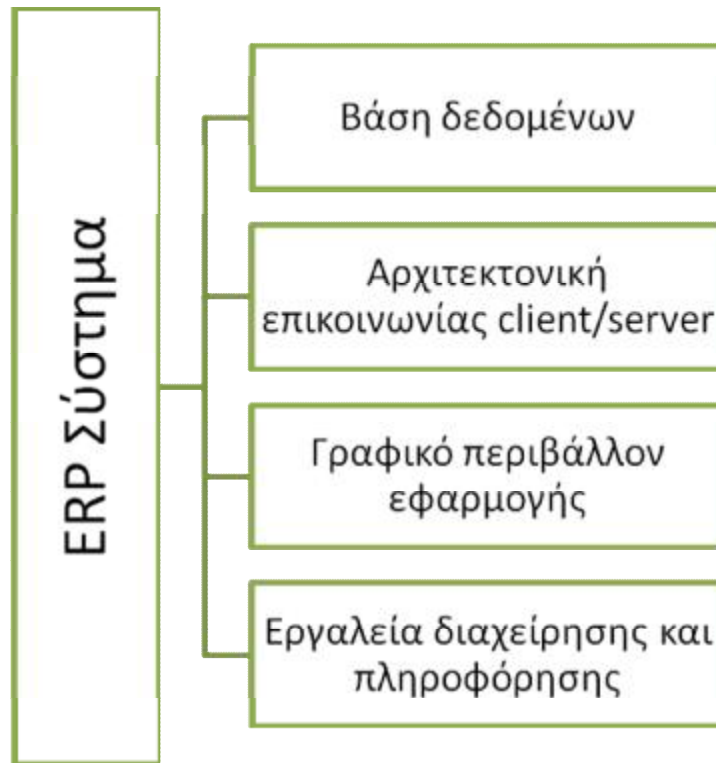
διαχείριση αυτών και την άμεση λήψη αποφάσεων σε περιπτώσεις όπου αυτό κρίνεται αναγκαίο. (Furht, B. & Escalante, A, 2010, Wagner, B., & Monk, E., 2008)

Η δεκαετία του 1990 σηματοδότησε την αλλαγή στη ροή της ιστορίας των συστημάτων επιχειρηματικών πόρων, καθώς εμφανιστήκαν τα συστήματα διαχείρισης επιχειρησιακών πόρων (Enterprise Resources Planning Systems – ERP), τα οποία είναι η επέκταση των MRP II. Η επέκταση των συστημάτων MRPII έγινε σε πολλούς τομείς, όπου τα νέα συστήματα κατάφεραν να συγχωνεύσουν λειτουργίες όπως ο σχεδιασμός προϊόντων, η διαχείριση αποθηκών, η διαχείριση έργων και ανθρώπινου δυναμικού και την εμπορική και την οικονομική διαχείριση μιας εταιρείας ή ενός οργανισμού. Τα νέα αυτά συστήματα φέρουν τον όρο «ολοκληρωμένα συστήματα πληροφορικής», και κύριος στόχος τους είναι να καλύψουν όλες τις λειτουργικές διεργασίες μιας επιχείρησης και την ενοποίηση όλων των λειτουργιών αυτής, προκειμένου να επιτευχθούν οι στόχοι της εταιρείας και να γίνει σωστά η οργάνωσή της.

1.4 Χαρακτηριστικά ERP συστημάτων

Στην προηγούμενη ενότητα παρουσιάστηκαν τα στοιχεία τα οποία είναι απαραίτητα για την παρουσίαση και την αναγνώριση του στόχου των πληροφοριακών συστημάτων ERP. Η παρούσα ενότητα επεκτείνει την παρουσίαση των συστημάτων αυτών παρουσιάζοντας τους τρόπους και τα μέσα, μέσα από τα οποία μπορούν να επιτευχθούν οι στόχοι της επιχείρησης, αλλά και η οργάνωσή της που είναι το βασικότερο στοιχείο των συστημάτων.

Όπως αναφέρθηκε και στην παραπάνω ενότητα, τα πληροφοριακά συστήματα ERP χρησιμοποιούν μια βάση δεδομένων στην οποία αποθηκεύονται όλα τα στοιχεία της επιχείρησης. Η λειτουργία των συστημάτων αυτών σχετίζεται με την εκπλήρωση των στόχων της εταιρείας που σχετίζονται με την αποτελεσματικότερη οργάνωση της παραγωγής ή παροχής υπηρεσιών και των σταδίων που προηγούνται και έπονται αυτών, αλλά και με την ποιότητα και την καλύτερη εξυπηρέτηση του πελάτη. Η βάση δεδομένων η οποία βρίσκεται στα συστήματα αυτά και αποτελεί το βασικό δομικό τους στοιχεία, περιέχει σχετικές πληροφορίες με κάθε επιχείρηση, πληροφορίες οι οποίες αφορούν την οργανωτική δομή της επιχείρησης καθώς και τις λειτουργίες της. Στην Εικόνα 1 παρουσιάζονται τα συστατικά στοιχεία τα οποία συνθέτουν ένα σύστημα ERP:



Εικόνα 1: Συστατικά συστήματος ERP.

Με τη βάση δεδομένων του κάθε συστήματος συνδέονται όλα τα επιμέρους στοιχεία και λειτουργίες αυτού, οι οποίες είναι σχετικές με τα επίπεδα τα οποία πρέπει να ελέγχονται μέσα από μια επιχείρηση. Τα στοιχεία αυτά παρουσιάζονται στην Εικόνα 2. Κάθε μια λειτουργία και ένας τομέας από τους παραπάνω που συνθέτουν ένα ERP σύστημα αντιστοιχεί σε έναν τομέα της επιχείρησης, ο οποίος υπάρχει εξολοκλήρου στο εσωτερικό αυτής.



Εικόνα 2: Λειτουργίες επιχείρησης τις οποίες διαχειρίζεται ένα ERP σύστημα.

Πιο συγκεκριμένα, οι τομές αυτοί περικλείουν τα ακόλουθα χαρακτηριστικά (Wagner, B., & Monk, E., 2008):

- **Διαχείριση πελατών**

Στον τομέα αυτό ανήκουν οι επικοινωνιακές σχέσεις της επιχείρησης οι οποίες αφορούν το κοινωνικό της προφίλ και τις σχέσεις τις με το εξωτερικό περιβάλλον. Για να μπορέσει μια επιχείρηση να αυξήσει το οικονομικό της όφελος αλλά και τους πελάτες της, θα πρέπει να είναι σε θέση να παρουσιάσει σωστά το εταιρικό της προφίλ και να είναι συνεπείς στις υποχρεώσεις της όσον αφορά το κοινωνικό της πλαίσιο, στοχεύοντας στην προσέγγιση νέων πελατών αλλά και στην ευχαρίστηση των υφιστάμενων. Ο τομέας αυτός στοχεύει στις πωλήσεις, το marketing και στην ανάπτυξη της επιχείρησης και τη δικτύωσή της αναφορικά με τις κοινωνικές σχέσεις.

- **Διαχείριση ανθρωπίνου δυναμικού**

Η διαχείριση ανθρωπίνου δυναμικού εμπεριέχει όλες οι διαδικασίες οι οποίες αφορούν το ανθρώπινο δυναμικό μιας επιχείρησης, και πιο συγκεκριμένα όλες τις οργανωτικές και διοικητικές διαδικασίες αυτού. Στον τομέα αυτό αποθηκεύονται και εμπεριέχονται όλα τα στοιχεία τα οποία είναι απαραίτητα για την διεξαγωγή όλων των εργασιών οι οποίες αφορούν το προσωπικό της εταιρίας, όπως είναι οι άδειες τους, οι καρτέλες με την εξέλιξή τους στην

εταιρίας, η εκπαίδευση του δυναμικού αυτού, οι προσλήψεις νέων ατόμων, και γενικότερα κάθε στοιχείο που είναι απαραίτητο για την οργάνωση και την ανάπτυξη του δυναμικού της επιχείρησης.

- **Οικονομική διαχείριση**

Τα στοιχεία τα οποία εμπεριέχονται στον τομέα αυτό αφορούν τη λογιστική αλλά και την οικονομική διαχείριση της εταιρίας. Στο κομμάτι αυτό συγκαταλέγονται όλα τα οικονομικά της στοιχεία, όπως είναι διάφορες πληρωμές και έξοδα αυτής, η μισθοδοσία των υπαλλήλων της, λογαριασμοί, αλλά και γενικότερα η διαχείριση των εισροών και των εκροών μιας επιχείρησης. Χαρακτηριστικά παραδείγματα των στοιχείων αυτών είναι τα εξής:

- Ø Πληρωτέοι λογαριασμοί
 - ο Πληρωμές προμηθευτών.
 - ο Ενσωμάτωση με τη διαχείριση εντολών αγορών.
- Ø Εισπρακτέοι λογαριασμοί
 - ο Βραχυπρόθεσμα οικονομικά προτερήματα, από πωλήσεων με πίστωση.
 - ο Ενσωμάτωση με τη γενική λογιστική.
- Ø Γενική Λογιστική
 - ο Διαχείριση όλων των λογαριασμών με βάση προκαθορισμένο λογιστικό σχέδιο.
- Ø Μισθοδοσία
 - ο Πληρωμές υπαλλήλων.

- **Διαχείριση πωλήσεων και προμηθειών**

Στην κατηγορία αυτή υπάγονται όλα τα στοιχεία τα οποία ασχολούνται με τη διαχείριση πωλήσεων, τις προμήθειες, τις διανομές τις οποίες κάνει μια επιχείρηση – εταιρία, τον προγραμματισμό αυτών, καθώς και ότι αφορά τις διαδικασίες οι οποίες εκτελούνται από την αρχή της διαδικασίας πωλήσεων, όπως οι παραγγελίες, μέχρι να φτάσει το εμπόρευμα στην επιχείρηση που το ζητά ή τον καταναλωτή. Χαρακτηριστικές διαδικασίες είναι οι ακόλουθες:

- Ø Ανάλυση ζήτησης.
 - ο Πρόβλεψη ζήτησης μεμονωμένων πελατών - επιχειρήσεων.
- Ø Προγραμματισμός απαιτήσεων διανομών.
 - ο Αιτήματα μεταφορών και ζήτησης τροφοδοσίας από την αποθήκη προς διάφορους χώρους της επιχείρησης.
 - ο Δημιουργία εντολών μεταφοράς και μετακίνησης με βάση τη ζήτηση.
- Ø Διαχείριση προμηθευτών
 - ο Διαχείριση βασικών στοιχείων προμηθευτών.
 - ο Εντολές προμήθειας, προσφοράς, παραγγελίες, κοστολόγηση, έλεγχος παραλαβών κ.α.
- Ø Διαχείριση μεταφορών
 - ο Μεταφορικοί πόροι και μέσα τα οποία βοηθούν στη διανομή των προϊόντων.

- **Διαχείριση παροχής υπηρεσιών**

Στη διαχείριση παροχής υπηρεσιών υπάρχουν οι διεργασίες οι οποίες αφορούν την ευρύτερη εξυπηρέτηση πελατών, τη διαχείριση προδιαγραφών και η διαχείριση υπηρεσιών τεχνικής υποστήριξης. Σε κάθε μια από αυτές τις ενότητες εμπεριέχονται στοιχεία τα οποία αφορούν τις υπηρεσίες που ασχολούνται με τις παραγγελίες των προϊόντων, τις εντολές επιστροφής, και άλλες υπηρεσίες. Πιο συγκεκριμένα, ο τομέας αυτός χωρίζεται στα ακόλουθα μέρη και εμπεριέχει την:

- Ø Εξυπηρέτηση πελατών
 - ο Διαχείριση επιστροφών.
 - ο Υπόσχεση παραγγελιών.
 - ο Τιμολόγηση προϊόντων και εκπτώσεις.
- Ø Διαχείριση προδιαγραφών
 - ο Εξειδικευμένες προδιαγραφές
 - ο Σύνδεση με διαχείριση τεχνικών προδιαγραφών
 - ο Πληροφορίες για την κοστολόγηση, υπόσχεση παραγγελιών κλπ.
- Ø Διαχείριση υπηρεσιών τεχνικής υποστήριξης
 - ο Αποστολή προειδοποιήσεων, δεξιότητες πόρων, αναφορά βλαβών.

Η ασφάλεια στο επίπεδο εφαρμογής απαιτεί μεγάλες προσπάθειες των εμπειρογνομόνων ERP για να προσφέρουν έναν αποτελεσματικό τρόπο για να εξασφαλιστούν τα επιχειρηματικά δεδομένα και οι διαδικασίες. Οι τεχνικοί θα επιλέξουν επίσης να ενεργοποιήσουν / απενεργοποιήσουν τις λειτουργίες ασφαλείας που παρέχει ο προμηθευτής βάσης δεδομένων σύμφωνα με τη συνολική λύση ασφαλείας. Οι VandeRiet, Janssen και Gruijter (1998) σε έρευνά τους συνόψισαν ορισμένες από τις πτυχές ασφαλείας σε ένα σύστημα ERP.

Το στοιχείο που χαρακτηρίζει ένα σύστημα ως ERP είναι η φιλοδοξία του να προσφέρει μια "ολοκληρωμένη" λύση όσο το δυνατόν περισσότερο όσον αφορά τη διαχείριση του πλήρους φάσματος υπηρεσιών. Επίσης, μια πραγματικά "πλήρης" λύση φέρει τα εξής πλεονεκτήματα: η ανταλλαγή δεδομένων γίνεται αποτελεσματικά σε όλη την επιχείρηση, η εξάλειψη δαπανηρών και δύσκολων στη διατήρηση ενοποιήσεων και η επιτάχυνση της αποδοτικότητας και των καμπυλών μάθησης μέσω μιας κοινής διασύνδεσης λογισμικού.

