



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ  
ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ  
ΠΑΤΡΑ**

**ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ & ΟΙΚΟΝΟΜΙΑΣ**

**ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ**

πρώην  
ΕΦΑΡΜΟΓΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗ ΔΙΟΙΚΗΣΗ & ΣΤΗΝ ΟΙΚΟΝΟΜΙΑ

**Ανάπτυξη Διαδραστικής Εφαρμογής Υπολογισμού και  
Παρακολούθησης Δρομολογίων και Δυναμικής Πληροφόρησης  
Επιβατών για smart phones**

**An interactive smart phone application development of bus route monitoring for  
passenger information.**

Πτυχιακή Εργασία

**ΚΑΡΑΓΕΩΡΓΑΣ ΠΑΝΑΓΙΩΤΗΣ  
ΑΜ. 12551**

**Επιβλέπων Καθηγητής: Δρ. Χρήστος Πιερρακέας**

Πάτρα, Σεπτέμβριος 2016

# Ευχαριστίες

---

Θα ήθελα να ευχαριστήσω τον καθηγητή μου Χρήστο Πιερρακέα για την εμπιστοσύνη που μου έδειξε εκφράζοντας την ιδέα του έτσι ώστε να γίνει πράξη. Επίσης θα ήθελα να ευχαριστήσω τους ανθρώπους που με στήριξαν σε αυτή μου την προσπάθεια και κυρίως τους δικούς μου ανθρώπους, τη μητέρα μου Δέσποινα και την αδερφή μου Ισμήνη για την ατελείωτη υπομονή και συμπαράσταση που έδωσαν κατά τη διάρκεια των σπουδών μου.

*Αφιερωμένη στον πατέρα μου Γιώργο!*

## Περίληψη

---

Ο σκοπός της πτυχιακής εργασίας είναι η ανάπτυξη μιας εφαρμογής για smartphones που τρέχουν το λογισμικό Android, από θεωρητικό επίπεδο μέχρι και πρακτικό. Η εφαρμογή που αναπτύσσεται περιλαμβάνεται στη συσκευή του τελικού χρήστη, με τη δυνατότητα αναζήτησης δρομολογίων και του αστικού λεωφορείου που θέλει να επιβιβαστεί ο χρήστης.

Στα κεφάλαια που ακολουθούν θα αναλυθεί λεπτομερικά όλη η μεθοδολογία που ακολουθήθηκε από την καταγραφή των απαιτήσεων και την περιγραφή του προβλήματος μέχρι τα διαγράμματα κλάσεων και περιπτώσεων χρήσης και την κωδικοποίησή τους. Επιπλέον θα δούμε αναλυτικά μέσω της κωδικοποίησης το λειτουργικό κομμάτι της εφαρμογής καθώς και το σχεδιασμό του user interface.

Είναι μια εφαρμογή η οποία λειτουργεί μέσω διαδικτύου, είτε με Wi-Fi, είτε μέσω Mb Internet και δίνει τη δυνατότητα στους χρήστες της να δουν ανα πάσα ώρα και στιγμή που βρίσκεται το λεωφορείο της επιλογής τους

# Abstract

---

The purpose of this thesis is the development of an application for smartphones that runs under the Android OS, and the analysis at the theoretical and practical level. The application developed is included in the routes for the proper city bus that the user wants to board.

In the following chapters we will present / describe in every single detail the methodology used from the recording of requirements and the problem description to the class diagrams and the use case diagrams and their decoding. Moreover, we will notice in the detail through the codification the operating part of the application as well as the user interface design.

It is an application that works over a network, or Wi-Fi or MB Internet and it enables users to check at any time the waiting time of the selected bus.

# Περιεχόμενα

---

Ευχαριστίες

Περίληψη

Abstract

<b>Κεφάλαιο 1: Εισαγωγή</b> .....	1
<b>Κεφάλαιο 2: Κινητές συσκευές</b> .....	2
2.1 Personal Digital Assistants (PDA) .....	3
2.2 Κινητά τηλέφωνα .....	9
2.3 Smartphones .....	14
2.4 Tablets .....	23
<b>Κεφάλαιο 3: Τεχνολογία εντοπισμού θέσης</b> .....	33
3.1 Εντοπισμός θέσης μέσω GPS .....	34
3.2 Εντοπισμός θέσης μέσω GSM .....	37
3.3 Διάφορες τεχνολογίες εντοπισμού θέσης .....	39
<b>Κεφάλαιο 4: Λογισμικό Android</b> .....	41
4.1 Αρχιτεκτονική λογισμικού android .....	43
4.2 Δυνατότητες ανάπτυξης εφαρμογών .....	45
<b>Κεφάλαιο 5: Μεθοδολογία ανάπτυξης εφαρμογής</b> .....	49
5.1 Εισαγωγή .....	50
5.2 Ανάπτυξη απαιτήσεων .....	50
5.3 Σχεδίαση .....	52
5.3.1 Unified Modeling Language (UML) .....	54
5.3.2 Διάγραμμα κλάσεων .....	56
5.3.3 Διάγραμμα περιπτώσεων χρήστη .....	60
<b>Κεφάλαιο 6: Περιγραφή εφαρμογής</b> .....	65
6.1 Αντικείμενο εφαρμογής .....	65
6.2 Περιγραφή προβλήματος .....	66
6.3 Καταγραφή απαιτήσεων εφαρμογής (BusLane) .....	67
6.4 Διάγραμμα κλάσεων .....	68
6.5 Διάγραμμα περίπτωσης χρήστη (BusLane) .....	69
6.6 Ανάλυση περιπτώσεων χρήσης (σενάριο) .....	70
6.7 Μοντέλο πρωτοτυποποίησης .....	74

<b>Κεφάλαιο 7: Υλοποίηση</b> .....	76
7.1 Περιγραφή Υλοποίησης .....	76
7.2 Κωδικοποίηση .....	77
7.3 Περιγραφή Βάσης Δεδομένων .....	101
7.4 Εμφάνιση χρηστών στο χάρτη .....	101
7.5 Εφαρμογή σε smartphones .....	104
<b>Κεφάλαιο 8: Σύνοψη</b> .....	109
8.1 Συμπεράσματα .....	109
8.2 Μελλοντικές επεκτάσεις με νέες υπηρεσίες .....	110

## **Παράρτημα**

## **Βιβλιογραφία**

## 1. Εισαγωγή

Η τεχνολογία εξελίσσεται όλο και περισσότερο και σε πολύ γρήγορους ρυθμούς. Τα πάντα πλέον γίνονται μέσω των κινητών συσκευών και του διαδικτύου. Στα παλιότερα χρόνια οι άνθρωποι δε θα μπορούσαν να διανοηθούν σε τι στάδιο και τι δυνατότητες θα είχε φτάσει η τεχνολογία. Η τεχνολογία από παλιά θεωρούνταν απλά χρήσιμη πληροφορία για αυτούς, ενώ σήμερα έχει γίνει αναπόσπαστο κομμάτι της καθημερινότητας. Στην ουσία καταλαμβάνει ένα μεγάλο μέρος της ζωής μας, εφόσον περιτριγυρίζομαστε από διαφορετικών ειδών τεχνολογίες, όπως για παράδειγμα οικιακές συσκευές, τηλέφωνα ασύρματα και ενσύρματα, κινητά τηλέφωνα ή smartphones, υπολογιστές, tablets και πολλά άλλα.

Θα λέγαμε πως από τη μια πλευρά είναι τρομακτικό σχετικά με το πόσες πολλές πληροφορίες μπορείς να λάβεις πατώντας ένα μόνο κουμπί. Από την άλλη πλευρά όμως θα λέγαμε πόσο γρήγορα και εύκολα μπορείς να λύσεις κάποια προβλήματα που αντιμετωπίζεις στην καθημερινότητα σου. Έτσι έχει έρθει ο καιρός πλέον που οι εφαρμογές και τα smartphones, «δουλεύουν» για εμάς, για να κάνουν τη ζωή μας ευκολότερη. Τα πάντα γίνονται πλέον μέσω μικρών εφαρμογών applications. Μπαίνεις σε μια εφαρμογή που έχει μια συγκεκριμένη λειτουργία, όπως για παράδειγμα να σερφάρεις στο διαδίκτυο, να πάρεις ένα τηλέφωνο, να στείλεις ένα email, να πληρώσεις ένα λογαριασμό μέσω e banking, να παραγγείλεις καφέ μέσω της εφαρμογής και πάρα πολλά άλλα.

Δεν είναι τυχαίο που μεγάλες εταιρίες έχουν δώσει σημασία στο application development. Και αυτό έχει γίνει γιατί όλοι μας χρησιμοποιούμε smartphones και παράλληλα τις εφαρμογές που είναι προ εγκατεστημένες ή συνήθως αναζητούμε applications που θα μας λύσουν κάποια προβλήματα της καθημερινότητά μας.

### 2. Κινητές συσκευές

Μια κινητή συσκευή είναι μια υπολογιστική συσκευή συνήθως σε μικρό μέγεθος, η οποία διαθέτει μια οθόνη απεικονίσεως με αφή για είσοδο δεδομένων και ένα πληκτρολόγιο μικρής κλίμακας το οποίο μπορεί να είναι ενσωματωμένο στη συσκευή αυτή.

Η φορητή υπολογιστική αυτή συσκευή διαθέτει ένα λειτουργικό σύστημα OS και μπορεί να τρέξει διάφορους τύπους λογισμικού εφαρμογών, που είναι γνωστές ως εφαρμογές. Οι περισσότερες κινητές συσκευές διαθέτουν Wi-Fi, Bluetooth και GPS δυνατότητες που μπορεί να επιτρέπουν συνδέσεις με το Διαδίκτυο ή και με άλλες συσκευές Bluetooth, όπως ένα αυτοκίνητο ή ένα μικρόφωνο με ακουστικά. Επίσης αυτές οι κινητές συσκευές περιέχουν διάφορους αισθητήρες όπως επιταχυνσιόμετρο, πυξίδα / μαγνητόμετρο και γυροσκόπιο που επιτρέπει την ανίχνευση του προσανατολισμού της κίνησης.

Στα τέλη της δεκαετίας του 2000 κατασκευάστηκαν τέτοιου είδους κινητές συσκευές, όπως για παράδειγμα προσωπικοί ψηφιακοί βοηθοί (PDA) ή άλλες παρόμοιες συσκευές, όπως οι υπολογιστές tablet ή τα smartphones στην πιο σύγχρονη εποχή.

Οι εταιρίες κινητής τηλεφωνίας εκμεταλλευόμενες τις τεχνολογίες που παρέχουν οι έξυπνες αυτές κινητές συσκευές (tablets, smartphones και άλλα) κατάφεραν να προσφέρουν υπηρεσίες κάνοντας αυτές τις συσκευές πλέον απαραίτητες για τη ζωή μας, για την εργασία μας και γενικά για την καθημερινότητά μας. Σε συνδυασμό λοιπόν τεχνολογίας και διαδικτύου οι εταιρίες αυτές έχουν μετατρέψει το κινητό τηλέφωνο σε ένα «μίνι» προσωπικό υπολογιστή διαστάσεων μιας παλάμης που έχει τη δυνατότητα πρόσβασης στο διαδίκτυο ανα πάσα ώρα και στιγμή από οποιοδήποτε μέρος, με την προϋπόθεση πως έχει ενεργοποιήσει ο χρήστης το Wi-Fi του ή το 3G ή 4G. Επιπλέον μια τέτοια συσκευή είναι σε θέση μέσω του γυροσκόπιου που διαθέτει να εντοπίζει την κίνηση του χρήστη μέσω της τεχνολογίας GPS που διαθέτει.

Γενικά μπορούμε να διακρίνουμε τις κινητές συσκευές στις παρακάτω κατηγορίες: Personal Digital Assistant, Κινητά τηλέφωνα, Smartphones, Tablets.

Personal Digital Assistant (PDA) (Προσωπικός Ψηφιακός Βοηθός) είναι ένας όρος για οποιαδήποτε μικρή κινητή συσκευή χειρός που παρέχει υπολογιστική και αποθήκευση και ανάκτησης πληροφοριών δυνατότητες για προσωπική ή επαγγελματική χρήση. Τα περισσότερα PDA έχουν ένα μικρό πληκτρολόγιο ενσωματωμένο στη συσκευή.

Κινητό τηλέφωνο ή απλά κινητό, ονομάζεται κατά κύριο λόγο το τηλέφωνο που δεν εξαρτάται από φυσική καλωδιακή σύνδεση με δίκτυο παροχής τηλεφωνίας και δεν εξαρτάται από κάποια τοπικά ασύρματη συσκευή εκπομπής ραδιοφωνικού σήματος χαμηλής συχνότητας. Τα κινητά τηλέφωνα χρησιμοποιούν τεχνολογία κυψελών (cells) και εκπέμπουν σε υψηλές συχνότητες. Για την εκπομπή και λήψη των σημάτων χρησιμοποιείται πλέον, αποκλειστικά ψηφιακή τεχνολογία με κωδικοποίηση.

Το smartphone είναι ένα κινητό τηλέφωνο βασισμένο σε ένα λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερη προηγμένη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με ένα απλό κινητό τηλέφωνο. Τα πρώτα smartphones



συνδιάζουν τις λειτουργίες ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός κινητού τηλεφώνου. Σε μεταγενέστερα μοντέλα προστέθηκαν οι λειτουργίες των φορητών media players, low-end compact ψηφιακές φωτογραφικές μηχανές, βιντεοκάμερες τσέπης, καθώς και μονάδες πλοήγησης GPS, με αποτέλεσμα να διαμορφωθεί σε μια πολυχρηστική συσκευή.

Ένας υπολογιστής ταμπλέτα ή tablet είναι ένας φορητός υπολογιστής ή προσωπικός ψηφιακός βοηθός και είναι σε μέγεθος μεγαλύτερο από ένα κινητό τηλέφωνο, ενσωματωμένο σε μια επίπεδη οθόνη αφής και κυριώς λειτουργεί αγγίζοντας την οθόνη αντί να χρησιμοποιεί ένα φυσικό πληκτρολόγιο.

## 2.1 Personal Digital Assistants (PDA)

Όπως αναφέραμε και παραπάνω με τον όρο Personal Digital Assistant ή PDA σε συντομογραφία είναι μια οποιαδήποτε μικρή κινητή συσκευή χειρός που παρέχει υπολογιστική και αποθήκευσής και ανάκτησης πληροφοριών δυνατότητες για προσωπική ή επαγγελματική χρήση. Ο όρος αυτός δημιουργήθηκε από την Personal Desktop Assistants, ένας όρος λογισμικού για μια εφαρμογή που ζητά από το χρήστη του υπολογιστή με την εισαγωγή δεδομένων και παρέχει γρήγορη αναφορά σε επαφές και σε άλλες λίστες. Οι Προσωπικοί Ψηφιακοί Βοηθοί (PDA) διακόπηκαν στις αρχές της δεκαετίας του 2010, μετά την ευρεία υιοθέτηση των smartphones.

Σχεδόν όλα τα PDA έχουν τη δυνατότητα σύνδεσης στο Internet. Ένα PDA διαθέτει ένα πρόγραμμα περιήγησης στο διαδίκτυο, καθώς και τα περισσότερα μοντέλα διαθέτουν δυνατότητες ήχου που επιτρέπει τη χρήση αναπαραγωγής πολυμέσων. Επίσης οι περισσότεροι προσωπικοί ψηφιακοί βοηθοί μπορούν να χρησιμοποιηθούν και ως κινητά τηλέφωνα και μπορούν να έχουν πρόσβαση στο διαδίκτυο μέσω Wi-Fi ή ασύρματα δίκτυα ευρείας ζώνης. Τα περισσότερα PDA χρησιμοποιούν οθόνη αφής.

Το πρώτο PDA κυκλοφόρησε το 1984 από την εταιρία Psion. Η εταιρία αυτή κατασκεύασε 3 μοντέλα PDA το 1991. Το Psion Series 3 το 1991, η Psion Series 3a το 1993, το 3g Psion Series το 1996 και τη Psion Series 3mx το 1998. Ο όρος PDA χρησιμοποιήθηκε για πρώτη φορά στις 7 Ιανουαρίου του 1992 από την Apple Computer, όταν ο CEO John Sculley στο Consumer Electronics Show στο Λας Βέγκας, στη Νεβάδα αναφέρθηκε στο μήλο του Νεύτωνα. Το 1994 η IBM παρουσίασε το πρώτο PDA με πλήρη λειτουργικότητα κινητό τηλέφωνο με το όνομα IBM Simon, το οποίο μπορεί επίσης να θεωρηθεί ως το πρώτο smartphone. Στη συνέχεια το 1996, η Nokia εισήγαγε ένα PDA με πλήρη λειτουργικότητα κινητό τηλέφωνο, το 9000 Communicator, το οποίο έγινε best-seller PDA του κόσμου. Το Communicator ίδρυσε μια νέα κατηγορία των PDAs: το “PDA τηλέφωνο”, που σήμερα ονομάζεται «smartphone». Μια άλλη κατηγορία που μπήκε στην αγορά αυτή ήταν η Palm, με μια σειρά προϊόντων PDA που ξεκίνησε το Μάρτιο του 1996.

Ένα τυπικό PDA έχει μια οθόνη αφής για την εισαγωγή δεδομένων, υποδοχή και κάρτα μνήμης για την αποθήκευση δεδομένων, <sup>1</sup>IrDA, Bluetooth και Wi-Fi. Ωστόσο, ορισμένα PDAs δεν έχουν οθόνη αφής γιατί περιέχουν προγραμματισμένα πλήκτρα, όπως ένα πληκτρολόγιο κατεύθυνσης και ένα αριθμητικό πληκτρολόγιο ή μπορεί να περιέχει και ένα πληκτρολόγιο αντίχειρα για την είσοδο. Αυτό εμφανίζεται συνήθως σε τηλέφωνα που είναι παρεμπιπτόντως και PDA.

Για να έχει τις προδιαγραφές που αναμένονται από ένα PDA, το λογισμικό της συσκευής περιλαμβάνει συνήθως ένα ημερολόγιο, ένα «βιβλίο διευθύνσεων» για τις επαφές, μια αριθμομηχανή και κάποιο είδος σημειωματάριου. Τα PDAs με ασύρματες συνδέσεις δεδομένων συνήθως περιλαμβάνουν ένα ηλεκτρονικό ταχυδρομείο και ένα πρόγραμμα περιήγησης στο διαδίκτυο. Πολλά από τα αρχικά PDAs, όπως το Apple Newton και το Palm Pilot, χαρακτηρίζονταν για μια οθόνη αφής για την αλληλεπίδραση του χρήστη, που έχει μόνο μερικά κουπιά, που συνήθως κρατούνται για τις συντομώσεις για συχνά χρησιμοποιούμενα προγράμματα. Πολλά από τα PDAs που έχουν και οθόνη αφής, συμπεριλαμβανομένων και των Windows Mobile συσκευές, περιέχουν και μια αποσπώμενη γραφίδα για να διευκολυνθεί η πραγματοποίηση επιλογών. Ο χρήστης αλληλεπιδρά με τη συσκευή πατώντας την οθόνη για να επιλέξει την εντολή που θέλει να πραγματοποιήσει η συσκευή ή σύροντας το δάχτυλο (ή με τη γραφίδα) στην οθόνη για να επιλέξουμε τις εντολές που θέλουμε να κάνει η συσκευή.

Κάποιες μέθοδοι εισαγωγής κειμένου σε PDA με οθόνη αφής περιλαμβάνουν:

- Ένα εικονικό πληκτρολόγιο που εμφανίζεται στην οθόνη αφής. Το κείμενο θα το εισάγουμε πατώντας το πληκτρολόγιο επί της οθόνης με το δάχτυλο ή τη γραφίδα.
- Ένα εξωτερικό πληκτρολόγιο που συνδέεται μέσω USB, θύρα υπέρυθρων ή Bluetooth.
- Η αναγνώριση χειρογράφου, όπου οι επιστολές ή οι λέξεις γράφονται στην οθόνη αφής, καθώς και το PDA μετατρέπει την είσοδο σε κείμενο. Αναγνώριση και υπολογισμός των χειρογράφων οριζόντιων και κάθετων τύπων, όπως «1+2=», μπορεί επίσης να είναι ένα χαρακτηριστικό γνωρίσμα.

Παρά την αυστηρή έρευνα και ανάπτυξη, οι τελικοί χρήστες βιώνουν ανάμεικτα αποτελέσματα με το σύστημα αναγνώρισης χειρογράφου. Ορισμένοι θεωρούν ότι είναι απογοητευτικό και ανακριβείς, ενώ άλλοι είναι ικανοποιημένοι με την ποιότητα της αναγνώρισης. Τα PDAs που περιέχουν οθόνη αφής προσφέρουν επίσης πλήρες

---

<sup>1</sup> Η IrDA (Infrared Data Association) (σύλλογος δεδομένων υπέρυθρων) είναι μια βιομηχανία με γνώμονα ομάδα συμφερόντων που ιδρύθηκε το 1993 από περίπου 50 εταιρείες. Η IrDA παρέχει τις προδιαγραφές για ένα πλήρες σύνολο των πρωτοκόλλων για ασύρματες επικοινωνίες μέσω υπέρυθρων και το όνομα “IrDA” αναφέρεται επίσης σε αυτό το σύνολο των πρωτοκόλλων. Ο κύριος λόγος για τη χρήση IrDA έχει την ασύρματη μεταφορά δεδομένων. Έτσι έχει εφαρμοστεί σε φορητές συσκευές, όπως κινητά τηλέφωνα, φορητούς υπολογιστές, φωτογραφικές μηχανές, εκτυπωτές, ιατρικές συσκευές. Κύρια χαρακτηριστικά αυτού του είδους των ασύρματων οπτικών επικοινωνιών είναι φυσικά η ασφαλή μεταφορά δεδομένων, Line-of-sight (LOS) και το πολύ χαμηλό ποσοστό σφάλματος bit (BER) που το καθιστά πολύ αποτελεσματικό. ποσοστό σφάλματος bit (BER) που το καθιστά πολύ αποτελεσματικό.

πληκτρολόγιο και τροχούς κύλισης ή περιστροφικά για να διευκολύνει την είσοδο των δεδομένων και πλοήγησης.

Πολλές οθόνες PDAs υποστήριζαν κάποιας μορφής εξωτερικού πληκτρολογίου, καθώς και εξειδικευμένα πτυσσόμενα πληκτρολόγια, τα οποία προσφέρουν ένα πλήρους μεγέθους πληκτρολόγιο είναι διαθέσιμα για πολλά μοντέλα. Το εξωτερικό πληκτρολόγιο μπορεί να συνδεθεί με τη συσκευή PDA με ένα καλώδιο ή μπορεί να συνδεθεί με την ασύρματη τεχνολογία, όπως για παράδειγμα τις υπέρηθρες ή το Bluetooth.

Πιο σύγχρονα PDAs, όπως το HTC HD2, η Apple με το iPhone, iPod Touch και τα Palm Pre, Palm Pre Plus, Palm Pixi, Palm Pixi Plus, το Google με το Android περιλαμβάνουν πιο προηγμένες μορφές οθόνης αφής, που μπορούν να εγγραφούν πολλαπλά αγγίγματα ταυτόχρονα. Αυτές οι “multi-touch” οθόνες επιτρέπουν πιο εξελιγμένες διεπαφές χρησιμοποιώντας διάφορες χειρονομίες που εγγράφονται με ένα ή περισσότερα δάχτυλα.

Όσον αφορά τις κάρτες μνήμης που υποστηρίζει ένα PDA, οι πρώτες συσκευές δεν είχαν τις υποδοχές για να τις υποστηρίξουν. Τα νεότερα μοντέλα έχουν κάποια μορφή υποδοχής κάρτας μνήμης Secure Digital ή Compact Flash υποδοχή. Παρόλο που έχουν σχεδιαστεί αυτές οι υποδοχές για την μνήμη η Secure Digital εισόδου / εξόδου (SDIO) και κάρτες όπως η CompactFlash είναι διαθέσιμα να παρέχουν αξεσουάρ, όπως Wi-Fi ή ψηφιακές φωτογραφικές μηχανές, εφόσον η συσκευή μπορεί να τις υποστηρίξει. Μερικές συσκευές έχουν επίσης μια θύρα USB, κυρίως για μονάδες USB flash. Επίσης μερικά PDAs χρησιμοποιούν microSD κάρτες, οι οποίες είναι συμβατές με ηλεκτρονικές κάρτες SD, αλλά έχουν πολύ μικρό φυσικό μέγεθος.

Οι συσκευές αυτές έχουν τη δυνατότητα να συνδεθούν με τον προσωπικό υπολογιστή ενός χρήστη μέσω σειριακών θυρών, πολλές συσκευές σήμερα συνδέονται μέσω ενός καλωδίου USB. Οι πρώτες συσκευές PDAs ήταν σε θέση να συνδέονται στο Internet μέσω ενός εξωτερικού modem συνδεδεμένο μέσω σειριακής θύρας του PDA ή στην υποδοχή «συγχρονισμού» ή απευθείας με τη χρήση μιας κάρτας επέκτασης που παρείχε μια Ethernet θύρα.

Οι περισσότερες σύγχρονες συσκευές PDA έχουν Bluetooth, ένα δημοφιλές πρωτόκολλο ασύρματης σύνδεσης για κινητές συσκευές. Με το Bluetooth υπάρχει η δυνατότητα σύνδεσης πληκτρολογίου, ακουστικών, δέκτες GPS, αλλά υπάρχει και η δυνατότητα μεταφοράς αρχείων μεταξύ συσκευών PDAs, που διαθέτουν τέτοιου είδους ασύρματες συνδέσεις. Επιπλέον υπάρχει και η δυνατότητα Wi-Fi τεχνολογίας που οι χρήστες αυτών των συσκευών είναι σε θέση να συνδεθούν στο διαδίκτυο. Παλαιότερα PDAs από τη δεκαετία του '90 έως το 2006 είχαν συνήθως μια IrDA επιτρέποντας μικρής εμβέλειας, line-of-sight ασύρματης επικοινωνίας. Πιο σύγχρονα μοντέλα χρησιμοποιούσαν την τεχνολογία αυτή η οποία αντικαταστάθηκε από το Bluetooth και το Wi-Fi. Με την τεχνολογία IrDA επιτρέπει η επικοινωνία μεταξύ δύο PDAs ή μεταξύ ενός PDA και οποιαδήποτε συσκευή με θύρα IrDA.

Επιπλέον τα περισσότερα PDAs μπορούν να συγχρονίσουν τα δεδομένα τους με εφαρμογές στον υπολογιστή του χρήστη. Με τη μέθοδο του συγχρονισμού επιτρέπεται από το χρήστη να ενημερώσει την επαφή, το χρονοδιάγραμμα ή άλλες πληροφορίες σχετικά με τον υπολογιστή τους, χρησιμοποιώντας λογισμικό όπως το Microsoft Outlook ή το ACT μεταφέρονται δεδομένα στο PDA ή μεταφέρονται ενημερωμένες πληροφορίες

από το PDA πίσω στον υπολογιστή. Ο συγχρονισμός αποτρέπει επίσης την απώλεια των πληροφοριών που είναι αποθηκευμένες στη συσκευή, αν χαθεί, κλαπεί ή καταστραφεί. Όταν το PDA επισκευαστεί ή αντικατασταθεί, μπορεί να «ξανά συγχρονίζονται» με τον υπολογιστή, την επαναφορά των δεδομένων του χρήστη. Επίσης μερικά PDAs μπορούν να συγχρονιστούν χρησιμοποιώντας τις δυνατότητες ασύρματης δικτύωσης τους, αντί να χρειάζεται να συνδεθεί απευθείας με έναν προσωπικό υπολογιστή μέσω καλωδίου. Συσκευές όπως της Apple με το λειτουργικό iOS, όπως το iPhone το iPod Touch και το iPad, μπορούν να χρησιμοποιήσουν το iCloud υπηρεσία για να συγχρονιστούν εφαρμογές όπως το ημερολόγιο, βιβλίο διευθύνσεων, λογαριασμό e-mail, σελιδοδείκτες στο Internet, καθώς και άλλα δεδομένα με έναν ή περισσότερους υπολογιστές Macintosh ή Windows με τη χρήση Wi-Fi ή δεδομένων κινητής τηλεφωνίας (mobile internet) συνδέσεις.

Συσκευές της Palm που τρέχουν webOS ή το λειτουργικό σύστημα Android της Google συγχρονίζονται κατά κύριο λόγο με το «σύννεφο» cloud. Για παράδειγμα, εάν το Gmail χρησιμοποιούνται πληροφορίες στις επαφές, το e-mail και το ημερολόγιο μπορεί να συγχρονιστεί ανάμεσα στο τηλέφωνο και τους διακομιστές της Google. Η RIM πουλάει το BlackBerry Enterprise Server για επιχειρήσεις, έτσι ώστε οι εταιρικοί χρήστες BlackBerry να συγχρονίσουν ασύρματα τα PDAs τους με την εταιρεία Microsoft Exchange Server, IBM Domino ή Novell GroupWise διακομιστές. E-mail, καταχωρίσεις ημερολογίου, επαφές, εργασίες και τα σημειώματα που διατηρούνται επί του διακομιστή της εταιρείας συγχρονίζονται αυτόματα με το BlackBerry. Τα πιο διαδεδομένα λειτουργικά συστήματα προ-εγκατεστημένο σε PDAs είναι:

- Palm OS.
- Microsoft Windows Mobile (Pocket PC) με Windows CE πυρήνα.
- WebOS.

Άλλα, σπάνια λειτουργικά συστήματα που χρησιμοποιούνται είναι:

- EPOC, τότε Symbian OS (σε κινητό τηλέφωνο και PDA combos).
- Linux (π.χ. VR3, iPAQ, Sharp Zaurus PDA, Opie GPE, Familiar Linux κ.λ.π.).
- Newton.
- QNX (επίσης για iPAQ).

Επίσης αρκετά PDAs περιλαμβάνουν δέκτες Global Positioning System (GPS). Αυτό είναι ιδιαίτερα συνηθισμένο σε συσκευές smartphones. Άλλα PDA είναι συμβατά με εξωτερικό GPS-δέκτη add-ons που χρησιμοποιούν επεξεργαστή και οθόνη του PDA για να εμφανιστούν πληροφορίες για τη θέση. Τα PDAs με τη λειτουργία GPS μπορεί να χρησιμοποιηθεί για την αυτοκινητοβιομηχανία στην τεχνολογία της πλοήγησης. Αυτό για ένα σύγχρονο αυτοκίνητο αποτελεί ένα σπάνιο εξοπλισμό. Έτσι λοιπόν το GPS PDA μπορεί επίσης να εμφανίζει τις κυκλοφοριακές συνθήκες, να εκτελέσει δυναμική δρομολόγηση και να δείχνουν τις τοποθεσίες.

Επιπλέον ορισμένες επιχειρήσεις και κυβερνητικοί οργανισμοί βασίζονται σε ενισχυμένα PDAs, που είναι γνωστή ως επιχείρηση ψηφιακών βοηθών (EDA), για τις κινητές εφαρμογές δεδομένων. Η EDA έχουν επιπλέον χαρακτηριστικά για τη σύλληψη των δεδομένων, όπως οι αναγνώστες barcode, αναγνώρισης ραδιοσυχνότητας (RFID), αναγνώρισης κάρτας μαγνητικής ταινίας ή αναγνώρισης έξυπνης κάρτας. Τυπικές εφαρμογές περιλαμβάνουν:

- Στρατιωτικός: κυρίως του Στρατού των ΗΠΑ.
- Διαχείριση Εφοδιαστικής Αλυσίδας σε αποθήκες.
- Πακέτο παράδοσης.
- Ιατρική περίθαλψη και τήρηση αρχείων στα νοσοκομεία.
- Εγκαταστάσεις συντήρησης και διαχείρισης.
- Επιβολής στάθμευσης.
- Ελέγχου πρόσβασης και ασφάλειας.
- Ανάγνωση μετρητών από επιχειρήσεις κοινής ωφέλειας.
- Εφαρμογές σερβιρίσματος και παραγγελιών σε εστιατόρια και χώρους φιλοξενίας.
- Έλεγχος των λοιμώξεων και αποπτείας στο περιβάλλον της υγειονομικής περίθαλψης.
- Κατανομή ταξί και δρομολόγησης.

Πολλές εταιρείες έχουν αναπτύξει προϊόντα PDA που στοχεύουν σε μοναδικές ανάγκες του ιατρικού επαγγέλματος, όπως βάσεις δεδομένων των ναρκωτικών, πληροφορίες για τη θεραπεία και ειδήσεις ιατρικής. Υπηρεσίες όπως AvantGo μεταφράζει ιατρικά περιοδικά σε PDA – αναγνώσιμη μορφή. Η WardWatch διοργανώνει ιατρικά αρχεία, παρέχοντας υπενθυμίσεις των πληροφοριών όπως οι θεραπευτικές αγωγές των ασθενών στους ιατρούς. Επίσης τα PDAs είναι χρήσιμα και στο κομμάτι της εκπαίδευσης, όταν οι αίθουσες επιτρέπουν τη ψηφιακή καταγραφή σημειώσεων. Οι μαθητές μπορούν να κάνουν ορθογραφικό έλεγχο να διορθώσουν και να τροποποιήσουν τις σημειώσεις τους ώστε να τις βάλουν σε μια τάξη στο PDA τους. Ορισμένοι εκπαιδευτικοί διανέμουν το υλικό των μαθημάτων μέσω του Διαδικτύου ή μέσω της λειτουργίας του file-sharing της συσκευής τους. Εκδότες βιβλίων, έχουν αρχίσει να δημοσιεύουν στο διαδίκτυο σημειώσεις ή και βιβλία μέσω του e-books, τα οποία μπορούν να φορτωθούν κατευθείαν σε ένα PDA, μειώνοντας πολύ τον αριθμό των βιβλίων που οι μαθητές πρέπει να έχουν μαζί τους.

Τέλος αυτές οι συσκευές (PDA) μπορούν να χρησιμοποιηθούν και για ψυχαγωγικές χρήσεις, όπως για παράδειγμα η αναπαραγωγή μουσικής με τη λειτουργία ενός MP3 player. Επιπλέον χρησιμοποιείτε για πληροφορίες πλοήγησης και εντοπισμού θέσης, με τις λειτουργίες ενός GPS.



Manufacturer	Dell
Type	PDA
Retail Availability	2004
Media	Secure Digital, CompactFlash
Operating System	Windows Mobile Classic 2003SE, 5, and 6
CPU	624 MHZ Intel XScale PXA 270
Display	3.7" VGA TFT screen w/ 16-bit color
Graphics	Intel 2700G
Input	Touchscreen, Tactile buttons
Connectivity	802.11b Wi-Fi, Bluetooth 1.2, Infra-Red
Power	rechargeable lithium- ionBattery

*Εικόνα 1. DELL X50 PDA και τεχνικά χαρακτηριστικά*



*Εικόνα 2. Acer n35 GPS PDA*

## 2.2 Κινητά τηλέφωνα

Κινητό τηλέφωνο ή απλά κινητό, ονομάζεται κατά κύριο λόγο το τηλέφωνο που δεν εξαρτάται από φυσική καλωδιακή σύνδεση με δίκτυο παροχής τηλεφωνίας και δεν εξαρτάται από κάποια τοπική ασύρματη συσκευή εκπομπής ραδιοφωνικού σήματος χαμηλής συχνότητας. Τα κινητά τηλέφωνα χρησιμοποιούν τεχνολογία κυψελών (cells) και εκπέμπουν σε υψηλές συχνότητες. Για την εκπομπή και λήψη των σημάτων χρησιμοποιείται πλέον, αποκλειστικά ψηφιακή τεχνολογία με κωδικοποίηση.

Εκτός από την τηλεφωνία, σύγχρονα κινητά τηλέφωνα υποστηρίζουν επίσης μια μεγάλη ποικιλία από άλλες υπηρεσίες, όπως η ανταλλαγή μηνυμάτων κειμένου, MMS, e-mail, πρόσβαση στο Internet, ασύρματη επικοινωνία μικρής εμβέλειας μέσω υπέρυθρες και Bluetooth, επιχειρηματικές εφαρμογές, παιχνίδια και φωτογραφίες. Τα κινητά τηλέφωνα που προσφέρουν αυτές και γενικότερες ικανότητες των υπολογιστών αναφέρονται ως smartphones όπως θα δούμε στην επόμενη ενότητα.

Το πρώτο χειρός κινητό τηλέφωνο αποδεικνύεται από τον John F. Mitchell και το Δρ. Martin Cooper της Motorola, το 1973 χρησιμοποιώντας μια φορητή συσκευή που ζύγιζε περίπου 2 κιλά. Το 1983, η DynaTAC 8000X ήταν η πρώτη σειρά της Motorola που ήταν διαθέσιμη στο εμπόριο. Από το 1983 έως το 2014, σε όλο τον κόσμο της κινητής τηλεφωνίας έχουμε άυξηση πάνω από 7 δισεκατομμύρια του παγκόσμιου πληθυσμού στη βάση της οικονομικής πυραμίδας. Το 2014 οι κατασκευαστές τηλεφώνων που κατέκτησαν μεγάλα μερίδια αγοράς και οι πωλήσεις τους απογειώθηκαν ήταν η Samsung, η Apple, η LG και η Nokia.

Η δημιουργία για φορητές τηλεφωνικές συσκευές άρχισε μετά τον Δεύτερο Παγκόσμιο Πόλεμο, με τις εξελίξεις που λαμβάνουν χώρα σε πολλές χώρες. Η εξέλιξη στην κινητή τηλεφωνία έχει εντοπιστεί σε διαδοχικές γενιές από τις αρχές του “0G” (μηδενικής γενιάς) υπηρεσίες όπως το σύστημα Bell’ s. Αυτά τα συστήματα “0G” υποστήριζε λίγες ταυτόχρονες κλήσεις και ήταν τότε πολύ ακριβό. Το πρώτο φορητό κινητό τηλέφωνο δημιουργήθηκε από τη Motorola το 1973. Το πρώτο εμπορικό αυτοματοποιημένο κυψελοειδές δίκτυο, ξεκίνησε στην Ιαπωνία από την NTT το 1979. Το 1981, ακολούθησε την ταυτόχρονη έναρξη στο σύστημα του κινητού τηλεφώνου στη Δανία, τη Φιλανδία, τη Νορβηγία και τη Σουηδία. Πολλές άλλες χώρες στη συνέχεια, ακολούθησαν στις αρχές έως τα μέσα του 1980. Αυτό το πρώτο generation σύστημα (“1G”) θα μπορούσε να υποστηρίξει περισσότερες ταυτόχρονες κλήσεις, αλλά εξακολουθεί να χρησιμοποιείται η αναλογική τεχνολογία.

Το 1991, η δεύτερη γενιά (2G) ψηφιακής κυψελωτής τεχνολογίας ξεκίνησε στη Φιλανδία από <sup>2</sup> Radiolinja για το <sup>3</sup> GSM πρότυπο, το οποίο προκάλεσε τον

---

<sup>2</sup> Radiolinja ήταν μια φιλανδική GSM εταιρεία που ιδρύθηκε στις 19 Σεπτεμβρίου του 1988. Στις 27 Μαρτίου του 1991, το πρώτο στον κόσμο GSM τηλεφώνημα έγινε στο δίκτυο Radiolinja. Το δίκτυο άνοιξε για εμπορική χρήση από την 1 Ιουλίου του 1991.

<sup>3</sup> GSM (Global System for Mobile Communications) είναι ένα πρότυπο που αναπτύχθηκε από το Ευρωπαϊκό Ινστιτούτο Τηλεπικοινωνιακών Προτύπων για να περιγράψει τα πρωτόκολλα δεύτερης γενιάς (2G) σε ψηφιακά κυκλοειδή δίκτυα που χρησιμοποιούνται από τα κινητά τηλέφωνα.

ανταγωνισμό στον κλάδο, καθώς οι νέοι επιχειρηματίες αμφισβητούσαν τα κατεστημένων φορέων εκμετάλλευσης δικτύων 1G.

Μετά από δέκα χρόνια, το 2001, η τρίτη γενιά (3G) ξεκίνησε στην Ιαπωνία από την NTT DoCoMo στην WCDMA πρότυπο. Αυτό ήταν η βάση για την εξέλιξη σε 3.5G, 3G+ ή βελτιώσεις 3G Turbo με βάση την πρόσβαση πακέτων υψηλής ταχύτητας (HSPA), επιτρέποντας δίκτυα UMTS να έχουν υψηλότερες ταχύτητες μεταφοράς δεδομένων. Μέχρι το 2009, είχε καταστεί σαφές ότι σε κάποια σημεία τα δίκτυα, 3G θα είναι συγκλονιστικά φορτωμένα από την αύξηση των εφαρμογών, όπως το streaming media. Ως εκ τούτου, η βιομηχανία άρχισε να ψάχνει σε τεχνολογίες 4<sup>ης</sup> γενιάς δεδομένων βελτιστοποιώντας την ταχύτητα έως και 10 φορές πάνω από τις υπάρχουσες τεχνολογίες 3G. Οι δύο πρώτες εμπορικά διαθέσιμες τεχνολογίες που τιμολογούνται ως 4G ήταν το πρότυπο WiMAX (που προσφερόταν στις ΗΠΑ από τη Sprint) και το LTE πρότυπο, που προσφέρθηκε για πρώτη φορά στη Σκανδιναβία από την TeliaSonera.

Όλα αυτά τα κινητά τηλέφωνα έχουν μια σειρά από κοινά χαρακτηριστικά, αλλά οι κατασκευαστές προσπαθούν να διαφοροποιήσουν τα προϊόντα τους μέσω της εφαρμογής πρόσθετων λειτουργιών, για να γίνουν πιο ελκυστικά για τους καταναλωτές. Αυτό έχει οδηγήσει σε μεγάλη καινοτομία στην ανάπτυξη των κινητών τηλεφώνων κατά τη διάρκεια των τελευταίων 20 ετών. Τα κοινά συστατικά που βρίσκονται σε όλα τα κινητά τηλέφωνα είναι:

- Μια μπαταρία που παρέχει την πηγή ενέργειας για τις λειτουργίες του τηλεφώνου.
- Ένας μηχανισμός εισόδου να επιτρέπει στο χρήστη να αλληλεπιδρά με το τηλέφωνο. Ο πιο κοινό μηχανισμός εισόδου είναι ένα πληκτρολόγιο, αλλά και οθόνες αφής που έχουν τα περισσότερα smartphones.
- Μια οθόνη που εμφανίζει μηνύματα κειμένου, επαφές και πολλά άλλα.
- Βασικές υπηρεσίες κινητής τηλεφωνίας που επιτρέπει στους χρήστες να πραγματοποιούν κλήσεις και αποστολή μηνυμάτων κειμένου.
- Όλα τα GSM τηλέφωνα χρησιμοποιούν μια κάρτα SIM για να επιτρέψει σε έναν λογαριασμό να προσθέσει διάφορες επαφές στο τηλέφωνό του. Ορισμένες 4 CDMA συσκευές έχουν επίσης μια παρόμοια κάρτα που ονομάζεται 5 R-UIM.

---

<sup>4</sup> CDMA (Πολλαπλή πρόσβαση διαίρεσης κώδικα) είναι μια μέθοδος πρόσβασης που χρησιμοποιείται από διάφορες ραδιοφωνικές τεχνολογίες επικοινωνίας. Το CDMA αποτελεί παράδειγμα της πολλαπλής πρόσβασης, στην οποία αρκετοί πομποί μπορούν να στείλουν πληροφορίες ταυτόχρονα σε ένα κανάλι επικοινωνίας. Αυτό επιτρέπει σε πολλούς χρήστες να μοιράζονται ένα εύρος συχνοτήτων

<sup>5</sup> R-UIM (Αφαιρούμενη χρήση στοιχείων ταυτότητας.) προέρχεται από τις λέξεις Removable User Identity Module. Είναι μια κάρτα που αναπτύχθηκε για cdmaOne / CDMA2000 ("CDMA") συσκευές που επεκτείνει τη GSM SIM κάρτα για CDMA τηλέφωνα και δίκτυα. Για εργασία σε δίκτυα CDMA, το R-UIM περιέχει μια πρώτη έκδοση της CSIM εφαρμογής. Η κάρτα SIM περιέχει επίσης (GSM) εφαρμογή, έτσι ώστε να μπορεί να λειτουργήσει και στα δύο δίκτυα. Είναι φυσικά συμβατό με τις GSM κάρτες SIM και μπορεί να χωρέσει σε υπάρχουσες συσκευές κινητής τηλεφωνίας, δεδομένου ότι αποτελεί επέκταση του προτύπου GSM.



- Ατομική GSM, WCDMA ταυτότητα από το διεθνές εξοπλισμό κινητής τηλεφωνίας (IMEI αριθμό).

Έτσι λοιπόν τα κινητά τηλέφωνα που συχνά αναφέρονται ως χαρακτηριστικά τηλέφωνα προσφέρουν βασικές τηλεφωνίες. Συσκευές με πιο προηγμένη ικανότητα των υπολογιστών με τη χρήση των εφαρμογών λογισμικού έγιναν γνωστές ως smartphones. Αρκετές σειρές τηλεφώνων για να αντιμετωπίσουν ένα συγκεκριμένο τμήμα της αγοράς όπως η RIM BlackBerry με έμφαση στις ανάγκες των επιχειρήσεων e-mail (εταιρικών πελατών), η σειρά της Sony Ericsson “Walkman” με εφαρμογές σχετικά με μουσική και η σειρά πάλι της Sony Ericsson “Cybershot” με δυνατότητες και έμφαση στη φωτογραφική μηχανή. Οι σειρές τηλεφώνων με δυνατότητες multimedia όπως το Nokia Nseries, η Palm Pre, η HTC Dream και η Apple με το iPhone. Παρ’ όλα αυτά η πιο συχνά χρησιμοποιούμενη εφαρμογή δεδομένων για κινητά τηλέφωνα είναι SMS μηνυμάτων κειμένου. Το πρώτο γραπτό μήνυμα SMS στάλθηκε από έναν υπολογιστή σε ένα κινητό τηλέφωνο το 1992 στο Ηνωμένο Βασίλειο, ενώ το πρώτο SMS από τηλέφωνο σε τηλέφωνο εστάλη στη Φιλανδία το 1993. Το πρώτο κινητό ειδήσεων υπηρεσιών, που υλοποιούνται μέσω SMS, ξεκίνησε στη Φιλανδία το 2000, και στη συνέχεια πολλές οργανώσεις που παρέχουν υπηρεσίες ειδήσεων μέσω SMS.

Το χαρακτηριστικό που απαιτούν τα GSM τηλέφωνα είναι ένα μικροσίπ που ονομάζεται Subscriber Identity Module ή σε συντομογραφία κάρτα SIM, για να λειτουργήσει. Η κάρτα SIM είναι περίπου το μέγεθος ενός μικρού γραμματόσημου και τοποθετείται συνήθως κάτω από την μπαταρία, στο πίσω μέρος της μονάδας. Η κάρτα SIM αποθηκεύει με ασφάλεια το κλειδί υπηρεσίας του συνδρομητή (IMSI) και χρησιμοποιείται για τον εντοπισμό και την ταυτοποίηση του χρήστη του κινητού τηλεφώνου. Επίσης επιτρέπει η SIM στους χρήστες να αλλάξουν τα τηλέφωνα με την απλή αφαίρεση της κάρτας SIM από το ένα κινητό τηλέφωνο και την εισαγωγή της σε κάποια άλλη συσκευή κινητού τηλεφώνου, με την προϋπόθεση ότι αυτό δεν εμποδίζεται από ένα κλείδωμα της κάρτας SIM.



*Εικόνα 3. Κάρτα SIM (Subscriber Identity Module)*

Στην κατηγορία με τα κινητά τηλέφωνα εντάσσονται και τα υβριδικά με multi-κάρτα τηλέφωνα. Ένα υβριδικό κινητό τηλέφωνο μπορεί να χωρέσει μέχρι και τέσσερις κάρτες SIM. Οι SIM και οι RUIIM κάρτες μπορούν να αναμιχθούν για να επιστρέψουν τόσο GSM και CDMA δίκτυα για να έχουν πρόσβαση. Από το 2010 και μετά που έγινε δημοφιλής στην Ινδία και στην Ινδονησία και άλλες αναδυόμενες αγορές που αποδίδεται στην επιθυμία να αποκτήσει το χαμηλότερο ποσοστό καλώντας on-net. Στο 3<sup>ο</sup> τρίμηνο του 2011, η Nokia αποστέλει 18 εκατομμύρια χαμηλού κόστους διπλής SIM του τηλεφώνου της σειράς της σε μια προσπάθεια να καλύψει το χαμένο έδαφος στην αγορά των smartphones.

Σε γενικές γραμμές τα κινητά τηλέφωνα χρησιμοποιούνται για διάφορους σκοπούς, μεταξύ των οποίων και τη διατήρηση της επαφής με τα μέλη της οικογένειας, τη διεξαγωγή των επιχειρήσεων που έχουν πρόσβαση σε ένα τηλέφωνο σε περίπτωση έκτακτης ανάγκης. Μερικοί άνθρωποι χρησιμοποιούν το κινητό τους τηλέφωνο για διάφορους σκοπούς, όπως για επαγγελματική και προσωπική χρήση. Πολλαπλές κάρτες SIM μπορούν επίσης να χρησιμοποιηθούν για να επωφεληθούν από τα οφέλη των διαφόρων σχεδίων, για παράδειγμα ένα συγκεκριμένο σχέδιο θα μπορούσε να παρέχει φθηνότερα τοπικές κλήσεις, υπεραστικές κλήσεις, διεθνές κλήσεις ή περιαγωγής δεδομένων δηλαδή internet. Το κινητό τηλέφωνο έχει επίσης χρησιμοποιηθεί σε μια ποικιλία από διαφορετικά πλαίσια στην κοινωνία, για παράδειγμα:

1. Μια μελέτη από τη Motorola έδειξε ότι ένας στους δέκα συνδρομητές κινητής τηλεφωνίας έχουν ένα δεύτερο τηλέφωνο που συχνά παραμένει μυστικό από τα άλλα μέλη της οικογένειας. Αυτά τα τηλέφωνα μπορούσαν να χρησιμοποιηθούν για συμμετοχή σε δραστηριότητες, συμπεριλαμβάνοντας εξωσυζυγικές σχέσεις ή παράνομες επιχειρηματικές συναλλαγές.
2. Επίσης ο ΟΗΕ ανέφερε ότι τα κινητά τηλέφωνα έχουν εξαπλωθεί πιο γρήγορα από οποιαδήποτε άλλη τεχνολογία και μπορεί να βελτιώσουν το βιοτικό επίπεδο των φτωχότερων ανθρώπων στις αναπτυσσόμενες χώρες παρέχοντας πρόσβαση σε πληροφορίες σε χώρους όπου το διαδίκτυο δεν είναι διαθέσιμο, ιδιαίτερα στις λιγότερο αναπτυγμένες χώρες. Η χρήση των κινητών τηλεφώνων γεννά επίσης μια πληθώρα πολύ μικρών επιχειρήσεων, με παροχή εργασιών όπως η πώληση χρόνου ομιλίας και την επισκευή των συσκευών.

Τα τελευταία χρόνια στα κινητά τηλέφωνα έχουν ενταχθεί και τα smartphones. Τα smartphones έχουν μια σειρά από ιδιαίτερα χαρακτηριστικά που έχουν ξεπεράσει κατά πολύ τη χρήση των προηγούμενων κινητών συσκευών. Ιδιαίτερη έμφαση τότε είχαν δώσει για τη διανομή περιεχομένου. Το 1998 μια από τις πρώτες εφαρμογές που υποστήριζε ένα κινητό τηλέφωνο τη διανομή και πώληση περιεχομένου πολυμέσων ήταν η πώληση των ringtones από την Randiolinja στη Φιλανδία. Λίγο αργότερα ένα άλλο περιεχόμενο πολυμέσων που εμφανίστηκε ήταν οι ειδήσεις, τα τηλεοπτικά παιχνίδια, αστέρια, ωροσκόπια, το περιεχόμενο της τηλεόρασης και άλλα.

Σε αρκετές χώρες, τα κινητά τηλέφωνα που χρησιμοποιούνται για την παροχή mobile banking υπηρεσιών, οι οποίες μπορεί να περιλαμβάνουν την ικανότητα να μεταφέρουν τις πληρωμές τους μετρητοίς με ένα ασφαλές μήνυμα κειμένου SMS. Για

παράδειγμα η <sup>6</sup>M-PESA mobile banking υπηρεσιών της Κένυας επιτρέπει στους πελάτες του παροχέα κινητής τηλεφωνίας Safaricom να κρατήσει ταμειακά υπόλοιπα που καταγράφονται στις κάρτες SIM τους. Μετρητά μπορούν να κατατεθούν ή να αποσυρθούν από τους λογαριασμούς M-PESA στο Safaricom κατάστημα λιανικής πώλησης σε όλη τη χώρα και μπορεί να μεταφερθεί ηλεκτρονικά από άτομο σε άτομο, καθώς χρησιμοποιείται για να πληρώσουν τους λογαριασμούς σε εταιρείες.

Τα κινητά τηλέφωνα χρησιμοποιούνται επίσης πολύ συχνά για τη συλλογή των δεδομένων θέσης. Όταν το τηλέφωνο είναι ενεργοποιημένο, η γεωγραφική θέση ενός κινητού τηλεφώνου μπορεί να καθοριστεί εύκολα χρησιμοποιώντας μια τεχνική γνωστή ως πολυπλεύριση. Έτσι λοιπόν οι κινήσεις ενός κινητού τηλεφώνου μπορεί να παρακολουθούνται από τον πάροχο των υπηρεσιών τους και εάν είναι επιθυμητό, από τις υπηρεσίες επιβολής του νόμου και της κυβέρνησής τους.



**Εικόνα 4. Κινητό τηλέφωνο  
NOKIA 3310**



**Εικόνα 5. Κινητό τηλέφωνο  
Sony Ericsson W660**

---

<sup>6</sup> M-Pesa : (M για κινητά, Pesa είναι Σουαζίλι για τα χρήματα) είναι ένα κινητό τηλέφωνο με βάση τη μεταφορά χρημάτων και της μικροχρηματοδότησης υπηρεσίας, που ξεκίνησε το 2007 από τη Vodafone για Safaricom και Vodacom, που είναι οι μεγαλύτεροι φορείς εκμετάλλευσης κινητών δικτύων στην Κένυα και την Τανζανία. Από τότε έχει επεκταθεί στο Αφγανιστάν, τη Νότια Αφρική, την Ινδία και το 2014 στην Ανατολική Ευρώπη. Η M – Pesa επιτρέπει στους χρήστες να καταθέσουν να τραβήξουν και να μεταφέρουν χρήματα εύκολα με μια φορητή συσκευή.

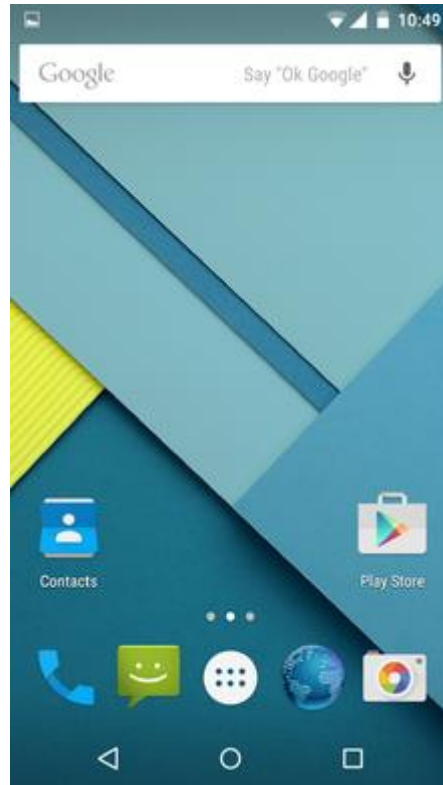
## 2.3 Smartphones

Το smartphone είναι ένα κινητό τηλέφωνο βασισμένο σε ένα λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με ένα απλό κινητό τηλέφωνο. Τα πρώτα smartphones συνδιάζουν τις λειτουργίες ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός κινητού τηλεφώνου. Σε μεταγενέστερα μοντέλα προστέθηκαν οι λειτουργίες των φορητών media players, low-end compact ψηφιακές φωτογραφικές μηχανές βιντεοκάμερες τσέπης, καθώς και μονάδες πλοήγησης GPS, με αποτέλεσμα να διαμορφωθεί μια πολύχρηστική συσκευή.

Πολλά σύγχρονα smartphones περιλαμβάνουν επίσης οθόνες αφής υψηλής ανάλυσης και web browsers που εμφανίζουν τυποποιημένες ιστοσελίδες, καθώς και βελτιστοποιημένες ιστοσελίδες για κινητά. Η πρόσβαση σε δεδομένα υψηλής ταχύτητας παρέχεται μέσω Wi-Fi και μέσω κινητών ευρυζωνικών υπηρεσιών. Τα τελευταία χρόνια, η ταχεία ανάπτυξη στην αγορά των εφαρμογών για κινητά και στο εμπόριο κινητών τηλεφώνων έχει γίνει οδηγός για την ευρεία υιοθέτηση των smartphones.

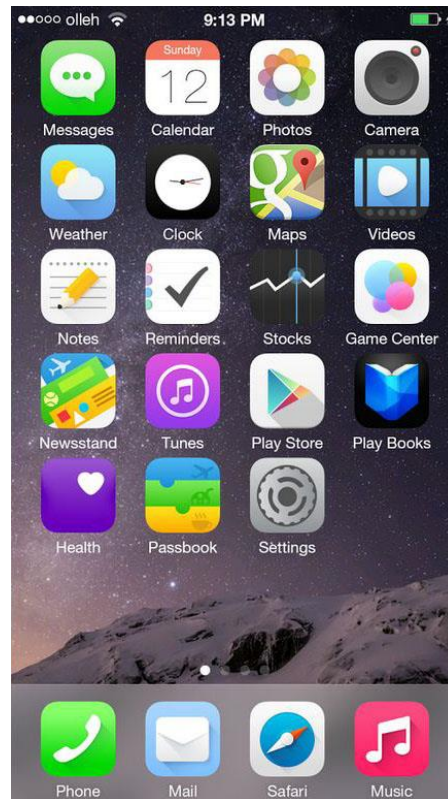
Τα λειτουργικά συστήματα (OS) των κινητών τηλεφώνων που χρησιμοποιούνται από τα σύγχρονα smartphones περιλαμβάνουν το Android, το iOS της Apple, το Symbian της Nokia, το BlackBerry OS της RIM, το Bada της Samsung, τα Windows Phone της Microsoft, το webOS της Hewlett-Packard, καθώς και ενσωματωμένες διανομές Linux όπως το Maemo και το MeeGo. Τέτοιου είδους λειτουργικά συστήματα μπορούν να εγκατασταθούν σε πολλά διαφορετικά μοντέλα κινητών τηλεφώνων και συνήθως κάθε συσκευή μπορεί να λάβει πολλές ενημερώσεις εκδόσεων λογισμικού λειτουργικού συστήματος κατά τη διάρκεια της λειτουργίας του. Μερικά άλλα επερχόμενα λειτουργικά συστήματα είναι το Firefox OS της Mozilla, το Ubuntu Phone της Canonical Ltd' s και το Tizen. Παρακάτω καταγράφονται πιο αναλυτικά τα πιο διαδεδομένα λειτουργικά συστήματα:

Android είναι μια open source πλατφόρμα που ιδρύθηκε τον Οκτώβριο του 2003 από τον Andy Rubin και υποστηρίζεται από την Google, μαζί με μεγάλα hardware και λογισμικά όπως η Intel, HTC, η Motorola και η Samsung που αποτελούν το Open Handset Alliance. Τον Οκτώβριο του 2008, το HTC κυκλοφόρησε το, το πρώτο κινητό τηλέφωνο που χρησιμοποιούν το Android. Η σουίτα λογισμικού που περιλαμβάνεται στο τηλέφωνο αποτελείται από την ενσωμάτωση με αποκλειστικές εφαρμογές της Google, όπως Χάρτες, Ημερολόγιο και το Gmail και ένα πλήρες πρόγραμμα περιήγησης στο διαδίκτυο. Το Android υποστηρίζει την εγκατάσταση εφαρμογών από τρίτους κατασκευαστές εφαρμογών που διατίθενται μέσω του Google Play, το οποίο ξεκίνησε τον Οκτώβριο του 2008, όπως το Android Market. Από το 4<sup>ο</sup> τρίμηνο του 2010, το Android έγινε best-seller πλατφόρμα smartphone.



*Εικόνα 6. Android Lollipop launcher 5.0 (2014)*

iOS είναι ένα κινητό λειτουργικό σύστημα που αναπτύχθηκε από την Apple και διανέμονται αποκλειστικά για το υλικό της Apple. Το 2007 η Apple παρουσίασε το iPhone, την πρώτη συσκευή που θα χρησιμοποιεί το iOS και ένα από τα πρώτα smartphones που πρέπει να χρησιμοποιήσετε ένα multi touch interface. Το πρώτο iPhone ήταν αξιοσημείωτο για τη χρήση του σε μεγάλη οθόνη αφής για άμεση εισαγωγή του χρήστη στο μενού του ως κύριο μέσο αλληλεπίδρασης το δάχτυλό του χρήστη, αντί για ένα στυλό, ή ένα τυπικό πληκτρολόγιο όπως ήταν τότε. Το 2008 η Apple εισήγαγε το App Store, γεγονός που επέτρεψε σε οποιοδήποτε χρήστη iPhone να εγκαταστήσει εφαρμογές από τρίτους κατασκευαστές που φτιάχνουν εφαρμογές για iOS. Το πρώτο έτος το App Store εκτοξεύτηκε στις 1.000.000.000 λήψεις και 75 δισεκατομμύρια στα μέσα του 2014. Τον Ιανουάριο του 2015, η Apple ανακοίνωσε ότι έχει πουλήσει ένα δισεκατομμύριο iOS συσκευές.



*Εικόνα 7. iOS 8.3 launcher (2015)*

Το 2010, η Microsoft παρουσίασε τα Windows Phone 7 με ένα περιβάλλον εργασίας χρήστη εμπνευσμένο από το «<sup>7</sup>Microsoft metro design language», για να αντικαταστήσει τα Windows Mobile. Τα Windows Phone 7 ενσωματώνονται με τις υπηρεσίες της Microsoft, όπως το Microsoft SkyDrive, το Office, το Xbox και το Bing, καθώς και μη υπηρεσίες της Microsoft όπως, το Facebook, το Twitter και ο λογαριασμός Google. Αυτή η πλατφόρμα λογισμικού εκτελεί τα Microsoft Mobile smartphones, και έχει λάβει κάποια θετική υποδοχή από τον τύπο της τεχνολογίας και έχουν επαινεθεί για τη μοναδικότητα και τη διαφοροποίηση της. Το 2012, η Microsoft κυκλοφόρησε τα Windows Phone 8, αντικαθιστώντας το παρελθόν της Windows CE-based αρχιτεκτονική με ένα βασισμένο στο Windows NT πυρήνα, με πολλά συστατικά από κοινού με τα Windows 8 να επιτρέπουν στις εφαρμογές να μεταφέρονται μεταξύ των δύο πλατφορμών.

---

<sup>7</sup> Microsoft metro design language: είναι ένα είδος τυπογραφίας βασισμένη στη σχεδιαστική γλώσσα από τη Microsoft, κυρίως για τις διεπαφές του χρήστη. Μια βασική αρχή του σχεδιασμού είναι η καλύτερη εστίαση σχετικά με το περιεχόμενο, που θα στηρίζονται περισσότερο για την τυπογραφία και λιγότερο για τα γραφικά.



*Εικόνα 8. Windows Phone 8.1 launcher (2014)*

Το 1999 η RIM κυκλοφόρησε την πρώτη συσκευή BlackBerry, παρέχοντας ασφαλείς επικοινωνίες push-email σε πραγματικό χρόνο σχετικά με τις ασύρματες συσκευές. Οι υπηρεσίες όπως το BlackBerry Messenger παρέχει την ενοποίηση όλων των επικοινωνιών σε ένα ενιαίο φάκελο εισερχόμενων. Το Σεπτέμβριο του 2012, η RIM ανακοίνωσε ότι 2.000.000.000 BlackBerry smartphone έχουν αποσταλεί. Από το Σεπτέμβριο του 2014 υπάρχουν περίπου 46 εκατομμύρια ενεργούς συνδρομητές υπηρεσιών BlackBerry. Πιο πρόσφατα, η RIM έχει μια μετάβαση της πλατφόρμας της, αλλάζοντας το όνομά της σε BlackBerry και την πραγματοποίηση νέων συσκευών σε μια νέα πλατφόρμα με το όνομα BlackBerry.



*Εικόνα 9. BlackBerry launcher*

Παρότι οι συσκευές που συνδυάζουν τηλεφωνία και πληροφορική είχαν ήδη επινοηθεί από το 1973 και διατέθηκαν προς πώληση το 1994, ο όρος «smartphone» δεν εμφανίστηκε μέχρι το 1997, όταν η Ericsson περιέγραψε το GS 88 “Penelope” ως «Smart Phone» δηλαδή «Εξυπνο Τηλέφωνο».

Η διάκριση μεταξύ των smartphones και των απλών κινητών τηλεφώνων μπορεί να είναι ασαφής και δεν υπάρχει επίσημος ορισμός για το ποιες είναι οι μεταξύ τους διαφορές. Μια από τις πιο σημαντικές διαφορές είναι ότι οι προηγμένες διεπαφές προγραμματισμού εφαρμογών (APIs) στα smartphones σχετικά με τη λειτουργία τρίτων εφαρμογών μπορούν να επιτρέψουν σε αυτές τις εφαρμογές να έχουν καλύτερη ανσωματώση στο λειτουργικό σύστημα και στο hardware του τηλεφώνου απ’ ότι συμβαίνει συνήθως στα απλά κινητά τηλέφωνα. Συγκριτικά, τα απλά κινητά τηλέφωνα τρέχουν συχνότερα σε ιδιόκτητο firmware, με υποστήριξη λογισμικού από τρίτους, μέσα από πλατφόρμες όπως η JAVA ME ή το BREW. Μια επιπλέον δυσκολία στη διάκριση μεταξύ smartphones και απλών κινητών τηλεφώνων είναι ότι, με την πάροδο του χρόνου, οι δυνατότητες των νέων μοντέλων των απλών κινητών τηλεφώνων μπορούν να υπερβούν εκείνες των τηλεφώνων που είχαν προωθηθεί ως smartphones στο παρελθόν. Ορισμένοι κατασκευαστές και πάροχοι χρησιμοποιούν τον όρο «smartphone» για τα υψηλής απόδοσης τηλέφωνα, με τις μεγάλες οθόνες και τα ακριβώς χαρακτηριστικά τους. Άλλοι προτιμούν τον όρο «phablet», αναγνωρίζοντας τη σύγκλισή τους με τους υπολογιστές tablet χαμηλής απόδοσης.

Στα πρώτα χρόνια που οι συσκευές συνδυάζαν τηλεφωνία και πληροφορική το 1973, ο Θεόδωρος Γεώργιος Παρασκευάκος κατοχύρωσε με δίπλωμα ευρεσιτεχνίας τις έννοιες



του συνδιαζμού νοημοσύνης, της επεξεργασίας δεδομένων και των οθονών οπτικής απεικόνισης με τα τηλέφωνα, περιγράφοντας έτσι τις κοινές πλέον δραστηριότητες των τραπεζικών συναλλαγών και την πληρωμή λογαριασμών κοινής ωφέλειας μέσω τηλεφώνου. Το πρώτο κινητό τηλέφωνο με ενσωματωμένα χαρακτηριστικά PDA ήταν ένα πρότυπο IBM που αναπτύχθηκε το 1992 και η επίδειξη του έγινε το ίδιο έτος στην εμπορική έκθεση βιομηχανίας πληροφορικής COMDEX. Μια ανανεωμένη έκδοση του προϊόντος διατέθηκε στο εμπόριο για τους καταναλωτές στις 16 Αυγούστου 1994 από την BellSouth, με την επωνυμία Simon Personal Communicator. Το Simon ήταν η πρώτη συσκευή που μπορεί να αναφέρεται ως «smartphone», έστω και αν αυτός ο όρος δεν είχε ακόμα επινοηθεί. Εκτός από την ικανότητα του να πραγματοποιηθεί και να δέχεται κλήσεις από κινητό τηλέφωνο, το Simon ήταν επίσης σε θέση να αποστέλλει και να λαμβάνει fax, email και σελίδες μέσω της οθόνης αφής του. Στο Simon περιλαμβάνονταν πολλές εφαρμογές, όπως βιβλίο διευθύνσεων, ημερολόγιο, χρονοπρογραμματιστής ραντεβού, αριθμομηχανή, παγκόσμιο ρολόι, παιχνίδια, ηλεκτρονικό σημειωματάριο, χειρόγραφες σημειώσεις, έξυπνα πληκτρολόγια αφής και πολλά άλλα.