Υπάρχει όμως ο κίνδυνος για τη συλλογή αυτή. Ενώ είναι φυσικά ευκολότερο να μεταφερθούν τα δεδομένα σε ένα ενιαίο αποθηκευτικό χώρο, είναι ακόμα πιο σημαντικό να μην υπάρξει κάποια παρεμβολή στο χώρο αυτό. Ουσιαστικά, όσο ευρύτερη είναι η εμβέλεια του συστήματος ERP, τόσο πιο σημαντικό είναι να γίνει απογραφή και να αντιμετωπιστούν όλοι οι κίνδυνοι ασφαλείας. Οι απειλές και τους κινδύνους ERP είναι αρκετοί και επιφυλάσσουν πολλές φορές άσχημα στοιχεία για μια επιχείρηση. Πιο συγκεκριμένα, τα δεδομένα τα οποία έχει μια επιχείρηση στη διάθεσή της αφορούν την οργάνωση, τη δομή, τη λειτουργία της, τις οικονομικές της απολαβές και ροές και σε περίπτωση επίθεσης όλα αυτά τα στοιχεία μπορεί να βρεθούν σε λάθος άτομα. Οι κίνδυνοι που ελλοχεύουν σε ένα τέτοιο σύστημα είναι οι εξής:

- Κίνδυνος 1: Το παρωχημένο μη υποστηριζόμενο λογισμικό μπορεί να οδηγήσει σε σφάλματα και ζητήματα ολοκλήρωσης.: Ο κίνδυνος αυτός υπάρχει καθώς οι παλαιότερες εκδόσεις λογισμικού δεν θα είναι συμβατές και δεν θα ενσωματωθούν με νεότερα προϊόντα. Σε περιπτώσεις όπου το λογισμικό δεν

υποστηρίζεται πλέον, είναι μικρότερες οι πιθανότητες να μπορέσει κάποιος να το αναβαθμίσει ή να το ενημερώσει εάν συμβεί κάποιο σφάλμα.

- Κίνδυνος 2: Η ανεπαρκής δυνατότητα αναφοράς μπορεί να οδηγήσει σε εξωτερική αναφορά και απώλεια ελέγχου δεδομένων. Ένας από τους κορυφαίους λόγους που οδήγησαν σε νέες αγορές ERP είναι ότι η έλλειψη λειτουργικότητας έχει προκαλέσει στους χρήστες να μην μπορούν να έχουν πρόσβαση και να αναλύουν δεδομένα στο σύστημά τους.
- Κίνδυνος 3: Το τεχνικό προσωπικό και οι πάροχοι έχουν πρόσβαση για να κάνουν αλλαγές μεγάλης κλίμακας στη συμπεριφορά του προγράμματος. Αρκετοί οργανισμοί επικεντρώνουν τη διαχείριση κινδύνων των επιχειρήσεων κυρίως στις εξωτερικές απειλές, τις διαδικασίες των κέντρων δεδομένων και την ασφάλεια των τελικών χρηστών. Ωστόσο, όσον αφορά την άμεση πρόσβαση του προγραμματιστή λογισμικού στο σύστημα, αυτός είναι ένας τομέας που συνήθως αξίζει περισσότερο έλεγχο. (Wei She, Bhavani Thuraisingham, 2007).

Οι διαδικασίες οι οποίες θα ενταχθούν στην ανάπτυξη της παρούσας εφαρμογής έχουν ελεγχθεί διεξοδικά και σχετίζονται με τους παραπάνω τομείς. Τονίζεται στο σημείο αυτό ότι οι διαδικασίες αυτές είναι οι βασικές και κάθε σύστημα μπορεί να εμπεριέχει περισσότερες ή λιγότερες από αυτές, ανάλογα με τους στόχους του, το αντικείμενο εργασίας του και τη χρήση του πληροφοριακού συστήματος την οποία επιθυμεί να κάνει. Επιπλέον, σημειώνεται ότι κάθε πληροφοριακό σύστημα παραμετροποιείται στα στεγανά της εκάστοτε επιχείρησης και έχει δυνατότητες επέκτασης με βάση την επέκταση αυτής.

2. ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ

Έχοντας παρουσιάσει και μελετήσει το σκοπό της παρούσας εργασίας, ο αναγνώστης, συνεχίζοντας την ανάγνωση του εγγράφου αυτού, θα πρέπει να εισέλθει σε πιο βαθιές και βασικά τεχνολογικές έννοιες οι οποίες αφορούν τον προγραμματισμό και την υλοποίηση μιας εφαρμογής σε περιβάλλον Android. Το κεφάλαιο αυτό παρουσιάζει λεπτομερώς όλα τα στοιχεία τα οποία έχουν χρησιμοποιηθεί για την ανάπτυξη της εφαρμογής και όλες τις απαραίτητες συνδέσεις αυτών, με στόχο να κατατοπίσει τον αναγνώστη σχετικά με τη δομή και τον τρόπο με τον οποίο αναπτύχθηκε η εφαρμογή. Σημειώνεται ότι στην παρούσα εργασία το λογισμικό το οποίο έχει αναπτυχθεί, χρησιμοποιείται μόνο σε συσκευές που φέρουν το λογισμικό Android. Η επιλογή αυτού βασίστηκε αποκλειστικά στο γεγονός ότι το χρησιμοποιεί η πλειονότητα των έξυπνων/φορητών συσκευών.

Στόχος της εφαρμογής που αναπτύχθηκε στα πλαίσια της παρούσας πτυχιακής εργασίας είναι να μπορεί ο κάθε υπάλληλος μιας εταιρείας να μπορεί να ενημερωθεί για τις ανάγκες τις επιχείρησης χωρίς περιορισμούς και από όλες τις έξυπνες συσκευές (tablets, κινητά, smart watches, κλπ).

2.1 Λογισμικό και εφαρμογές Android

Το λογισμικό Android αποτελεί το σημαντικότερο και ευρέως χρησιμοποιούμενο λειτουργικό σύστημα τα τελευταία χρόνια καθώς χρησιμοποιείται κατά κόρον σε συσκευές κινητής τηλεφωνίας και γενικότερα σε έξυπνες συσκευές. Η χρήση του είναι εύκολη και με αρκετά πλεονεκτήματα, όπως είναι η βελτιστοποίηση, η ευκολία στη χρήση του, η αυξημένη αποδοτικότητά του και η επεκτασιμότητά του. Το κέλυφος του συστήματος είναι παρόμοιο με εκείνο του λειτουργικού συστήματος Linux, κάτι το οποίο του επιτρέπει να έχει καλύτερη δομή και να χρησιμοποιεί καλύτερα τους πόρους του εκάστοτε συστήματος. Η πρώτη παρουσίαση του λογισμικού έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance. (John Horton, 2015).

Για την ανάπτυξη των εφαρμογών χρησιμοποιείται η γλώσσα προγραμματισμού Java. Η κοινότητα του λογισμικού έχει αναπτύξει μια πληθώρα βιβλιοθηκών προκειμένου να υποστηρίξουν τις διάφορες απαιτήσεις των χρηστών και προγραμματιστών για την κάλυψη όλων των αναγκών. Με τη βοήθεια διαφόρων εργαλείων επιτρέπεται στους προγραμματιστές η εύκολη και άμεση συγγραφή κώδικα για τη δημιουργία μιας εφαρμογής. Μερικά από τα πλεονεκτήματα του λογισμικού αυτού είναι τα ακόλουθα:

- Αποτελεί εργαλείο ανάπτυξης ανοιχτού κώδικα.
- Εύκολη ανάπτυξη εφαρμογών.
- Ανοιχτή και ενημερωμένη κοινότητα προγραμματιστών.
- Εύκολο περιβάλλον χρήσης.
- Ισχυρό λειτουργικό σύστημα με αυξημένη επεκτασιμότητα και ασφάλεια. Η εταιρία ανάπτυξης του λογισμικού είναι η Google η οποία παρέχει αξιοπιστία και άμεση ενημέρωση του λογισμικού για τυχόν σφάλματα.
- Αυξημένος αριθμός βιβλιοθηκών με αρκετές δυνατότητες, όπως η χρήση πολυμέσων.

Από την αρχική ημερομηνία χρήσης του και έπειτα, έχουν αναπτυχθεί αρκετές εκδόσεις του Android. Στον πίνακα που ακολουθεί παρουσιάζονται οι εκδόσεις αυτές καθώς και τα στοιχεία τους. (John Horton, 2015)

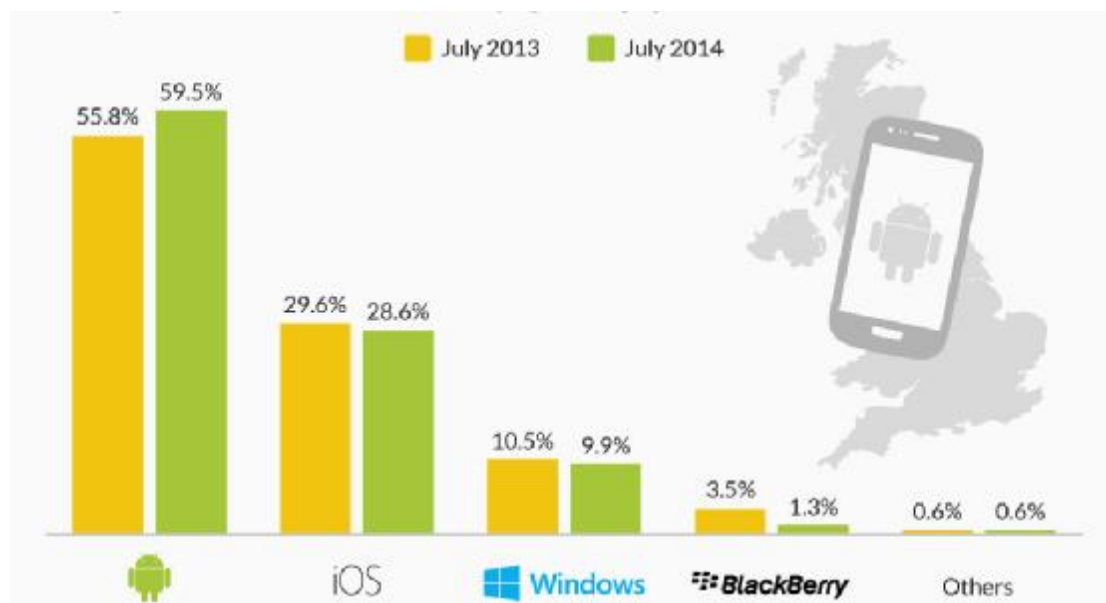
Κωδικό όνομα Νούμερο έκδοσης Ημερομηνία αρχικής κυκλοφορίας

| | | |
|--------------------|-------------|---------------------|
| N/A | 1.0 | 23 Σεπτεμβρίου 2008 |
| | 1.1 | 9 Φεβρουάριου 2009 |
| Cupcake | 1.5 | 27 Απριλίου 2009 |
| Donut | 1.6 | 15 Σεπτεμβρίου 2009 |
| Eclair | 2.0 – 2.1 | 26 Οκτωβρίου 2009 |
| Froyo | 2.2 – 2.2.3 | 20 Μαΐου 2010 |
| Gingerbread | 2.3 – 2.3.7 | 6 Δεκεμβρίου 2010 |
| Honeycomb | 3.0 – 3.2.6 | 22 Φεβρουάριου 2011 |
| Ice Cream Sandwich | 4.0 – 4.0.4 | 18 Οκτωβρίου 2011 |
| Jelly Bean | 4.1 – 4.3.1 | 9 Ιουλίου 2012 |
| KitKat | 4.4 – 4.4.4 | 31 Οκτωβρίου 2013 |

| | | |
|--------------------|-------------|--------------------------|
| Lollipop | 5.0 – 5.1.1 | 12 Νοεμβρίου 2014 |
| Marshmallow | 6.0 – 6.0.1 | 5 Οκτωβρίου 2015 |
| Nougat | 7.0 | 22 Αυγούστου 2016 |

Πίνακας 1: Πίνακας εκδόσεων λογισμικού Android (John Horton, 2015).

Η τελευταία έκδοση του λειτουργικού χρησιμοποιείται από αρκετούς χρήστες καθώς είναι ολοκληρωμένη και καλύπτει αρκετά από τα σφάλματα των προηγούμενων εκδόσεων. Τα στατιστικά από το 2014 (Niall McCarthy, 2014) που ακολουθούν παρουσιάζουν τη χρήση του εκάστοτε λογισμικού σε έξυπνες συσκευές στο Ηνωμένο Βασίλειο. Στην εργασία αυτή, η επιλογή ανάπτυξης εφαρμογής για το περιβάλλον Android έγινε με βάση τα στατιστικά αυτά δεδομένα, καθώς απ' ότι παρουσιάζεται το λογισμικό αυτό χρησιμοποιείται από την πλειονότητα των χρηστών ανά τον κόσμο.



Εικόνα 3: Στατιστικά στοιχεία για το λογισμικό Android.(Niall McCarthy, 2014)

Οι εφαρμογές που αναπτύσσονται για το Android αποτελούν μια ευρύτερη κατηγορία προγραμμάτων τα οποία χρησιμοποιούνται σε αρκετές κινητές συσκευές, όπως είναι οι έξυπνες συσκευές (smart phones ή tablet). Οι εφαρμογές για κινητά τηλέφωνα ονομάζονται και εφαρμογές λογισμικού και έχουν σχεδιαστεί για ένα συγκεκριμένο λειτουργικό σύστημα, καθώς πρέπει να είναι συμβατές με αυτό για να χρησιμοποιηθούν. Για να μπορούν οι χρήστες άμεσα, εύκολα και απρόσκοπτα να λάβουν τις εφαρμογές που επιθυμούν, έχουν αναπτυχθεί διάφορες πλατφόρμες για κάθε λογισμικό, όπως το Apple AppStore, Google Play.

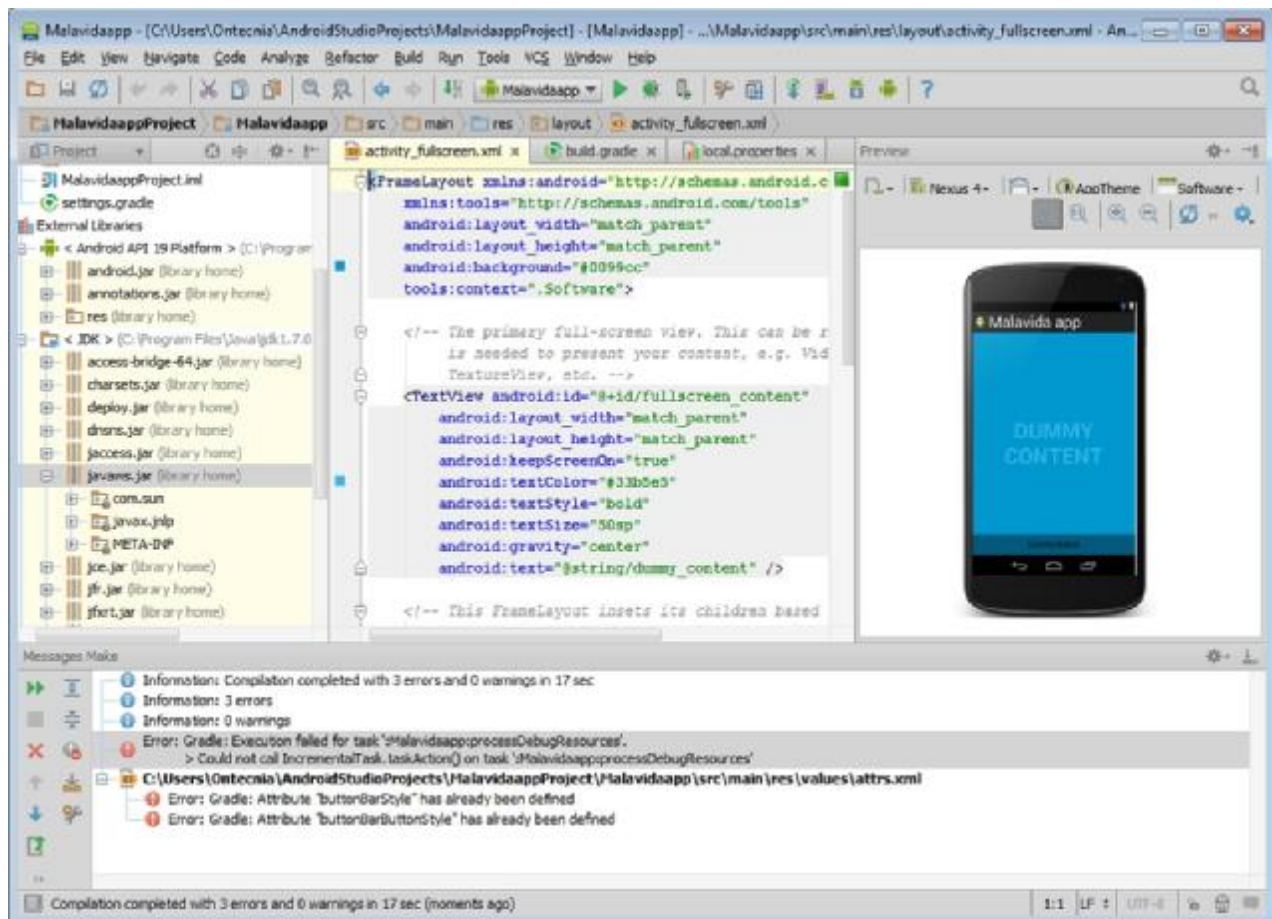
Η ανάπτυξη των εφαρμογών αυτών είναι η επέκταση και η ανάπτυξη των έξυπνων συσκευών, η ύπαρξη ευχέρειας, ευκολίας και η ανάπτυξη της αποτελεσματικότητας καθώς

και η αύξηση της παραγωγικότητας των επιχειρήσεων, χαρακτηριστικά τα οποία ζητούνταν από τους χρήστες κατά κόρον. Όλα τα πλεονεκτήματα του λειτουργικού αυτού σηματοδοτούν την ευκολία χρήσης του στις έξυπνες συσκευές και την προτίμηση των καταναλωτών. Το εύχρηστο λογισμικό δίνει πολλές δυνατότητες όχι μόνο στους χρήστες, αλλά και στους προγραμματιστές με τη χρήση του κατάλληλου περιβάλλοντος για την ανάπτυξη των εφαρμογών που απαιτούνται.

2.2 Περιβάλλον Android Studio

Για την ανάπτυξη εφαρμογών στο περιβάλλον αυτό χρησιμοποιούνται αρκετές προγραμματιστικές πλατφόρμες. Η πλατφόρμα με το μεγαλύτερο ποσοστό χρησιμότητας για την ανάπτυξη των εφαρμογών είναι το Android Studio, το οποίο αποτελεί ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE – Integrated Development Environment). Η πλατφόρμα αυτή βασίζεται στο λογισμικό της JetBrains' IntelliJ IDEA. Η πλατφόρμα αυτή πλέον είναι διαθέσιμη για λειτουργικά συστήματα Windows, Mac OS X και Linux, όπου μπορούν όλοι οι προγραμματιστές – χρήστες να το χρησιμοποιήσουν και να αναπτύξουν εφαρμογές σε όποιο περιβάλλον επιθυμούν.

Εκτός από την πλατφόρμα αυτή όμως, μπορεί να χρησιμοποιηθεί το εργαλείο Eclipse, το οποίο έχει αναπτυχθεί για την ανάπτυξη πολλαπλών εφαρμογών και όχι μόνο εφαρμογών Android. Στην Εικόνα 4 που ακολουθεί παρουσιάζεται ενδεικτικά το περιβάλλον του Android Studio.



Εικόνα 4: Απεικόνιση πλατφόρμας Android Studio.

2.3 Java γλώσσα προγραμματισμού

Η γλώσσα προγραμματισμού Java χρησιμοποιείται κυρίως στην ανάπτυξη εφαρμογών για έξυπνες συσκευές. Η γλώσσα αυτή κατατάσσεται στις αντικειμενοστραφείς γλώσσες προγραμματισμού και χρησιμοποιείται για γενικούς σκοπούς. Στόχος της γλώσσας αυτής είναι να μπορεί ο κώδικας ο οποίος παράγεται να χρησιμοποιηθεί σε οποιαδήποτε πλατφόρμα και χωρίς να πρέπει να υπάρχει ανάγκη για μεταγλώττιση του αρχείου και ανεξάρτητα από την αρχιτεκτονική του υπολογιστή στον οποίο εκτελούνται οι εφαρμογές. Το σημαντικό δηλαδή στοιχείο της γλώσσας αυτής είναι οι αλληλεξαρτήσεις του συστήματος οι οποίες δεν είναι τόσο σημαντικές για το περιβάλλον αυτό.

Από το 2016, η Java είναι στη λίστα με μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού σε χρήση, ειδικά για εφαρμογές webclient-server. Η Sun Microsystems ήταν η πρώτη εταιρία η οποία ανέπτυξε και κυκλοφόρησε τη Java το 1995 ως βασική συνιστώσα της πλατφόρμας Java της Sun Microsystems. Μεγάλο μέρος της γλώσσας προέρχεται από τη σύνταξη της από C και C++. Η τελευταία έκδοση είναι η Java 8 η οποία είναι η μόνη έκδοση η οποία υποστηρίζεται αυτήν τη στιγμή από την Oracle, αν και οι

προηγούμενες εκδόσεις υποστηρίζονται τόσο από την Oracle, όσο και από άλλες εταιρείες σε εμπορική βάση.

2.4 Στοιχεία Extensible Markup Language (XML)

Πέρα από τη Java χρησιμοποιείται και η XML για την ανάπτυξη εφαρμογών για περιβάλλον Android και κυρίως του γραφικού της περιβάλλοντος. Η XML- Extensible Markup Language. Η γλώσσα αυτή αποτελείται από ένα σύνολο κανόνων και χρησιμοποιείται κυρίως για την ηλεκτρονική κωδικοποίηση κειμένων. Επιπλέον, σημειώνεται ότι η γλώσσα αυτή σχεδιάστηκε για λόγους απλότητας και ευκολίας στην ανάπτυξη και χρήση διαδικασιών σε εφαρμογές. Αποτελεί μορφοποίηση δεδομένων και διαθέτει υποστήριξη Unicode για όλες τις γλώσσες προγραμματισμού.

Η σχεδίαση της εστιάζει κυρίως σε κείμενα, ωστόσο χρησιμοποιείται κυρίως και για λόγους αναπαράστασης αυθαίρετων δομών δεδομένων, οι οποίες χρησιμοποιούνται από υπηρεσίες διαδικτύου. Αρκετές προγραμματιστικές διεπαφές εφαρμογών χρησιμοποιούνται από προγραμματιστές για να προσπελαίνουν τα δεδομένα XML, αλλά και διάφορα συστήματα σχημάτων τα οποία είναι σχεδιασμένα για να βοηθούν στον ορισμό γλωσσών, που προκύπτουν από τη χρήση και την παραμετροποίηση.

Στην ενότητα αυτή θα παρουσιαστούν στοιχεία εμφάνισης τα οποία είναι διαθέσιμα για την παραγωγή οπτικών χαρακτηριστικών τα οποία θα χρησιμοποιηθούν για την ανάπτυξη της εφαρμογής. Χαρακτηριστικά, θα παρουσιαστούν η εμφάνιση, τα κουμπιά (εισαγωγή και λειτουργία αυτών), η εισαγωγή και η επεξεργασία κειμένου, οι ενέργειες οι οποίες εκτελούνται, καθώς και το απαραίτητο αρχείο Android Manifest.

2.4.1 Εμφάνιση (Layout)

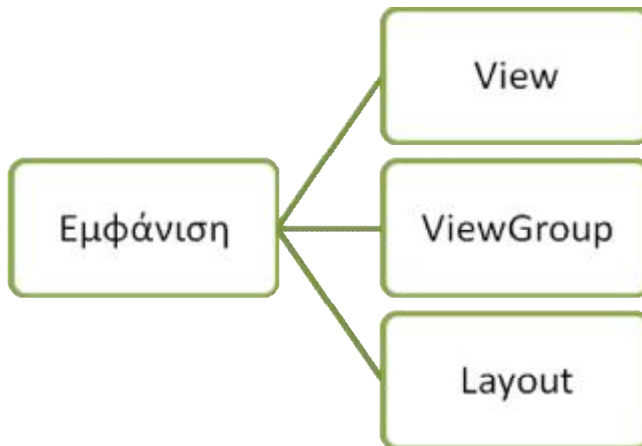
Το βασικό δομικό κομμάτι για την διεπαφή χρήστη στις εφαρμογές Android είναι το αντικείμενο¹ της προβολής (View). Το αντικείμενο **View** αποτελεί τη βασική κλάση² για τα widgets³ που χρησιμοποιούνται για την δημιουργία διεπαφής χρήστη έτσι ώστε να δημιουργηθεί ένα διαδραστικό περιβάλλον. Το **ViewGroup** είναι μία υποκλάση⁴ του View το οποίο παρέχει μη ορατά στον χρήστη εμφωλευμένα πεδία με στοιχεία Views και ViewGroups. Το τρίτο επίπεδο της διεπαφής είναι τα **layouts** τα οποία αποτελούν υποκλάσεις της γενικότερης κλάσης ViewGroup και καθορίζουν το πως θα φαίνεται η διεπαφή. Η παραπάνω διαβάθμιση παρουσιάζεται στην ακόλουθη εικόνα (Εικόνα 5):

¹Ένα αντικείμενο ορίζεται ως ένα σύνολο απόπεδία (*fields*) στη γλώσσα προγραμματισμού Java.

² Αντικείμενα του ίδιου τύπου ομαδοποιούνται σε μια κλάση (*class*). Σε κάθε κλάση ορίζονται τα πεδία και τις μεθόδους του κάθε αντικειμένου.

³Τα widgets είναι μια βασική πτυχή της προσαρμογής της αρχικής οθόνης.

⁴ Η υποκλάση είναι μέρος της κύριας κλάσης και έχει τη δυνατότητα να κληρονομεί τα χαρακτηριστικά της.



Εικόνα 5: Διαβάθμιση στοιχείων εμφάνισης.