Το 1996, η Nokia κυκλοφόρησε το Nokia 9000, μέρος της σειράς Nokia Communicator, το οποίο ήταν το τηλέφωνο με τις μεγαλύτερες πωλήσεις εκείνης της εποχής. Ήταν ένα τηλέφωνο palmtop σε ύψος υπολογιστή, σε συνδιασμό με ένα PDA από την HP. Στα πρώτα πρωτότυπα, οι δύο συσκευές είχαν συνδεθεί μαζί με έναν σύνδεσμο και το εγχείρημα περιγράφηκε ως μια σχεδίαση clamshell. Όταν άνοιγε η οθόνη των 640×200 pixels ήταν στην εσωτερική άνω επιφάνεια και ένα φυσικό πληκτρολόγιο τύπου QWERTY βρισκόταν στο κάτω μέρος. Τα e-mail και η περιήγηση στο διαδίκτυο με βάση το κείμενο παρέχονταν μέσω του λειτουργικού συστήματος GEOS V3.0. Στα τέλη της δεκαετίας του 1990 όμως, η συντριπτική πλειοψηφία των κινητών τηλεφώνων διέθεταν μόνο τις βασικές λειτουργίες του τηλεφώνου, συνεπώς πολλοί άνθρωποι κουβαλούσαν επίσης μια ξεχωριστή ειδική συσκευή PDA που έτρεχε τις πρώτες εκδόσεις των λειτουργικών συστημάτων, όπως το Palm OS, το BlackBerry OS ή τα Windows CE/Pocket PC. Αυτά τα λειτουργικά συστήματα θα εξελίσσονταν αργότερα σε λειτουργικά συστήματα κινητών τηλεφώνων που θα στελέχωναν μερικά από τα high – end smartphones.

Στις αρχές του 2001, η Palm, Inc εισήγαγε το Kyocera 6035, το πρώτο smartphone στις Ηνωμένες Πολιτείες. Η συσκευή αυτή συνδύαζε ένα PDA με ένα κινητό τηλέφωνο και λειτουργούσε στο δίκτυο της Verizon Wireless. Υποστήριζε επίσης περιορισμένη περιήγηση στο διαδίκτυο και το 2004, η HP κυκλοφόρησε το iPAQ h6315, μια συσκευή που συνδύαζε το προηγούμενο PDA τους, το HP 2215, με δυνατότητες κινητού τηλεφώνου.

Τα χαρακτηριστικά των smartphones που τα κάνουν ξεχωριστά σε σχέση με άλλες κινητές συσκευές και τηλέφωνα είναι ο επεξεργαστής τους και η οθόνη τους. Ο επεξεργαστής είναι το κεντρικό εξάρτημα που επεξεργάζεται δεδομένα σε ένα ηλεκτρονικό υπολογιστή, ελέγχει τη λειτουργικότητα και εκτελεί βασικές λειτουργίες διασύνδεσης και μεταβίβασης εντολών. Οι επεξεργαστές δεν σχετίζονται αποκλειστικά με τους ηλεκτρονικούς υπολογιστές καθώς πλέον ενσωματώνονται σε κάθε είδους συσκευής στην οποία απαιτείται ύπαρξη υπολογιστικής ικανότητας.

Οι οθόνες των smartphones ποικίλλουν σε μεγάλο βαθμό τόσο σε μέγεθος όσο και σε ανάλυση. Τα πιο συνηθισμένα μεγέθη οθόνης κυμαίνονται από 3 ίντσες ως πάνω από 5 ίντσες και υπάρχουν διάφοροι τύποι οθονών όπως: LCD, LED, OLED, AMOLED, IPS

και άλλα. Ορισμένες συσκευές με οθόνη 6 ή 8 ιντσών τρέχουν σε λειτουργικά συστήματα κινητών τηλεφώνων και έχουν την ικανότητα να πραγματοποιούν τηλεφωνικές κλήσεις, όπως το Huawei Ascend Mate, το Asus Fonepad και το Samsung Galaxy Note 8.0. Έχουν ακουστεί ορισμένα επιχειρήματα σχετικά με την εργονομία, ότι η αύξηση των μεγεθών οθόνης αρχίζει να επηρεάζει αρνητικά τη χρηστικότητα. Οι συχνότερες αναλύσεις για οθόνες smartphone ποικίλλουν από 240 × 320 (QVGA) pixels έως 1080 × 1920 (Full HD), με την πλειοψηφία των τηλεφώνων Android, τα οποία έχουν συνήθως ανάλυση ίση ή μεγαλύτερη από 1280 × 720 (720p HD) και το iPhone 6 και iPhone 6 plus αντίστοιχα έχουν 1334 × 750 και 1920 × 1080.

Από τα τέλη του 2012 και τις αρχές του 2013, έχει προκύψει στο χώρο της βιομηχανίας μια τάση μετακίνησης προς τις οθόνες smartphone με πλήρη HD χαρακτηριστικά (1080 pixels) για τις high-end συσκευές. Με το πέρασμα των ετών, η πυκνότητα των pixels στις οθόνες των smartphones γίνεται όλο και μεγαλύτερη. Η Apple έλαβε το αναγνωρισμένο σήμα για την οθόνη υψηλής πυκνότητας με την επωνυμία Retina. Η ανάλυση υψηλής πυκνότητας βολεύει για μικρούς, περίπλοκους χαρακτήρες (κυρίως για κείμενα ανατολικής Ασίας) και αιχμηρές άκρες (ιδίως γραμμικό σχέδιο και γραμματισειρές Aliased).

Η ζήτηση των έξυπνων τηλεφώνων έχει ξεπεράσει το υπόλοιπο της αγοράς της κινητής τηλεφωνίας. Σύμφωνα με μια έρευνα του 2012, περίπου το ήμισυ των καταναλωτών των κινητών τηλεφώνων στις ΗΠΑ έχουν στην κατοχή τους smartphones που αντιπροσωπεύουν περίπου το 70% του συνόλου των κινητών συσκευών των ΗΠΑ από το 2013. Στις ηλικίες από 25 έως 34 ετών, η ιδιοκτησία smartphone φέρεται να φθάνει το 62%. Το <sup>8</sup>NPD Group αναφέρει ότι, στις ΗΠΑ, το μερίδιο πωλήσεων των smartphone συσκευών κατά το 3<sup>ο</sup> τρίμηνο του 2011 έφθασε σε 59% για τους καταναλωτές άνω των 18 ετών.

Η ευρωπαϊκή αγορά κινητής τηλεφωνίας, όπως μετράται από ενεργούς συνδρομητές από τα κορυφαία 50 δίκτυα, αποτελείται από 860 εκατομμύρια χρήστες. Σύμφωνα με μια έκθεση της Olswang στις αρχές του 2011, το ποσοστό υιοθέτησης των smartphone έχει ως εξής: από το Μάρτιο του 2011, το 22% των Βρετανών καταναλωτών είχαν smartphone, με το ποσοστό αυτό να αυξάνεται στο 31% στις ηλικίες μεταξύ 24 έως 35 ετών. Στην Κίνα, τα smartphones αντιπροσώπευαν περισσότερο από το ήμισυ 51% των αποστολών συσκευών κατά το δεύτερο τρίμηνο του 2012. Παγκοσμίως σε μερίδιο επί των κερδών, τα smartphones υπερβαίνουν κατά πολύ το μερίδιο των απλών κινητών τηλεφώνων. Σύμφωνα με μια έρευνα του Canaccord Genuity τον Νοέμβριο του 2011, η Apple κατέχει το 52% των συνολικών κερδών της βιομηχανίας κινητής τηλεφωνίας, ενώ κατέχει μόνο το 4,2% της παγκόσμιας αγοράς συσκευών. Ομοίως η HTC και η RIM κατασκευάζουν μόνο smartphones και τα παγκόσμια μερίδια κέρδους τους είναι στο 9% και 7% αντίστοιχα. Η Samsung, στη δεύτερη θέση με 29% μετά την Apple, κατασκευάζει τόσο smartphones όσο και απλά κινητά τηλέφωνα, αλλά δεν εναφέρει την κατανομή διαχωρισμού των κερδών της μεταξύ των δύο ειδών συσκευών.

---

<sup>8</sup> NPD Group: είναι μια εταιρεία έρευνας αγοράς. Ο όμιλος NPD κατατάσσεται σταθερά μεταξύ των κορυφαίων 25 εταιρειών έρευνας αγοράς στην ανεξάρτητη Honomichl Top 50 της έκθεσης, η οποία τα μέσα ενημέρωσης και η έρευνα της βιομηχανίας αναγνωρίζουν ως μια αξιόπιστη πηγή πληροφοριών σχετικά με την έρευνα αγοράς της βιομηχανίας.

Μέχρι το τέλος του Νοεμβρίου του 2011, τα smartphones με ενσωματωμένη φωτογραφική μηχανή λάμβαναν το 27% των φωτογραφιών, μια σημαντική αύξηση από το 17% το 2010. Για πολλούς ανθρώπους, τα smartphones έχουν αντικαταστήσει τις παραδοσιακές φωτογραφικές μηχανές και κάμερες. Μια μελέτη που πραγματοποιήθηκε τον Σεπτέμβριο του 2012 κατέληξε στο συμπέρασμα ότι 4 στους 5 ιδιοκτήτες smartphone χρησιμοποιούν τη συσκευή για τα ψώνια τους. Στις αρχές του 2013, οι πωλήσεις smartphone ξεπέρασαν παγκοσμίως εκείνες των κινητών τηλεφώνων παλαιότερων στυλ.

Από το λανσάρισμα του μοντέλου Communicator το 1996 μέχρι το 2011, η Nokia ήταν κυρίαρχη στην αγορά των smartphones, αν και πρόσφατα εισήλθαν και άλλοι ανταγωνιστές στην αγορά. Με βάση μια έκθεση της Strategy Analytics, η Samsung ξεπέρασε τη Nokia σε αποστολές smartphone με περίπου 27,8 εκατομμύρια μονάδες να έχουν αποσταλεί κατά το 3<sup>ο</sup> τρίμηνο του 2011. Η Apple ξεπέρασε παγκοσμίως τη Nokia με βάση τα έσοδα και τα κέρδη για πρώτη φορά το 2<sup>ο</sup> τρίμηνο του 2011, με το μερίδιο επί των κερδών της Apple της συνολικής παγκόσμιας αγοράς smartphone να αυξάνεται στο 66,3%.

Μεταξύ του 2<sup>ου</sup> τριμήνου του 2010 και του 2<sup>ου</sup> τριμήνου του 2011, οι παγκόσμιες πωλήσεις Symbian smartphone της Nokia μειώθηκαν σημαντικά από 38,1% σε 15,2%, ενώ οι πωλήσεις smartphone της Samsung αυξήθηκαν σημαντικά σε όλο τον κόσμο από 5% έως 17,5%. Από το 1<sup>ο</sup> τρίμηνο του 2011, η Nokia είχε ήδη ανακοινώσει σχέδια μετάβασης σε Windows Phone. Τα Samsung smartphones χρησιμοποιούν ένα ευρύ χαρτοφυλάκιο λειτουργικών συστημάτων, συμπεριλαμβανομένου του δικού τους λειτουργικού συστήματος Bada, μαζί με τα Android και τα Windows Mobile. Το μερίδιο αγοράς μεταξύ των κατασκευαστών smartphone δεν ισοδυναμεί με το μερίδιο αγοράς των λειτουργικών συστημάτων, δεδομένου ότι ορισμένα λειτουργικά συστήματα είναι διαθέσιμα μόνο προς έναν κατασκευαστή και άλλα έχουν λάβει άδεια από πολλούς κατασκευαστές. Το iPhone της Apple, το Symbian της Nokia και το BlackBerry της RIM είναι προς το παρόν διαθέσιμα μόνο από ενιαίους κατασκευαστές. Το λειτουργικό σύστημα Android της Google και τα λειτουργικά συστήματα για κινητά της Microsoft είναι πλατφόρμες που έχουν λάβει άδεια και χρησιμοποιούνται από διάφορους κατασκευαστές. Ως αποτέλεσμα, οι κατασκευαστές smartphones που χρησιμοποιούν εξουσιοδοτημένα λειτουργικά συστήματα διαμοιράζουν όλοι το συνολικό μερίδιο αγοράς του συγκεκριμένου λειτουργικού συστήματος μεταξύ τους, ενώ το συνολικό μερίδιο για ένα λειτουργικό σύστημα ενός ενιαίου κατασκευαστή κατέχει μόνο ο εν λόγω κατασκευαστής. Το λειτουργικό σύστημα Symbian της Nokia ήταν διαθέσιμο στο παρελθόν από διάφορους κατασκευαστές υπό εξουσιοδοτημένο μοντέλο, αργότερα ήταν διαθέσιμο κατά κύριο λόγο μόνο από τη Nokia.

Κατά το 1<sup>ο</sup> τρίμηνο του 2012, μετά από 14 χρόνια στην αγορά, η Samsung ξεπέρασε τη Nokia σε μονάδες πώλησης. Η Samsung βρέθηκε επίσης στην πρώτη θέση με πωλήσεις 44,5 εκατομμυρίων smartphones ή 30,6% του μεριδίου αγοράς, ενώ πωλήθηκαν 35,1 εκατομμύρια iPhone ή 24,1% του μεριδίου αγοράς.

Παρόλα αυτά το 2013, η QSAIpha ξεκίνησε την παραγωγή ενός smartphone σχεδιασμένο εξ ολοκλήρου γύρω από την ασφάλεια, την κρυπτογράφηση και την προστασία της ταυτότητας. Τον Δεκέμβριο του 2013, η πρώτη curved στον κόσμο OLED smartphones τεχνολογία εισήχθησαν στη λιανική αγορά με την πώληση του Samsung Galaxy και LG G Flex. Στις αρχές του 2014, τα smartphones έχουν αρχίσει να χρησιμοποιούν Quand HD (2K) 2560 × 1440 σε 5.5' οθόνη έως και 534 ppi σε

συσκευές όπως το G3 LG η οποία είναι μια σημαντική βελτίωση σε σχέση με της Apple Retina Display. Η Quad HD χρησιμοποιείται σε προηγμένες τηλεοράσεις και ηλεκτρονικών υπολογιστών οθόνες, αλλά με 110 ppi ή λιγότερο σε τέτοιες μεγάλες οθόνες. Και τέλος από το 2013, μέσω των υψηλά τεχνολογικών μέσων έχει αντιμετωπιστεί και το πρόβλημα του νερού και της σκόνης στα smartphones όπως το Sony Xperia Z, το Sony Xperia Z3 και το Samsung Galaxy S5 που είναι πιο ανθεκτικά ως προς το νερό σε σχέση με τα υπόλοιπα smartphones.

Τέλος όπως και τα κινητά τηλέφωνα έτσι και τα smartphones μπορούν να στέλνουν και να λαμβάνουν δεδομένα πολύ πιο γρήγορα σε σχέση με τα παλαιότερα τηλέφωνα. Η βιομηχανία χρησιμοποιεί διαφορετικά πρότυπα μετάδοσης δεδομένων όπως το 2G που εισήχθη το 1991 με ταχύτητα 56Kbit / s, το 3G που εισήχθη το 2002 με ταχύτητα περίπου 200Kbit / s έως 14Mbit / s και τέλος το 4G που χρησιμοποιείται από το 2010 μέχρι σήμερα με ταχύτητες που εκτιμώνται από 100Mbit / s και 1Gbit /s.



*Εικόνα 10. iPhone 6 & iPhone 6 plus*



*Εικόνα 11. Samsung Galaxy S6 Edge*



*Εικόνα 12. Nokia Lumia 730*

## 2.4 Tablets

Tablet είναι ένας φορητός υπολογιστής ή προσωπικός ψηφιακός βοηθός και είναι σε μέγεθος μεγαλύτερο από ένα κινητό τηλέφωνο ή από ένα smartphone. Είναι ενσωματωμένο σε μια επίπεδη οθόνη αφής και λειτουργεί κυρίως αγγίζοντας την οθόνη αντί να χρησιμοποιείται ένα φυσικό πληκτρολόγιο. Τα tablets είναι εξοπλισμένα με αισθητήρες, συμπεριλαμβανομένων κάμερες, ένα μικρόφωνο και ένα επιταχυνσιόμετρο. Διαθέτουν συνήθως στην οθόνη ένα pop-up εικονικό πληκτρολόγιο.

Τα tablets μπορεί να περιλαμβάνουν και φυσικά κουμπιά για βασικές λειτουργίες, όπως η ένταση του ήχου και το κουμπί power on που θέτει σε λειτουργία το tablet. Τα tablets pc μπορούν να ταξινομηθούν σε διάφορες κατηγορίες, ανάλογα με την παρουσία και την φυσική εμφάνιση του πληκτρολογίου. Οι κατηγορίες slates και booklets δεν έχουν το τυπικό πληκτρολόγιο που έχει ένα pc ή laptop και συνήθως διαθέτουν ένα εικονικό πληκτρολόγιο που προβάλλεται στην οθόνη αφής για την εισαγωγή και συγγραφή κειμένου. Σε αντίθεση με τις κατηγορίες Hybrids και Convertibles που διαθέτουν και αυτές οι κατηγορίες ένα εικονικό πληκτρολόγιο έχουν τη δυνατότητα να προσφέρουν και ένα τυπικό πληκτρολόγιο για την εισαγωγή και συγγραφή κειμένου.

Το Σεπτέμβριο του 1989 κυκλοφόρησε το πρώτο εμπορικό tablet με την ονομασία GRIDPad. Το 1991, η AT & T κυκλοφόρησε το πρώτο τους EO Personal Communicator, αυτό ήταν ένα από τα πρώτα εμπορικά διαθέσιμα tablets που λάνσαρε η GO Corporation της PenPoint OS με hardware της AT & T, συμπεριλαμβανομένης και της δικής τους AT & T Hobbit CPU. Το 1992, η Atari έβγαλε το Stylus που αργότερα μετονομάστηκε σε ST-Pad που ήταν πρωτότυπο για τους προγραμματιστές, διότι βασιζόταν στην πλατφόρμα TOS / GEM Atari ST.

Η Apple Computers ξεκίνησε τον προσωπικό ψηφιακό βοηθό το 1993. Το σύστημα που χρησιμοποίησε η Apple ήταν το Newton OS και τα υλικά που κατασκευάζονταν ήταν από τη Motorola και ενσωματώνοντας ένα επεξεργαστή ARM που η Apple είχε αναπτύξει σε συνεργασία με την Acorn Computers. Το 1996, η Palm Inc κυκλοφόρησε το πρώτο Palm OS με βάση PalmPilot αφής και γραφίδα με βάση το PDA. Η Intel το 1999 ανακοίνωσε το StrongARM υπολογιστή tablet που βασίζεται σε επεξεργαστή οθόνη αφής, με την επωνυμία WebPad.

Το 2000, η Νορβηγική εταιρεία Screen Media AS και η Γερμανική εταιρεία Dosch & Amand GmbH κυκλοφόρησε το «FreePad». Ήταν βασισμένο σε Linux για να χρησιμοποιηθεί το πρόγραμμα περιήγησης Opera. Η πρόσβαση στο internet παρέχεται από DECT OMAP που είναι διαθέσιμη μόνο στην Ευρώπη και παρείχαν μέχρι ασύρματη πρόσβαση 10Mbit /s. Η συσκευή είχε 16 MB μνήμη αποθηκευτικού χώρου και μνήμη RAM 32 MB. Το FreePad πωλήθηκαν στη Νορβηγία και τη Μέση Ανατολή αλλά η εταιρεία διαλύθηκε το 2003.

Το Απρίλιο του 2000 η Microsoft εγκαινίασε το Pocket PC 2000 με λειτουργικό σύστημα Windows CE 3.0. Οι συσκευές που κατασκευάζονται από διάφορους κατασκευαστές βασίζονται σε ένα συνδυασμό υλικού x86, MIPS, ARM και SuperH. Το 2002, η Microsoft προσπάθησε να καθορίσει το Microsoft Tablet PC ως ένα φορητό υπολογιστή για την εργασία στον τομέα των επιχειρήσεων, αν και αυτές οι συσκευές απέτυχαν, κυρίως λόγω των τιμών τους. Η Nokia είχε κάποια σχέδια πριν το 2000 για ένα internet tablet, το οποίο κατασκευάστηκε το 2001 και ήταν από τα πρώτα μοντέλα της, το Nokia M510 το οποίο έτρεχε λειτουργικό σύστημα EPOC και διέθετε πρόγραμμα περιήγησης της Opera, ηχεία και μια οθόνη των 10 ιντσών 800 × 600 pixels, αλλά δεν κυκλοφόρησε ποτέ λόγω του φόβου πως η αγορά δεν ήταν έτοιμη για μια τέτοια τεχνολογία. Το 2005 η Nokia κυκλοφόρησε τελικά το πρώτο της Tablet το Nokia 770 με λειτουργικό σύστημα Linux που ονομάζεται Maemo, χρησιμοποιώντας τον όρο internet tablet αναφέρεται σε μια φορητή συσκευή πληροφοριών που θα εστιάζει στη χρήση του διαδικτύου και των μέσων μαζικής ενημέρωσης. Έτσι κατασκευάστηκαν άλλες δύο συσκευές με χαρακτηριστικά ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός Ultra – Mobile PC (UMPC), το Nokia N900 που τρέχει <sup>9</sup>Maemo και το N9 που τρέχει <sup>10</sup>Meego.

---

<sup>9</sup> Maemo είναι μια πλατφόρμα λογισμικού που αναπτύχθηκε από τη Nokia και στη συνέχεια παραδόθηκε στο ίδρυμα Hilton για smartphones και tablet που έχουν πρόσβαση στο διαδίκτυο. Βασίζεται στο Debian GNU / Linux. Η πλατφόρμα περιλαμβάνει το λειτουργικό σύστημα Maemo και Maemo SDK. Το Maemo βασίζεται κυρίως σε open-source κώδικα και έχει αναπτυχθεί από τη Maemo για συσκευές της Nokia σε συνεργασία με πολλά έργα ανοιχτού κώδικα.

<sup>10</sup> Το λειτουργικό σύστημα Meego δημιουργήθηκε για να λειτουργήσει ταυτόχρονα σε υπολογιστές, κινητά τηλέφωνα, tablets και σε οχήματα. Το λειτουργικό αυτό δυστυχώς σταμάτησε να αναπτύσσεται. Ωστόσο είναι διαθέσιμο μέχρι την έκδοση 1.2.

Στις μέρες μας η πλατφόρμα που έχει κυριαρχήσει σε υπολογιστές και tablet είναι το Android. Το 2008, τα πρώτα σχέδια για Android tablet έκαναν την εμφάνισή τους. Τα πρώτα μοντέλα κυκλοφόρησαν το 2009, μεταξύ αυτών ήταν η Archos 5 που ήταν ένα μοντέλο τσέπης με 5 ιντσών οθόνη αφής, που κυκλοφόρησε για πρώτη φορά με ένα ιδιόκτητο λειτουργικό σύστημα και αργότερα το 2009 κυκλοφόρησε με το Android 1.4. Η Camangi WebStation κυκλοφόρησε το 2<sup>ο</sup> τρίμηνο του 2009 το πρώτο LTE Android tablet ενώ περισσότερα προϊόντα ακολούθησαν το 2010 περιμένοντας αρκετοί κατασκευαστές το Android Honeycomb, που ήταν ειδικά προσαρμοσμένο για χρήση με tablets, το οποίο έκανε το ντεμπούτο του το Φεβρουάριο του 2011.

Από το 2010 μέχρι σήμερα η Apple κάνει συχνά την εμφάνισή της με νέα κατηγορία συσκευών, τα iPads που είναι από τα πιο πετυχημένα tablets στην αγορά κάνοντας το ντεμπούτο τους το 2010 μπαίνοντας σε σκληρό ανταγωνισμό με της Samsung Galaxy Tab. Το 2013 η Samsung ανακοίνωσε ένα tablet που έχει τη δυνατότητα να τρέχει το Android και το Windows 8 ταυτόχρονα, συγχρονίζοντας τα δεδομένα μεταξύ των δύο αυτών λειτουργικών συστημάτων.

Εν κατακλείδι στα μέσα του 20<sup>ου</sup> αιώνα αναπτύχθηκε στις δύο τελευταίες δεκαετίες μέχρι και το 2010 το πρώτο σύγχρονο tablet, το iPad. Σύμφωνα με το περιοδικό PC Magazine το Μάρτιο του 2012 ανέφερε ότι το 31% των Αμερικανών χρηστών έχουν στην κατοχή τους ένα tablet που το χρησιμοποιούν κυρίως για серφάρισμα στο internet αλλά και για ενημερωτικό σκοπό για ειδήσεις και άλλα. Το iPad είναι ένα από τα υψηλότερα σε πωλήσεις προϊόντα της Apple, με 100 εκατομμύρια πωλήσεις μεταξύ του Απριλίου του 2010 και του Οκτωβρίου του 2012 αλλά το μερίδιο αγοράς στο iPad το 2013 μειώθηκε σε 36% σε σχέση με τα tablets με λειτουργικό Android που οι πωλήσεις τους αυξήθηκαν έως 62%, με 121 εκατομμύρια συσκευές που πωλήθηκαν στην αγορά.

Ένα από τα πιο βασικά στοιχεία μεταξύ των υπολογιστών tablet είναι ο οθόνη αφής και το interface γενικά της συσκευής. Με την οθόνη αφής επιτρέπει στο χρήστη να πλοηγηθεί εύκολα πληκτρολογώντας στο εικονικό πληκτρολόγιο τα δεδομένα που θέλει να εισάγει στην οθόνη. Το σύστημα αυτό πρέπει να ανταποκρίνεται στο άγγιγμα που κάνει ο χρήστης με το δάχτυλό του πάνω στην οθόνη της συσκευής. Οι οθόνες αφής χωρίζονται σε δύο κατηγορίες: α) resistive touchscreens και β) capacitive touchscreens.

A) Οι resistive touchscreens είναι οθόνες αφής που επιτρέπουν ένα υψηλό επίπεδο ακρίβειας κατά το άγγιγμα τους. Αυτό γίνεται γιατί μια τέτοια οθόνη αφής αποτελείται από δύο εύκαμπτα φύλλα επικαλυμμένα με ένα ωμικό υλικό και χωρίζονται από ένα κενό αέρος ή microdots. Υπάρχουν δύο διαφορετικοί τύποι μεταλλικών στρωμάτων, ο πρώτος τύπος ονομάζεται Matrix, στην οποία τα ηλεκτρόδια είναι χωρισμένα όπως για παράδειγμα σε γυαλί ή πλαστικό. Ο δεύτερος τύπος ονομάζεται Αναλογική η οποία αποτελείται από διαφανή ηλεκτρόδια χωρίς σχηματοποίηση. Όταν λοιπόν ο χρήστης πιέζει την οθόνη αφής, τα δύο φύλλα πιέζονται μαζί. Σε αυτά τα δύο φύλλα υπάρχουν οριζόντιες και κάθετες γραμμές που όταν ενωθούν εμφανίζεται η ακριβή τοποθεσία της αφής. Επειδή η οθόνη αφής ανιχνεύει την είσοδο από την επαφή με οποιοδήποτε αντικείμενο (δάχτυλο, γραφίδα / στυλό) είναι ένα είδος «παθητικής» τεχνολογίας.

B) Οι capacitive touchscreens είναι οθόνες αφής λιγότερο ακριβές αλλά πιο ευαίσθητες ως προς τις συσκευές. Αυτές οι οθόνες βασίζονται σε χωριτική σύζευξη και απαιτούν ένα αγωγίμο υλικό όπως για παράδειγμα ένα δάχτυλο για την είσοδο και την περιύγηση της συσκευής. Πολλοί τύποι αισθητήρων χρησιμοποιούν χωρητική ανίχνευση, συμπεριλαμβανομένων των αισθητήρων για την ανίχνευση και τη μέτρηση της

εγγύτητας, τη θέση ή τη μετατόπιση, την υγρασία, την επιταχυνση και άλλα. Οι Digital συσκευές αναπαραγωγής ήχου, κινητά τηλέφωνα και υπολογιστές tablet χρησιμοποιούν χωρητική αντίληψη οθόνες αφής (capacitive touchscreens) ως συσκευές εισόδου αντικαταστάοντας το ποντίκι και το πληκτρολόγιο του υπολογιστή. Τα σημερινά tablets έχουν capacitive οθόνη αφής με multi-touch, σε αντίθεση με προηγούμενες συσκευές που χρησιμοποιούσαν τη γραφίδα σαν είσοδο δεδομένων και περιήγησης του χρήστη στη συσκευή. Από το 2007 και μετά, με την τεχνολογία σε χωρητικές οθόνες όπως αναφέραμε και πιο πάνω και με την επιτυχία του iPhone με την multi-touch οθόνη του οι περισσότερες συσκευές έχουν χωρητική οθόνη αφής.

Τα περισσότερα tablets από τα μέσα του 2010 χρησιμοποιούν επεξεργαστή ARM για μεγαλύτερη διάρκεια ζωής της μπαταρίας. Η «οικογένεια ARM Cortex» είναι για αρκετά ισχυρές εργασίες, όπως η περιήγηση στο διαδίκτυο, η παραγωγή φωτογραφιών σε επαγγελματικό επίπεδο καθώς και οι λειτουργίες παιχνιδιών για κινητά τηλέφωνα. Όπως και με τα smartphones οι περισσότερες εφαρμογές tablet παρέχονται από «προμηθευτές λογισμικού» και κάνοντας ένα κλικ υπάρχει η δυνατότητα αυτόματης ενημέρωσης (update) της εφαρμογής ή εγκατάσταση μιας νέας εφαρμογής. Αυτές τις εφαρμογές μπορούμε να τις βρούμε από τα on line καταστήματα εφαρμογών για παράδειγμα Google Play για Android λογισμικό και το App Store για λογισμικό iOS καθώς υπάρχουν κοινά χαρακτηριστικά ανάμεσα στις εφαρμογές των smartphones και των tablets διότι χρησιμοποιούν ίδιο λειτουργικό σύστημα.

Τα υλικά που αποτελείται ένα tablet είναι:

1. Μια υψηλής ευκρίνειας αντιθαμβωτική οθόνη.
2. Μια ασύρματη σύνδεση στο internet που συνήθως γίνεται με ένα Wi-Fi.
3. Μια κάμερα εμπρός ή και πίσω από τη συσκευή για φωτογραφίες και βίντεο.
4. Μικρό βάρος για πιο εύκολη μεταφορά της συσκευής και μεγαλύτερη διάρκεια ζωής της μπαταρίας από μια συγκριτικά μεγέθους φορητού υπολογιστή.
5. Bluetooth για σύνδεση περιφερειακών συσκευών και την επικοινωνία με τις τοπικές συσκευές.
6. Εξωτερικές θύρες USB για σύνδεση του πληκτρολογίου με τη συσκευή.
7. Σημαντικότερο τέλος είναι πως στα tablets μπορούν να περιέχονται υλικά που του παρέχουν λειτουργικότητα, όπως για παράδειγμα η κάμερα, το GPS και η τοπική αποθήκευση δεδομένων.

Ως προς το λογισμικό τους τα tablets αποτελούνται από:

1. Ένα κινητό πρόγραμμα περιήγησης στο διαδίκτυο
2. Reader για ανάγνωση σε ψηφιακά βιβλία, περιοδικά, σημειώσεις και άλλο περιεχόμενο.
3. Ικανότητα κατεβάσματος και εγκατάστασης εφαρμογών όπως για παράδειγμα παιχνίδια, εφαρμογές εκπαίδευσης ή και υπηρεσιών κοινής ωφέλειας.
4. Ένα φορητό media player στο οποίο γίνεται αναπαραγωγή μουσικής και βίντεο.
5. Σύνδεση και συγχρονισμός στα e-mail του χρήστη καθώς και σύνδεση με ευκολη πρόσβαση στα social media.



6. Λειτουργίες όπως έχει ένα κινητό τηλέφωνο όπως για παράδειγμα συγγραφή μηνυμάτων, αναπαραγωγή ήχου ή φωνής στη διαδικασία του μεγαφώνου και άλλα.
7. Τέλος υπάρχει και η δυνατότητα βίντεο για τηλεδιάσκεψη.

Όλα τα δεδομένα που συλλέγει η συσκευή ή που κατεβάζει ο χρήστης τις διάφορες εφαρμογές αποθηκεύονται στη συσκευή. Αναλυτικά έχουμε:

1. Μια μνήμη ram της συσκευής.
2. Θύρες για συνδεσή με εξωτερικές συσκευές αποθήκευσης δεδομένων.
3. Διάφορες μορφές αποθήκευσης που παρέχουν τα cloud όπως το Google Drive για παράδειγμα στο Android και το iCloud σε λειτουργικό iOS, που παρέχουν υπηρεσίες για δημιουργία αντιγράφων ασφαλείας και τον συγχρονισμό δεδομένων μεταξύ των συσκευών.

Επιπλέον εκτός από το πληκτρολόγιο και την οθόνη αφής ορισμένα tablet μπορούν να χρησιμοποιήσουν τις παρακάτω μεθόδους εισόδου:

1. Η συσκευή μπορεί να συνδεθεί με USB με ένα επιταχυνσιόμετρο. Εκεί ανιχνεύει τη φυσική κίνηση και τον προσανατολισμό της ταμπλέτας. Αυτό επιτρέπει στην οθόνη αφής να περιστρέφεται είτε οριζόντια είτε κάθετα. Επιπλέον, γέροντας το tablet με όποια κλίση θέλουμε μπορεί να χρησιμοποιηθεί ως είσοδος, για παράδειγμα για να κατευθύνουν ένα παιχνίδι οδήγησης.
2. Το φως του περιβάλλοντος και οι αισθητήρες εγγύτητας, εντοπίζουν εάν η συσκευή είναι κοντά σε κάτι, ιδίως στο αυτί μας ή στο μάτι μας προσαρμόζοντας τη συσκευή και τις ρυθμίσεις της έτσι ώστε να είναι λιγότερο βλαβερό για το χρήστη.
3. Η αναγνώριση φωνής.
4. Αναγνώριση χειρονομιών
5. Αναγνώριση χαρακτήρων για να γράψει ένα κείμενο στο tablet, που μπορεί να αποθηκευτεί ως οποιοδήποτε άλλο κείμενο όταν αποθηκεύεται αντί ο χρήστης να χρησιμοποιεί το πληκτρολόγιο.
6. Συμβατό επίπεδο εικοινωνίας με άλλες συμβατές συσκευές, όπως το <sup>11</sup>ISO / IEC 14443.

---

<sup>11</sup> ISO / IEC 14443 αναγνωριστικές κάρτες ή κάρτες ολοκληρωμένων κυκλωμάτων – κάρτες εγγύτητας είναι ένα διεθνές πρότυπο που καθορίζει τις κάρτες εγγύτητας που χρησιμοποιείται για την αναγνώριση, καθώς και τα πρωτόκολλα μετάδοσης για την επικοινωνία με αυτό.

Δύο κύριες Intel ×86 αρχιτεκτονικής κυριαρχούν στην αγορά των tablet. Η αρχιτεκτονική της ARM, της Intel και της AMD x86. Η αρχιτεκτονική της Intel ×86, συμπεριλαμβανομένων και το ×86 και ×64 έχει τροφοδοτήσει τους υπολογιστές της IBM από το 1981 και τους υπολογιστές Macintosh της Apple από το 2006. Οι επεξεργαστές έχουν ενσωματωθεί με τα tablet PCs με την πάροδο των χρόνων και γενικά προσφέρουν μεγαλύτερη απόδοση σε συνδυασμό με την ικανότητα να τρέξει πλήρεις εκδόσεις Microsoft, μαζί με την επιφάνεια εργασίας των Windows και τις διάφορες εφαρμογές τρίτων κατασκευαστών.