Υπάρχουν πολλά layout που μπορούν να χρησιμοποιηθούν για την δημιουργία διεπαφής χρήστη, όπως είναι τα παρακάτω (Εικόνα 6):

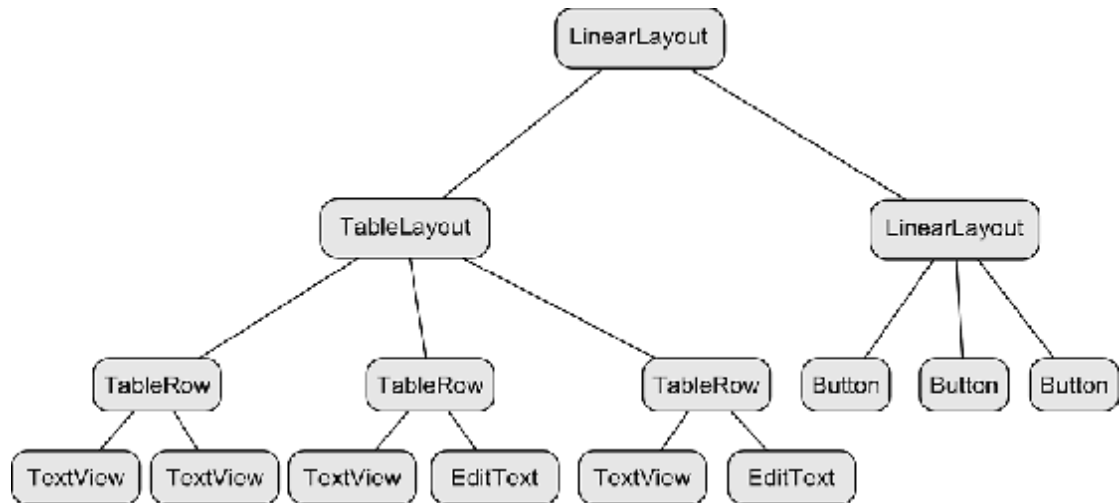
- **Absolute Layout:** ο χρήστης καθορίζει ακριβώς την τοποθεσία του κάθε αντικειμένου δίνοντας τις συντεταγμένες.

```
<AbsoluteLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="fill_parent">
```
- **Frame Layout:** το χρησιμοποιεί ο χρήστης για να δείξει κάτι στην οθόνη που επιθυμεί..

```
<FrameLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="fill_parent">
```
- **Linear Layout:** αφορά τον τρόπο με τον οποίο παρουσιάζονται τα αντικείμενα στην οθόνη. Στην περίπτωση αυτή τα αντικείμενα τοποθετούνται κάθετα, δηλαδή το ένα κάτω από το άλλο.

```
<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:orientation="vertical" >
```
- **Relative Layout:** με τη χρήση του στοιχείου αυτού σημειώνεται η θέση του στοιχείου στην οθόνη.


```
<RelativeLayoutxmlns:android="http://schemas.android.com/apk/res/android"  
android:layout_width="fill_parent"  
android:layout_height="fill_parent"  
android:paddingLeft="16dp"  
android:paddingRight="16dp" >
```



Εικόνα 6: Διαβάθμιση Layout στο Android. (Griffiths, D., & Griffiths, D., 2017)

2.4.2 Εισαγωγή κουμπιού (Button)

Το αντικείμενο Button (Εικόνα 7) είναι ένα κουμπί το οποίο ο χρήστης μπορεί να πατήσει για να ξεκινήσει μία διαδικασία. Στην εφαρμογή που έχει δημιουργηθεί για τις

ανάγκες της εργασίας αυτής αναπτύχθηκαν πολλά κουμπιά τα οποία είτε αλλάζουν οθόνη, είτε καλούν συγκεκριμένες διαδικασίες.

`<Button`

`android:layout_width="100dp"`

`android:layout_height="wrap_content"`

`android:text="OK"`

`android:layout_x="50px"`

`android:layout_y="361px" />`

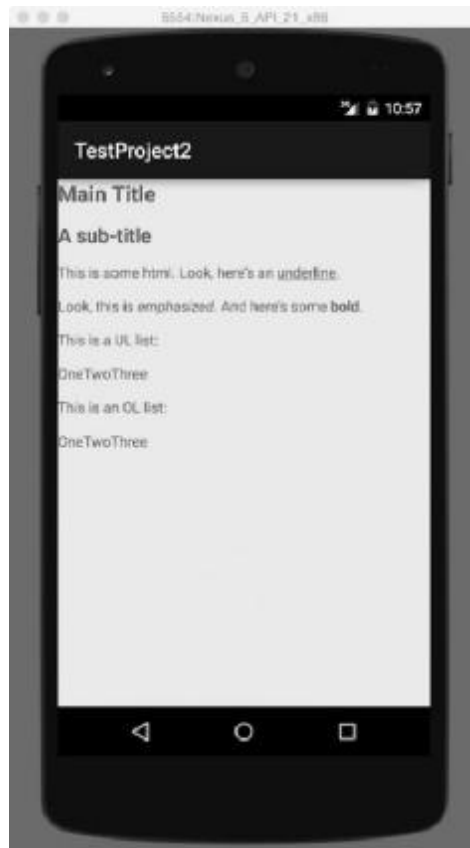


Εικόνα 7: Εικόνα κουμπιού σε μια εφαρμογή.

2.4.3 Εισαγωγή κειμένου (Text View)

Το αντικείμενο εισαγωγής κειμένου χρησιμοποιείται για την παρουσίαση κειμένου στην οθόνη της εφαρμογής (Εικόνα 8).

```
<TextView  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="Tutorialspoint"  
android:id="@+id/textView2"/>
```



Εικόνα 8: Εμφάνιση Text view σε μια εφαρμογή.

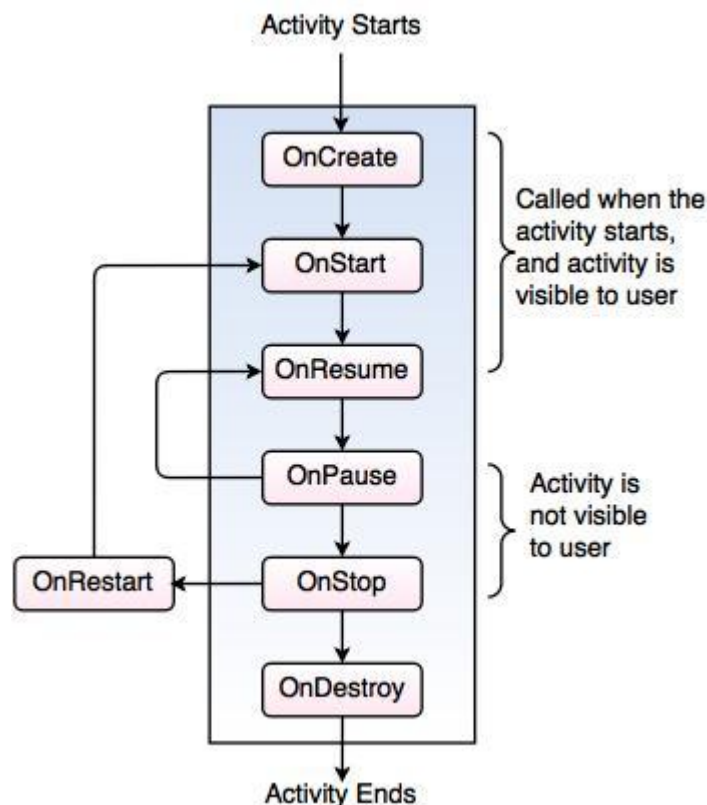
2.4.4 Ενέργειες (Activity)

Μία ενέργεια Activity αποτελεί μία στοχευόμενη διαδικασία την οποία μπορεί να κάνει ο χρήστης. Η ενέργεια δηλαδή αφορά την αλληλεπίδραση της εφαρμογής με τον χρήστη. Το εκάστοτε Activity μπορεί να καταλάβει χωρικά όλη την οθόνη προκειμένου να μπορεί ο χρήστης να αλληλεπιδράσει με αυτή σε όλο το εύρος του κινητού. Ωστόσο, μπορούν να

χρησιμοποιηθούν και Activities σαν αναδυόμενα παράθυρα, ή εμφωλευμένα προκειμένου να ικανοποιηθούν οι στόχοι της εφαρμογής. Κάθε φορά που ένα καινούργιο Activity ξεκινάει τοποθετείται στην κορυφή μιας στοίβας και το προηγούμενο κατεβαίνει πιο κάτω στην στοίβα, όπου και θα μείνει μέχρι να κλείσει το πρώτο στην σειρά Activity. Οι ενέργειες εκτελούνται σειριακά και ανάλογα με την σειρά που βρίσκονται στην στοίβα αυτή.

Υπάρχουν τέσσερις δυνατές καταστάσεις για κάθε Activity (Εικόνα 9):

- Ενεργό (active), λέγεται αυτό που είναι ενεργοποιημένο στην οθόνη.
- Σε παύση (paused), ονομάζεται η κατάσταση στην οποία η ενέργεια είναι ενεργή, αλλά δεν μπορεί ο χρήστης να αλληλεπιδράσει με αυτή. Για παράδειγμα όταν ο χρήστης είναι στο διαδίκτυο και δέχεται κλήση.
- Σταματημένη (stopped), είναι η κατάσταση της ενέργειας όταν είναι μπλοκαρισμένη από μια άλλη διαδικασία. Στην περίπτωση αυτή, τα στοιχεία του πρώτου Activity παραμένουν στο σύστημα αλλά δεν φαίνεται το παράθυρο πλέον στον χρήστη.
- Εάν ένα Activity είναι σε παύση ή σταματημένο το σύστημα μπορεί να το σβήσει από την μνήμη είτε ζητώντας να σβήσει, είτε τερματίζοντας το μόνο του. Όταν ο χρήστης χρησιμοποιήσει αυτό το Activity ξανά τότε πρέπει να ξεκινήσει από την αρχή.



Εικόνα 9: Διάγραμμα μεθόδων στο Activity. (Griffiths, D., & Griffiths, D., 2017)

Εκτός από τις κλάσεις, υπάρχουν και δύο μέθοδοι που χρησιμοποιούν όλες οι υποκλάσεις ενός Activity, η `onCreate(Bundle)` και η `onPause()`. Η μέθοδος `onCreate()` χρησιμοποιείται για την αρχικοποίηση των μεταβλητών και τη δημιουργία των γεγονότων που θέλουμε να ξεκινήσουν με την έναρξη της εφαρμογής. Αντίθετα, η δεύτερη μέθοδος καλείται όταν μια άλλη διεργασία λαμβάνει χώρα όταν πρόκειται να σταματήσει παροδικά η εκτέλεση ενός γεγονότος.

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
}
```

```
protected void onPause() {  
    super.onPause();  
}
```

2.4.5 Android Manifest

Απαραίτητο στοιχείο κάθε εφαρμογής είναι το αρχείο `AndroidManifest.xml`. Το αρχείο αυτό περιέχει όλες τις βασικές πληροφορίες που χρειάζεται το σύστημα για να ξεκινήσει την εφαρμογή πριν τρέξει οποιοδήποτε κομμάτι κώδικα και να αναγνωρίσει όλα τα απαραίτητα αρχεία τα οποία θα χρησιμοποιήσει κατά την εκτέλεσή της.

Το αρχείο Manifest είναι υπεύθυνο για:

- Την ονομασία της εφαρμογής. Το όνομα του πακέτου είναι ξεχωριστό και χρησιμοποιείται ως αναγνωριστικό της εφαρμογής.
- Την περιγραφή των συστατικών της εφαρμογής, όπως οι ενέργειες activities που παρουσιάστηκαν παραπάνω.
- Τον καθορισμό του ελάχιστου `AndroidAPIlevel`⁵ που χρειάζεται η εφαρμογή.
- Τον καθορισμό των βιβλιοθηκών που χρειάζεται η εφαρμογή.

Το αρχείο αυτό είναι επεξεργάσιμο από τον χρήστη καθώς αυτός μπορεί να αλλάζει όλα τα απαραίτητα στοιχεία μιας εφαρμογής τα οποία πρέπει να γνωστοποιήσει και στο λογισμικό της συσκευής. Ένα από τα σημαντικότερα στοιχεία αυτά είναι τα δικαιώματα που θα έχει η εφαρμογή και τι απαιτείται προκειμένου να ολοκληρώσει τη λειτουργία της.

⁵ Το `AndroidAPIlevel` είναι το ιστορικό εκδόσεων του λειτουργικού συστήματος Android.

3. ΣΤΟΙΧΕΙΑ ΣΧΕΔΙΑΣΜΟΥ ΕΦΑΡΜΟΓΗΣ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ANDROID

Για να γίνει μια σωστή και ολοκληρωμένη εφαρμογή πρέπει πρώτα να μελετηθούν οι απαιτήσεις του χρήστη και να σχεδιαστεί η δομή της εφαρμογής (Εικόνα 10). Κατά το σχεδιασμό της εφαρμογής είναι απαραίτητη η κατανόηση της λειτουργίας της, η ανάλυση των απαιτήσεών της, ο σχεδιασμός της αρχιτεκτονικής της καθώς και η ανάλυση και η δημιουργία της πρωταρχικής της δομής. Το κεφάλαιο αυτό λαμβάνει το ρόλο του οδηγού, καθώς παρουσιάζει στον αναγνώστη τα στάδια και τα βήματα τα οποία απαιτούνται για την σχεδίαση μιας ολοκληρωμένης εφαρμογής.



Εικόνα 10: Βήματα σχεδιασμού εφαρμογής.

3.1 Απαιτήσεις δημιουργίας εφαρμογής

Στις απαιτήσεις της εφαρμογής συμπεριλαμβάνονται τα χαρακτηριστικά τα οποία πρέπει να λάβει υπόψη του ο σχεδιαστής για να αναπτύξει την εφαρμογή σωστά και να ικανοποιεί τους στόχους της εργασίας. Τα στοιχεία αυτά ποικίλουν και πρέπει να καταγραφούν με ακρίβεια και λεπτομέρεια, ώστε να μπορέσει το τελικό αποτέλεσμα της εφαρμογής να είναι

άρτιο και να μπορεί να το χρησιμοποιήσει ο χρήστης χωρίς κανένα πρόβλημα. Στην ακόλουθη ενότητα παρουσιάζονται και αναλύονται μια προς μια αυτές οι απαιτήσεις.

| Απαιτήσεις | Λεπτομέρειες |
|--|--|
| Λειτουργία και χωρίς πρόσβαση στο διαδίκτυο. | Οι υπηρεσίες οι οποίες παρέχει η εφαρμογή θα μπορούν να είναι προσβάσιμες και εκτός διαδικτύου. |
| Ενημέρωση χρήστη για βασικά στοιχεία της επιχείρησης. | Ο χρήστης θα πρέπει να έχει την επιλογή να ενημερώνεται για βασικά στοιχεία της επιχείρησης αρχικά, τα οποία θα αντλούνται από το εκάστοτε κυρίως πρόγραμμα της επιχείρησης. |
| Επιλογή υπηρεσίας από τον χρήστη. | Οι υπηρεσίες οι οποίες θα πρέπει να παρέχονται στην εφαρμογή θα είναι οι βασικότερες έννοιες τις οποίες θα χρειαζόταν ένας χρήστης για την ενημέρωση της κατάστασης της επιχείρησής του. |
| Ασφάλεια και έλεγχος βαθμού δικαιωμάτων χρήστη. | Προκειμένου να αποφευχθεί κάποια διαρροή των στοιχείων της επιχείρησης από την εφαρμογή, θα πρέπει τα άτομα τα οποία θα έχουν πρόσβαση στην εφαρμογή να ταυτοποιήσουν τα στοιχεία τους. |
| Εύχρηστο και απλό περιβάλλον | Η εφαρμογή θα πρέπει να αποτελείται από ένα εύχρηστο και λειτουργικό περιβάλλον, το οποίο θα μπορεί εύκολα και χωρίς ιδιαίτερες γνώσεις να χειριστεί οποιοσδήποτε χρήστης. |
| Φορητή εφαρμογή. | Θα πρέπει ο χρήστης να μπορεί να χρησιμοποιήσει την εφαρμογή τόσο σε κινητό τηλέφωνο, όσο και σε συσκευή tablet. |

Πίνακας 2: Παρουσίαση και ανάλυση απαιτήσεων εφαρμογής.

3.2 Λειτουργία εφαρμογής

Η εφαρμογή που έχει υλοποιηθεί έχει σχεδιαστεί να λειτουργεί σε περιβάλλον Android. Ο χρήστης θα πρέπει να είναι σε θέση να μπορεί να χρησιμοποιήσει την εφαρμογή σε οποιαδήποτε κινητή συσκευή με λογισμικό android (είτε σε κινητό τηλέφωνο, ή σε συσκευή tablet). Ακολουθώντας τις απαιτήσεις που παρουσιάστηκαν παραπάνω, η εφαρμογή θα πρέπει να είναι ασφαλής και να μπορεί να πιστοποιείται η είσοδος του κάθε χρήστη, έτσι ώστε να μην μπορούν τα δεδομένα της επιχείρησης να είναι προσβάσιμα από τον οποιοδήποτε. Σε κάθε χρήστη θα αντιστοιχεί ένα όνομα χρήστη κι ένας κωδικός πρόσβασης, με τη χρήση – εισαγωγή των οποίων θα γίνεται η είσοδός τους στο σύστημα.

Εφόσον επιτευχθεί η είσοδος των χρηστών στο σύστημα, τότε θα έχουν πρόσβαση στα απαραίτητα δεδομένα. Για αρχή, η παρούσα εφαρμογή θα φέρει μόνο τις βασικές καρτέλες – στοιχεία τα οποία είναι απαραίτητα για την βασική ενημέρωση του χρήστη. Τα στοιχεία αυτά είναι οι πωλήσεις της επιχείρησης, τα οικονομικά της στοιχεία, το προσωπικό της καθώς και τα προϊόντα τα οποία υπάρχουν στην αποθήκη αυτής.

Μέσα από την εκτέλεση των βασικών αυτών διαδικασιών και τη λήψη των στοιχείων τα οποία θα εμπεριέχονται στην εφαρμογή, θα μπορεί ο χρήστης να ενημερώνεται άμεσα και απρόσκοπτα για την πορεία της εταιρίας του ημερησίως ή ακόμη και εβδομαδιαίως ή γενικότερα όποτε το επιθυμεί. Σημειώνεται στο σημείο αυτό ότι η βάση δεδομένων της εφαρμογής δεν θα είναι διαδικτυακή, καθώς το επίπεδο ασφαλείας το οποίο προτείνεται δεν μπορεί να καλυφθεί σε μια περίπτωση απομακρυσμένης επικοινωνίας. Για την αρχική επικοινωνία χρησιμοποιείται μια βάση δεδομένων, η οποία σε ιδανικές συνθήκες⁶ θα μπορούσε να επικοινωνεί με το πληροφοριακό σύστημα της επιχείρησης. Η βάση δεδομένων χρησιμοποιείται για την αποθήκευση όλων των διαδικασιών της επιχείρησης. Πιο συγκεκριμένα, πέρα από τα στοιχεία του χρήστη, στη βάση δεδομένων αποθηκεύονται όλα στοιχεία που αφορούν τις πωλήσεις, το προσωπικό, τα αποθέματα αλλά και τα έσοδα και έξοδα αυτής. Στην ενότητα που ακολουθεί γίνεται ανάλυση των στοιχείων και των πινάκων που περιέχει η βάση δεδομένων.

Αναφορικά με τα παραπάνω, οι δομές – υπηρεσίες οι οποίες προτείνεται να έχει η εφαρμογή είναι οι εξής:

- Ενημέρωση πωλήσεων
- Ενημέρωση εργασιακού προσωπικού
- Ενημέρωση οικονομικών στοιχείων επιχείρησης
- Ενημέρωση αποθήκης – προϊόντων

⁶ Με τον όρο «ιδανικές συνθήκες» εννοείται η περίπτωση όπου θα υπάρχει αυξημένη ασφάλεια μεταξύ της επικοινωνίας της βάσης δεδομένων με την εφαρμογή και τα στοιχεία της επιχείρησης, ώστε να μην υπάρξει κάποια διαρροή των στοιχείων αυτών.

3.3 Σχεδιασμός εφαρμογής

Κατά το σχεδιασμό της βάσης δεδομένων ελήφθησαν υπόψη οι παραπάνω απαιτήσεις καθώς και η κεντρική ιδέα και στόχοι της εργασίας, οι οποίοι έχουν παρουσιαστεί. Πιο συγκεκριμένα, η εφαρμογή θα δίνει αρχικά την επιλογή στον χρήστη να εισάγει τα στοιχεία του για την ταυτοποίηση του λογαριασμού του. Έπειτα, θα δίνει τη δυνατότητα στο χρήστη να επιλέξει την υπηρεσία που θέλει ο ίδιος, σχετικά με τις εργασίες που θέλει να μελετήσει για την επιχείρησή του.

Η βάση δεδομένων η οποία έχει αναπτυχθεί έχει ως στόχο να καλύψει όλες τις ανάγκες της εφαρμογής και αν μπορέσει να βοηθήσει το χρήστη και να τον ενημερώσει σχετικά με το τι γίνεται στην επιχείρησή του. Χαρακτηριστικά αναφέρεται ότι η βάση δεδομένων αποτελείται από τέσσερις πίνακες, οι οποίοι συμπεριλαμβάνουν όλα τα απαραίτητα για την εφαρμογή χαρακτηριστικά και έχει ως στόχο την άμεση κάλυψη των αναγκών για την εκάστοτε καρτέλα που έχει δημιουργηθεί (Εικόνα 11). Η επικοινωνία της εφαρμογής με τη βάση δεδομένων γίνεται με τη χρήση εντολών MySQL.

Στην παρούσα ενότητα δίνεται μια παρουσίαση της βάσης δεδομένων και των περιεχομένων της καθώς και ένα διάγραμμα, σύμφωνα με το οποίο παρουσιάζονται οι τα στοιχεία της βάσης δεδομένων, όπως αυτά είναι δομημένα στους πίνακές τους. Σημειώνεται ότι η παρούσα ενότητα αντικατοπτρίζει όλα όσα αναφέρθηκαν και αναλύθηκαν στο κεφάλαιο αυτό και προετοιμάζει το έδαφος για την παρουσίαση της τελικής εφαρμογής έτσι όπως λειτουργεί και καλύπτει τις ανάγκες και τους στόχους της εργασίας αυτής. Στον πίνακα που ακολουθεί (Πίνακας 3) παρουσιάζονται τα στοιχεία που αντιστοιχούν στον κάθε πίνακα. Τα ονόματα των πινάκων είναι σε αντιστοιχία με τα ονόματα της εκάστοτε καρτέλας.

| <i>Πίνακας Βάσης δεδομένων</i> | <i>Στοιχεία πίνακα</i> |
|--------------------------------|---|
| <i>Αποθήκη</i> | <ul style="list-style-type: none">• Id: Αύξων αριθμός και αναγνωριστικό αναζήτησης προϊόντος.• Name: όνομα προϊόντος το οποίο υπάρχει στην αποθήκη.• Perigrafi: περιγραφή προϊόντος.• Topothesisia: Στοιχεία διαδρόμου και ραφιού ή επιπλέον χαρακτηριστικά σχετικά με την τοποθεσία του.• Apothema: Ο αριθμός του αποθέματος το οποίο υπάρχει στην αποθήκη. |
| <i>Υπάλληλοι</i> | <ul style="list-style-type: none">• Id: Αύξων αριθμός και αναγνωριστικό αναζήτησης υπαλλήλου.• Username: Είναι το όνομα χρήστη το οποίο θα είναι το αναγνωριστικό του υπαλλήλου για την είσοδό του στο σύστημα.• Password: Στο πεδίο αυτό |

| | |
|------------------|---|
| | <p>αποθηκεύεται ο κωδικός του υπαλλήλου για την είσοδό του στο σύστημα.</p> <ul style="list-style-type: none"> • Name: Όνομα υπαλλήλου. • Klados: Αναφορά στον κλάδο ή στον τομέα στον οποίο ανήκει ο υπάλληλος. |
| <i>Πωλήσεις</i> | <ul style="list-style-type: none"> • Id: Αύξων αριθμός εισαγωγών. • Meres(5): Στόχος του πίνακα αυτού είναι να παρουσιαστούν οι συνολικές πωλήσεις της επιχείρησης και τις 5 ημέρες της εβδομάδας. Ο αριθμός των ημερών είναι καθαρά ενδεικτικός. |
| <i>Λογιστικά</i> | <ul style="list-style-type: none"> • Id: Αύξων αριθμός εισαγωγών. • Esoda: Στο πεδίο αυτό αντιστοιχούν τα στοιχεία του πίνακα τα οποία παρουσιάζουν τα συνολικά έξοδα της επιχείρησης. Επιλέχθηκε για τις ανάγκες της εργασίας να παρουσιάζονται τα εβδομαδιαία έξοδα της εταιρίας. • Exoda: Η ίδια λογική ακολουθείται και στο πεδίο αυτό, καθώς παρουσιάζονται τα συνολικά εβδομαδιαία έξοδα της επιχείρησης. |

Πίνακας 3: Στοιχεία πινάκων βάσης δεδομένων.

| Apothiki | Ypalliloi | Pwliseis | Logistika |
|--|---|--|--|
| <ul style="list-style-type: none"> 🔑 Name: VARCHAR(255) 🔑 id: INTEGER 🔑 Perigr: VARCHAR(255) 🔑 Topothesia: VARCHAR(255) 🔑 Apothema: INTEGER | <ul style="list-style-type: none"> 🔑 id: INTEGER 🔑 Name: VARCHAR(255) 🔑 Klados: VARCHAR(255) 🔑 Username: VARCHAR(255) 🔑 Password: VARCHAR(255) | <ul style="list-style-type: none"> 🔑 id: INTEGER 🔑 Deutera: INTEGER 🔑 Triti: INTEGER 🔑 Tetarti: INTEGER 🔑 Pempti: INTEGER 🔑 Paraskeui: INTEGER | <ul style="list-style-type: none"> 🔑 id: INTEGER 🔑 Esoda: INTEGER 🔑 Eksoda: INTEGER |

Εικόνα 11: Παρουσίαση τελικών πινάκων βάσης δεδομένων.

Στις εικόνες που ακολουθούν (Εικόνα 12 – Εικόνα 14) παρουσιάζονται οι συναρτήσεις οι οποίες δημιουργούν, αναβαθμίζουν και βοηθούν στην επεξεργασία των δεδομένων της βάσης.