Όσα tablet δεν τρέχουν λειτουργικό Windows ×86, περιλαμβάνουν την <sup>12</sup>JooJoo. Η Intel ανακοίνωσε το 2010 τα σχέδιά της να εισέλθει στην αγορά των tablet με τον <sup>13</sup>Atom και τον Οκτώβριο του 2013, η Intel ανακοίνωσε τα σχέδια της για την κατασκευή FPGA-based quad πυρήνες για ARM και ×86 επεξεργαστές.

Η ARM ήταν η επιλογή της αρχιτεκτονικής της CPU για τους κατασκευαστές των smartphones (95% ARM), PDA, ψηφιακές φωτογραφικές μηχανές (80% ARM), set-top boxes, routers DSL, έξυπνες τηλεοράσεις (70% ARM), συσκευές αποθήκευσης και υπολογιστές tablet (95% ARM).

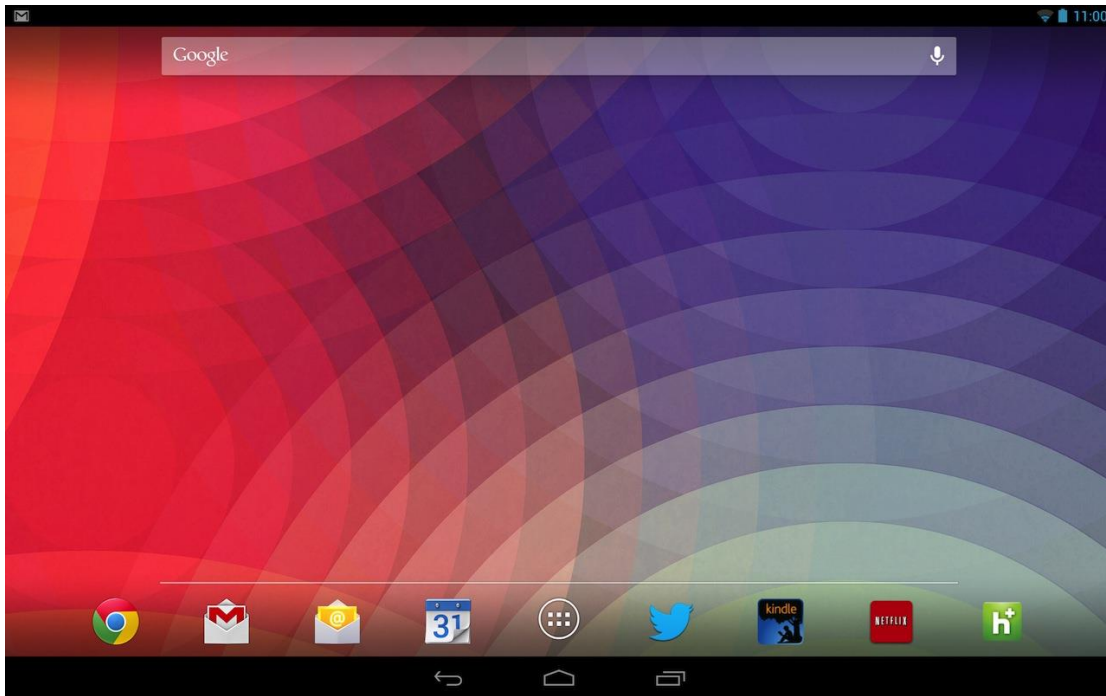
Τα tablet όπως και τα συμβατικά PC, τρέχουν σε πολλαπλά λειτουργικά συστήματα. Αυτά τα λειτουργικά συστήματα έρχονται σε δύο κατηγορίες, με βάση την επιφάνεια εργασίας που βασίζονται σε κινητά τηλέφωνα όπως το OS. Τα tablet OS-based Desktop είναι σήμερα παχύτερα και βαρύτερα και απαιτούν περισσότερο αποθηκευτικό χώρο, περισσότερη ψήξη και δίνουν λιγότερη διάρκεια ζωής της μπαταρίας, αλλά μπορεί να τρέξει εφαρμογές υψηλής έντασης επεξεργαστή όπως για παράδειγμα το Adobe Photoshop. Σε αντίθετη περίπτωση τα tablet τρέχουν μόνο εφαρμογές για κινητά τηλέφωνα και smartphones. Στο τέλος του 1<sup>ου</sup> τριμήνου του 2013, η GlobalWebIndex σημείωσε πως σε δύο χρόνια χρήσης του tablet αυξήθηκε η ζήτησή του κατά 282% με 156 εκατομμύρια χρήστες του Android tablet και 122 εκατομμύρια χρήστες του iPad, που αποτελούν το 75%. Μέχρι το τέλος του έτους 2013, η Gartner διαπίστωσε ότι πωλήθηκαν 121 εκατομμύρια tablet με λειτουργικό Android, 70 εκατομμύρια tablet με λειτουργικό iOS (iPad) και 4 εκατομμύρια tablet με λειτουργικό σύστημα Windows.

---

<sup>12</sup> Η JooJoo είναι ένα λειτουργικό Linux

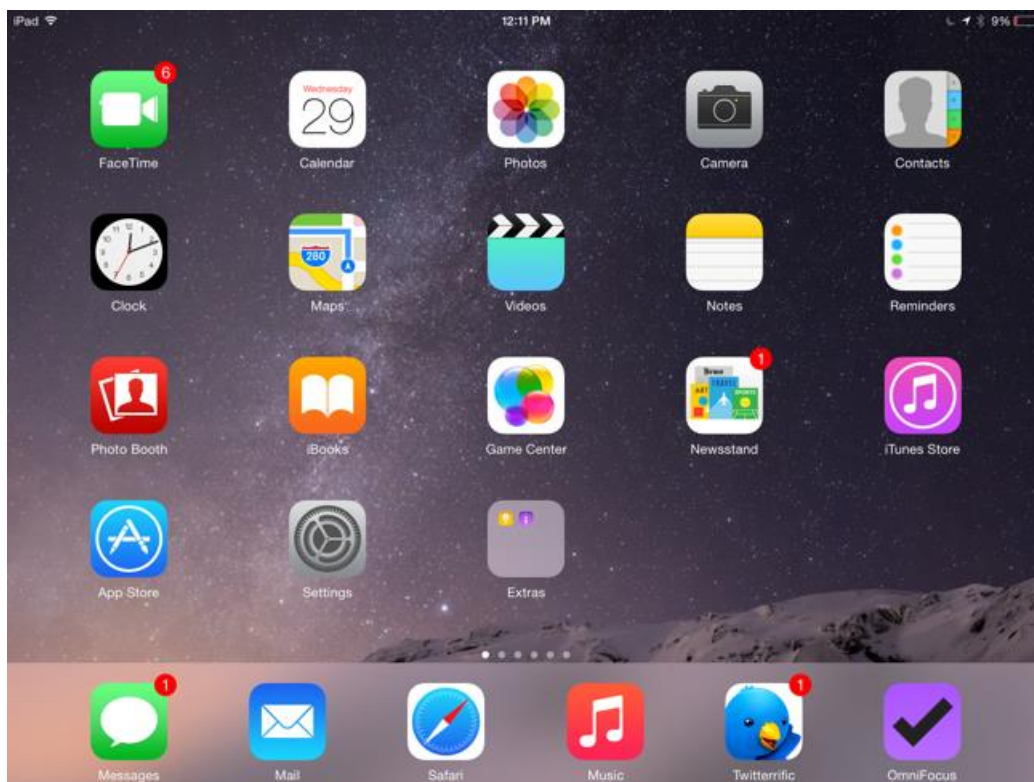
<sup>13</sup> Intel Atom είναι η εμπορική ονομασία για μια σειρά από εξαιρετικά χαμηλής τάσης IA-32 και ×86 -64 επεξεργαστές ή μικροεπεξεργαστές από την Intel που αρχικά σχεδιάστηκε σε 45 nm συμπληρωματικού μετάλλου-οξειδίου-ημιαγωγού (CMOS) με επακόλουθα μοντέλα, με την κωδική ονομασία Cedar, χρησιμοποιώντας ένα 32 nm διαδικασία. Οι επεξεργαστές Atom χρησιμοποιούνται κυρίως σε notebooks, nettops, ενσωματωμένες εφαρμογές που κυμαίνονται από την υγειονομική περίθαλψη στην προηγμένη ρομποτική και κινητές συσκευές Internet.

Πιο αναλυτικά, το Android είναι ένα λειτουργικό σύστημα που βασίζεται στη Linux, που προσφέρει η Google ως ανοικτού κώδικα και έχει σχεδιαστεί κυρίως για φορητές συσκευές όπως smartphones και υπολογιστές tablet. Οι εταιρίες όπως η Motorola και η Lenovo καθυστέρησαν την εγκατάσταση του Android στα tablet τους παρά μόνο μετά το 2011, όταν το Android 3.0 (Honeycomb) και οι νεότερες εκδόσεις υποστηρίζουν μεγαλύτερα μεγέθη οθόνης και έχουν πρόσβαση και στην υπηρεσία της Google Play.



***Εικόνα 13. User Interface σε tablet με λειτουργικό σύστημα Android.***

Το iPad τρέχει λειτουργικό σύστημα iOS, το οποίο δημιουργήθηκε για το iPhone και το iPod Touch. Παρά το γεγονός ότι χτίστηκε στην ίδια εφαρμογή του Unix όπως το MacOS, το user interface είναι τελείως διαφορετικό. Το iPad έχει σχεδιαστεί έτσι ώστε ο χρήστης να έχει κύριο μέσω τα δάχτυλά του για τη χρήση της συσκευής, πράγμα που σημαίνει πως δεν απαιτείται κανένα είδος γραφίδας όπως σε προηγούμενα μοντέλα tablet. Η Apple εισήγαγε τις multi-touch κινήσεις, όπως η μετακίνηση με τα δύο δάχτυλα για να κάνει ο χρήστης μεγέθυνση ή σμίκρυνση αντίστοιχα. Τέλος το iOS είναι χτισμένο με την αρχιτεκτονική ARM.



**Εικόνα 14. User Interface σε iPad με λειτουργικό σύστημα iOS 8**

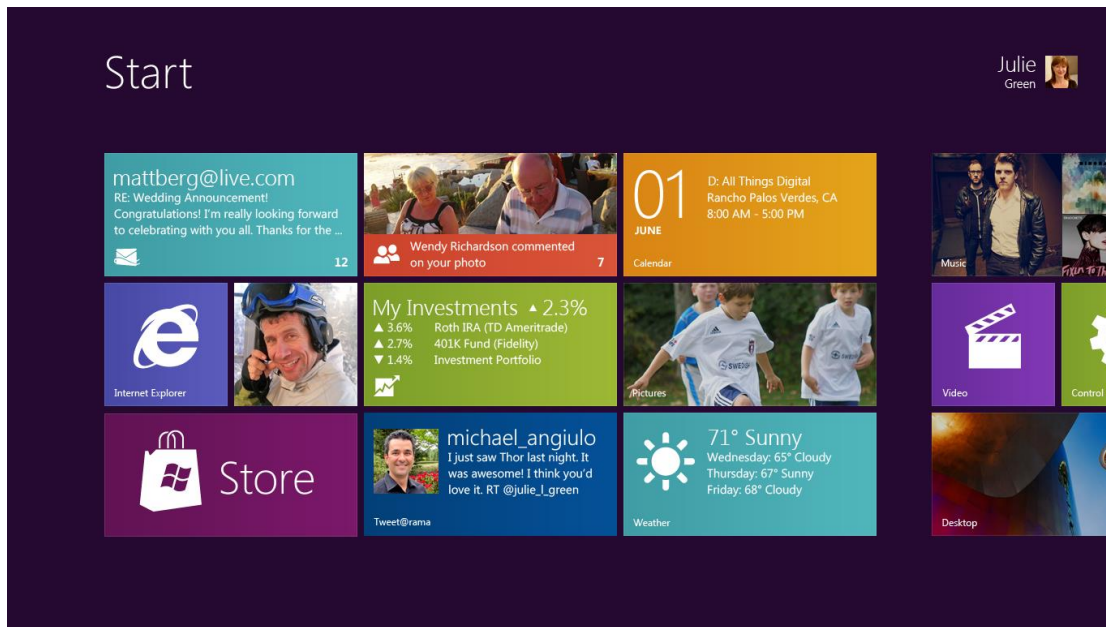
Μετά από τα Windows για <sup>14</sup>Pen Computing για τα Windows 3.1 το 1991, τα tablet της Microsoft έτρεχαν Windows XP. Σύμφωνα με τη Microsoft, το 2001, «η Microsoft Tablet PC» ως κύριο μέσω του χρήστη με τη συσκευή ήταν ένα στυλό-γραφίδα, καθώς είχε και τη λειτουργικότητα της φωνητικής αναγνώρισης. Στο tablet Pc χρησιμοποιήθηκε το ίδιο υλικό από τους φορητούς υπολογιστές αλλά προστέθηκε και η εισαγωγή πένα – γραφίδα. Τα tablet που εκτελούν Windows θα μπορούσε να χρησιμοποιήσει την οθόνη αφής για το ποντίκι και το χέρι για τη γραφή καθώς υποστηρίζουν και κινήσεις πλέον έκτος της χρήσης γραφίδα. Μετά το tablet Pc, η Microsoft ανακοίνωσε την έκδοση Ultra-Mobile PC το 2006 φέρνοντας tablet που τρέχουν Windows με μικρότερο touch-centric μορφή.

Το 2008 η Microsoft παρουσίασε ένα πρότυπο ενός tablet με δύο οθόνες που ονομάστηκε Microsoft Courier αλλά δεν υλοποιήθηκε ποτέ. Επιπλέον ένα μοντέλο της Asus Eee Pad το 2010 έτρεχε λογισμικό Windows CE μπαίνοντας στην αγορά για να αλλάξει τα δεδομένα και στο Android. Τον Οκτώβριο του 2012, η Microsoft κυκλοφόρησε τα Windows 8, το οποίο διαθέτει σημαντικές αλλαγές σε διάφορες πτυχές της διεπαφής του χρήστη. Το λειτουργικό σύστημα αυτό καθιέρωσε επίσης ένα “on line κατάστημα εφαρμογών” windows store, για καλύτερη χρήση του tablet. Έτσι λοιπόν

---

<sup>14</sup> Pen Computing ήταν μια σουίτα λογισμικού για τα Windows 3,1x, χρησιμοποιώντας το λογισμικό αυτό για να ενσωματώσουν την πένα – γραφίδα έτσι ώστε ο χρήστης να περιηγείται στη συσκευή.

μέχρι το πρώτο εξάμηνο του 2014, τα tablet που τρέχουν λειτουργικό Windows έχουν αυξηθεί κατά 33%.



**Εικόνα 15. User Interface σε tablet με λειτουργικό σύστημα Windows 8**

Εν κατακλείδι, η χρήση tablet αυξήθηκε σε μεγάλο βαθμό από το 2010 από τις επιχειρήσεις, καθώς έχουν αρχίσει να τα χρησιμοποιούν σε συνέδρια, εκδηλώσεις και εμπορικές εκθέσεις. Το 2012, η Intel ανέφερε πως τα tablet βελτίωσαν την παραγωγικότητα περίπου σε 19,000 υπαλλήλους κατά μέσο όρο 57 λεπτών την ημέρα. Στις ΗΠΑ και τον Καναδά, εκτιμάται ότι το 60% των online καταναλωτών θα κατέχουν ένα tablet από το 2017 και στην Ευρώπη, το 42% των online καταναλωτών θα κατέχουν από ένα tablet.

Από τις αρχές του 2013, το 29% των ΗΠΑ είναι σε σύνδεση στο διαδίκτυο από το tablet τους. Από τις αρχές του 2014 το ποσοστό που οι Αμερικανοί καταναλωτές είναι σε σύνδεση με το tablet τους έχει αυξηθεί κατά 44% και μέσα σε αυτό το ποσοστό εντάσσονται και πολλά παιδιά. Μια έρευνα του 2014 διαπίστωσε πως τα tablet είναι πιο χρήσιμα ως προς τα παιχνίδια. Σε παιδιά κάτω των 12 ετών χρησιμοποιούνται περισσότερο τα tablet από ότι χρησιμοποιούνται τώρα τα επιτραπέζια παιχνίδια, τα παζλ, τα αυτοκινητάκια, οι κούκλες και τα μπλοκ ζωγραφικής. Παρ'όλα αυτά η πλειοψηφία των γονέων ισχυρίζεται πως το tablet είναι μόνο για μερικές φορές ένα παιχνίδι. Από το 2014, σχεδόν τα δύο τρίτα της Αμερικής από 2 έως και 10 ετών έχουν πρόσβαση σε ένα tablet. Η μεγάλη χρήση των tablet από τους ενήλικες είναι σαν μια προσωπική δυνατότητα σύνδεσης στο Internet TV.



*Εικόνα 16. iPad mini*



*Εικόνα 17. Samsung Galaxy Tab 3*

### 3. Τεχνολογία εντοπισμού θέσης

Οι τεχνολογίες εντοπισμού θέσης είναι νέες για το ευρύ κοινό, αλλά και για τις ίδιες τις εταιρείες, παρ' όλα αυτά είναι πάρα πολύ αξιόπιστες και αναπτύχθηκαν σε πολύ σύντομο χρονικό διάστημα. Τα μέσα επικοινωνίας εντοπισμού θέσεις χρησιμοποιούν τεχνολογίες όπως το Παγκόσμιο Σύστημα Θεσιδεσίας <sup>15</sup>(GPS), φορητούς υπολογιστές, το κινητό τηλέφωνο, τα Συστήματα Γεωγραφικών Πληροφοριών (GIS), την εφαρμογή Google Maps και άλλα. Το GPS επιτρέπει τον ακριβή γεωγραφικό προσδιορισμό μιας τοποθεσίας και οι φορητοί υπολογιστές επιτρέπουν τη σύνδεση των διαδραστικών μέσων με την τοποθεσία αυτή. Το Σύστημα Γεωγραφικών Πληροφοριών παρέχει πιο αυθαίρετη πληροφορία σχετικά με τη γεωλογική κατάσταση μια τοποθεσίας. Οι χάρτες της Google προσφέρουν την οπτική αναπαράσταση ενός συγκεκριμένου τόπου.

Άλλη μια τεχνολογία εντοπισμού θέσεις είναι η τεχνολογία GSM (Global System for Mobile Communications). Η εύρεση της θέσης ενός κινητού τηλεφώνου ανάλογα με τη θέση της κυψέλης που βρίσκεται είναι ένας άλλος τρόπος για τον εντοπισμό ενός αντικειμένου ή ενός προσώπου. Αυτή η τεχνολογία χρησιμοποιείται ως εξής. Το κινητό τηλέφωνο κατά τη διάρκεια της λειτουργίας του επικοινωνεί με την πιο κοντινή κεραία εκπομπής σε πολύ τακτά χρονικά διαστήματα. Η διαδικασία αυτή ονομάζεται GSM Localization και στηρίζεται στην τεχνική multilateration. Η τεχνική αυτή βασίζεται στην ακρίβεια του υπολογισμού Time Difference Of Arrival (TDOA) ενός σήματος που εκπέμπεται από το κινητό προς τρεις κεραίες εκπομπής του παρόχου. Με το συνδυασμό των χρόνων αυτών μπορεί να εντοπιστεί η θέση του κινητού τηλεφώνου καθώς και του χρήστη με πολύ μικρή πιθανότητα λάθους. Στη συνέχεια περιγράφονται αναλυτικά οι βασικοί τρόποι εντοπισμού θέσης που χρησιμοποιούνται από την αγορά, καθώς θα δούμε και άλλες τεχνολογίες εντοπισμού θέσης.

---

<sup>15</sup> Το GPS (Global Positioning System) είναι το παγκόσμιο σύστημα εντοπισμού γεωγραφικής θέσης ακινήτου ή κινητού χρήστη, το οποίο βασίζεται σε ένα «πλέγμα» εικοσιτεσσάρων δορυφόρων της Γης εφοδιασμένων με ειδικές συσκευές εντοπισμού που ονομάζονται «πομποδέκτες GPS».

### 3.1 Εντοπισμός θέσης μέσω GPS

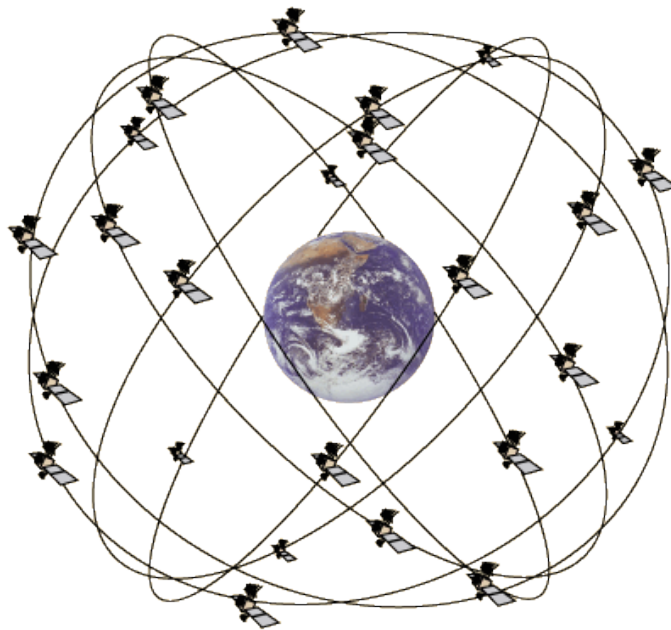
Το GPS (Global Positioning System), Παγκόσμιο Σύστημα Στιγματοθέτησης, ή Θεσιθεσίας είναι ένα παγκόσμιο σύστημα εντοπισμού γεωγραφικής θέσης, (στίγματος), ακινήτου ή κινητού χρήστη, το οποίο βασίζεται σε ένα “πλέγμα” εικοσιτεσσάρων δορυφόρων της Γης, που είναι εφοδιασμένοι με ειδικές συσκευές εντοπισμού, οι οποίοι ονομάζονται “πομποδέκτες GPS”. Οι πομποδέκτες αυτοί παρέχουν ακριβείς πληροφορίες για τη θέση ενός σημείου, το υψόμετρό του, την ταχύτητά του και την κατεύθυνση της κίνησής του. Επίσης, σε συνδυασμό με λογισμικό χαρτογράφησης μπορούν να απεικονίσουν γραφικά τις πληροφορίες αυτές. Το σύστημα αυτό ξεκίνησε από το Υπουργείο Άμυνας των ΗΠΑ και ονομάστηκε NAVSTAR GPS (Navigation Signal Timing and Global Positioning System). Το δορυφορικό αυτό σύστημα ρυθμίζεται καθημερινά από τη Βάση της Πολεμικής Αεροπορίας Schriever με κόστος 400.000.000 \$ το χρόνο.

Το σύστημα εντοπισμού θέσης GPS σχηματίζει ένα παγκόσμιο δίκτυο, με εμβέλεια που καλύπτει ξηρά, θάλασσα και αέρα. Εξαιτίας αυτής της έκτασης του, είναι απαραίτητο η διαχώρισή του σε επιμέρους τμήματα. Τα τμήματα αυτά είναι:

1. Διαστημικό τμήμα: Αποτελείται από το δίκτυο 24-32 δορυφόρων. Οι δορυφόροι αυτοί καλύπτουν με το σήμα τους ολόκληρο τον πλανήτη, αποδεικνύοντας έτσι τη φιλοσοφία και τη λειτουργία του GPS, δηλαδή τη διαθεσιμότητα του σε κάθε σημείο της Γης. Όλοι οι δορυφόροι βρίσκονται σε ύψος 20.200 χιλιομέτρων πάνω από την επιφάνεια της θάλασσας, εκτελώντας δύο περιστροφές γύρω από τη Γη κάθε 24 ώρες.
2. Επίγειο τμήμα ελέγχου: Οι δορυφόροι όπως είναι αναμενόμενο, είναι πιθανό να αντιμετωπίσουν ανα πάσα ώρα και στιγμή κάποιο πρόβλημα στη σωστή λειτουργία τους. Οι έλεγχοι που πραγματοποιούνται σε αυτούς είναι για τη σωστή τους ταχύτητα, το σωστό τους υψόμετρο και την επάρκεια σε ηλεκτρική ενέργεια. Παράλληλα, εφαρμόζονται όλες οι διορθωτικές κινήσεις που αφορούν στο σύστημα χρονομέτρησης των δορυφόρων, ώστε να αποτρέπονται οι λανθασμένες πληροφορίες στους χρήστες του συστήματος. Το τμήμα επίγειου ελέγχου αποτελείται από ένα επανδρωμένο κέντρο και τέσσερα μη επανδρωμένα κέντρα, εγκατεστημένα σε ισάριθμες περιοχές του πλανήτη. Οι περιοχές αυτές είναι οι εξής: α)Κολοράντο (ΗΠΑ) β) Χαβάη (Ανατολικός Ειρηνικός Ωκεανός γ)Ascension Island (Ατλαντικός Ωκεανός) δ) Diego Garcia (Ινδικός Ωκεανός) ε)Kwajalein (Δυτικός Ειρηνικός Ωκεανός). Ο κυριότερος σταθμός βάσης είναι του Κολοράντο, ο οποίος είναι μάλιστα και ο μοναδικός που βρίσκεται στην ξηρά, αναλαμβάνοντας τον έλεγχο της σωστής λειτουργίας των τεσσάρων σταθμών, καθώς και τον συντονισμό τους. Παρατηρώντας τις θέσεις των σταθμών πάνω σε ένα παγκόσμιο χάρτη, παρατηρεί κανείς ότι η διάταξη τους ακολουθούν μια γραμμή παράλληλη με τα γεωγραφικά μήκη της Γης.



3. Το τμήμα τελικού χρήστη: Απαρτίζεται από τους χιλιάδες χρήστες δεκτών GPS. Οι δέκτες αυτοί μπορούν να χρησιμοποιηθούν κατά τη διάρκεια μιας πεζοπορίας, σε οχήματα ή σε θαλάσσια σκάφη. Για να προσφέρουν όσο το δυνατό περισσότερες πληροφορίες, οι δέκτες συνδυάζονται με ειδικό λογισμικό που προβάλλει ένα χάρτη στην οθόνη της συσκευής GPS. Είναι ένα λογισμικό που λαμβάνει από τους δορυφόρους τις πληροφορίες για τα στίγματα των σημείων που βρίσκονται οι δέκτες και τις μετατρέπει σε κατανοητή μορφή για τον άνθρωπο, πληροφορώντας το χρήστη για την ακριβή γεωγραφική του θέση.



*Εικόνα 18: Αναπαράσταση σχεδίου του συστήματος GPS*

Η εξάπλωση του GPS οφείλεται στις προσιτές τιμές των φορητών συσκευών GPS, για πεζούς ή οχήματα και των γενικών υπολογιστικών συσκευών όπως τα PDA με ενσωματωμένο δέκτη GPS. Ένας φορητός δέκτης αποτελείται από:

1. Την εσωτερική δορυφορική κεραία, η οποία λαμβάνει το σήμα από τους δορυφόρους. Επιπλέον, λαμβάνει σήμα και από ανακλάσεις όπως για παράδειγμα σε τοίχους κάνοντας δυνατή τη λήψη σε δρόμους που περιβάλλονται από ψηλά κτήρια. Υπάρχουν και αρκετοί δέκτες που έχουν υποδοχή και για εξωτερική κεραία. Οι εξωτερικές δορυφορικές κεραίες δίνουν καλύτερη λήψη, λόγω της

δυνατότητας τοποθέτησης πάνω από πιθανά εμπόδια, για παράδειγμα στην οροφή ενός αυτοκινήτου.

2. Τον κυρίως δέκτη GPS ο οποίος χρησιμοποιεί κυκλώματα με ειδικές τεχνικές επεξεργασίας σήματος ώστε να ξεχωρίζει τα ασθενή σήματα από τους δορυφόρους. Ο κύριος δέκτης αποτελείται από το αναλογικό τμήμα εισόδου και το ψηφιακό, το οποίο περιέχει σύνθετο ψηφιακό υλικό που συνήθως είναι κάποιο εξειδικευμένο ολοκληρωμένο κύκλωμα τύπου ASIC. Αυτό το hardware χρησιμοποιεί πολύ εξελιγμένους αλγόριθμους επεξεργασίας, για να εξάγει χρήσιμο στίγμα σε συνθήκες δύσκολης λήψης.
3. Τον κυρίως μικροελεγκτή, την οθόνη απεικόνισης και το υπόλοιπο hardware επικοινωνίας με το χρήστη της συσκευής. Ο μικροελεγκτής επεξεργάζεται το στίγμα που λαμβάνει από τον κυρίως δέκτη GPS. Το αποτέλεσμα της επεξεργασίας αυτής είναι πιο κατανοητή μορφή στίγματος για τον άνθρωπο, εμφανίζοντας σε οθόνη με δυνατότητες γραφικών πάνω σε ψηφιακό χάρτη, μαζί με άλλες πληροφορίες όπως την ταχύτητα κίνησης, την ώρα και το υψόμετρο.

Οι μικροελεγκτές έχουν κάνει δυνατή την ύπαρξη διάφορων απλών και εξελιγμένων βοηθημάτων εύρεσης θέσης και πλοήγησης. Για παράδειγμα μπορούμε να βλέπουμε τη διαδρομή που έχουμε ήδη κάνει, να μεγθύνουμε το χάρτη ή να εισάγουμε προορισμό και ο δέκτης να βρίσκει πάντα τη βέλτιστη διαδρομή. Σχεδόν πάντα υπάρχει η δυνατότητα ορισμού στο χάρτη με σημεία ενδιαφέροντος, όπως καταστήματα, αξιοθέατα και άλλα. Στις πιο σύγχρονες συσκευές GPS υπάρχει και η δυνατότητα φωνητικών οδηγιών κατά τη διαδικασία της πλοήγησης. Μελλοντικά οι φορητοί δέκτες GPS θα βρίσκουν μεγάλη εφαρμογή και σε άτομα με αναπηρία όπως οι τυφλοί που θα έχουν τη δυνατότητα να ακούν τις φωνητικές οδηγίες από το GPS.



**Εικόνα 19: Συσκευή GPS**

## 3.2 Εντοπισμός θέσης μέσω GSM

Global System for Mobile communications (Παγκόσμιο Σύστημα Κινητών Επικοινωνιών) ή αλλιώς GSM είναι ένα Ευρωπαϊκό ψηφιακό σύστημα κινητής τηλεφωνίας, που το Ευρωπαϊκό Τηλεπικοινωνιακό Σύστημα το 1982 άρχισε την μελέτη για την δημιουργία ενός κοινού Ευρωπαϊκού ψηφιακού συστήματος κινητής τηλεφωνίας δεύτερης γενιάς (2G). Το GSM είναι ένα κυψελοειδές ψηφιακό σύστημα κινητής τηλεφωνίας δεύτερης γενιάς το οποίο χρησιμοποιεί ηλεκτρομαγνητικά σήματα και την τεχνική πολλαπλής πρόσβασης με διαχωρισμό του διαθέσιμου φάσματος συχνοτήτων σε ένα αριθμό καναλιών και τη διαίρεση αυτών σε χρονοθυρίδες για την μετάδοση σημάτων.

Η εμβέλεια ενός δικτύου GSM σε μια γεωγραφική περιοχή για να γίνει διαμελίζεται σε μικρότερες περιοχές που λέγονται κυψέλες, οι οποίες εφάπτονται μεταξύ τους με κάθε κυψέλη να έχει και ένα σταθμό βάσης, συνθέτοντας έτσι μια δομή κυψελών, επαναλαμβάνοντας τη μέθοδο αυτή όσες φορές χρειάζεται για την κάλυψη της μιας περιοχής. Έτσι αυξάνεται η χωρητικότητα του δικτύου αλλά πρέπει η ισχύς κάθε κυψέλης να μην υπερχειλίζει άλλες κυψέλες της ίδιας δομής, ενώ για να μη δημιουργείται ενδοκαναλική παρεμβολή σε γειτονικές κυψέλες η επαναχρησιμοποίηση συχνοτήτων πρέπει να σχεδιάζεται έτσι ώστε να απέχουν επαρκή απόσταση οι κυψέλες μιας δομής που έχουν την ίδια συχνότητα με τις κυψέλες άλλης δομής. Η ενδοκαναλική παρεμβολή μειώνεται όσο αυξάνεται ο αριθμός των κυψελών της δομής.

Ένα GSM δίκτυο λοιπόν χωρίζεται σε τρία βασικά μέρη:

1. Τον Κινητό Σταθμό (Mobile Station): Έχει έναν πομπό-δέκτη, κεραία, οθόνη και την κάρτα SIM. Η μέγιστη ισχύς εκπομπής στην Ευρώπη μιας κινητής μονάδας είναι στα 2 Watt.
2. Το Βασικό Υποσύστημα Σταθμού (Base Station Subsystem): Το BSS διαχειρίζεται τις κλήσεις σε μια γεωγραφική περιοχή όπου καλύπτεται από ένα σύνολο κεραιών διαφόρων μεγεθών σε σειρά και κάθε τέτοια κεραία εξυπηρετεί και από μια κυψέλη. Το BSS χωρίζεται στο βασικό σταθμό πομπό-δέκτη Base Transceiver Station (BTS) και στο βασικό σταθμό ελέγχου Base Station Controller (BSC).
  - Το Βασικό Υποσύστημα Σταθμού (BTS) φροντίζει την επικοινωνία μεταξύ του δικτύου GSM και του κινητού σταθμού. Ένα BTS μπορεί να ελέγχει μια ή περισσότερες κεραίες. Όταν ένας χρήστης X θέλει να πραγματοποιήσει μια κλήση σε έναν άλλο συνδρομητή Ψ, ο σταθμός βάσης μεταβιβάζει το σήμα με το αίτημα του X για αναζήτηση και εντοπισμό του άλλου συνδρομητή Ψ στο τηλεπικοινωνιακό κέντρο της εταιρείας του X. Το κέντρο της εταιρείας εντοπίζει την κυψέλη στην οποία βρίσκεται ο Ψ και στέλνει το σήμα στον πλησιέστερο σταθμό βάσης. Από εκεί, με τη χρήση των διαθέσιμων συχνοτήτων, στέλνει το σήμα στο κινητό Ψ και έτσι μπορεί να επικοινωνήσει μαζί του ο X.
  - Το BSC (Base Station Controller) ή Βασικός Σταθμός Ελέγχου, ελέγχει τα σήματα παίρνοντας τα από ένα ή περισσότερα BTS, ενώ εκχωρεί και απελευθερώνει κανάλια. Τα σήματα που λαμβάνει τα κατευθύνει στο Mobile Switching Centre και όταν χρειάζεται μετατρέπει τα 16 kbps

φωνής που είναι στην κινητή τηλεφωνία σε 64kbps που χρησιμοποιείται στη σταθερή τηλεφωνία.

3. Το Υποσύστημα Δικτύου μεταγωγής (NNS – Network Switching Subsystem) αποτελείται από το Κέντρο Διανομής Mobile Switching Center (MSC) που είναι υπεύθυνο για την διασύνδεση, τον έλεγχο και την δρομολόγηση εισερχόμενων και εξερχόμενων κλήσεων μεταξύ του δικτύου κινητής τηλεφωνίας και ενός άλλου δικτύου. Όταν ένα MSC συνδέεται με ένα δίκτυο σταθερής τηλεφωνίας θα πρέπει να δέχεται 64kbps φωνής, όταν όμως ο MSC συνδέεται με ένα δίκτυο κινητής τηλεφωνίας τότε θα πρέπει να γνωρίζει που βρίσκεται τη συγκεκριμένη στιγμή ο χρήστης.

Η εναλλαγή μιας κλήσης σε διαφορετική κυψέλη που βρίσκεται σε εξέλιξη ονομάζεται handover και αυτό γίνεται επειδή η κινητή μονάδα βρίσκεται εν κινήση. Υπάρχουν τέσσερις διαφορετικοί τύποι handover στο GSM που αφορούν, σε κανάλια που είναι στην ίδια κυψέλη, σε κυψέλες που βρίσκονται υπό τον έλεγχο του ίδιου Βασικού σταθμού ελέγχου (BSC), σε κυψέλες που βρίσκονται στον έλεγχο διαφορετικών σταθμών ελέγχου αλλά και στον ίδιο κινητό σταθμό μεταγωγής δικτύου (MSC) και σε κυψέλες σε διαφορετικούς σταθμούς μεταγωγής δικτύων (MSC). Οι δύο πρώτοι τύποι λέγονται εσωτερικά handovers και χρησιμοποιούν τον ίδιο Βασικό Σταθμό Ελέγχου (BSC), το MSC ενημερώνεται μόνο όταν ολοκληρωθεί το handover. Οι άλλοι δύο τύποι handovers λέγονται εξωτερικά handovers τα MSCs. Επιπλέον τα handovers, μπορούν να ενεργοποιηθούν από το ίδιο το κινητό ή το MSC σαν μια λύση για την αυξημένη κίνηση σε μια κυψέλη, την ώρα που δεν απασχολείται, το κινητό ελέγχει τα κανάλια επικοινωνίας με 16 γειτονικές κυψέλες και δημιουργεί μια λίστα με τις 6 πιο πιθανές κυψέλες για handover που χρησιμοποιούνται για τον αλγόριθμο του handover. Ο αλγόριθμος μικρότερης απόδοσης επιτρέπει την αλλαγή της ισχύς στο handover έτσι ώστε όταν το σήμα χάσει την απόδοσή του μέχρι ενός ορίου, η ισχύς του κινητού να αυξηθεί. Σε περίπτωση που η ισχύς είναι αυξημένη το σήμα δε βελτιώνεται αλλά δημιουργείται ένα νέο handover.

Για να μπορέσει ένας χρήστης να χρησιμοποιήσει το δίκτυο αυτό, θα πρέπει πρώτα να πιστοποιηθεί. Για να γίνει αυτό, κάθε κινητό τηλέφωνο διαθέτει ένα μυστικό κλειδί το οποίο βρίσκεται στην κάρτα SIM και στο Κέντρο Πιστοποίησης (AC). Όταν το κινητό ενεργοποιηθεί το Κέντρο Πιστοποίησης στέλνει ένα τυχαίο αριθμό στο κινητό και με το μυστικό κλειδί και με έναν αλγόριθμο κρυπτογράφησης, δημιουργείται ένας νέος αριθμός. Το κινητό τότε στέλνει πίσω στο κέντρο πιστοποίησης τον αριθμό αυτό και με τη σειρά του ελέγχει εάν είναι ο ίδιος αριθμός με αυτόν που φτιάχτηκε. Εάν ο αριθμός είναι ίδιος τότε ο χρήστης πιστοποιήθηκε διαφορετικά εάν δεν είναι ίδιος τότε ειδοποιείτε ο χρήστης πως η διαδικασία εγγραφής ήταν ανεπιτυχής.

Κάθε κινητό τηλέφωνο έχει τη δική του ταυτότητα. Η ταυτότητα αυτή είναι ένας μοναδικός 16ψήφιος αριθμός για κάθε συσκευή που αντιστοιχεί στη μάρκα του κινητού, στον αριθμό σειράς της συσκευής, στοιχεία κατόχου, ημερομηνία αγοράς συσκευής και άλλα. Ένα δίκτυο GSM αποθηκεύει σε 3 διαφορετικές λίστες τις “ταυτότητες” των συσκευών των συνδρομητών της. Η πρώτη λίστα είναι η λευκή λίστα που υπάρχουν όλες τις “ταυτότητες” των κινητών που λειτουργούν φυσιολογικά και έχουν συνδεθεί στο δίκτυο με ασφάλεια. Η δεύτερη λίστα είναι η γκρι λίστα που υπάρχουν οι “ταυτότητες” των συσκευών που είναι υπό παρακολούθηση λόγω πιθανόν προβλημάτων. Και η τρίτη λίστα είναι η μαύρη λίστα που υπάρχουν οι “ταυτότητες” των κινητών που έχουν

δηλωθεί από τους κατόχους τους σαν κλεμμένα και στην περίπτωση αυτή διενεργείται παρακολούθηση των συσκευών αυτών.



*Εικόνα 20: GSM (Global System for Mobile communications)*

### 3.3 Διάφορες τεχνολογίες εντοπισμού θέσης

Μια άλλη τεχνολογία εντοπισμού θέσης είναι η LBS (Location Based Services) που χρησιμοποιούν την τοποθεσία των δεδομένων για να ελέγχουν χαρακτηριστικά. Η LBS είναι μια υπηρεσία πληροφοριών που έχει πολλές χρήσεις σήμερα στην κοινωνική δικτύωση ως ψυχαγωγία, που είναι προσβάσιμη από φορητές συσκευές μέσω δικτύου κινητής τηλεφωνίας και η οποία χρησιμοποιεί πληροφορίες σχετικά με τη γεωγραφική θέση της κινητής συσκευής.

Επιπλέον στην τεχνολογία LBS περιλαμβάνονται υπηρεσίες προσδιορισμού θέσης ενός προσώπου ή ενός αντικειμένου, για παράδειγμα το πλησιέστερο μηχάνημα ATM ή την παρακολούθηση και τον εντοπισμό ενός οχήματος. Επιπροσθέτως άλλες τεχνολογίες όπως το WLAN, το Bluetooth και οι υπέρυθρες που χρησιμοποιούνται για να συνδεθούν μεταξύ τους οι συσκευές σε τοπικές υπηρεσίες, παρέχουν και αυτές με τη σειρά τους στο χρήστη την τοποθεσία που βρίσκεται καθώς και άλλες υπηρεσίες σε σχέση με την περιοχή. Η τεχνολογία του Bluetooth η οποία είναι σχετικά καινούργια περιλαμβάνεται σε όλες τις συσκευές κινητής τηλεφωνίας και όχι μόνο, το οποίο διαθέτει ένα μοναδικό

αναγνωριστικό το οποίο μπορεί να αναγνωρίσει μια άλλη συσκευή και να συνδεθούν μεταξύ τους.

Η τεχνολογία του Bluetooth δημιουργήθηκε κατά κύριο λόγο για να υπάρχει “επικοινωνία” μεταξύ δύο συσκευών για ανταλλαγή αρχείων. Ωστόσο υπάρχουν δύο τρόποι αναζήτησης τοποθεσίας με την τεχνολογία Bluetooth. 1) Binary location, είναι μια προσέγγιση κατά την οποία σε κάθε δωμάτιο ενός κτιρίου υπάρχει μια εγκατάσταση ενός συστήματος Access Point (AP) με το οποίο το κινητό τηλέφωνο μπορεί να συνδεθεί και έτσι να εντοπισθεί ο χώρος στον οποίο βρίσκεται. 2) Analog location, είναι μια προσέγγιση κατά την οποία τα συστήματα Access Point (AP) είναι εγκατεστημένα σε συγκεκριμένα σημεία του κτιρίου πιο αραιά σε σχέση με τον πρώτο τρόπο και η απόσταση από κάθε AP μετράται από το σύστημα. Με βάση τη διαδικασία τριγωνισμού και τις μετρήσεις που υπολογίστηκαν από το σύστημα τότε εντοπίζεται κατά προσέγγιση η θέση του χρήστη.

Σύμφωνα με τα παραπάνω αυτό λειτουργεί μόνο σε τοπικούς χώρους και όχι σε μεγάλες αποστάσεις καθώς πρέπει να υπάρχει μεγάλος συντονισμός όλως των Access Point.

### 4. Λογισμικό Android

Το λογισμικό Android είναι ένα λειτουργικό σύστημα με βάση τον πυρήνα Linux που αναπτύχθηκε από την Google. Με τη διεπαφή τού χρήστη που βασίζεται στη άμεση επικοινωνία του χρήστη με τη συσκευή με την κίνηση του χεριού του, το Android έχει σχεδιαστεί κυρίως για την οθόνη αφής σε φορητές συσκευές όπως, smartphones, tablets, με εξειδικευμένες διεπαφές χρήστη για τηλεοράσεις όπως Android TV, ρολόγια χειρός τα επωνομαζόμενα Android Wear. Το λειτουργικό αυτό σύστημα χρησιμοποιεί εισροές αφής που αντιστοιχούν σε πραγματικές δράσεις ανάμεσα στο χρήστη και στη συσκευή, όπως για παράδειγμα το σύρσιμο και το άγγιγμα του χρήστη στην οθόνη αφής για να κάνει η συσκευή μια συγκεκριμένη λειτουργία. Η οθόνη αφής έχει επίσης χρησιμοποιηθεί και σε κονσόλες παιχνιδιών, σε ψηφιακές φωτογραφικές μηχανές, σε υπολογιστές, περιφεριακά και άλλα. Από το 2015, το Android έχει τα περισσότερα εγκατεστημένα συστήματα από όλα τα λειτουργικά συστήματα.

Τον Ιούλιο του 2013 το κατάστημα Google Play είχε πάνω από ένα εκατομμύριο εφαρμογές Android και πάνω από 50 δισεκατομμύρια εφαρμογές έχουν ληφθεί από τους χρήστες. Μέσα από μια έρευνα που έγινε τον Απρίλιο έως το Μάιο του 2013 σχετικά με τα κινητά τηλέφωνα και το λογισμικό Android διαπιστώθηκε πως το 71% των προγραμματιστών εφαρμογών, δημιούργησαν εφαρμογές για την πλατφόρμα Android. Από μια άλλη έρευνα του 2015 διαπιστώθηκε πως το 40% των επαγγελματιών προγραμματιστών δίνουν προτεραιότητα στην κατασκευή εφαρμογών που βασίζεται στην πλατφόρμα, Android μιας και οι περισσότεροι χρήστες smartphones, έχουν συσκευή που τρέχει το λογισμικό Android.

Ο πηγαίος κώδικας του Android δίνεται από την Google αν και οι περισσότερες συσκευές που τρέχουν το λογισμικό Android κυκλοφορούν με ένα συνδυασμό ανοικτού κώδικα και κλειστού λογισμικού που συμπεριλαμβάνουν ένα ιδιόκτητο λογισμικό που αναπτύχθηκε πρόσθετα από την Google και από το αρχικό Android.

Το Android, ιδρύθηκε στο Πάλο Άλτο της Καλιφόρνιας τον Οκτώβριο του 2003 από τον Andy Rubin, Rich Miner, Nick Sears και Chris White για την ανάπτυξη, έξυπνων φορητών συσκευών που είναι περισσότερο ενήμεροι για την τοποθεσία και τις προτιμήσεις του ιδιοκτήτη του. Το πρώτο πράγμα που ήθελε να ανατύξει η εταιρεία ήταν ένα προηγμένο λειτουργικό σύστημα για ψηφιακές φωτογραφικές μηχανές. Όταν διαπιστώθηκε πως η αγορά δεν ήταν μεγάλη για μια τέτοια ενέργεια τότε η εταιρεία προσπάθησε να μπει στην παραγωγή ενός λειτουργικού συστήματος smartphone που θα ανταγωνιζόταν το Symbian και το Microsoft Windows Mobile.

Τον Ιούλιο του 2005 η Google απέκτησε το Android για 50 εκατομμύρια δολάρια. Με την κίνηση αυτή η Google σκόπευε να μπει στο χώρο της κινητής τηλεφωνίας. Στις 5 Νοεμβρίου του 2007, η Open Handset Alliance που είναι μια κοινοπραξία των εταιρειών τεχνολογίας, συμπεριλαμβανομένων και της Google και οι κατασκευαστές συσκευών, όπως η HTC, η Sony και η Samsung και ασύρματη φορείς όπως η Sprint Nextel και η T-Mobile και κατασκευαστές chipset όπως η Qualcomm και Texas

Instruments αποκάλυψαν πως στόχος είναι η ανάπτυξη ανοιχτών προτύπων για τις κινητές συσκευές. Η πρώτη συσκευή που διατίθεται στο εμπόριο και τρέχει το λογισμικό Android ήταν το HTC Dream που κυκλοφόρησε στις 22 Οκτωβρίου του 2008. Το 2010, η Google εγκαινίασε το Nexus μια σειρά από συσκευές smartphones και tablets που τρέχει το λειτουργικό σύστημα Android. Η HTC συνεργάστηκε με την Google για να βγάλει στην αγορά το πρώτο smartphone Nexus One. Τα Nexus λειτουργούν σαν ναυαρχίδα του Android αποδεικνύοντας έτσι το πιο πρόσφατο λογισμικό και hardware χαρακτηριστικά του Android.

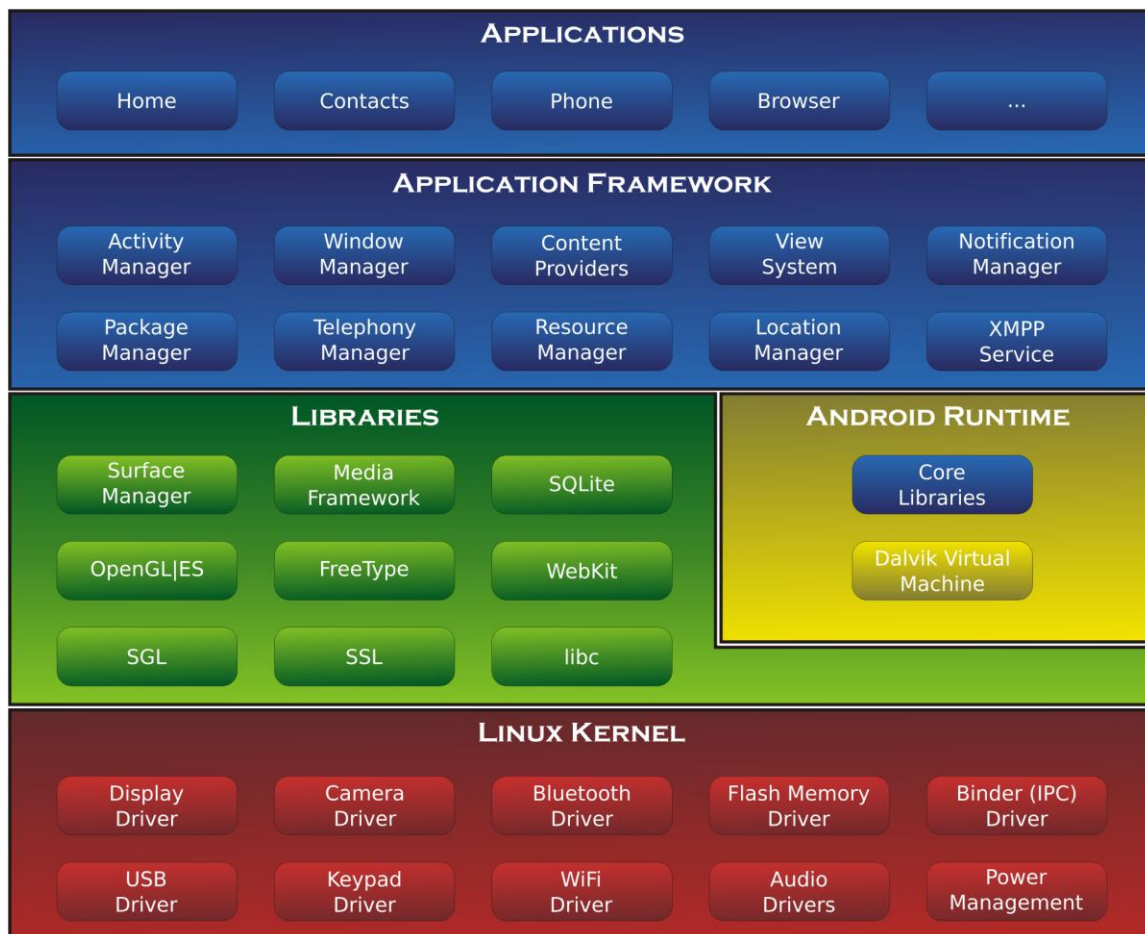
Από το 2008, το Android έχει δει πολλές ενημερώσεις που βελτιώνουν σταδιακά το λειτουργικό σύστημα, προσθέτοντας νέα χαρακτηριστικά και διορθώσεις σφαλμάτων σε προηγούμενες εκδόσεις. Κάθε σηματική έκδοση έχει το όνομα ενός γλυκού, με αλφαβητική σειρά, όπως για παράδειγμα η έκδοση 1.5 “Cupcake” ακολούθησε η 1.6 “Donut”. Η έκδοση 4.4.4 “KitKat” εμφανίστηκε ως μια ενημέρωση ασφαλείας, που κυκλοφόρησε στις 19 Ιουνίου του 2014. Η έκδοση Android 5.0 “Lollipop” κυκλοφόρησε στις 14 Νοεμβρίου του 2014.

Τέλος το 2014 η Google ξεκίνησε το Android One, ένα τυποποιημένο smartphone με στόχο κυρίως τους πελάτες στον αναπτυσσόμενο κόσμο. Το smartphone Android One τρέχει την τελευταία έκδοση του Android 5.1. Από τις 3 Μαρτίου του 2015, η νεότερη έκδοση του λειτουργικού συστήματος Android 5.1 είναι διαθέσιμη για επιλεγμένες συσκευές, συμπεριλαμβανομένου του Android One series, το Nexus 6 και το Nexus 9.



## 4.1 Αρχιτεκτονική Λογισμικού Android

Το Android αποτελείται από ορισμένες συνιστώσες λογισμικού που συνθέτουν ένα ολοκληρωμένο και ενιαίο σύστημα. Το σύστημα αυτό παρέχει όλα τα μέσα που απαιτούνται για τη χρήση εφαρμογών χωρίζοντάς το σε 4 επίπεδα και 5 ομάδες συνιστωσών που περιγράφονται παρακάτω, ξεκινώντας από το χαμηλότερο επίπεδο προς το υψηλότερο επίπεδο όπως δείχνει η παρακάτω εικόνα.



*Εικόνα 21: Αρχιτεκτονικό διάγραμμα του Android*

Αναλυτικά το πρώτο επίπεδο είναι το Linux Kernel. Το Android βασίζεται στον πυρήνα Linux για βασικές λειτουργίες της συσκευής όπως για παράδειγμα η διαχείριση των drivers της όπως για παράδειγμα camera driver, flash memory driver, audio driver και άλλα. Η διαχείριση των διεργασιών της και η διαχείριση της μνήμης της συσκευής καθώς και τη διαχείριση διεπαφών δικτύου που διαθέτει η συσκευή, όπως για παράδειγμα GSM, WiFi, Bluetooth, HSDPA και άλλα.

Το δεύτερο επίπεδο είναι το Native Libraries που είναι οι βιβλιοθήκες του Android οι οποίες είναι γραμμένες σε γλώσσες προγραμματισμού C και C++ και μπορούν να χρησιμοποιηθούν μέσω κατάλληλου interface της Java. Μερικές από τις κυριότερες βιβλιοθήκες είναι 1) η βιβλιοθήκη Surface Manager για τη δημιουργία παραθύρων διαστάσεων και τρισδιάστατων γραφικών. 2) Η βιβλιοθήκη Media Framework που περιέχει αποκωδικοποιητές για παραγωγή αρχείων πολυμέσων όπως MPEG, MP3 και άλλα. 3) Η βιβλιοθήκη SQLite για την υποστήριξη της βάσης δεδομένων SQL και 4) η βιβλιοθήκη WebKit για την υποστήριξη των φυλλομετρητών (browsers).

Στο δεύτερο επίπεδο πάλι αλλά σε διαφορετική συνιστώσα όμως είναι το Android Runtime. Όπως φαίνεται στην παραπάνω εικόνα το Android Runtime αποτελείται από:

α) Βασικές βιβλιοθήκες για τη διεπαφή των εφαρμογών Java με το περιβάλλον της συσκευής στην οποία εκτελούνται. β) Τη Dalvic Virtual Machine η οποία είναι υπεύθυνη για τη δημιουργία εκτελέσιμων αρχείων των εφαρμογών προκειμένου να τα τρέξει το λειτουργικό σύστημα. Κάθε εφαρμογή του Android είναι γραμμένη σε γλώσσα προγραμματισμού Java και τώρα υποστηρίζεται πλέον και η γλώσσα προγραμματισμού Json, την οποία όμως το λειτουργικό σύστημα δεν την αντιλαμβάνεται απευθείας. Για το λόγο αυτό η Dalvic Virtual Machine δημιουργεί εκτελέσιμα αρχεία .dex (Dalvic Executable) τα οποία εκτελούνται από το λειτουργικό σύστημα. Κάθε αρχείο εκτελείται από τη δική του Virtual Machine, ακόμα και όταν εκτελούνται παράλληλα μεταξύ τους, έτσι δεν επηρεάζονται τα προγράμματα και σε περίπτωση κάποιου σφάλματος σε ένα πρόγραμμα δεν προκαλείται κάποιο πρόβλημα στα υπόλοιπα.

Το τρίτο επίπεδο είναι το Application Framework. Οι εφαρμογές έχουν πρόσβαση στις βασικές βιβλιοθήκες του συστήματος μέσω των κατάλληλων διεπαφών και μέσω του Application Framework μπορούν να παρέχουν λειτουργίες και υπηρεσίες προς άλλες εφαρμογές, εφόσον η πολιτική ασφαλείας δεν περιορίζει το Application Framework. Μερικές από τις βασικές λειτουργίες που περιλαμβάνονται στο Application Framework είναι:

- **View System:** Επιτρέπει τη χρήση λιστών, πεδίων κειμένου, πλαισίων, κουμπιών και άλλα.
- **Content Providers:** Επιτρέπει στις εφαρμογές την πρόσβαση σε δεδομένα άλλων εφαρμογών ή τον διαμοιρασμό των δικών τους δεδομένων, όπως για παράδειγμα οι επαφές.
- **Resource Manager:** Παρέχει την πρόσβαση σε γραφικά και σε αρχεία σχετικά με τη διάταξη των στοιχείων του γραφικού περιβάλλοντος.
- **Notification Manager:** Διαχειρίζεται τα μηνύματα των εφαρμογών που εμφανίζονται στην status bar, όπως για παράδειγμα εισερχόμενα μηνύματα, ραντεβού, ειδοποιήσεις από Facebook, Instagram και άλλα.
- **Activity Manager:** Διαχειρίζεται τον κύκλο ζωής των εφαρμογών και παρέχει τη δυνατότητα μετάβασης στις προγενέστερες καταστάσεις τους.

Το τέταρτο και τελευταίο επίπεδο είναι το Application, που εκεί βρίσκονται οι εφαρμογές που θα χρησιμοποιεί ο χρήστης στη συσκευή του. Μερικές από τις πιο γνωστές εφαρμογές είναι: ο browser που χρησιμοποιείται για το серφάρισμα στο internet, email client, αποστολή και λήψη sms, ημερολόγιο, παιχνίδια, διαχείριση επαφών, προβολή χαρτών και εαν υπάρχει διαθέσιμος δέκτης GPS τότε εμφανίζεται και το στίγμα του χρήστη πάνω στο χάρτη, office και πολλές άλλες. Όλες αυτές οι εφαρμογές είναι γραμμένες σε γλώσσα προγραμματισμού Java όπως αναφέραμε και παραπάνω καθώς έχει μπει στο κομμάτι αυτό και η γλώσσα προγραμματισμού Json και μπορούν να τρέχουν πολλές εφαρμογές παράλληλα χωρίς να υπάρχει κάποιο πρόβλημα επηρεάζοντας τη μια εφαρμογή με την άλλη.

## 4.2 Δυνατότητες Ανάπτυξης Εφαρμογών

Οι εφαρμογές έχουν την ίδια ακριβώς ισχύ, καμία δεν προωθείται περισσότερο ή λιγότερο από κάποια άλλη, ούτε παρεμποδίζεται η μια εφαρμογή από την άλλη. Αυτό σημαίνει πως οι εφαρμογές που είναι εγκατεστημένες ήδη στο smartphone δε διαφέρουν σε τίποτα σε σχέση με αυτές που φτιάχνουν οι προγραμματιστές και οι developers καθώς έχουν και αυτοί πρόσβαση στο δομικό μέρος της συσκευής. Το πλεονέκτημα του Android είναι πως έχει τη δυνατότητα multi-tasking, δηλαδή να τρέχει πολλές εφαρμογές ταυτόχρονα, ενώ δεν απαιτείται να κλείσει καμία εφαρμογή. Όταν ο χρήστης θελήσει να βγει από μια εφαρμογή τότε αυτή η εφαρμογή εξακολουθεί να τρέχει στο background. Η υλοποίηση εφαρμογών στην πλατφόρμα Android είναι εύκολη χρησιμοποιώντας τα ανοιχτά λογισμικά όπως το eclipse μαζί με το Android SDK καθώς και το λογισμικό Android Studio. Όταν ο προγραμματιστής αντιληφθεί τη λογική ανάπτυξης των εφαρμογών τότε σε καμία περίπτωση δε θα είναι δύσκολη η ανάπτυξη της εφαρμογής. Οι εφαρμογές στο Android αποτελούνται από τα παρακάτω Components.

**Activities:** Επιτρέπουν τη διεπαφή χρήστη για να χειριστεί την αλληλεπίδραση του με την έξυπνη οθόνη του τηλεφώνου. Τα Activities αντιπροσωπεύουν μια μόνο οθόνη με μια διεπαφή χρήστη που εκτελεί διάφορες ενέργειες στην οθόνη. Για παράδειγμα, μια εφαρμογή ηλεκτρονικού ταχυδρομείου μπορεί να έχει μια δραστηριότητα (activity) που εμφανίζει τη λίστα των νέων μηνυμάτων ηλεκτρονικού ταχυδρομείου, μια άλλη δραστηριότητα (activity) είναι για να συνθέσει ένα μήνυμα ηλεκτρονικού ταχυδρομείου, και μια άλλη δραστηριότητα (activity) είναι για την ανάγνωση των e-mails. Μια δραστηριότητα (activity) υλοποιείται ως μια υποκατηγορία των δραστηριοτήτων (activities) και συντάσσεται ως εξής:

```
public class MainActivity extends Activity {  
  
}
```

Η κλάση MainActivity είναι η μοναδική κλάση δραστηριότητας Activity μιας εφαρμογής. Παρ'όλα αυτά υπάρχουν εφαρμογές που έχουν αρκετές δραστηριότητες – συνήθως κάθε δραστηριότητα παριστά μια διαφορετική οθόνη μέσα στην εφαρμογή. Η κλάση MainActivity επεκτείνει (δηλαδή κληρονομεί από) την κλάση Activity. Όταν δημιουργείται ένα project, το ADT Plugin παράγει αυτή την κλάση σαν μια δευτερεύουσα κλάση της Activity και παρέχει το κέλυφος μιας επικαλυμμένης μεθόδου onCreate, την οποία πρέπει να επικαλύπτει κάθε δευτερεύουσα κλάση της Activity.

**Services:** Είναι υπηρεσίες που επεξεργάζονται και εκτελούνται στο background μιας εφαρμογής, έτσι ώστε να εκτελεστεί η εφαρμογή σε μακροχρόνιες εργασίες. Για παράδειγμα, μια υπηρεσία μπορεί να παίζει μουσική στο παρασκήνιο ενώ ο χρήστης βρίσκεται σε διαφορετική εφαρμογή, ή θα μπορούσε να φέρει τα δεδομένα μέσω του δικτύου χωρίς να εμποδίζει την αλληλεπίδραση του χρήστη σε μια άλλη δραστηριότητα. Η υπηρεσία (service) αυτή υλοποιείται ως υποκατηγορία της υπηρεσίας και συντάσσεται ως εξής:

```
public class MyService extends Service {  
  
}
```

Όπου η κλάση MyService είναι κλάση δραστηριότητας μιας εφαρμογής η οποία κάνει μια λειτουργία που έχουμε ορίσει εμείς, η οποία επεκτείνει (δηλαδή κληρονομεί από) την κλάση Service.

**Broadcast Receivers Components:** Είναι ένα component που ανταποκρίνεται στις ανακοινώσεις μεταδόσεων σε όλο το σύστημα. Όπως για παράδειγμα να συνδεθούν και να αποσυνδεθούν ακουστικά από τη συσκευή ή όταν η οθόνη ενεργοποιείται ή απενεργοποιείται ή όταν η μπαταρία είναι χαμηλή. Όλα αυτά προέρχονται από το σύστημα. Τα Broadcast Receivers δεν έχουν καμία διεπαφή χρήστη, αλλά μπορεί να δημιουργηθεί μια ειδοποίηση στη γραμμή κατάστασης. Σκοπός του είναι να κάνει ελάχιστη ποσότητα εργασίας. Στην ουσία το broadcast receiver είναι σαν αδρανείς συνιστώσες εφαρμογών που μπορούν να δηλώσουν σε διάφορα συστήματα ή εφαρμογές τα συμβάντα τους. Όταν πραγματοποιηθεί κάποιο συμβάν το σύστημα ειδοποιεί όλα τα καταχωρημένα broadcast receivers και τα εμφανίζει σε συγκεκριμένες δράσεις που θα κοινοποιηθούν στον χρήστη έτσι ώστε να εκτελεστεί κάποια άλλη δραστηριότητα και συντάσσεται ως εξής:

```
public class MyReceiver extends BroadcastReceiver {  
  
    public void onReceive(context,intent){  
  
    }  
  
}
```

Καλούμε την κλάση MyReceiver η οποία επεκτείνει στην κλάση BroadcastReceiver η οποία είναι μια αφηρημένη κλάση με τη μέθοδο onReceiver (). Κάθε φορά που συμβαίνει ένα συμβάν το Android καλεί τη μέθοδο onReceiver () για όλους τους καταχωρημένους Broadcast. Παρατηρείτε πως το αντικείμενο intent διέρχεται από όλες τις απαιτούμενες πρόσθετες πληροφορίες και από το πλαίσιο του αντικειμένου Context.

**Content Providers:** Ένας content provider διαχειρίζεται την πρόσβαση σε ένα κεντρικό μέρος αποθήκευσης των δεδομένων. Ο πάροχος (provider) είναι μέρος της εφαρμογής Android, το οποίο συχνά παρέχει το δικό του UI για την εργασία με τα δεδομένα. Ωστόσο οι πάροχοι περιεχομένου (content providers) προορίζονται κυρίως για να χρησιμοποιηθούν από άλλες εφαρμογές, οι οποίες έχουν πρόσβαση χρησιμοποιώντας ένα αντικείμενο παρόχου πελάτη (client). Και οι δύο μαζί προσφέρουν μια συνεπή τυπική διασύνδεση με τα δεδομένα που χειρίζονται επίσης την επικοινωνία μεταξύ διεργασιών για την καλύτερη και ασφαλή πρόσβαση στα δεδομένα και συντάσσεται ως εξής:

```
public class MyContentProvider extends ContentProvider {  
    public void onCreate(){  
    }  
}
```

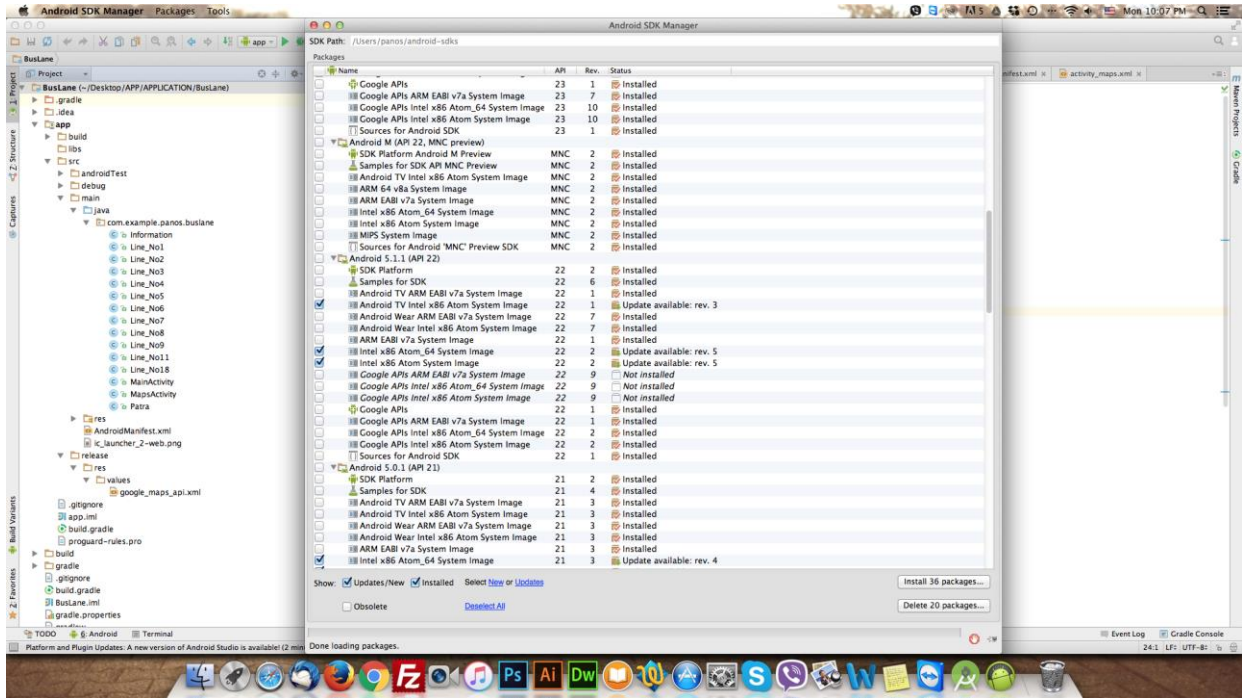
Καλούμε την κλάση MyContentProvider η οποία επεκτείνει στην κλάση ContentProvider που πρέπει να εφαρμόσει ένα τυποποιημένο σύνολο από APIs που επιτρέπουν άλλες εφαρμογές να πραγματοποιούν συναλλαγές. Η μέθοδος onCreate () καλείται όταν η δραστηριότητα δημιουργήθηκε μόνο μια φορά σε ολόκληρο τον κύκλο ζωής της δραστηριότητας.