```

private static final String logistic_eksoda = "logistic_eksoda";
private static final String logistic_upol = "logistic_upol";

public DBHelper(Context context) { super(context, Db_Name, null, Db_Version); }

//creating table
@Override
public void onCreate(SQLiteDatabase db) {
    // writing command for sqlite to create table with required columns
    String Create_Table = "CREATE TABLE " + Table_Name + " (" + User_id
        + " INTEGER PRIMARY KEY AUTOINCREMENT," + User_name + " TEXT," + User_password + " TEXT" + ")";
    db.execSQL(Create_Table);
    String Create_Table1 = "CREATE TABLE " + Table_Apothiki + " (" + Proion_id + " INTEGER PRIMARY KEY AUTOINCREMENT," + Proion_name
    db.execSQL(Create_Table1);
    String Create_Table2 = "CREATE TABLE " + Table_Ipalisi + " (" + Ipalisi_id + " INTEGER PRIMARY KEY AUTOINCREMENT," + Ipalisi_name
    db.execSQL(Create_Table2);
    String Create_Table3 = "CREATE TABLE " + Table_Polisais + " (" + Polisais_id + " INTEGER PRIMARY KEY AUTOINCREMENT," + Polisais_name
    db.execSQL(Create_Table3);
    String Create_Table4 = "CREATE TABLE " + Table_Logistic + " (" + Logistic_id + " INTEGER PRIMARY KEY AUTOINCREMENT," + Logistic_name
    db.execSQL(Create_Table4);
}

```

Εικόνα 12: Παρουσίαση συνάρτησης onCreate.

```

//Add new User by calling this method
public void addUser(User user) {
    // getting db instance for writing the user
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues cv = new ContentValues();
    // cv.put(User_id,usr.getId());
    cv.put(User_name, user.getName());
    cv.put(User_password, user.getPassword());
    //inserting row
    db.insert(Table_Name, null, cv);
    //close the database to avoid any leak
    db.close();
}

public void addApothiki(Apothiki apth) {
    // getting db instance for writing the user
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues cv = new ContentValues();
    // cv.put(Proion_id, apth.getId());
    cv.put(Proion_name, apth.getName());
    cv.put(Proion_perigrafi, apth.getPerigrafi());
    cv.put(Proion_Topothesia, apth.getTopothesia());
    cv.put(Proion_apothema, apth.getApothema());
    //inserting row
    db.insert(Table_Apothiki, null, cv);
    //close the database to avoid any leak
    db.close();
}

```

Εικόνα 13: Εισαγωγή πεδίων - τιμών σε κάθε πίνακα.

```

public ArrayList<Apothiki> getDataApothiki() {
    SQLiteDatabase db = this.getReadableDatabase();
    ArrayList<Apothiki> apothikiList = new ArrayList<>();
    Cursor result = db.rawQuery("select * from " + Table_Apothiki, null);
    while (result.moveToNext()) {
        apothikiList.add(new Apothiski(result.getInt(result.getColumnIndex(Proion_id)),result.getString(result.getColumnIndex(Proion_n
    )
    }
    return apothikiList;
}

public ArrayList<Ipalili> getDataIpalili() {
    SQLiteDatabase db = this.getReadableDatabase();
    ArrayList<Ipalili> ipalililist = new ArrayList<>();
    Cursor result = db.rawQuery("select * from " + Table_Ipalili, null);
    while (result.moveToNext()) {
        ipalililist.add(new Ipalili(result.getInt(result.getColumnIndex(Ipalili_id)),result.getString(result.getColumnIndex(Ipalili
    )
    }
    return ipalililist;
}

public ArrayList<Poliseis> getDataPoliseis() {
    SQLiteDatabase db = this.getReadableDatabase();
    ArrayList<Poliseis> poliseislist = new ArrayList<>();
    Cursor result = db.rawQuery("select * from " + Table_Poliseis, null);
    while (result.moveToNext()) {
        poliseislist.add(new Poliseis (result.getInt(result.getColumnIndex(Poliseis_id)),result.getString(result.getColumnIndex(Po

```

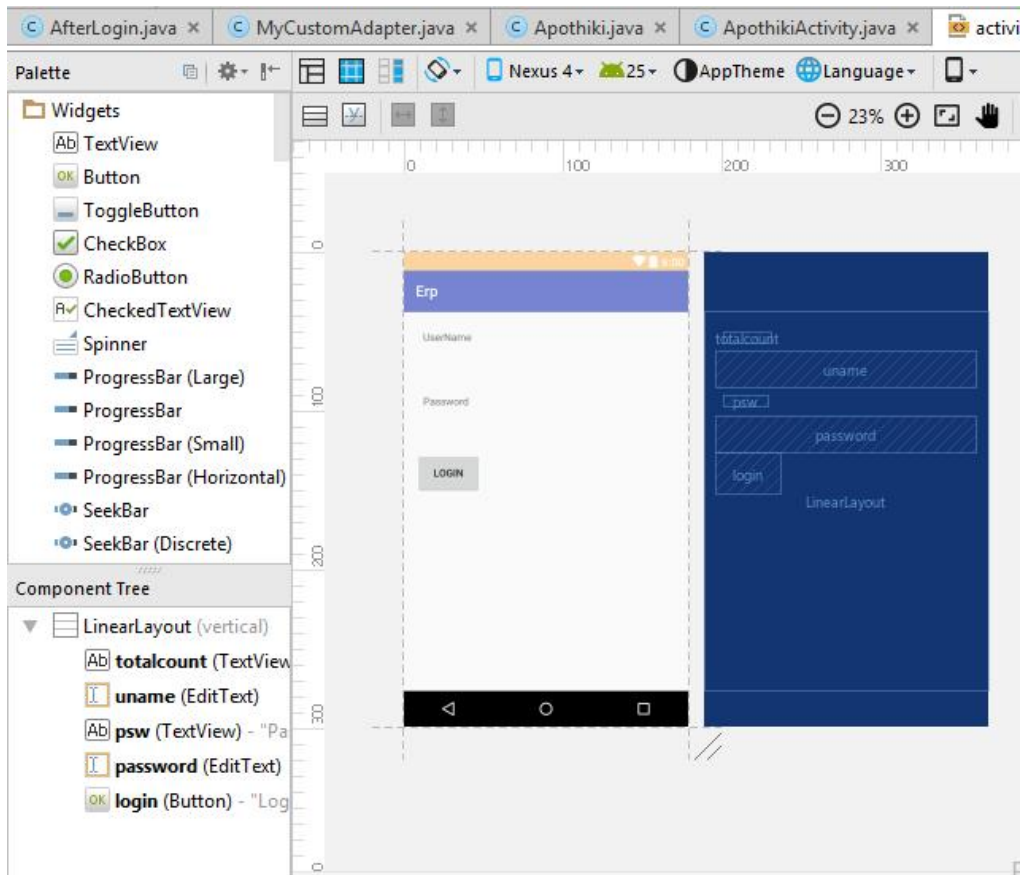
Εικόνα 14: Χρήση συναρτήσεων get για τη λήψη στοιχείων από τους πίνακες και την παρουσίαση των δεδομένων. Εκτελούνται MySQL ερωτήματα.

3.4 Ανάπτυξη εφαρμογής

Στην παρούσα ενότητα γίνεται παρουσίαση του τρόπου με τον οποίο αναπτύχθηκε η εργασία, με κυρίως στόχο την παρουσίαση του προγραμματιστικού σκέλους. Αρχικά, σημειώνεται ξανά η χρήση του προγραμματιστικού εργαλείου AndroidStudio, το οποίο χρησιμοποιήθηκε για την ανάπτυξη της παρούσας εφαρμογής σε περιβάλλον Android.

Το πρώτο βήμα ανάπτυξης αφορά την αρχική οθόνη της εφαρμογής, η οποία θα πρέπει να δέχεται τα στοιχεία εισόδου του πελάτη, να κάνει τον απαραίτητο έλεγχο και σε περίπτωση που ο πελάτης αυτός είναι καταχωρημένος με τη βάση δεδομένων, να τον μεταφέρει στην επόμενη κεντρική οθόνη.

Το γραφικό περιβάλλον της πρώτης οθόνης, παρουσιάζεται στην Εικόνα 15 και Εικόνα 16. Σημειώνεται ότι μόλις ο χρήστης πατήσει το κουμπί “Login”, τότε το σύστημα σε περίπτωση εισαγωγής σωστών στοιχείων, τον μεταφέρει στην επόμενη οθόνη. Σε περίπτωση λανθασμένων στοιχείων, ο χρήστης παραμένει στην αρχική οθόνη της εφαρμογής χωρίς δικαιώματα για να εξερευνήσει το υπόλοιπό της.



Εικόνα 15: Αρχική οθόνη εφαρμογής παρουσιασμένη μέσα από το πρόγραμμα AndroidStudio.

```

package user.erp;

import ...

public class MainActivity extends AppCompatActivity {
    EditText uname,pswd;
    Button login;
    DBHelper db;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        uname=(EditText)findViewById(R.id.uname);
        pswd=(EditText)findViewById(R.id.password);
        login=(Button)findViewById(R.id.login);
        login.setOnClickListener((v) -> {
            String name=uname.getText().toString();
            String password=pswd.getText().toString();
            int id= checkUser(new User(name,password));
            if(id!=-1)
            {
                Toast.makeText(MainActivity.this,"User Does Not Exist",Toast.LENGTH_SHORT).show();
            }
            else
            {
                Intent myIntent = new Intent(MainActivity.this, AfterLogin.class);
                MainActivity.this.startActivity(myIntent);
                Toast.makeText(MainActivity.this,"User Exist "+name,Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```

Εικόνα 16: Στοιχεία αρχικής οθόνης για την είσοδο του χρήστη.

```
package user.crp;

//this is model class
public class User {
    //variables
    int id;
    String name;
    String password;
    // Constructor with two parameters name and password
    public User(String name,String password)
    {
        this.name=name;
        this.password=password;
    }
    //Parameter constructor containing all three parameters
    public User(int id,String name,String ped)
    {
        this.id=id;
        this.name=name;
        this.password=ped;
    }
    //getting id
    public int getId() { return id; }
    //setting id
    public void setId(int id) { this.id = id; }
    //getting name
    public String getName() { return name; }
    //setting name
    public void setName(String name) { this.name = name; }
    //getting password
    public String getPassword() { return password; }
    //setting password
}
```

Εικόνα 17: Στοιχεία χρηστών και ο τρόπος δόμησής τους στη βάση δεδομένων.

Προκειμένου να ξεκινήσει σωστά η δομή της εργασίας, μετά την αρχική οθόνη έγινε η ανάπτυξη της βάσης δεδομένων (Εικόνα 11 - Εικόνα 14), όπως περιγράφεται στην προηγούμενη παράγραφο. Η βάση δεδομένων αποθηκεύει όλα τα απαραίτητα στοιχεία τόσο για το χρήστη, όσο και για τις υπόλοιπες καρτέλες οι οποίες θα χρησιμοποιηθούν και των οποίων τα στοιχεία πρέπει να είναι αποθηκευμένα. Σημειώνεται ότι στα αρχεία στα οποία έχει δομηθεί η βάση εμπεριέχονται όλες οι βασικές απαραίτητες συναρτήσεις, όπως η διαγραφή, η εισαγωγή και η επιλογή των στοιχείων τα οποία θα κληθούν μέσα από τις αντίστοιχες κατηγορίες.

Εφόσον έχει υλοποιηθεί η βάση δεδομένων και έχουν σημειωθεί οι κατηγορίες, πρέπει να γίνουν τα απαραίτητα αρχεία προκειμένου να δημιουργηθούν οι κατηγορίες της εφαρμογής και να αλληλεπιδρούν. Πρώτο μέλημα στη συνέχεια της ανάπτυξης είναι η δημιουργία της κύριας και αρχικής οθόνης της εφαρμογής η οποία παρουσιάζει τις καρτέλες στις οποίες μπορεί ο χρήστης να βρει τις επιθυμητές πληροφορίες. Η οθόνη αυτή αποτελείται από τέσσερα κουμπιά καθένα από τα οποία έχουν διαφορετική χρήση και λειτουργία (Εικόνα 18).

```
package user.ecp;

import ...

public class AfterLogin extends AppCompatActivity {
    Button apothiki;
    Button ipalili;
    Button poliseis;
    Button logistic;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.after_login_main);

        apothiki = (Button) findViewById(R.id.button);
        apothiki.setOnClickListener(new View.OnClickListener() {

            Intent myIntent = new Intent(AfterLogin.this, ApothikiActivity.class);
            AfterLogin.this.startActivity(myIntent);

        });

        ipalili = (Button) findViewById(R.id.button2);
        ipalili.setOnClickListener(new View.OnClickListener() {

            Intent myIntent = new Intent(AfterLogin.this, IpaliliActivity.class);
            AfterLogin.this.startActivity(myIntent);

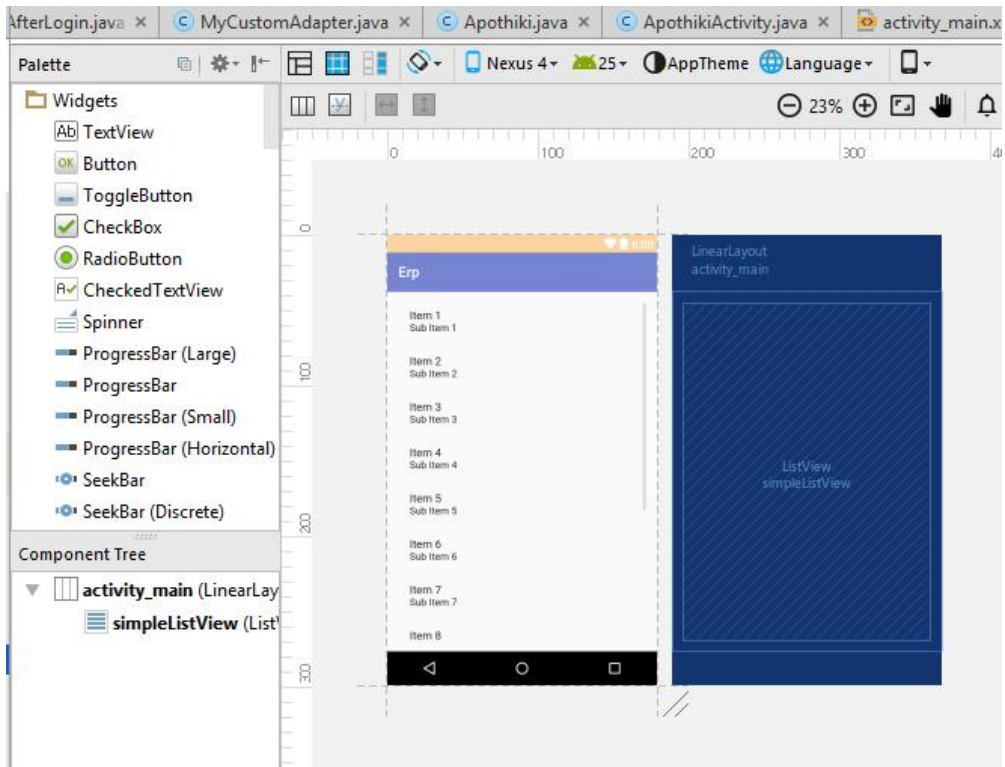
        });

        poliseis = (Button) findViewById(R.id.button3);
        poliseis.setOnClickListener(new View.OnClickListener() {
```