Ωστόσο υπάρχουν και πρόσθετα συστατικά τα οποία θα χρησιμοποιηθούν στην κατασκευή της εφαρμογής. Αυτά τα συστατικά είναι τα παρακάτω:

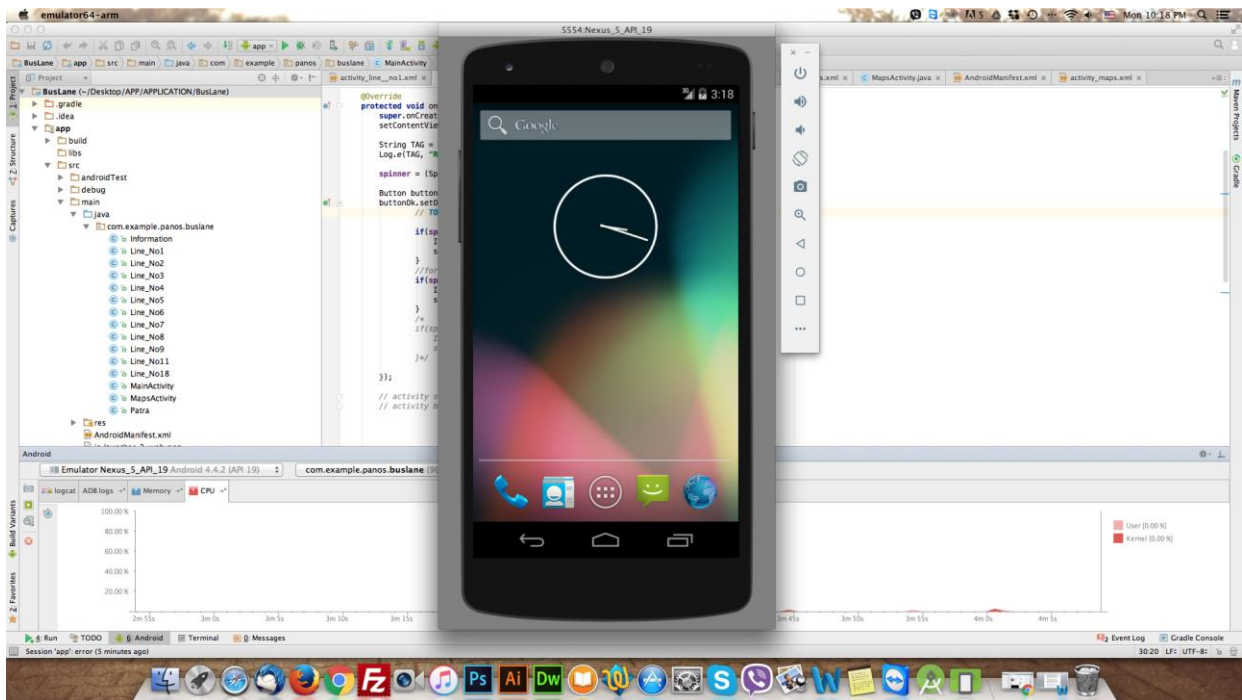
- Fragments: Αντιπροσωπεύει ένα τμήμα της διεπαφής του χρήστη σε μια δραστηριότητα.
- Views: Στοιχεία UI που έχουν συνταχθεί στην οθόνη συμπεριλαμβανομένων των πλήκτρων κ.λ.π.
- Layouts: Διάταξη των στοιχείων στην οθόνη και εμφάνιση των views ανάλογα με το layout.
- Intents: Είναι μηνύματα και συστατικά μαζί.
- Resources: Εξωτερικά στοιχεία όπως strings, drawable εικόνες και άλλα.
- Manifest: Είναι το αρχείο ρυθμίσεων για την εφαρμογή.

Το σημαντικό γεγονός που ευνοεί την ανάπτυξη εφαρμογών είναι πως το πακέτο Android SDK (Software Development Kit) συνεργάζεται με το eclipse και με το Android Studio και ο προγραμματιστής μπορεί εύκολα να βλέπει τις αλλαγές στον emulator που κάνει στον κώδικά του, χωρίς να κάνει συνέχεια εγκατάσταση στο κινητό του για να δει τις αλλαγές. Επιπλέον ο emulator είναι αξιόπιστος καθώς έχει ακριβώς την ίδια συμπεριφορά αν η εφαρμογή εγκατασταθεί σε ένα κινητό τηλέφωνο Android. Τέλος

ένα άλλο σημαντικό πλεονέκτημα είναι το γεγονός πως σε αναβαθμίσεις του λογισμικού, η εφαρμογή εξακολουθεί να δουλεύει κανονικά, χωρίς την επανασχεδίαση κάποιων σημαντικών κομματιών στον κώδικα που αφορά την αλληλεπίδραση της εφαρμογής του κινητού τηλεφώνου.



**Εικόνα 22: Εγκατάσταση Android SDK με Android Studio**



**Εικόνα 23: Άνοιγμα emulator στο Android Studio**

### 5. Μεθοδολογία Ανάπτυξης Εφαρμογής

Αναλύοντας στα προηγούμενα κεφάλαια τις τεχνολογίες από τα πρώτα PDA και τα κινητά τηλέφωνα, μέχρι τα smartphones και τα tablets καθώς και τις διάφορες τεχνολογίες εντοπισμού θέσεις και τα δομημένα μέρη της αρχιτεκτονικής του λογισμικού Android, στο κεφάλαιο αυτό θα αναλύσουμε τη μεθοδολογία ανάπτυξης της εφαρμογής. Μια μεθοδολογία ανάπτυξης ενός λογισμικού παρέχει μια συστηματική προσέγγιση στη διαδικασία της ανάλυσης, σχεδίασης, της κατασκευής και της εξέλιξης ενός έργου πληροφορικής. Μια μεθοδολογία είναι ουσιαστικά μια σειρά από στάδια τα οποία περιγράφουν συγκεκριμένες εργασίες.

Στο κεφάλαιο αυτό θα αναφερθούμε στην ανάπτυξη των απαιτήσεων της εφαρμογής, στη σχεδίασή της, με τη βοήθεια της UML (Unified Modeling Language) που είναι μια γραφική γλώσσα για την ανάλυση και τη σχεδίαση συστημάτων και εφαρμογών. Η UML είναι ουδέτερη σε σχέση με τις μεθοδολογίες και μπορεί να χρησιμοποιηθεί με διάφορους τρόπους, όπως οι Unified Process, eXtreme Programming, Catalysis και άλλες. Η πιο διαδεδομένη είναι η ενοποιημένη προσέγγιση (Unified Process). Τα βασικά στάδια της Unified Process είναι τα παρακάτω:

Η σύλληψη (inception) είναι η πρώτη φάση της ενοποιημένης προσέγγισης όπου παρουσιάζεται η αρχική ιδέα του συστήματος ή της εφαρμογής, μέχρι το σημείο που είναι αρκετά καλά θεμελιωμένη έτσι ώστε να προχωρήσει στην επόμενη φάση της επεξεργασίας.

Η λεπτομερής επεξεργασία (elaboration) είναι η δεύτερη φάση, στην οποία περιγράφεται ο σκοπός του συστήματος καθώς και η υψηλού επιπέδου αρχιτεκτονική του. Σε αυτή τη φάση προσδιορίζονται και οι απαιτήσεις του συστήματος.

Η κατασκευή (construction) είναι η τρίτη φάση, όπου σχεδιάζεται και κατασκευάζεται το λογισμικό ή η εφαρμογή.

Η μετάβαση (transition) είναι η τέταρτη φάση, όπου το λογισμικό ή η εφαρμογή υπόκειται σε έλεγχο πριν την παράδοση στους χρήστες. Η φάση της μετάβασης σηματοδοτεί την έναρξη της φάσης της συντήρησης λογισμικού και όχι το τέλος της διαδικασίας ανάπτυξης.

## 5.1 Εισαγωγή

Στο κεφάλαιο αυτό θα ασχοληθούμε με την μεθοδολογία ανάπτυξης της εφαρμογής, θα δώσουμε ορισμούς για την ανάπτυξη των απαιτήσεων της, καθώς και για τη σχεδιάσή της με τη χρήση μιας γραφικής γλώσσας, τη UML (Unified Modeling Language). Θα αναλύσουμε τη UML καθώς και τα διαγράμματα κλάσεων και περιπτώσεων χρήσης. Αυτή η διαδικασία ανάπτυξης των απαιτήσεων, με τη βοήθεια της UML δίνει τη βάση της εφαρμογής για την υλοποίησή της έτσι ώστε να υπάρχει μικρή πιθανότητα σφαλμάτων κατά τη διαδικασία του προγραμματισμού και της ανάπτυξής της.

Με την έννοια ανάπτυξη των απαιτήσεων του συστήματος εννοούμε τις διεργασίες που πρέπει να κάνει το σύστημα σε πιο αναλυτική μορφή έτσι ώστε να είναι κατανοητό ως προς τον κατασκευαστή, για το τι θα κάνει το σύστημα και ποιά θα είναι η εφαρμογή του. Όταν κατασκευάζουμε λοιπόν μια εφαρμογή ή ένα λογισμικό, το πρώτο που πρέπει να συλλάβουμε, με όσο το δυνατό μεγαλύτερη σαφήνεια, είναι οι εργασίες που αυτό θα πρέπει να κάνει, καθώς και άλλα χαρακτηριστικά που είναι επιθυμητό να έχει, όπως για παράδειγμα η εμφάνιση, οι επιδώσεις, ο τρόπος χρήσης, η ασφάλεια και άλλα.

## 5.2 Ανάπτυξη Απαιτήσεων

Οι απαιτήσεις ενός συστήματος είναι οι περιγραφές των υπηρεσιών που παρέχονται από το σύστημα καθώς και οι λειτουργικοί περιορισμοί του. Οι απαιτήσεις αυτές αντανακλούν τις ανάγκες των πελατών για ένα σύστημα που βοηθά στην επίλυση, μερικών προβλημάτων όπως ο έλεγχος μια συσκευής ή η ανεύρεση πληροφοριών. Κάποια από τα προβλήματα που προκύπτουν κατά τη διαδικασία παραγωγής των απαιτήσεων είναι αποτέλεσμα της αδυναμίας να γίνει σαφής διαχωρισμός μεταξύ αυτών των διαφορετικών επιπέδων περιγραφής. Για να διακρίνονται οι απαιτήσεις χωρίζονται σε δύο κατηγορίες, απαιτήσεις χρήστη (user requirements) για τις αφηρημένες απαιτήσεις υψηλού επιπέδου και απαιτήσεις συστήματος (system requirements) για τη λεπτομερή περιγραφή για το τι πρέπει να κάνει το σύστημα.

Οι απαιτήσεις χρήστη (user requirements) είναι δηλώσεις σε φυσική γλώσσα και διαγράμματα των υπηρεσιών που αναμένεται να παρέχει το σύστημα και των περιορισμών του κάτω από τους οποίους πρέπει να λειτουργεί.

Οι απαιτήσεις συστήματος (system requirements) περιγράφουν με λεπτομέρειες τις λειτουργίες, τις υπηρεσίες και τους λειτουργικούς περιορισμούς του συστήματος. Το έγγραφο των απαιτήσεων του συστήματος πρέπει να είναι ακριβές.

Οι απαιτήσεις πρέπει να γράφονται με διαφορετικά επίπεδα λεπτομέρειας γιατί χρησιμοποιούνται συνήθως με διαφορετικό τρόπο από κάθε κατηγορία αναγνώστη. Οι αναγνώστες των απαιτήσεων χρήστη δεν ασχολούνται με το πως θα υλοποιηθεί το σύστημα και για το λόγο αυτό δεν ενδιαφέρονται για τις λεπτομερείς δυνατότητές τους. Οι αναγνώστες όμως των απαιτήσεων συστήματος πρέπει να γνωρίζουν με μεγάλη ακρίβεια τις λεπτομέρειες για το τι θα κάνει το σύστημα επειδή συμμετέχουν άμεσα στην υλοποίηση του συστήματος.



Οι απαιτήσεις ενός συστήματος κατατάσσονται σε λειτουργικές απαιτήσεις, μη λειτουργικές απαιτήσεις και σε απαιτήσεις πεδίου (domain requirements).

Λειτουργικές απαιτήσεις (functional requirements): Πρόκειται για δηλώσεις που ορίζουν ποιες υπηρεσίες παρέχει το σύστημα, πώς θα πρέπει να αντιδρά σε συγκεκριμένες εισόδους και πώς θα πρέπει να συμπεριφέρεται σε συγκεκριμένες καταστάσεις. Οι απαιτήσεις αυτές εξαρτώνται από τον τύπο του λογισμικού που αναπτύσσεται από τους χρήστες του λογισμικού και από τη γενική προσέγγιση που ακολουθεί ο οργανισμός όταν γράφει απαιτήσεις. Όταν εκφράζονται οι απαιτήσεις από το χρήστη συνήθως περιγράφονται με πιο αφηρημένο τρόπο, σε σχέση με την καταγραφή των απαιτήσεων του συστήματος που το περιγράφουν με λεπτομέρειες. Η έλλειψη ακρίβειας στις προδιαγραφές των απαιτήσεων είναι η αιτία των προβλημάτων στην τεχνολογία λογισμικού. Για κάποιον που αναπτύσσει ένα τέτοιο σύστημα μπορεί να ερμηνεύσει μια ασαφή απαίτηση με τέτοιο τρόπο ώστε να απλοποιήσει την υλοποίησή της, αυτό όμως μπορεί να μη το θέλει ο πελάτης. Σε αυτήν την περίπτωση χρειάζεται να οριστούν νέες απαιτήσεις και να γίνουν αλλαγές στο σύστημα. Προφανώς αυτό καθυστερεί την παράδοση του έργου και αυξάνει το κόστος.

Οι προδιαγραφές των λειτουργικών απαιτήσεων σε ένα σύστημα θα πρέπει να είναι πλήρεις και συνεπείς. Ο όρος πληρότητα (completeness) σημαίνει ότι θα πρέπει να έχουν οριστεί όλες οι υπηρεσίες που απαιτούνται από το χρήστη. Ο όρος συνέπεια (consistency) σημαίνει ότι οι απαιτήσεις δεν πρέπει να έχουν ορισμούς που αντιφάσκουν. Στην πράξη όμως είναι πρακτικά αδύνατο να επιτευχθεί και πληρότητα και συνέπεια, σε μεγάλα και σύνθετα συστήματα. Αυτό συμβαίνει γιατί είναι πολύ εύκολο να γίνουν λάθη και παραλείψεις όταν γράφονται οι προδιαγραφές για τέτοιου είδους συστήματα. Επίσης οι ενδιαφερόμενοι του συστήματος έχουν διαφορετικές ανάγκες, με αποτέλεσμα να δημιουργείται μια σύγχυση στην καταγραφή των απαιτήσεων ως συνέπεια να υπάρχουν αντιφάσεις. Αυτές οι αντιφάσεις μπορεί να μην είναι εμφανείς όταν καθορίζονται αρχικά οι απαιτήσεις και αυτό έχει ως αποτέλεσμα να περιλαμβάνονται στις προδιαγραφές αντιφατικές απαιτήσεις. Το πρόβλημα αυτό μπορεί να προκύψει σε βαθύτερη ανάλυση του συστήματος ή όταν παραδοθεί το σύστημα στον πελάτη.

Μη λειτουργικές απαιτήσεις (non-functional requirements): Πρόκειται για περιορισμούς στις υπηρεσίες ή τις λειτουργίες που προσφέρει το σύστημα. Περιλαμβάνουν χρονικούς περιορισμούς, περιορισμούς της διαδικασίας ανάπτυξης και πρότυπα. Οι μη λειτουργικές απαιτήσεις έχουν εφαρμογή στο σύστημα ως σύνολο και δεν αφορούν μεμονομένα χαρακτηριστικά ή υπηρεσίες του. Επομένως, μπορεί να καθορίζουν την απόδοση, την προστασία από εξωτερικούς κινδύνους, τη διαθεσιμότητα ή και άλλες ανακλύπτουσες ιδιότητες του συστήματος. Αυτό σημαίνει ότι συχνά είναι πιο κρίσιμες από τις μεμονωμένες λειτουργικές απαιτήσεις. Οι μη λειτουργικές απαιτήσεις δεν αφορούν μόνο το σύστημα λογισμικού που θα αναπτυχθεί. Κάποιες μη λειτουργικές απαιτήσεις μπορεί να θέτουν τους περιορισμούς στη διαδικασία της ανάπτυξης του συστήματος. Οι μη λειτουργικές απαιτήσεις προκύπτουν από τις ανάγκες των χρηστών, από περιορισμένο προϋπολογισμό, από πολιτικές της εταιρείας, από την ανάγκη διαλειτουργικότητας με άλλα συστήματα λογισμικού ή υλικού ή από εξωτερικούς παράγοντες όπως οι κανονισμοί προστασίας ή η νομοθεσία περί προσωπικού απορρήτου.

Απαιτήσεις πεδίου (domain requirements): Πρόκειται για απαιτήσεις που προέρχονται από το πεδίο εφαρμογής του συστήματος και όχι από συγκεκριμένες ανάγκες των χρηστών του, αντανακλώντας χαρακτηριστικά και περιορισμούς αυτού του

πεδίου. Μπορεί να είναι λειτουργικές ή μη λειτουργικές απαιτήσεις. Οι απαιτήσεις πεδίου είναι σημαντικές επειδή συχνά αντανακλούν θεμελιώδεις έννοιες του πεδίου εφαρμογής. Αν δεν ικανοποιούνται οι απαιτήσεις το σύστημα δε μπορεί να λειτουργήσει σωστά.

Η ανάπτυξη των απαιτήσεων έχει σημαντικά πλεονεκτήματα. Εάν συγκρίνουμε τις αλλαγές ενός συστήματος σε σχέση με τον πραγματικό κόσμο τότε θα διαπιστώσουμε πως ο πραγματικός κόσμος αλλάζει λιγότερο συχνά, σε σχέση με τις απαιτήσεις ενός συστήματος λογισμικού. Βασίζοντας το σύστημά μας σε εννοιολογικές οντότητες του πραγματικού κόσμου διασφαλίζουμε με αυτόν τον τρόπο μια σχετική σταθερότητα στη δομή του υπο ανάπτυξη συστήματός. Με τον τρόπο αυτό και με το μοντέλο του πεδίου εφαρμογής θα μας δώσει το πλαίσιο σχεδιασμού της συμπεριφοράς του συστήματος, επιτυγχάνοντας με τον τρόπο αυτό να εντοπίσουμε την ανάλυση μέσα στο πλαίσιο λειτουργίας του συστήματος. Αυτό θα έχει ως αποτέλεσμα να σχεδιάσουμε πιο συγκεκριμένες περιπτώσεις χρήσης (όπως θα δούμε στη συνέχεια), να πετύχουμε συνεκτικό σχεδιασμό και να μειώσουμε το συνολικό χρόνο ανάλυσης.

## 5.3 Σχεδίαση

Το επόμενο βήμα μετά την καταγραφή και ανάλυση των απαιτήσεων είναι η ανάλυση ευρωστίας που μας βοηθάει να μεταβούμε από την ανάλυση του «τι κάνει το σύστημα;» στο σχεδιασμό του «πώς θα το κανει;». Είναι ένα καθοριστικό βήμα σε κάθε μεθοδολογία και ανάπτυξης λογισμικού. Σχεδιάζοντας λεπτομερή διαγράμματα συνεργασίας γλιτώνουμε τριπλάσιο ή και τετραπλάσιο χρόνο κατά το σχεδιασμό των διαγραμμάτων ακολουθίας. Η σχεδίαση, περιλαμβάνει τον εντοπισμό και την ταξινόμηση των αντικειμένων του συστήματος σε τρεις κατηγορίες:

1. *Αντικείμενα οντοτήτων*, τα οποία αναπαριστούν το πεδίο εφαρμογής. Σε αυτή την κατηγορία περιλαμβάνουν βάσεις δεδομένων, δομές δεδομένων και αρχεία όπου αποθηκεύονται τα δεδομένα που χρησιμοποιούνται σε κάθε σενάριο. Τα δεδομένα προκύπτουν από το μοντέλο του πεδίου εφαρμογής.
2. *Συνοριακά αντικείμενα*, τα οποία υλοποιούν την επικοινωνία των χειριστών με το σύστημα. Σε αυτή την κατηγορία περιλαμβάνουν αντικείμενα οθόνης, για παράδειγμα, παράθυρα, μενού και άλλα. Για τον εντοπισμό τους βασιζόμαστε στο πρωτότυπο του συστήματος και στα αποτελέσματα του σχεδιασμού των περιπτώσεων χρήσης.
3. *Αντικείμενα ελέγχου*, τα οποία υλοποιούν τις υπηρεσίες του συστήματος και συνδέουν τα συνοριακά αντικείμενα με τα αντικείμενα οντοτήτων. Σε αυτά τοποθετούμε τις πολιτικές επεξεργασίας και διάθεσης των δεδομένων, ώστε εάν γίνουν κάποιες αλλαγές, να γίνουν τοπικά και να μην έχουν επίδραση στη συνολική λειτουργία του συστήματος. Οι ελεγκτές λοιπόν, χρησιμοποιούνται ακριβώς επειδή δεν έχουμε ακόμη αρκετές πληροφορίες για να αποδώσουμε τη συμπεριφορά που αναπαριστούν σε συγκεκριμένες κλάσεις.

Η αποτελεσματική διαχείριση ενός έργου λογισμικού εξαρτάται από την λεπτομερή σχεδίαση του έργου. Ο υπεύθυνος του έργου πρέπει να προβλέπει τα προβλήματα που μπορεί να προκύψουν και να ετοιμάζει ενδεχόμενες λύσεις γι' αυτά.

Στο ξεκίνημα του έργου καταστρώνεται ένα πλάνο, το οποίο χρησιμεύει ως οδηγός για τα μετέπειτα στάδια. Το αρχικό αυτό πλάνο θα πρέπει να είναι το καλύτερο δυνατό με δεδομένες τις διαθέσιμες πληροφορίες και όσο το δυνατό περισσότερα διαθέσιμα στοιχεία υπάρχουν. Ο σχεδιασμός είναι μια επαναληπτική διαδικασία, η οποία ολοκληρώνεται μόνο όταν ολοκληρωθεί το ίδιο το έργο. Εφόσον γίνονται διαθέσιμες περισσότερες πληροφορίες κατά την διάρκεια του σχεδιασμού του έργου, το πλάνο πρέπει να ενημερώνεται σε τακτά χρονικά διαστήματα. Στην αρχή της διαδικασίας του σχεδιασμού θα πρέπει να λάβετε υπόψη, τους περιορισμούς που επηρεάζουν το έργο, όπως για παράδειγμα, την απαιτούμενη ημερομηνία παράδοσης, το διαθέσιμο προσωπικό, τον συνολικό προϋπολογισμό κ.λ.π. Επιπλέον καθοριστικό παράγοντα παίζουν οι παράμετροι του έργου όπως για παράδειγμα η δομή, το μέγεθος και η κατανομή των λειτουργιών. Μετά από αυτό το βήμα ορίζεται ένα ορόσημο της προόδου του έργου και τα παραδοτέα. Στη συνέχεια αυτή η διαδικασία εισέρχεται σε ένα βρόχο. Μέσα στον βρόχο εκτιμάται το χρονοδιάγραμμα του έργου και οι δραστηριότητες που ορίζονται στο χρονοδιάγραμμα ξεκινούν ή δίνεται άδεια να συνεχιστούν. Μετά από κάποιο χρονικό διάστημα θα πρέπει να ελέγξετε την πρόοδο και να σημειώσετε τις αποκλίσεις από το προγραμματισμένο χρονοδιάγραμμα.

Αν το έργο καθυστερεί ίσως χρειαστεί να διαπραγματευτούμε πάλι τους περιορισμούς και τα παραδοτέα με τον πελάτη. Εάν η διαπραγμάτευση αυτή δεν είναι επιτυχής και το χρονοδιάγραμμα δεν ικανοποιεί τον πελάτη ίσως χρειαστεί μια τεχνική επισκόπηση του έργου, έτσι ώστε να βρεθεί μια εναλλακτική λύση που να ανταποκρίνεται στους περιορισμούς του έργου και να ικανοποιεί το χρονοδιάγραμμα που έχουμε ορίσει. Παρακάτω παρουσιάζεται ο ψευδοκώδικας που αντιπροσωπεύει μια διαδικασία σχεδιασμού για την ανάπτυξη λογισμικού.

*Καθορισμός των περιορισμών του έργου*

*Πραγματοποίηση αρχικών εκτιμήσεων για τις παραμέτρους του έργου*

*Ορισμός των ορόσημων και των παραδοτέων του έργου*

**While** το έργο δεν έχει ολοκληρωθεί ή ακυρωθεί **loop**

*Δημιουργία χρονοδιαγράμματος έργου*

*Έναρξη δραστηριοτήτων σύμφωνα με το χρονοδιάγραμμα*

*Αναμονή (για λίγο)*

*Επιθεώρηση της προόδου του έργου*

*Αναθεώρηση εκτιμήσεων για τις παραμέτρους του έργου*

*Ενημέρωση του χρονοδιαγράμματος του έργου*

*Επαναδιαπραγμάτευση των περιορισμών του έργου και των παραδοτέων*

**If** (προκύψουν προβλήματα) **then**

*Έναρξη τεχνικής επισκόπησης και πιθανής αναθεώρησης*

**End if**

**End loop**

### 5.3.1 Unified Modeling Language (UML)

Η UML είναι μια γραφική γλώσσα απεικόνισης ή μοντελοποίησης ενός πληροφοριακού συστήματος βασισμένο σε αντικείμενα (αντικειμενοστρεφούς συστήματος). Χρησιμοποιείται για οπτικοποίηση που όλα τα συντακτικά στοιχεία αναπαριστούνται με γραφικά συμβολα. Κάθε γραφικό σύμβολο έχει μια ορισμένη σημασιολογία. Επίσης διευθετεί τις προδιαγραφές όλων των σημαντικών αποφάσεων της ανάλυσης, του σχεδιασμού και της υλοποίησης που πρέπει να ληφθούν κατά την ανάπτυξη ενός λογισμικού συστήματος. Ως προς την κατασκευή όλα τα μοντέλα μπορούν να συνδεθούν άμεσα με διάφορες γλώσσες προγραμματισμού και καλύπτει την τεκμηρίωση όλων των λεπτομερειών της αρχιτεκτονικής ενός συστήματος. Τέλος παρέχει τη δυνατότητα διατύπωσης των απαιτήσεων και των απαιτούμενων ελέγχων των συστατικών ενός λογισμικού συστήματος.

Η μοντελοποίηση ενός συστήματος παρέχει τη δυνατότητα αφαίρεσης των ασήμαντων λεπτομερειών και εστιάζει στις λεπτομέρειες που είναι σημαντικές για το σύστημα και για τον κατασκευαστή του. Ο τρόπος λοιπόν που κατασκευάζεται ένα σύστημα, είναι άρρηκτα συνδεδεμένος με την τεχνολογία που θα χρησιμοποιηθεί για να αναπτυχθεί το σύστημα. Σήμερα τα περισσότερα συστήματα που αναπτύσσονται είναι σχεδόν στο σύνολό τους αντικειμενοστρεφή, επειδή το <sup>16</sup>αντικειμενοστρεφές λογισμικό είναι πιο εύκολο στην αρχική του σύλληψη, αφού τα αντικείμενα είναι οντότητες του υπαρκτού κόσμου του πεδίου προβλήματος. Επίσης είναι ευκολότερο στην εξέλιξή του. Με τη χρήση των διασυνδέσεων (interfaces), μπορεί να επιτρέψει την εξέλιξη του λογισμικού με ιδιαίτερη ευκολία για τη παραγωγή λογισμικού. Η χρήση των διασυνδέσεων δεν δεσμεύει τη χρήση ενός αντικειμένου με τις λεπτομέρειες της υλοποίησης του κάνοντας εφικτή την αντικατάστασή του από ένα άλλο συμβατό αντικείμενο. Επιπλέον η αντικειμενοστρεφή προσέγγιση επιτρέπει τη δημιουργία λογισμικού με βάση τα συστατικά (components). Τέλος στις σύγχρονες τεχνολογίες κατασκευής καταναλωμένων συστημάτων στη σύνδεση επιχειρήσεων (Business – to – Business – B2B) και στη σύνδεση επιχειρηματικών εφαρμογών (Enterprise Application Integration – EAI) όπως οι υπηρεσίες του παγκόσμιου ιστού, που έχουν σαν τεχνολογικό υπόβαθρο αντικειμενοστρεφείς γλώσσες προγραμματισμού, όπως Java, C# κ.α.

Άρα είναι σχεδόν αδύνατο να ασχοληθεί σήμερα κάποιος με την ανάπτυξη λογισμικού χωρίς να γνωρίζει να αναπτύσσει αντικειμενοστρεφή συστήματα. Είναι αναγκαία η γνώση της UML για τη μοντελοποίηση αυτών των συστημάτων αφού είναι η πρότυπη γλώσσα μοντελοποίησης αντικειμενοστρεφών συστημάτων.

---

<sup>16</sup> Αντικειμενοστρεφές σύστημα: Αποτελείται από αλληλεπιδρώντα αντικείμενα που διατηρούν μια δική τους τοπική κατάσταση και παρέχουν λειτουργίες. Οι διαδικασίες αντικειμενοστρεφούς σχεδιασμού περιλαμβάνουν το σχεδιασμό κλάσεων αντικειμένων και των σχέσεων μεταξύ αυτών των κλάσεων. Αυτές οι κλάσεις ορίζουν τα αντικείμενα του συστήματος και τις αλληλεπιδράσεις τους.

Όπως αναφέραμε και παραπάνω η UML είναι μια γλώσσα απεικόνισης ή μοντελοποίησης ενός πληροφοριακού συστήματος βασισμένου σε αντικείμενα. Οι βασικοί τύποι διαγραμμάτων που παρέχει η UML για τη μοντελοποίηση και σχεδίαση του συστήματος είναι:

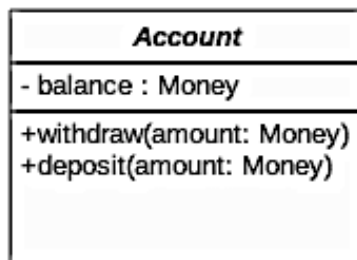
1. Διάγραμμα περιπτώσεων χρήσης (Use Case Diagram): Αποτυπώνει τις προδιαγραφές του υπό κατασκευή συστήματος.
2. Διάγραμμα κλάσεων (Class Diagram): Αποτυπώνει τις κλάσεις του συστήματος καθώς και τις στατικές σχέσεις μεταξύ αυτών των κλάσεων.
3. Διαγράμματα συμπεριφοράς (Behavior Diagrams): Αποτυπώνουν τη δυναμική συμπεριφορά του συστήματος. Στα διαγράμματα συμπεριφοράς ανήκουν και τα παρακάτω διαγράμματα.
  - 3.1 Διάγραμμα καταστάσεων (Statechart Diagram): Περιγράφει τις καταστάσεις ενός αντικειμένου του συστήματος και τον τρόπο με τον οποίο το αντικείμενο αυτό αλλάζει κατάσταση ως ανάδραση σε συμβάντα.
  - 3.2 Διάγραμμα δραστηριοτήτων (Activity Diagram): Παρέχει έναν τρόπο για την αποτύπωση των δραστηριοτήτων που λάμβάνουν χώρα στο σύστημα, συμπεριλαμβανομένων και των παράλληλων δραστηριοτήτων που μπορεί να συμβαίνουν σε αυτό.
  - 3.3 Διάγραμμα αλληλεπίδρασης (Interaction Diagram): Δύο ισοδύναμοι τύποι διαγραμμάτων που χρησιμοποιούνται για την αποτύπωση της αλληλεπίδρασης των αντικειμένων του συστήματος:
    - 3.3.1 Διάγραμμα ακολουθίας (Sequence Diagram): Δίνει έμφαση στη χρονική ακολουθία της αλληλεπίδρασης μεταξύ των αντικειμένων και απεικονίζει τις ανταλλαγές των μηνυμάτων που λαμβάνουν χώρα μεταξύ αυτών για την επίτευξη κάποιου συγκεκριμένου στόχου στο πλαίσιο μιας περίπτωσης χρήσης.
    - 3.3.2 Διάγραμμα συνεργασίας (Collaboration Diagram): Το διάγραμμα αυτό είναι ισοδύναμο με το διάγραμμα ακολουθίας με την έννοια ότι αποτυπώνονται σε αυτό οι ίδιες πληροφορίες. Η διαφορά είναι πως απεικονίζονται οι σύνδεσμοι μεταξύ των αντικειμένων που αλληλεπιδρούν και δεν είναι τόσο ξεκάθαρη η αποτύπωση της χρονικής σειράς των μηνυμάτων.
4. Διαγράμματα υλοποίησης (Implementation Diagrams): Δύο τύποι διαγραμμάτων σχετικοί με την υλοποίηση του συστήματος:
  - 4.1 Διάγραμμα συστατικών (Component Diagram): Δίνει τη δυνατότητα να εμφανίσουμε συστατικά με τις δημόσιες διασυνδέσεις που αυτά παρέχουν.
  - 4.2 Διάγραμμα διάταξης (Deployment Diagram): Δείχνει τη διάταξη των συστατικών ενός συστήματος κατά τον χρόνο εκτέλεσης και τους κόμβους στους οποίους είναι τοποθετημένα αυτά τα συστατικά.

Το κάθε διάγραμμα περιγράφει το σύστημα από διαφορετική οπτική γωνία. Έτσι χρησιμοποιούμε τα διαγράμματα κλάσεων για την περιγραφή των στατικών σχέσεων μεταξύ των κλάσεων, τα διαγράμματα συμπεριφοράς για την περιγραφή της δυναμικής συμπεριφοράς του συστήματος, και τα διαγράμματα υλοποίησης για την καταγραφή των λεπτομερειών υλοποίησης του συστήματος.