Εικόνα 18: Διαδικασία ταυτοποίησης στοιχείων της εφαρμογής και εμφάνιση των κατηγοριών εφόσον γίνει επιτυχημένη είσοδος χρήστη.

Μόλις δημιουργηθούν οι παραπάνω κατηγορίες, συνεχίζεται η υλοποίηση της εφαρμογής δημιουργώντας τα στοιχεία των κατηγοριών. Η πρώτη κατηγορία «αποθήκη» φέρει στοιχεία σχετικά με την αποθήκη της επιχείρησης. Για να δομηθεί η συγκεκριμένη κατηγορία αλλά και γενικότερα οι κατηγορίες της εφαρμογής αυτής ακολουθήθηκε το μοντέλο που παρουσιάζεται παρακάτω.

Αρχικά δημιουργούνται τα εμφανισιακά στοιχεία των κατηγοριών, το οποίο είναι κοινό για όλες τις κατηγορίες. Τα στοιχεία τα οποία καλούνται από τη βάση δεδομένων παρουσιάζονται σε λίστα στο χρήστη, προκειμένου να είναι πιο ευδιάκριτα και εύκολα στην αναζήτηση (Εικόνα 19). Επιπλέον, πίσω από την παρουσίαση των στοιχείων υπάρχει και ο αντίστοιχος κώδικας ο οποίος συνδέεται με τη βάση δεδομένων και εκτελεί την εντολή “Select”, η οποία χρησιμοποιείται για την αναζήτηση και τη λήψη των επιθυμητών δεδομένων (Εικόνα 20).



Εικόνα 19: Παρουσίαση οθόνης η οποία χρησιμοποιείται σε όλες τις κατηγορίες για την παρουσίαση των στοιχείων της εκάστοτε κατηγορίας.

```

package user.erp;

import ...

public class MyCustomAdapter extends ArrayAdapter {
    private Context context;
    private ArrayList<Apothiki> apoth;

    public MyCustomAdapter(Context context, int textViewResourceId, ArrayList objects) {...}

    private class ViewHolder
    {
        TextView Name;
        TextView Perigrafi;
        TextView Topothesia;
        TextView Apothema;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent)
    {
        ViewHolder holder=null;
        if (convertView == null)
        {
            LayoutInflater vi = (LayoutInflater)context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            convertView = vi.inflate(R.layout.row, null);

            holder = new ViewHolder();
            holder.Name = (TextView) convertView.findViewById(R.id.apName);
            holder.Perigrafi = (TextView) convertView.findViewById(R.id.apPerigrafi);
            holder.Topothesia=(TextView)convertView.findViewById(R.id.apTopothesia);
            holder.Apothema=(TextView)convertView.findViewById(R.id.apApothema);
            convertView.setTag(holder);
        }
    }
}

```

Εικόνα 20: Κώδικας που χρησιμοποιείται για την παρουσίαση των στοιχείων τα οποία λαμβάνονται από τη βάση δεδομένων.

Ενδεικτικά στην παρούσα παράγραφο παρουσιάζονται τα στοιχεία τα οποία αφορούν τη δημιουργία της υπηρεσίας Αποθήκη, ωστόσο σημειώνεται ότι η λογική ανάπτυξης όλων των υπηρεσιών είναι η ίδια. Αρχικά, εφόσον έχει δημιουργηθεί το περιβάλλον όλων των κουμπιών, δημιουργείται ένα αρχείο το οποίο να αφορά την κάθε υπηρεσία, προκειμένου να γίνει η λήψη των αντίστοιχων δεδομένων από κάθε καρτέλα χωριστά. Το αρχείο το οποίο φέρει τις πληροφορίες για την εκάστοτε καρτέλα και στην περίπτωση αυτή με την υπηρεσία της Αποθήκης, παρουσιάζεται στην Εικόνα 21. Στη συνέχεια, στο αρχείο της υπηρεσίας γίνεται η εισαγωγή των τιμών που υπάρχουν στην υπηρεσία αυτή με τη χρήση της εντολής Insert (Εικόνα 22).



```
AfterLogin.java | MyCustomAdapter.java | Apothiki.java | MainActivity.java | User.java
package user.erp;

/*this is model class
public class Apothiki {
    //variables
    int pid;
    String pname;
    String perigrifi;
    String topothesis;
    String apothema;
    public Apothiki(){

    }

    // Constructor with two parameters name and password
    public Apothiki(Integer id, String name, String perigrifi, String topothesis, String apothema) {
        this.pid= id;
        this.pname = name;
        this.perigrifi = perigrifi;
        this.topothesis = topothesis;
        this.apothema = apothema;
    }

    //getting id
    public int getId() { return pid; }

    //setting id
    public void setId(int id) { this.pid = id; }
```

Εικόνα 21: Αρχείο το οποίο φέρει τις πληροφορίες της υπηρεσίας Αποθήκη.

```
package user.erp;

import ...

public class ApothikiActivity extends AppCompatActivity {
    MyCustomAdapter myCustomAdapter=null;
    ListView listView=null;
    DBHelper db=null;
    ArrayList<Apothiki> apothiki=null;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.apothiki);

        //
        // Create/Open a SQLite database
        // and fill with dummy content
        // and close it
        //
        db = new DBHelper(ApothikiActivity.this);
        // db.deleteAll();
        db.addApothiki(new Apothiki(0, "Proion 1", "Perigrifi 1", "Diadromos 1 Rafi 1", "10"));
        db.addApothiki(new Apothiki(0, "Proion 2", "Perigrifi 2", "Diadromos 1 Rafi 2", "5"));
        db.addApothiki(new Apothiki(0, "Proion 3", "Perigrifi 3", "Diadromos 1 Rafi 3", "5"));
    }
}
```

Εικόνα 22: Εισαγωγή στοιχείων από την υπηρεσία Αποθήκη στη βάση δεδομένων. Τα στοιχεία αφορούν τα προϊόντα τα οποία βρίσκονται στην αποθήκη της επιχείρησης.

Τέλος, με το πάτημα του πλήκτρου επιλογής του χρήστη, ξεκινά η διαδικασία κατά την οποία γίνεται κλήση των κατάλληλων στοιχείων και λήψη τους από τη βάση δεδομένων, προκειμένου να παρουσιαστούν στην κατάλληλη καρτέλα – ενότητα. Τα στοιχεία τα οποία φέρει η κάθε κατηγορία υπάρχουν στη βάση δεδομένων, και σε περίπτωση εισαγωγής νέου στοιχείου, τα στοιχεία εισάγονται κατά τη φόρτωση της καρτέλας με το ξεκίνημα της εφαρμογής.

Σημειώνεται ότι στο αρχείο AndroidManifest έχουν σημειωθεί οι απαραίτητες ενέργειες, κατά την έναρξη υλοποίησης της εφαρμογής, οι οποίες σχετίζονται με:

- Εισαγωγή στοιχείων για την αρχική οθόνη της εφαρμογής
- Πρόσβαση της εφαρμογής στο διαδίκτυο
- Επικοινωνία και ανταλλαγή στοιχείων της εφαρμογής με βάση δεδομένων που έχει υλοποιηθεί
- Σημείωση της κάθε καρτέλας – υπηρεσίας που παρέχεται (του κάθε αρχείου που θα χρησιμοποιηθεί για την παρουσίαση των στοιχείων), προκειμένου να γνωστοποιηθεί στο κέλυφος της εφαρμογής ποια είναι τα αρχεία από τα οποία θα αντλούνται δεδομένα.

4. ΕΦΑΡΜΟΓΗ

Εφόσον έχει γίνει παρουσίαση του σχεδιασμού της εφαρμογής, στο παρόν κεφάλαιο θα παρουσιαστεί η υλοποίηση, η λειτουργία και λειτουργικότητα της εφαρμογής μέσα από εικόνες. Στο σημείο αυτό πρέπει να σημειωθεί ότι τα δεδομένα τα οποία υπάρχουν στη βάση είναι όσα έχουν ληφθεί με τη λήψη της εφαρμογής.

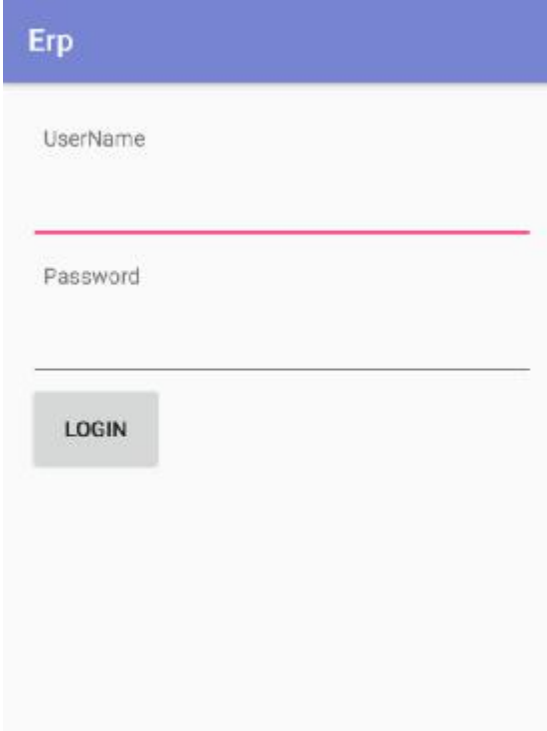
Εφόσον έχει γίνει σωστά η είσοδος του χρήστη, θα μπορεί να επιλέξει μια από τις ακόλουθες κατηγορίες:

- *Ενημέρωση αποθήκης:* στην καρτέλα αυτή θα μπορεί ο χρήστης να δει τον αριθμό των αποθεμάτων κάθε προϊόντος στην αποθήκη.
- *Απασχολούμενα άτομα ανά κλάδο:* στην καρτέλα αυτή ο χρήστης θα μπορεί να ενημερωθεί σχετικά με τα άτομα τα οποία απασχολεί στην επιχείρηση.
- *Πωλήσεις της εταιρείας:* στο πλαίσιο αυτό ο χρήστης θα μπορεί να ενημερωθεί για τις πωλήσεις της εταιρείας ημερησίως αλλά και μηνιαίως καθώς και να λάβει κάποια στατιστικά δεδομένα.
- *Λογιστικά:* στην καρτέλα αυτή ο χρήστης θα μπορεί να ελέγξει τις κινήσεις στο λογιστήριο της επιχείρησης.

4.1 Αρχική οθόνη

Στην αρχική οθόνη (Εικόνα 23) ο χρήστης καλείται να εισάγει τα στοιχεία για την ταυτοποίηση του, τα οποία είναι απαραίτητα για την είσοδό του στην υπηρεσία. Σε περίπτωση που τα δεδομένα τα οποία εισάγει ο χρήστης είναι λανθασμένα, τότε του αποτρέπεται η είσοδος από την εφαρμογή. Σε αντίθετη περίπτωση, δίνονται οι κάτωθι επιλογές στο χρήστη (Εικόνα 24):

- Υπηρεσία Αποθήκη
- Υπηρεσία Υπάλληλοι
- Υπηρεσία Πωλήσεις
- Υπηρεσία Λογιστικά



The image shows a login form for an ERP system. At the top, there is a blue header with the text "Erp". Below the header, there are two input fields: "UserName" and "Password". The "Password" field is currently empty and has a red underline. Below the input fields, there is a grey button labeled "LOGIN".

Εικόνα 23: Εισαγωγή στοιχείων για την είσοδο του χρήστη.