### 5.3.2 Διάγραμμα Κλάσεων (Class Diagram)

Τα διαγράμματα κλάσεων είναι ο πρώτος τύπος διαγράμματος της UML που θα δούμε και ο οποίος έχει άμεση σχέση με τα αντικειμενοστρεφή συστήματα. Τα αντικειμενοστρεφή συστήματα, λειτουργούν ως μια συλλογή συνεργαζόμενων αντικειμένων. Τα αντικείμενα αυτά αποτελούν στιγμιότυπα κλάσεων που ανήκουν σε ένα σύστημα και με τις μεταξύ τους συσχετίσεις. Κατά τη διαδικασία της ανάλυσης των προδιαγραφών του συστήματος, οι κατασκευαστές αρχίζουν να συλλέχουν πληροφορίες για το πεδίο του προβλήματος του συστήματος. Αυτές οι πληροφορίες και οι έννοιες του πεδίου προβλήματος καταγράφονται σε ένα διάγραμμα κλάσεων, το οποίο ονομάζεται μοντέλο του πεδίου προβλήματος (problem domain model). Στο μοντέλο αυτό καταγράφονται ως κλάσεις οι έννοιες του πεδίου προβλήματος, χωρίς να υπάρχει δέσμευση πως οι κλάσεις αυτές θα υπάρχουν και στο λογισμικό. Τέλος, υπάρχει και η δυνατότητα εμφάνισης συσχετίσεων μεταξύ των κλάσεων στο μοντέλο. Κατά τη διαδικασία της επεξεργασίας του μοντέλου του πεδίου προβλήματος οι περισσότερες κλάσεις θα υπάρχουν αλλά πιθανό οι συσχετίσεις και οι λειτουργίες του μπορεί να αλλάξουν και να βελτιωθούν. Οι λειτουργίες των κλάσεων του λογισμικού αποτελούν μεθόδους των αντικειμένων αυτών των κλάσεων, οι οποίες καλούνται από άλλα αντικείμενα διαφορετικών συνεργαζόμενων κλάσεων στο πλαίσιο της υλοποίησης ενός σεναρίου μιας περίπτωσης χρήσης. Οι κλάσεις είναι η βάση της κατασκευής οποιουδήποτε αντικειμενοστρεφούς συστήματος.

Συμβολίζονται με ένα παραλληλόγραμμο που έχει τρία διαμερίσματα. Στο πρώτο διαμέρισμα αναγράφεται το όνομα της κλάσης, στο δεύτερο διαμέρισμα απεικονίζονται οι ιδιότητες και στο τρίτο και τελευταίο διαμέρισμα οι λειτουργίες. Στην παρακάτω εικόνα απεικονίζεται ένα παράδειγμα μιας κλάσης για ένα τραπεζικό λογαριασμό μιας τράπεζας, που στο πρώτο διαμέρισμα αναγράφεται το όνομα της κλάσης «Account», στο δεύτερο διαμέρισμα αναγράφονται οι ιδιότητες της κλάσης, που στο παράδειγμά μας είναι το balance για το υπόλοιπο του λογαριασμού και στο τρίτο διαμέρισμα αναγράφονται οι λειτουργίες της κλάσης που στο παράδειγμά μας είναι withdraw για την ανάληψη και deposit για την κατάθεση.



Εικόνα 24 : Σύμβολο κλάσης

Σε μια κλάση οι ιδιότητες είναι ιδιωτικές (private) και συμβολίζεται με το σύμβολο – και οι λειτουργίες είναι δημόσιες (public) και συμβολίζεται με το +. Ιδιωτικά είναι τα χαρακτηριστικά μιας κλάσης που δεν είναι προσπελάσιμα από άλλες κλάσεις, ενώ δημόσια είναι αυτά που είναι προσπελάσιμα και από άλλες κλάσεις. Όταν τα δεδομένα είναι ιδιωτικά σημαίνει πως τα αντικείμενα άλλων κλάσεων δεν έχουν τη δυνατότητα να προσπελάσουν απευθείας τις ιδιότητες μιας κλάσης, αλλά χρησιμοποιούν γι' αυτό μεθόδους της κλάσης. Η μέθοδος αυτή ονομάζεται αρχή απόκρυψης των δεδομένων (information hiding principle) και έχει δύο πλεονεκτήματα:

Το πρώτο πλεονέκτημα είναι πως η εσωτερική παράσταση των δεδομένων μπορεί να αλλάξει κατά βούληση χωρίς αυτό να επηρεάσει τους χρήστες των αντικειμένων της κλάσης, εφόσον η δημόσια διασύνδεση και η σημασιολογία της παραμένουν ίδιες. Το δεύτερο πλεονέκτημα είναι πως οι μέθοδοι της κλάσης μπορούν να εφαρμόσουν πολιτικές ελέγχου για την προσπέλαση της κλάσης. Για παράδειγμα, μπορούν να ελέγξουν αν οι παράμετροι κλήσης είναι έγκυρες υπό κάποια έννοια και να εκτελέσουν τον κώδικα τους μόνο στην περίπτωση αυτή, προκαλώντας εξαιρέσεις στην αντίθετη περίπτωση. Η UML συντάσσεται ως εξής:

#### **Προσδιορισμός-πρόσβασης όνομα-ιδιότητας:**

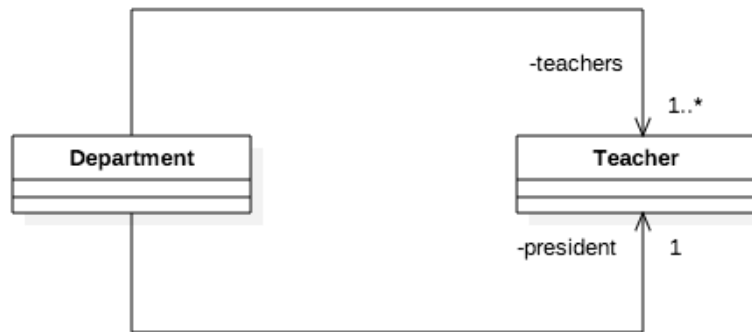
***Τύπος[πολλαπλότητα διάταξη]=Αρχική –τιμή{συμβολοσειρά ιδιοτήτων}***

Ο *τύπος* είναι κάποιος τύπος δεδομένων όπως για παράδειγμα String, int, Boolean και άλλα. Η *πολλαπλότητα* μπορεί να είναι κάποιο διάστημα τιμών με κάτω και πάνω όριο όπως για παράδειγμα 1...5. Επίσης μπορεί να είναι κάποια συγκεκριμένη αριθμητική τιμή όπως για παράδειγμα το 1. Αν το πάνω όριο είναι αστερίσκος τότε υποδειλώνει άπειρες τιμές, για παράδειγμα 1...\*. Αν πάλι είναι μόνο αστερίσκος \* αυτό σημαίνει 0 ή περισσότερα. Στην περίπτωση που έχουμε πολλαπλότητα 0..1 τότε αυτό σημαίνει 0 ή 1 και τέλος εαν είναι ένας συγκεκριμένος αριθμός όπως για παράδειγμα το 11 ή κάποια συγκεκριμένη περιοχή τιμών, όπως 2...4. Η *διάταξη* μπορεί να μην υπάρχει ή να είναι unordered. Την λέξη unordered τη χρησιμοποιούμε όπου θεωρούμε πως δεν υπάρχει διάταξη. Σε αντίθετη περίπτωση όπου υπάρχει διάταξη χρησιμοποιούμε τη λέξη ordered. Η διάταξη αφορά την περίπτωση που έχουμε πολλαπλότητα μεγαλύτερη από 1. Για παράδειγμα εαν θέλουμε να διατηρήσουμε μια συλλογή με τα ονόματα των φοιτητών ενός τμήματος με αλφαβητική σειρά, η σύνταξη σε UML θα είναι: **-students : string[\* ordered]**. Η *αρχική-τιμή* είναι η τιμή που αποδίδεται στη συγκεκριμένη ιδιότητα, μετά τη δημιουργία ενός αντικειμένου της κλάσης. Τέλος η *συμβολοσειρά ιδιοτήτων* περιέχει για τη συγκεκριμένη δήλωση. Μια ιδιότητα που χρησιμοποιείται συχνά είναι η {frozen}, που χαρακτηρίζει μια ιδιότητα ως αμετάβλητη. Από τη στιγμή που πάρει κάποια τιμή για πρώτη φορά, στη συνέχεια δε μεταβάλλεται. Για παράδειγμα, εαν θέλουμε να δηλώσουμε μια σταθερά για το επιτόκιο καταθέσεων μιας τράπεζας. Το επιτόκιο είναι πραγματικός αριθμός και εφόσον πάρει κάποια τιμή, η τιμή αυτή διατηρείται, είναι δηλαδή σταθερά. Η δήλωση αυτής της ιδιότητας σε UML θα ήταν:

**-interestRate : double = 0.035{frozen}.**

Οι κλάσεις συσχετίζονται με διάφορους τρόπους. Πριν όμως αναφερθούμε στους τρόπους συσχετίσεων των κλάσεων θα δούμε τι είναι συσχετίσεις κλάσεων. Μια συσχέτιση (association) μεταξύ δύο κλάσεων απεικονίζει μια στατική σχέση μεταξύ των δύο κλάσεων. Η ανάγκη μιας συσχέτισης προκύπτει από τη διαπίστωση πως για τη

λειτουργία μιας κλάσης απαιτείται η συνεργασία της με μια ή περισσότερες άλλες κλάσεις. Αν αυτή η συνεργασία απαιτείται να είναι σε μόνιμη βάση, τότε χρησιμοποιούμε συσχέτιση, αν είναι παροδική τότε χρησιμοποιούμε εξάρτηση. Η συσχέτιση αναπαριστάται τραβώντας μια γραμμή μεταξύ των δυο κλάσεων. Σε μια συσχέτιση μπορούμε να έχουμε τα εξής στοιχεία. Το όνομα της συσχέτισης που μπορεί να είναι μια λέξη που υποδηλώνει το νόημα της συσχέτισης. Τα ονόματα όταν αναγράφονται πάνω στη συσχέτιση, θα πρέπει να τοποθετούνται στο μέσω του συνδέσμου έτσι ώστε, να μη συγχέεται με τα ονόματα που πιθανώς θα υπάρχουν στα άκρα των συσχετίσεων. Σε κάθε άκρο μιας συσχέτισης μπορούμε να βάλουμε ένα όνομα που υποδηλώνει το ρόλο αυτής της κλάσης στη συσχέτιση. Υπάρχει περίπτωση η μια κλάση να συσχετίζεται με μια άλλη κλάση με δυο διαφορετικές συσχετίσεις. Στην περίπτωση αυτή είναι απαραίτητο να υποδηλώσουμε ποιός είναι ο ρόλος της πρώτης συσχέτισης και ποιός ο ρόλος της δεύτερης. Στην παρακάτω εικόνα απεικονίζεται μια κλάση «Department» για ένα τμήμα εκπαιδευτικού ιδρύματος και μια κλάση «Teacher» για τους εκπαιδευτικούς του τμήματος. Επιπλέον φαίνονται και δύο συσχετίσεις. Η μια αφορά τους εκπαιδευτικούς του τμήματος (teachers) και η άλλη αφορά τον πρόεδρο του τμήματος που είναι επίσης εκπαιδευτικός (president).



**Εικόνα 25: Πάνω από μια συσχετίσεις μεταξύ των κλάσεων**

Στη συνέχεια, στην παραπάνω εικόνα στα άκρα των κλάσεων είναι ο πολλαπλότητα. Η πολλαπλότητα (multiplicity) αφορά ένα άκρο μιας συσχέτισης και είναι το πλήθος των αντικειμένων που μπορεί πιθανώς να συσχετίζονται με ένα αντικείμενο της άλλης κλάσης, κατά τη διάρκεια της εκτέλεσης του προγράμματος. Επιπλέον η πλοϊμότητα (navigability) σε μια συσχέτιση συμβολίζεται με ένα βέλος στη συσχέτιση και υποδηλώνει πλοϊμότητα μόνο προς τη φορά του βέλους. Η πλοϊμότητα αφορά τη δυνατότητα που έχουμε από μια κλάση να ανακτήσουμε αντικείμενα της άλλης σε μια συσχέτιση. Για παράδειγμα στην εικόνα 23 υπάρχει πλοϊμότητα μόνο από την κλάση «Department» προς την κλάση «Teacher». Έχοντας ένα αντικείμενο Department μπορούμε να βρούμε ποιοι διδάσκουν σε αυτό το τμήμα ή ποιος είναι πρόεδρος του τμήματος. Έχοντας όμως ένα αντικείμενο Teacher δεν μπορούμε να βρούμε σε ποιο τμήμα διδάσκει.



Η γενίκευση είναι μια ειδική μορφή συσχέτισης, κατά την οποία μια γενική κλάση υποτελεί τη βάση για τη δήλωση μιας ή περισσότερων ειδικότερων, υπό κάποια έννοια κλάσεων. Η γενική κλάση ονομάζεται υπερκλάση και οι ειδικές κλάσεις ονομάζονται υποκλάσεις. Η γενίκευση στις περισσότερες γλώσσες προγραμματισμού υλοποιείται με το μηχανισμό της κληρονομικότητας ή της επέκτασης με τη φράση `extends` στην Java. Η φιλοσοφία είναι πως η γενική κλάση παρέχει λειτουργίες, ιδιότητες, συσχετίσεις και άλλα που είναι χρήσιμες σε όλες τις υποκλάσεις. Από την άλλη πλευρά όμως, οι υποκλάσεις επεκτείνουν τη λειτουργικότητα της υπερκλάσης και παρέχουν επιπλέον λειτουργίες όπου αυτό είναι απαραίτητο ή εξειδικεύουν τη συμπεριφορά τους. Το σύμβολο της γενίκευσης είναι ένα βέλος που δείχνει από την ειδική κλάση στη γενική.



**Εικόνα 25: Σύμβολο γενίκευσης**

Η συναρμολόγηση (aggregation) και η σύνθεση (composition) είναι δύο ειδικές περιπτώσεις συσχετίσεων. Και οι δύο υποδηλώνουν τη συσχέτιση μιας κλάσης με κάποια άλλη κλάση που αποτελεί μέρος της. Η συναρμολόγηση έχει μόνο έναν επιπλέον περιορισμό σε σχέση με μια συνηθισμένη συσχέτιση. Δεν επιτρέπεται η κυκλική συσχέτιση του μέρους με το όλο, αλλά μόνο μια συσχέτιση από το όλο προς το μέρος. Αντιθέτως, η σύνθεση είναι μια ισχυρή μορφή συσχέτισης με δύο επιπλέον σημασιολογικά στοιχεία που διαφοροποιείται από τις συνηθισμένες συσχετίσεις. Το πρώτο είναι πως περιέχει αποκλειστικά τα μέρη του, δηλαδή κάποιο άλλο δεν μπορεί να περιέχει το ίδιο αντικείμενο. Το δεύτερο είναι πως υπάρχει μια σχέση «ζωής και θανάτου» μεταξύ του όλου και των μερών του, δηλαδή τα μέρη του δημιουργούνται και καταστρέφονται ταυτόχρονα με το όλο.

Η συναρμολόγηση συμβολίζεται με μια συσχέτιση από το όλο προς το μέρος, στην οποία τοποθετείται ένας άσπρος ρόμβος στην πλευρά του άλλου. Η σύνθεση συμβολίζεται με μια συσχέτιση από το όλο προς το μέρος στην οποία τοποθετείται ένας μαύρος ρόμβος στην πλευρά του όλου.



**Εικόνα 27: Σύμβολο συναρμολόγησης**

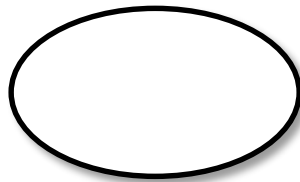


**Εικόνα 28: Σύμβολο σύνθεσης**

### 5.3.3 Διάγραμμα Περιπτώσεων Χρήσης (Use Case Diagram)

Όπως αναφερθήκαμε και πιο πάνω, το βασικότερο θέμα στην κατασκευή ενός πληροφοριακού συστήματος είναι η κατασκευή του ορθού συστήματος που ικανοποιεί τις απαιτήσεις των χρηστών του συστήματος και των άλλων συμμετεχόντων. Το μοντέλο περιπτώσεων χρήσης δίνει έμφαση στη λειτουργικότητα ενός συστήματος, όπως αυτή είναι ορατή από τους εξωτερικούς χρήστες του. Μια περίπτωση χρήσης διαμερίζει τη λειτουργικότητα ενός συστήματος σε συναλλαγές που έχουν νόημα για τους χρήστες του συστήματος (χειριστές). Τα βασικά διαγραμματικά στοιχεία των διαγραμμάτων περιπτώσεων χρήσης είναι τα παρακάτω.

Η περίπτωση χρήσης αναπαριστά ένα στόχο για έναν εξωτερικό χειριστή του συστήματος. Οι χειριστές ενός συστήματος μπορεί να είναι άνθρωποι όπως για παράδειγμα ένας γραμματέας ή ένας ταμίας, αλλά ενδέχεται να είναι και εξωτερικά συστήματα όπως ένα διατραπεζικό σύστημα συναλλαγών που είναι απαραίτητα για τη λειτουργία του υπό ανάπτυξη συστήματος. Το σύμβολο για μια περίπτωση χρήσης είναι η έλλειψη, μέσα στην οποία αναγράφεται το όνομα της περιπτώσης χρήσης όπως φαίνεται παρακάτω.

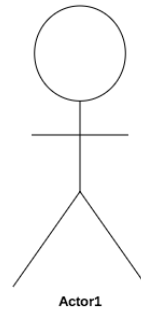


*Εικόνα 29: Σύμβολο περίπτωσης χρήσης*

Μια περίπτωση χρήσης αποτελεί ένα στόχο υψηλού επιπέδου για τον χειριστή της, δηλαδή του αποδίδει κάποιο από αποτέλεσμα που έχει αξία γι' αυτόν. Όπως για παράδειγμα, η περίπτωση χρήσης «Ανάληψη Μετρητών» είναι κάτι που έχει αξία για έναν πελάτη ενός μηχανήματος ATM μιας τράπεζας. Σε μια περίπτωση χρήσης περιλαμβάνει συνήθως πολλά εναλλακτικά σενάρια, τα οποία ονομάζονται επεκτάσεις (extensions). Η ιδέα είναι πως μια περίπτωση χρήσης έχει ένα κύριο σενάριο όπου πάνε καλά και αρκετές επεκτάσεις στις οποίες κάτι δεν πάει καλά ή προκύπτει μια εξαίρεση. Για παράδειγμα, στην περίπτωση ανάληψης μετρητών θα είχαμε ένα σενάριο όπου όλα θα πήγαιναν καλά όπως το ότι θα υπήρχαν αρκετά χρήματα, ο χρήστης θα έδινε σωστά τον κωδικό του pin του, θα ζητούσε ένα ποσό που θα καλυπτόταν από το υπόλοιπο του λογαριασμού του και πολλά άλλα και αρκετά σενάρια στο οποία κάτι δε θα πήγαινε καλά, σύμφωνα με τα παραπάνω, όπως για παράδειγμα μετά από 3 διαδοχικές λάθος προσπάθειες για την εισαγωγή του pin, το ATM θα μπορούσε να δεσμεύσει την κάρτα, άρα θα έπρεπε να χειριστούμε με κάποιο τρόπο αυτά τα σενάρια για το καλύτερο δυνατό αποτέλεσμα.

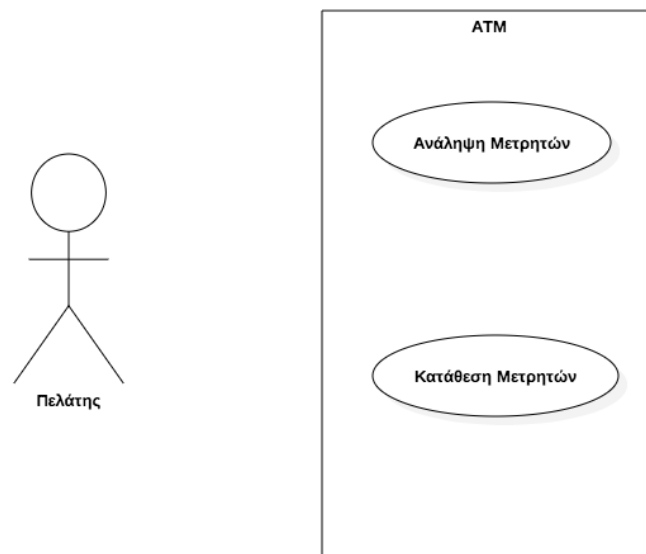
Ο χειριστής ενός συστήματος μπορεί να είναι άνθρωπος ή ένα υποσύστημα. Ένας χειριστής συμβολίζεται στο διάγραμμα περιπτώσεων χρήσης στη UML, με μια φιγούρα που απισημαίνει το ρόλο που παίζει στα πλαίσια μιας περίπτωσης χρήσης. Οι χειριστές

παίζουν πολύ σημαντικό ρόλο για τη διαδικασία εύρεσης των προδιαγραφών, γιατί αποτελούν την πηγή από την οποία προκύπτουν οι προδιαγραφές του συστήματος. Σε κάθε σύστημα που αναλύεται με το διάγραμμα περιπτώσεων χρήσης εκφράζεται η ερώτηση, «ποιές λειτουργίες πρέπει να προσφέρει το σύστημα;» και η απάντηση είναι «αυτές που χρειάζονται για την ικανοποίηση των στόχων των χειριστών του». Άρα είναι σημαντικό να βρούμε τους χειριστές του συστήματος είτε είναι άνθρωποι είτε είναι υποσυστήματα επειδή οι στόχοι τους θα μας υποδείξουν τις προδιαγραφές του συστήματος. Ο συμβολισμός για τους χειριστές στο διάγραμμα των περιπτώσεων χρήσης είναι :



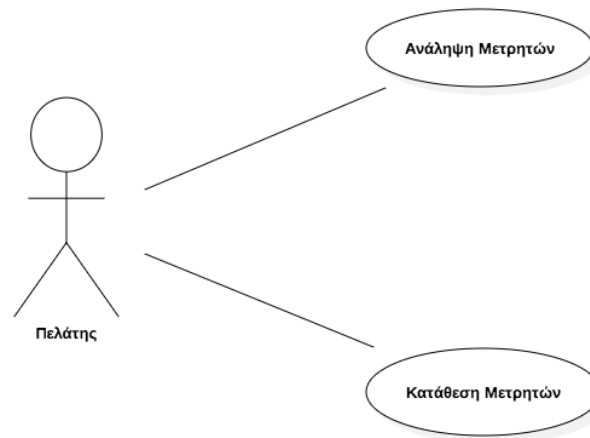
**Εικόνα 30: Σύμβολο χειριστή**

Για να διακρίνουμε τις προδιαγραφές που βρίσκονται μέσα στα πλαίσια του υπό ανάπτυξη συστήματος περιλαμβάνουμε τις περιπτώσεις χρήσης μέσα σε ένα πλαίσιο με το όνομα του συστήματος και έξω από αυτές τους χειριστές του συστήματος όπως φαίνεται παρακάτω:



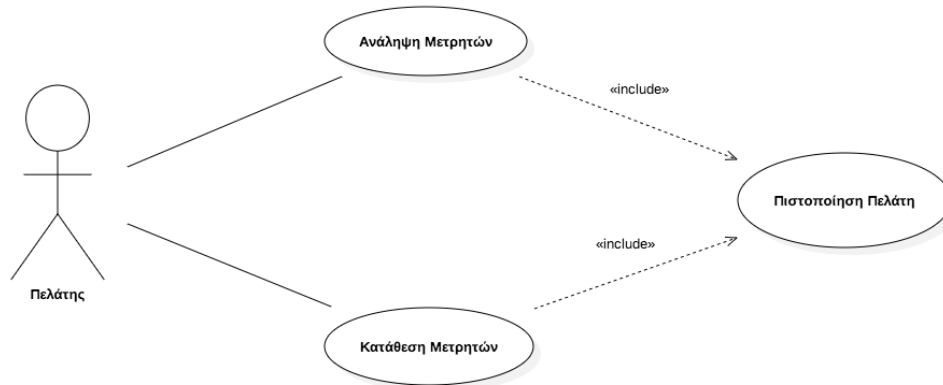
**Εικόνα 31: Πλαίσιο συστήματος διαγράμματος περιπτώσεων χρήσης**

Μια σχέση υποδηλώνει τη σχέση του χειριστή με μια περίπτωση χρήσης. Ο χειριστής πιθανό να είναι ο βασικός (primary) για μια περίπτωση χρήσης, αλλά να σχετίζονται επίσης με αυτήν και οι άλλοι χειριστές ή εξωτερικά συστήματα. Ένας πελάτης ενός ATM για παράδειγμα είναι ο κύριος χειριστής της περίπτωσης χρήσης «Ανάληψη μετρητών». Η σχέση αυτή συμβολίζεται με μια γραμμή που ενώνει τον χειριστή με την περίπτωση χρήσης. Στο παράδειγμά μας με το ATM ένας πελάτης σχετίζεται με τις περιπτώσεις χρήσης «Ανάληψη Μετρητών» και «Κατάθεση Μετρητών».



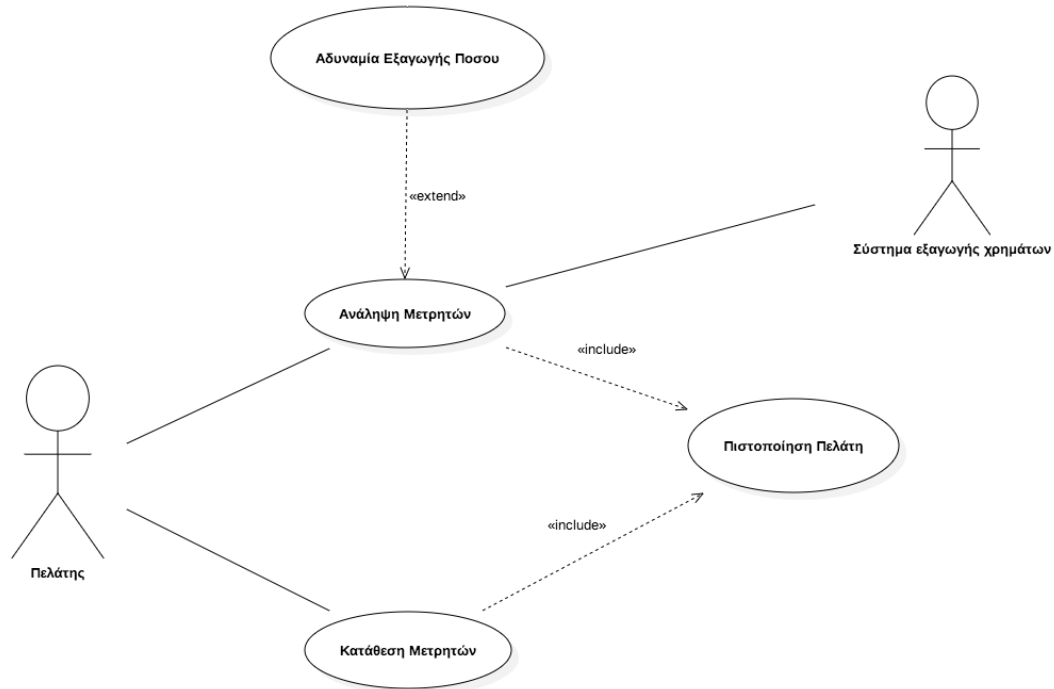
**Εικόνα 32: Σχέση χειριστή με περιπτώσεις χρήσης**

Μια ειδική περίπτωση σχέσης στην οποία σχετίζονται δύο περιπτώσεις χρήσης είναι η *συμπερίληψη* (include). Η έννοια της συμπερίληψης είναι υποχρεωτική, δηλαδή πάντα μια περίπτωση χρήσης θα συμπεριλαμβάνει την άλλη. Η συμπερίληψη χρησιμοποιείται όταν η ίδια η λειτουργικότητα περιλαμβάνεται σε περισσότερες από μια περιπτώσεις χρήσης. Ακολουθώντας το παράδειγμά με το ATM όπως αναφέραμε και παραπάνω στις περιπτώσεις χρήσης «Ανάληψη Μετρητών» και «Κατάθεση Μετρητών» θα μπορούσε να συμπεριλάβουμε και στις δυο μια περίπτωση χρήσης όπως για παράδειγμα «Πιστοποίηση Πελάτη», που θα αφορά την πιστοποίηση του πελάτη. Για το λόγο που η πιστοποίηση είναι υποχρεωτική γι' αυτές αλλά και για άλλες περιπτώσεις χρήσης, το προτιμότερο είναι να δείξουμε την πιστοποίηση ως μια ξεχωριστή περίπτωση χρήσης, η οποία συμπεριλαμβάνεται από όλες τις άλλες περιπτώσεις χρήσης που χρειάζονται πιστοποίηση του πελάτη.



**Εικόνα 33: Σχέση συμπερίληψης μεταξύ περιπτώσεων χρήσης**

Όπως η σχέση έτσι και η επέκταση (extend) καθορίζει μια σχέση μεταξύ δύο περιπτώσεων χρήσης, στην οποία σχέση μια περίπτωση χρήσης επεκτείνεται προαιρετικά από μια άλλη, ανάλογα με τις επιλογές ή την κατάσταση κάποιου χειριστή. Αυτή η σχέση προκύπτει σε συγκεκριμένα σημεία της λειτουργικότητας της βασικής ροής που ονομάζονται σημεία επέκτασης (extension points). Για παράδειγμα, αν θέλει ο χρήστης να κάνει ανάληψη μετρητών τότε θα πρέπει να επικοινωνήσει με το εξωτερικό «Σύστημα Χειρισμού Χρημάτων», το οποίο είναι υπεύθυνο για την εξαγωγή των μετρητών. Το σύστημα αυτό μπορεί να θεωρηθεί ένας από τους χειριστές της περίπτωσης χρήσης «Ανάληψη Μετρητών» και ενδέχεται να μην έχει χαρτονομίσματα για την εξαγωγή του ποσού που του ζήτησε ο πελάτης. Σε αυτή την περίπτωση, η περίπτωση χρήσης «Ανάληψη Μετρητών» επεκτείνεται από την περίπτωση χρήσης «Αδυναμία Εξαγωγής Ποσού» που είναι η ίδια περίπτωση χρήσης μέχρι το σημείο που ο πελάτης ζητήσει το ποσό και ο χειριστής «Σύστημα Εξαγωγής Χρημάτων» διαπιστώσει πως αδυνατεί να εξάγει το συγκεκριμένο ποσό.



**Εικόνα 34: Σχέση επέκτασης μεταξύ περιπτώσεων χρήσης**

Εν κατακλείδι στο συγκεκριμένο κεφάλαιο αναφερθήκαμε στη μεθοδολογία ανάπτυξης μιας εφαρμογής ή ενός συστήματος με τη χρήση της γλώσσας μοντελοποίησης, τη UML. Πριν τον σχεδιασμό των διαγραμμάτων ιδιαίτερη βάση δίνεται στην καταγραφή των απαιτήσεων και στην ανάπτυξη του από την πλευρά του χρήστη και από την πλευρά του συστήματος. Εφόσον γίνει σωστά η καταγραφή και η ανάλυση των απαιτήσεων τότε το επόμενο βήμα είναι να προχωρήσουμε στο σχεδιασμό του συστήματος ή της εφαρμογής, με τη βοήθεια του διαγράμματος κλάσεων και με το διάγραμμα των περιπτώσεων χρήσης όπου κάναμε μια εισαγωγή και αναλύσαμε. Στα επόμενα κεφάλαια θα αναφερθούμε και θα αναλύσουμε ένα ολοκληρωμένο σύστημα μιας εφαρμογής από την καταγραφή των προβλημάτων και των απαιτήσεων έως και τη μοντελοποίησή του.

### 6. Περιγραφή Εφαρμογής

Στο κεφάλαιο αυτό θα επικεντρωθούμε κυρίως στην ανάλυση και στο σχεδιασμό μιας εφαρμογής που θα υλοποιηθεί με βάση την πλατφόρμα Android. Πριν προχωρήσουμε στον “πυρήνα” της εφαρμογής που είναι ο κώδικας της, πρώτα θα περιγράψουμε την εφαρμογή, θα καταγράψουμε τις προδιαγραφές του συστήματος ως προς τις μη λειτουργικές απαιτήσεις της εφαρμογής και τις λειτουργικές απαιτήσεις του χρήστη καθώς επίσης θα σχεδιάσουμε με τη UML την εφαρμογή, με βάση το διάγραμμα κλάσεων και το διάγραμμα περίπτωσης χρήσης.

Η περιγραφή του προβλήματος είναι το βασικότερο στάδιο που πρέπει να περάσει μια εφαρμογή ή ένα σύστημα. Βοηθάει τον κατασκευαστή να καταλάβει τι θέλει να κάνει το σύστημα ή η εφαρμογή του πελάτη του. Μετά την περιγραφή του προβλήματος, ο κατασκευαστής προχωράει στο επόμενο στάδιο που είναι η καταγραφή των απαιτήσεων, από το χρήστη και από το σύστημα. Όταν λέμε καταγραφή των απαιτήσεων από το χρήστη εννοούμε, τι θέλει να κάνει το σύστημα ή η εφαρμογή για να καλύψει τις ανάγκες του. Όταν λέμε καταγραφή των απαιτήσεων από το σύστημα ή την εφαρμογή εννοούμε πως πρέπει να είναι από λειτουργικότητα και πως πρέπει να συμπεριφέρεται το σύστημα.

#### 6.1 Αντικείμενο Εφαρμογής

Το αντικείμενο της εφαρμογής που θα αναλύσουμε είναι μια διαδραστική εφαρμογή υπολογισμού και παρακολούθησης δρομολογίων και δυναμικής πληροφόρισης επιβατών για smart phones πάνω στην πλατφόρμα Android. Με βάση την εφαρμογή αυτή ο χρήστης που την έχει εγκαταστήσει στο smartphone του, θα έχει τη δυνατότητα μέσω του δικτύου των αστικών λεωφορείων, για την πόλη της Πάτρας που θα είναι συνδεδεμένη με την εφαρμογή αυτή, να γνωρίζει ανα πάσα ώρα και στιγμή που βρίσκεται το λεωφορείο, έτσι ώστε ο χρήστης – επιβάτης να πάει στην επιθυμητή στάση που θέλει εγκαίρως.

## 6.2 Περιγραφή Προβλήματος

Ένα μέσο, όπως για παράδειγμα, ένα αστικό λεωφορείο κινείται στη διαδρομή που πραγματοποιεί δρομολόγια σε τακτά χρονικά διαστήματα και στέλνει τις συντεταγμένες του από την έναρξη της διαδρομής έως και την λήξη της διαδρομής ή στις επιμέρους στάσεις, στην περίπτωση που σταματά σε ενδιάμεσους προορισμούς σε πραγματικό χρόνο.

Η εφαρμογή υποστηρίζει χρήστες – επιβάτες, οι οποίοι αναμένουν σε σημεία της διαδρομής με σκοπό να επιβιβαστούν στο διερχόμενο λεωφορείο της γραμμής που επιθυμούν. Η εφαρμογή αυτή δίνει τη δυνατότητα στους χρήστες – επιβάτες της να γνωρίζουν και να βλέπουν μέσω του smartphone τους που βρίσκεται το λεωφορείο σε τακτά χρονικά διαστήματα και να εκτιμά το χρονική απόσταση του διερχόμενου λεωφορείου από το σημείο που βρίσκονται οι χρήστες – επιβάτες. Η εφαρμογή αυτή χρησιμοποιεί βάση δεδομένων στην οποία καταχωρούνται τα στοιχεία που στέλνει το κινούμενο λεωφορείο κάθε φορά που εκτελεί συγκεκριμένη διαδρομή ανά λεπτό και μέσω ενός αλγόριθμου υπολογισμού που χρησιμοποιεί τα καταχωρηθέντα στοιχεία εκτιμάται ο χρόνος μέχρι τον χρήστη – επιβάτη.