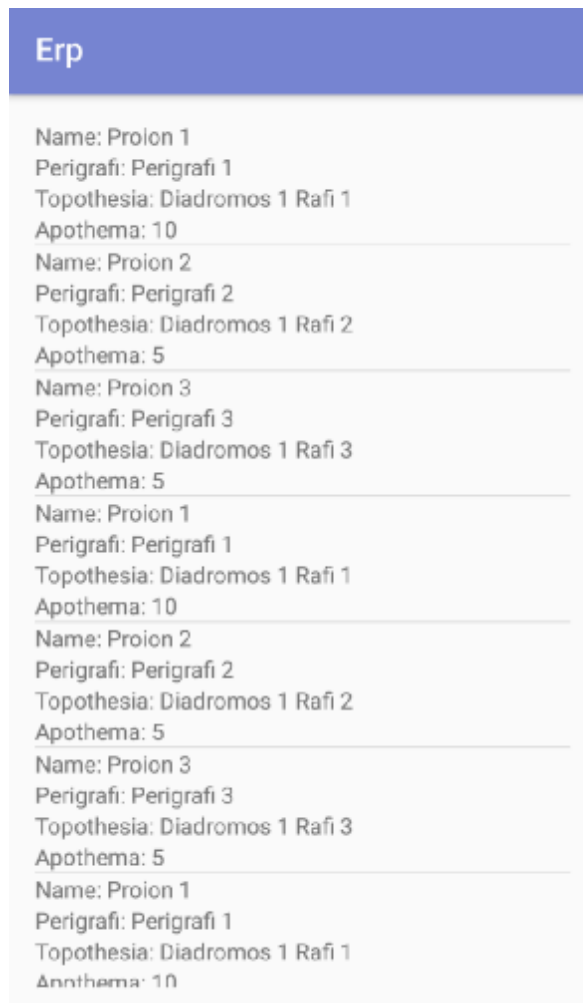


Εικόνα 24: Επιλογές από υπηρεσίες που δίνονται στην εφαρμογή.

4.2 Υπηρεσία “Αποθήκη”

Με την επιλογή της υπηρεσίας αυτής ο χρήστης θα μπορεί να ελέγξει τα αποθέματα της αποθήκης και να δει τι λείπει και τι όχι, ώστε να ο χρήστης να είναι ενημερωμένος ανά πάσα ώρα και στιγμή η αποθήκη της εταιρίας (Εικόνα 25). Τα πεδία τα οποία εμφανίζονται αφορούν:

- Το όνομα του προϊόντος
- Την περιγραφή του
- Την τοποθεσία του στην αποθήκη και
- Το απόθεμα του προϊόντος το οποίο εμφανίζεται



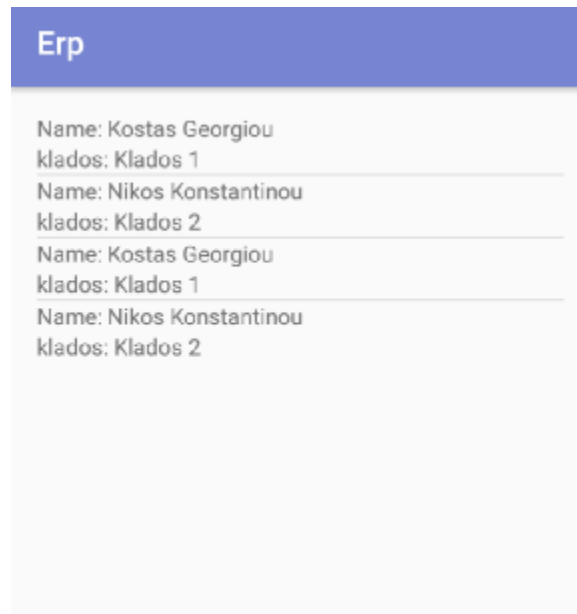
| Erp |
|--|
| Name: Proion 1 Perigrifi: Perigrifi 1 Topothesia: Diadromos 1 Rafi 1 Apothema: 10 |
| Name: Proion 2 Perigrifi: Perigrifi 2 Topothesia: Diadromos 1 Rafi 2 Apothema: 5 |
| Name: Proion 3 Perigrifi: Perigrifi 3 Topothesia: Diadromos 1 Rafi 3 Apothema: 5 |
| Name: Proion 1 Perigrifi: Perigrifi 1 Topothesia: Diadromos 1 Rafi 1 Apothema: 10 |
| Name: Proion 2 Perigrifi: Perigrifi 2 Topothesia: Diadromos 1 Rafi 2 Apothema: 5 |
| Name: Proion 3 Perigrifi: Perigrifi 3 Topothesia: Diadromos 1 Rafi 3 Apothema: 5 |
| Name: Proion 1 Perigrifi: Perigrifi 1 Topothesia: Diadromos 1 Rafi 1 Apothema: 10 |

Εικόνα 25: Υπηρεσία Αποθήκη.

4.3 Υπηρεσία “Υπάλληλοι”

Η υπηρεσία Υπάλληλοι είναι αρμόδια για την παρουσίαση των υπαλλήλων οι οποίοι εργάζονται στην επιχείρηση. Στην καρτέλα αυτή εμφανίζεται όχι μόνο το όνομα του υπαλλήλου, αλλά και ο τομέας – κλάδος στον οποίο ανήκει ο εκάστοτε υπάλληλος (Εικόνα

26). Η υπηρεσία αυτή βοηθά στην καταγραφή των υπαλλήλων και παράλληλα στην άμεση ενημέρωση του χρήστη για τον κάθε υπάλληλο της εταιρείας.



Εικόνα 26: Υπηρεσία Υπάλληλοι.

4.4 Υπηρεσία “Πωλήσεις”

Στην υπηρεσία των πωλήσεων παρουσιάζονται οι πωλήσεις της κάθε μέρας στην επιχείρηση (Εικόνα 27). Με βάση τις πωλήσεις αυτές θα μπορεί ο χρήστης να λαμβάνει όλα τα δεδομένα τα οποία είναι απαραίτητα για να μπορέσει να ελέγξει την καθημερινή πορεία της επιχείρησής του και να παράγει τα κατάλληλα στατιστικά δεδομένα ή να πάρει τις κατάλληλες αποφάσεις όταν πρέπει. Οι πωλήσεις και παράλληλα και η ημερομηνία στην

οποία γίνονται αυτές παρουσιάζονται ανά μέρα και χωρίζονται σε κάθε εβδομάδα για να μπορεί ο χρήστης να έχει παράλληλα ένα ιστορικό πωλήσεων.

| Erp | |
|---------------------|----|
| Date: November 2017 | |
| Deytera: | 5 |
| Triti: | 10 |
| Tetarti: | 4 |
| Pempiti: | 7 |
| Paraskeyi: | 16 |
| Date: November 2017 | |
| Deytera: | 15 |
| Triti: | 0 |
| Tetarti: | 3 |
| Pempiti: | 8 |
| Paraskeyi: | 1 |
| Date: November 2017 | |
| Deytera: | 5 |
| Triti: | 10 |
| Tetarti: | 4 |
| Pempiti: | 7 |
| Paraskeyi: | 16 |
| Date: November 2017 | |
| Deytera: | 15 |
| Triti: | 0 |
| Tetarti: | 3 |
| Pempiti: | 8 |
| Paraskeyi: | 1 |

Εικόνα 27: Υπηρεσία Πωλήσεις.

4.5 Υπηρεσία “Λογιστικά”

Στην υπηρεσία λογιστικών στην εφαρμογή αυτή υπάρχουν τα εβδομαδιαία έξοδα της επιχείρησης τα οποία είναι απαραίτητα για να μπορέσει ο χρήστης να κατανοήσει το βαθμό των οικονομικών απολαβών της εταιρίας (Εικόνα 28). Τα έσοδα και τα έξοδα σημειώνεται ότι είναι τα συνολικά εβδομαδιαία έξοδα της επιχείρησης. Τέλος, στο τέλος κάθε ενότητας παρουσιάζεται ο τελικός τζίρος της εβδομάδας ο οποίος υπολογίζεται από την αφαίρεση των εξόδων από τα έσοδα της επιχείρησης.

| Erp |
|--------------|
| Esoda: 10000 |
| Eksoda: 8000 |
| Tziros: 2000 |
| Esoda: 12000 |
| Eksoda: 9000 |
| Tziros: 3000 |
| Esoda: 10000 |
| Eksoda: 8000 |
| Tziros: 2000 |
| Esoda: 12000 |
| Eksoda: 9000 |
| Tziros: 3000 |

Εικόνα 28: Υπηρεσία Λογιστικά.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η πληροφορία σήμερα αποτελεί κατευθυντήριο μοχλό αλλά και πηγή κίνησης και λειτουργίας για όλες τις επιχειρήσεις του κόσμου. Οι πληροφορίες μπορεί να προέρχονται από οποιαδήποτε πηγή, να έχουν οποιαδήποτε δομή, σημασία ή και αντικείμενο, αποτελούν όμως το σημαντικότερο κομμάτι στην λειτουργία της επιχείρησης. Κάθε επιχείρηση λαμβάνει τις πληροφορίες τις οποίες θέλει, τις επεξεργάζεται και χρησιμοποιεί τα στοιχεία ή τα αποτελέσματα της επεξεργασίας αυτής προς όφελός της. Ωστόσο, η διαδικασία αυτή είναι αρκετά χρονοβόρα και απαιτεί αυξημένο αριθμό ατόμων. Για το λόγο αυτό τα τελευταία χρόνια έχουν αναπτυχθεί τα πληροφοριακά συστήματα.

Τα πληροφοριακά συστήματα αποτελούνται από ένα σύνολο διαδικασιών και αυτοματοποιημένων συστημάτων, τα οποία έχουν ως στόχο να συλλέξουν, να ανακτήσουν, να επεξεργαστούν και να αποθηκεύσουν κάθε σημαντική πληροφορία για το σύστημα. Τα συστήματα αυτά αποτελούνται από λογισμικό, υλικό και τηλεπικοινωνιακό σκέλος και μπορούν να ενταχθούν σε κάθε επιχείρηση.

Η παρούσα εργασία έχει ως στόχο να καλύψει το κενό το οποίο εμφανίζεται μεταξύ των πληροφοριακών συστημάτων μιας επιχείρησης και της απομακρυσμένης παρακολούθησής της. Η εφαρμογή αυτή έχει ως στόχο την ενημέρωση του χρήστη, μέσα από τα απαραίτητα στοιχεία ασφαλείας, σε αντικείμενα όπως οι πωλήσεις, τα οικονομικά, τα λογιστικά και το δυναμικό της επιχείρησης. Στόχος της εφαρμογής είναι να μπορεί ο χρήστης απομακρυσμένα να ελέγξει τα αποθέματα τα οποία υπάρχουν στην αποθήκη, τα συνολικά έξοδα και έσοδα της επιχείρησης καθώς και τα άτομα τα οποία εργάζονται στην εταιρία ανά τομέα/κλάδο.

Η εφαρμογή αυτή μπορεί να έχει αρκετές μελλοντικές επεκτάσεις, οι οποίες έχουν ως στόχο την κάλυψη πολλών ακόμα τομέων ενός εμπορικού ERP συστήματος. Επιπρόσθετα η εφαρμογή θα μπορούσε μελλοντικά και να επεξεργαστεί τα στοιχεία τα οποία υπάρχουν στις υπάρχουσες καρτέλες, ώστε να υποκαθιστά σε κάποιες περιπτώσεις το ERP πρόγραμμα της εταιρίας. Επιπλέον, μια ακόμη επέκταση της εφαρμογής θα μπορούσε να είναι η σύνδεση της εφαρμογής με κάποια διαδικτυακή βάση δεδομένων, η οποία θα συνδέεται απευθείας με το κεντρικό σύστημα.

Αρκετές επιχειρήσεις πλέον ανεβάζουν τα δεδομένα τους στο cloud και χρησιμοποιούν τις υπηρεσίες αυτού, ώστε να μπορεί κάθε συνεργαζόμενη επιχείρηση να λάβει δεδομένα και να ενημερωθεί άμεσα. Επομένως, μια ακόμη προοπτική επέκτασης της παρούσας εφαρμογής θα μπορούσε να αποτελεί η επικοινωνία και η λήψη δεδομένων αυτής από το cloud. Ωστόσο, παραμένει το θέμα ασφαλείας, το οποίο είναι αρκετά σημαντικό καθώς οι διαδικτυακές επικοινωνίες είναι ακόμη ένα ευάλωτο κομμάτι ασφαλείας.

ΑΝΑΦΟΡΕΣ

Furht, B. & Escalante, A. (2010). Handbook of Cloud Computing. New York: Springer.

Griffiths, D., & Griffiths, D. (2017). Head First Android Development: A Brain-Friendly Guide. " O'Reilly Media, Inc."

John Horton (2015), “Android Programming for Beginners”, Packt.

Laudon, K. & Price, J. (2009). Πληροφοριακά Συστήματα Διοίκησης. Κλειδάριθμος.

Niall McCarthy (2014). “UK Smartphone Sales Are Dominated By Android”. Online at <https://www.statista.com/chart/2700/uk-smartphone-sales-are-dominated-by-android/>

Van de Riet, R., Janssen, W., and de Gruijter, P. (1998). “Security Moving from Database Systems to ERP Systems”, Database and Expert Systems Applications, Proceedings, pp. 273-280, August.

Wagner, B., & Monk, E. (2008). Enterprise resource planning. Cengage Learning

Wei She, Bhavani Thuraisingham (2007), “Security for Enterprise Resource Planning Systems”, Information Systems Security, 16:152–163.