Η εκτίμηση γίνεται μέσω του χρόνου εκτίμησης της απόστασης των δύο σημείων, του τυχαίου λεωφορείου και του επιβάτη στην παρούσα χρονική στιγμή την οποία βρίσκουμε υπολογίζοντας τα αντίστοιχα προηγούμενα σημεία στη βάση δεδομένων.

Ο χρήστης – επιβάτης με την εκκίνηση της εφαρμογής (BusLane) επιλέγει από πτυσσόμενο μενού την πόλη στην οποία επιθυμεί να κάνει αναζήτηση και μετά επιλέγει την αντίστοιχη γραμμή – διαδρομή που επιθυμεί. Η αναζήτηση αυτή γίνεται αυτόματα, αρκεί ο χρήστης να βρίσκεται σε δίκτυο. Στην περίπτωση που ο χρήστης δεν είναι ακριβώς στο δρόμο που διέρχεται το λεωφορείο τότε η εφαρμογή υπολογίζει το πιο κοντινό προς το χρήστη – επιβάτη σημείο, στο δρόμο για την εκτίμηση.

Ιδιαίτερη προσοχή πρέπει να δοθεί στην εφαρμογή ώστε να είναι όσο το δυνατό πιο απλή και εύχρηστη με σκοπό να μπορεί να χρησιμοποιείται από τους χρήστες χωρίς καμία ιδιαίτερη δυσκολία.

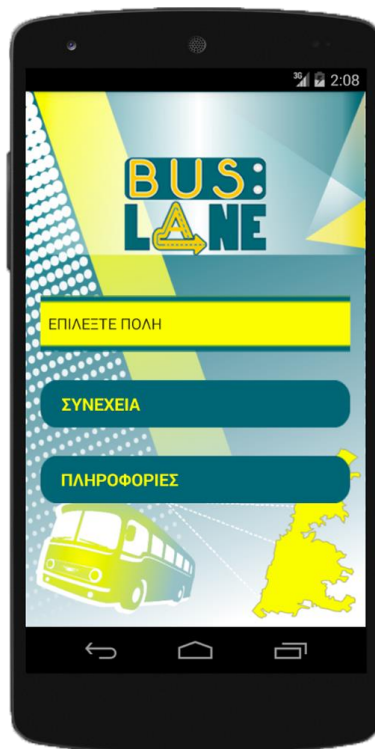
Τα πλεονεκτήματα της εφαρμογής αυτής είναι πολλά, αλλά κυρίως δημιουργούν αύξηση επιβατικότητας κίνησης και καλύτερη οργάνωση, ελαχιστοποιώντας την ταλαιπωρία των επιβατών.



## 6.3 Καταγραφή Απαιτήσεων Εφαρμογής (BusLane)

Ζητείται η κατασκευή μιας εφαρμογής με το όνομα “BusLane” η οποία θα ενημερώνει τους χρήστες – επιβάτες σχετικά με το που βρίσκεται και σε πόση ώρα θα φτάσει το λεωφορείο της επιλογής τους σε αυτούς. Πιο συγκεκριμένα η εφαρμογή θα πρέπει να έχει τις παρακάτω δυνατότητες για τους χρήστες – επιβάτες των λεωφορείων.

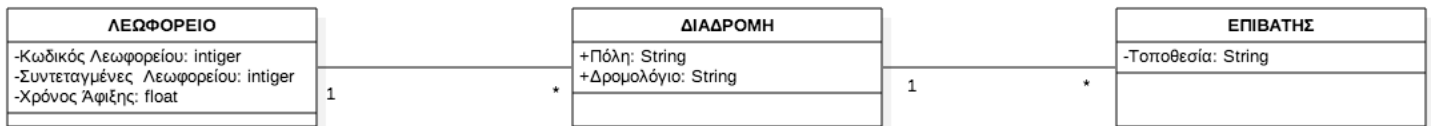
1. Η εφαρμογή με το όνομα BusLane πρέπει να υποστηρίζεται στην πλατφόρμα λογισμικού, Android.
2. Θα πρέπει να λειτουργεί σε όλες τις πρόσφατες εκδόσεις των λογισμικών.
3. Ανοίγοντας την εφαρμογή BusLane ο χρήστης θα έχει τη δυνατότητα να δει ένα όμορφο γραφικό περιβάλλον με το συνδυασμό των αποχρώσεων του κίτρινου, του γαλάζιου και του πράσινου.
4. Στην έναρξη της εφαρμογής θα υπάρχει ένα πτυσσόμενο μενού καθώς και άλλα δύο κουμπιά (buttons), το ένα κάτω από το άλλο, στο ίδιο ακριβώς μέγεθος, όπως φαίνεται στην παρακάτω εικόνα. Ο χρήστης μέσω πτυσσόμενου μενού θα μπορεί να επιλέξει την πόλη και τη γραμμή του λεωφορείου καθώς θα έχει και τη δυνατότητα να επιλέξει και κατεύθυνση γραμμής (Αφετηρία – Τερματισμός).



*Εικόνα 35: Αναπαράσταση σε screenshot στο άνοιγμα της εφαρμογής*

## 6.4 Διάγραμμα Κλάσεων

Παρακάτω απεικονίζεται το διάγραμμα κλάσεων με βάση τη UML και οι οντότητες της εφαρμογής που περιγράψαμε παραπάνω.

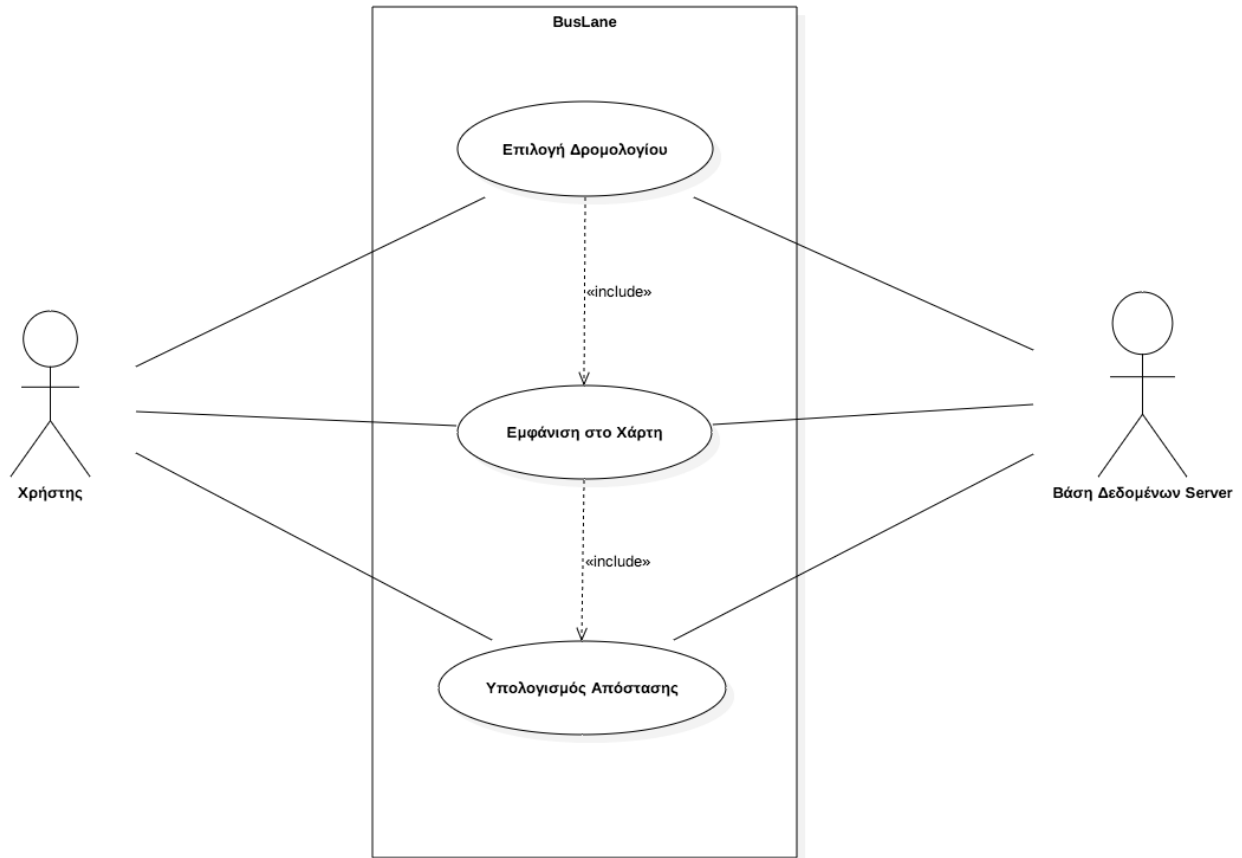


*Εικόνα 36: Διάγραμμα κλάσεων*

Σύμφωνα με τις προδιαγραφές της εφαρμογής έχουμε 3 οντότητες. Το «ΛΕΩΦΟΡΕΙΟ», τη «ΔΙΑΔΡΟΜΗ» και τέλος τον «ΕΠΙΒΑΤΗ». Οι ιδιότητες της κλάσης ΛΕΩΦΟΡΕΙΟ που είναι private (-) είναι ο κωδικός λεωφορείου, οι συντεταγμένες του λεωφορείου και ο χρόνος άφιξης του λεωφορείου. Οι κλάσεις «ΛΕΩΦΟΡΕΙΟ» και «ΔΙΑΔΡΟΜΗ» συσχετίζονται μεταξύ τους με πολλαπλότητα πολλά προς ένα 1..\*. Ενώ οι κλάσεις «ΔΙΑΔΡΟΜΗ» και «ΕΠΙΒΑΤΗΣ» συσχετίζονται μεταξύ τους με πολλαπλότητα ένα προς πολλά 1..\*.

## 6.5 Διάγραμμα Περίπτωσης Χρήσης (BusLane)

Παρακάτω απεικονίζεται το διάγραμμα περιπτώσεων χρήσης με βάση τη UML, του χρήστη με την εφαρμογή – BusLane



*Εικόνα 37: Διάγραμμα περίπτωσης χρήσης*

## 6.6 Ανάλυση περιπτώσεων χρήσης (σενάριο)

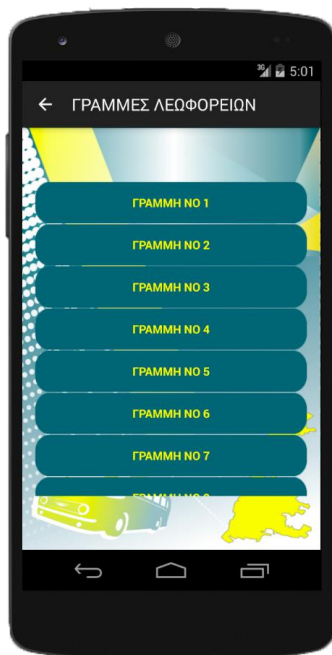
Με βάση τις απαιτήσεις της εφαρμογής οι οποίες αναλύθηκαν παραπάνω, θα αναλύσουμε τώρα τι ακριβώς θα κάνει η εφαρμογή και θα καταγράψουμε το σενάριο χρήσης της.

1. Ο χρήστης επιλέγει μια πόλη που επιθυμεί από το πτυσσόμενο μενού που υπάρχει στην αρχική οθόνη για να μπορέσει να δει το λεωφορείο που θέλει να επιβιβαστεί και μετά πατάει το κουμπί «ΣΥΝΕΧΕΙΑ» για να μεταβεί στις γραμμές των λεωφορείων, όπως φαίνεται στις εικόνες.



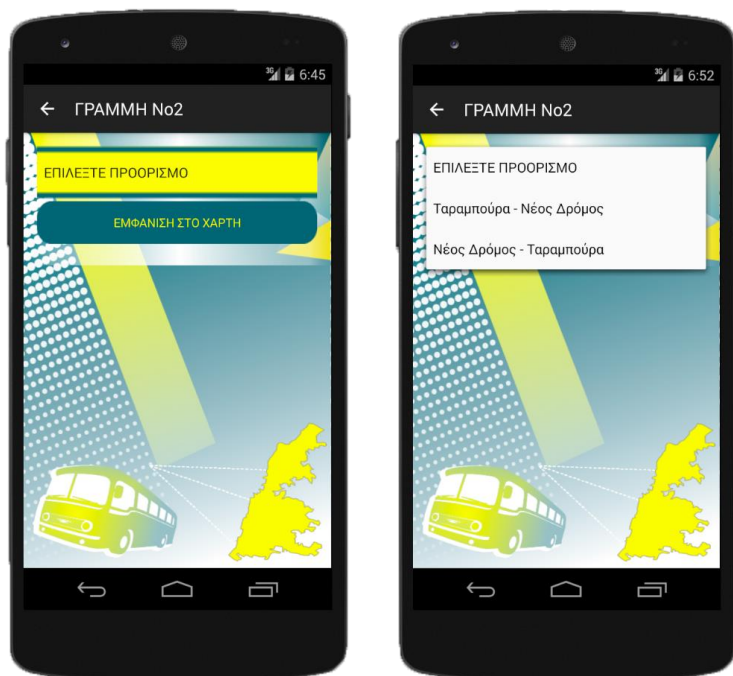
*Εικόνα 38: Αναπαράσταση σε screenshot, η εφαρμογή BusLane τη στιγμή που ο χρήστης επιλέγει πόλη*

2. Όταν ο χρήστης επιλέξει την πόλη και πατήσει «ΣΥΝΕΧΕΙΑ» τότε θα μεταβεί σε μια δεύτερη οθόνη που θα υπάρχουν όλες οι γραμμές των λεωφορείων της πόλης που επέλεξε ο χρήστης.



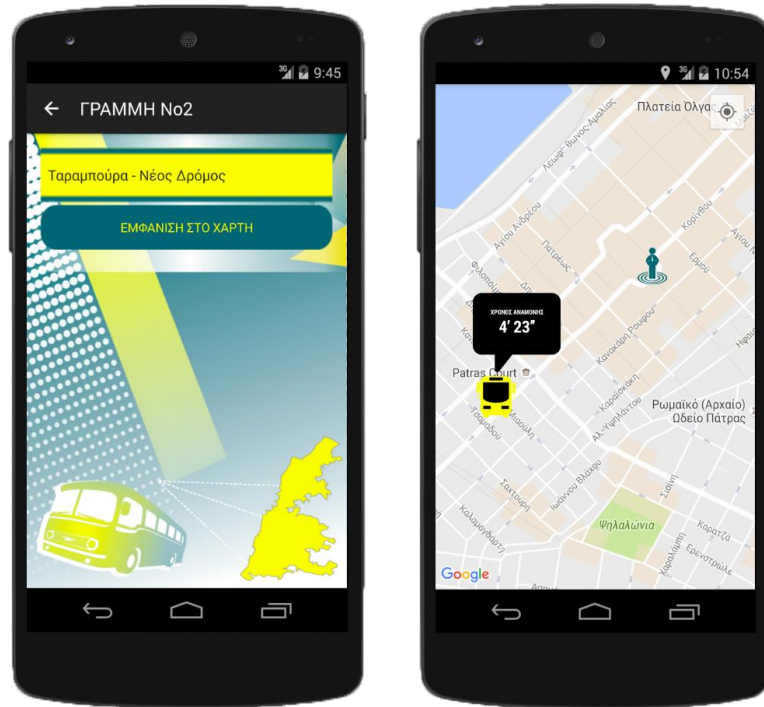
**Εικόνα 39: Αναπαράσταση σε screenshot τη στιγμή που ο χρήστης επιλέγει γραμμή λεωφορείου**

3. Στην οθόνη αυτή υπάρχει η δυνατότητα ο χρήστης σε περίπτωση που κάνει κάποιο λάθος να μεταβεί στην προηγούμενη οθόνη πατώντας το κουμπί με το βέλος που δείχνει προς τα πίσω που βρίσκετε δίπλα από τον τίτλο «ΓΡΑΜΜΕΣ ΛΕΩΦΟΡΕΙΩΝ».
4. Στην οθόνη με τις διαθέσιμες γραμμές λεωφορείων, θα υπάρχουν συγκεντρωτικά, όλες οι γραμμές σε μορφή κουμπιών, η μια κάτω από την άλλη. Στην περίπτωση που υπάρχουν πολλές γραμμές λεωφορείων ο χρήστης θα έχει τη δυνατότητα να κάνει κάθετο scroll, έτσι ώστε να βρει τη γραμμή που επιθυμεί.
5. Πατώντας τη διαθέσιμη γραμμή λεωφορείου που θέλει (είναι σε μορφή κουμπιού όπως είπαμε και παραπάνω), ο χρήστης μεταβαίνει σε μια τρίτη οθόνη που έχει τον τίτλο της γραμμής στη μαύρη μπάρα που θα υπάρχει στο επάνω μέρος της οθόνης.
6. Όταν ο χρήστης επιλέξει τη γραμμή που επιθυμεί για να δει τις πληροφορίες που θέλει σχετικά με το συγκεκριμένο λεωφορείο, τότε θα μεταβεί στην οθόνη που αφορά τις πληροφορίες της συγκεκριμένης γραμμής – δρομολογίου. Σε αυτή την οθόνη θα εμφανιστούν ένα πτυσσόμενο μενού «ΕΠΙΛΕΞΤΕ ΠΡΟΟΡΙΣΜΟ». Όταν το πατήσει ο χρήστης θα εμφανίζεται μια λίστα (drop down list) όπου θα υπάρχουν οι διαδρομές της συγκεκριμένης γραμμής. Στη λίστα αυτή θα εμφανίζονται οι διαδρομές από την έναρξη του δρομολογίου και το αντίστροφο. Με αυτόν τον τρόπο ο χρήστης θα περιορίζει τη δυνατότητα της αναζήτησης του και θα είναι πιο ξεκάθαρο να δει ακριβώς που θα βρίσκεται το λεωφορείο της επιλογής του.



*Εικόνα 40: Αναπαράσταση σε screenshot τη στιγμή που ο χρήστης επιλέγει προορισμό*

7. Όταν ο χρήστης επιλέξει τον προορισμό που επιθυμεί για να έχει τη δυνατότητα να δει το λεωφορείο που βρίσκεται, πρέπει να πατήσει το κουμπί «ΕΜΦΑΝΙΣΗ ΣΤΟ ΧΑΡΤΗ» και τότε θα εμφανιστεί ένας χάρτης. Μέσα στο χάρτη εμφανίζεται το στίγμα του συγκεκριμένου λεωφορείου καθώς και το στίγμα του χρήστη, υπολογίζοντας αυτόματα μεταξύ λεωφορείου και χρήστη το χρόνο αναμονής.



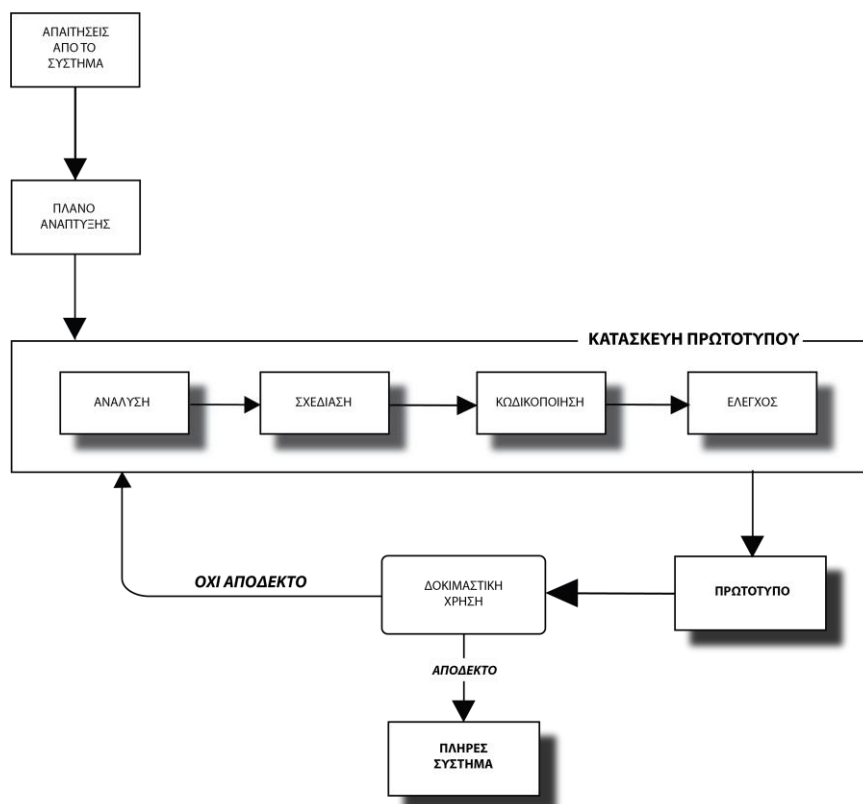
*Εικόνα 41: Αναπαράσταση σε screenshot, τη στιγμή που ο χρήστης περιμένει το λεωφορείο της επιλογής του.*

8. Για να ενεργοποιηθεί το στίγμα του χρήστη, θα πρέπει ο χρήστης να ενεργοποιήσει την τοποθεσία στο κινητό του. Εάν πάλι ο χρήστης δεν έχει ενεργοποιήσει την τοποθεσία, μπορεί να την ενεργοποιήσει μέσα από την εφαρμογή, στο στάδιο του χάρτη. Πάνω στο χάρτη – πάνω δεξιά, θα υπάρχει ένα κουμπί που, πατώντας το ενεργοποιείται η τοποθεσία του χρήστη.
9. Τέλος για να λειτουργήσει αυτή η εφαρμογή θα πρέπει ο χρήστης να βρίσκεται σε δίκτυο. Είτε μέσω Wi-Fi είτε μέσω mobile internet.

Αυτή η εφαρμογή για να λειτουργήσει αποτελεσματικά, λαμβάνει δεδομένα από τις συντεταγμένες των λεωφορείων που εκπέμπουν μέσω κάποιου πομπού. Αυτά τα δεδομένα είναι αποθηκευμένα σε μια βάση δεδομένων που υπάρχει σε έναν server. Μέσω του αρχείου php που πρέπει να βάλουμε στον server, θα επικοινωνήσει με τη βάση δεδομένων και θα λάβουμε τα δεδομένα που επιθυμούμε, τα οποία θα εμφανίσουμε στην εφαρμογή μέσω κατάλληλου κώδικα.

## 6.7 Μοντέλο Πρωτοτυποποίησης

Το μοντέλο πρωτοτυποποίησης (prototyping model) υλοποιεί την ανάπτυξη του λογισμικού σε τμήματα που ονομάζονται πρωτότυπα και όχι εξ' ολοκλήρου. Οι διαδικασίες ανάπτυξης επαναλαμβάνονται για ένα τμήμα του συστήματος κάθε φορά και για το λόγο αυτό το μοντέλο χαρακτηρίζεται ως επαναληπτικό. Κάθε πρωτότυπο περιλαμβάνει τις βασικές από τις λειτουργίες που προορίζεται να εκτελεί το Λογισμικό και τίθεται σε δοκιμασία από τον πελάτη. Από τις δοκιμασίες αυτές συλλέγονται παρατηρήσεις και πληροφορίες και η διαδικασία κατασκευής νέου πρωτοτύπου επαναλαμβάνεται μέχρις ότου το ένα πρωτότυπο να ικανοποιεί τις απαιτήσεις, δηλαδή να εκτελεί τις επιθυμητές λειτουργίες του λογισμικού με τρόπο ικανοποιητικό και για να γίνεται αποδεκτό από τον πελάτη. Από το σημείο αυτό και μετά μπορούν να προστεθούν και οι υπόλοιπες λειτουργίες, ώστε το λογισμικό να είναι ολοκληρωμένο. Παρακάτω απεικονίζεται σε διάγραμμα το μοντέλο πρωτοτυποποίησης και τα στάδια του.



Εικόνα 42: Το μοντέλο της πρωτοτυποποίησης



Ένα σημαντικό πλεονέκτημα του μοντέλου αυτού, είναι η δυνατότητα απόκτησης άποψης για την εφαρμογή Λογισμικού νωρίτερα απ' ότι στο μοντέλο του καταρράκτη. Αυτό μπορεί να γλιτώσει την ανάπτυξη από καθυστερήσεις ή ακόμα και από ολική αποτυχία, τα οποία θα επέρχονταν αν ο κατασκευαστής αναγκαζόταν να οπισθοδρομήσει την ανάπτυξη, ενώ αυτή είχε προχωρήσει πολύ. Παράλληλα, ιδιαίτερη σημασία αποκτά η διοίκηση του έργου η οποία πρέπει να εξασφαλίζει την υλοποίηση του πρωτότυπου και την εύκολη τροποποίηση του. Κάθε κατασκευή πρωτότυπου μπορεί να θεωρηθεί ως ένα μικρό έργο Λογισμικού το οποίο κατασκευάζεται με διαδικασίες που μπορούν να ακολουθούν άλλα μοντελα κύκλου ζωής, όπως αυτού του καταρράκτη.

Με βάση τις παραπάνω παρατηρήσεις το μοντέλο πρωτοτυποποίησης χρησιμοποιείται στην ανάπτυξη εφαρμογών λογισμικού για τις απαιτήσεις από τις οποίες δεν υπάρχει βεβαιότητα στην αρχή της ανάπτυξης, οπότε δεν μπορούν να συμφωνηθούν και να παγιωποιηθούν. Τέτοιες είναι εφαρμογές που κατασκευάζονται για πρώτη φορά ή που είναι στενά εξαρτημένες από τον πελάτη, χωρίς να υπάρχει αποδεκτό προηγούμενο παράδειγμα. Ωστόσο, το μέγεθος των εφαρμογών αυτών δεν μπορεί να είναι ιδιαίτερα μεγάλο, διότι ο χρόνος ανάπτυξης κάθε πρωτότυπου μεγαλώνει και η απαιτούμενη ευελιξία μειώνεται.

Για όλους τους παραπάνω λόγους το μοντέλο αυτό χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής μας.

### 7. Υλοποίηση

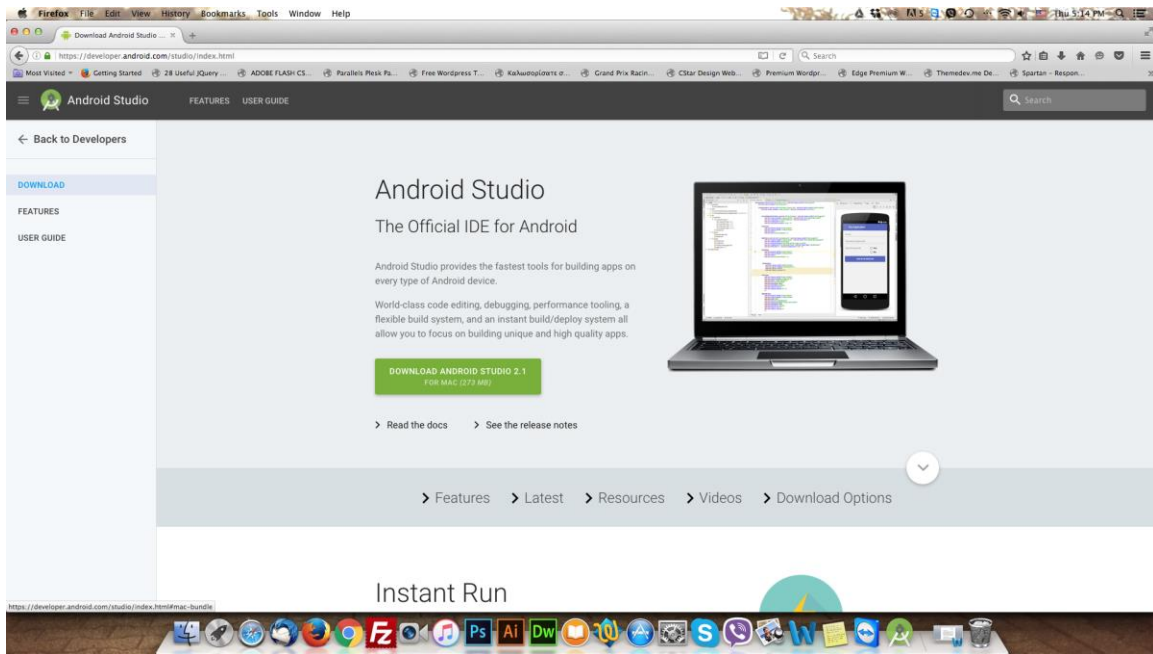
Σε αυτό το κεφάλαιο θα ασχοληθούμε κυρίως με το κεντρικό σώμα της εφαρμογής, με τον τρόπο που έχει δομηθεί η εφαρμογή καθώς και με την κωδικοποίησή της ως προς το user interface και τη λειτουργικότητά της. Θα δούμε μέσα από τον κώδικα πως ο χρήστης μπορεί να επιλέξει την πόλη που επιθυμεί, τη γραμμή καθώς και τον προορισμό του λεωφορείου που θέλει να επιβιβαστεί.

#### 7.1 Περιγραφή Υλοποίησης

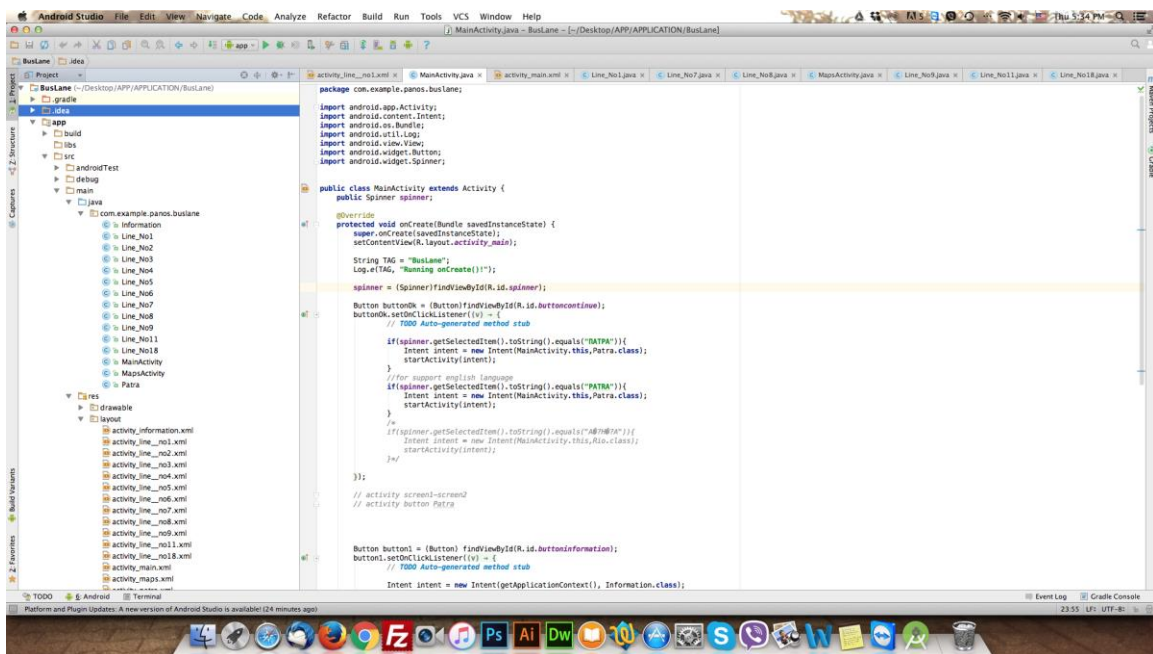
Πρίν προχωρήσουμε στην υλοποίηση της εφαρμογής, στα προηγούμενα κεφάλαια αναλύσαμε και καταγράψαμε τις προδιαγραφές και τις απαιτήσεις της εφαρμογής. Με βάση το μοντέλο πρωτοτυποποίησης προσπαθήσαμε να φτιάξουμε ένα πρότυπο της εφαρμογής BusLane. Ακολούθησαν διαγράμματα κλάσεων (class diagram) και σενάριο περιπτώσεων χρήσης (uses cases diagram). Εφόσον σχεδιάστηκαν και αναλύθηκαν σωστά τα διαγράμματα και οι απαιτήσεις της εφαρμογής και κατόπιν συνεννόησης με τον υπεύθυνο του project το επόμενο βήμα είναι η υλοποίηση της εφαρμογής προγραμματίζοντάς την. Στον «πυρήνα» της εφαρμογής θα δούμε κώδικες σε γλώσσα προγραμματισμού java για τη λειτουργικότητα της εφαρμογής αλλά κυρίως θα δούμε και κώδικα σε γλώσσα xml για τη μορφοποίηση και την εμφάνιση της εφαρμογής. Επιπλέον θα δούμε και τη MySQL για την αποθήκευση των δεδομένων της εφαρμογής στη βάση δεδομένων που χρειάστηκε να φτιάξουμε, καθώς και php για την επικοινωνία του server με τη βάση δεδομένων.

## 7.2 Κωδικοποίηση

Για να υλοποιηθεί η εφαρμογή πάνω στην πλατφόρμα Android χρειάστηκε να εγκαταστήσουμε το Android Studio στον υπολογιστή μας από τη διεύθυνση <https://developer.android.com/studio/index.html>



*Εικόνα 43: Screenshot του Android Studio*



*Εικόνα 44: Screenshot του προγράμματος Android Studio*

Για την υλοποίηση της εφαρμογής χρειάστηκαν να δημιουργήσουμε τις ακόλουθες κλάσεις σε κώδικα java. MainActivity.class, Patra.class, Line\_No1.class, Line\_No2.class, Line\_No3.class, Line\_No4.class, Line\_No5.class, Line\_No6.class, Line\_No7.class, Line\_No8.class, Line\_No9.class, Line\_No11.class, Line\_No18.class, Information.class, MapsActivity.class. Η κάθε κλάση συνοδεύεται με το αντίστοιχο xml αρχείο που έχει το ίδιο όνομα με αυτό της κλάσης. Για παράδειγμα η κλάση MainActivity.java συνοδεύεται από το αρχείο activity\_main.xml. Το αρχείο .xml είναι η μορφοποίηση της κλάσης .java. Δηλαδή το xml αρχείο τροποποιείται σύμφωνα με το πως θέλουμε να φαίνεται η εφαρμογή στα smartphones ή στα tablets για τη συγκεκριμένη κλάση. Για παράδειγμα έχουμε την κλάση, MainActivity.java και ο κώδικας σε java είναι ο ακόλουθος:

```
package com.example.panos.buslane;
```

```
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Spinner;
```

```
public class MainActivity extends Activity {  
    public Spinner spinner;
```

```
    @Override
```

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

```
        String TAG = "BusLane";  
        Log.e(TAG, "Running onCreate!");
```

```
        spinner = (Spinner)findViewById(R.id.spinner);
```

```
        Button buttonOk = (Button)findViewById(R.id.buttoncontinue);  
        buttonOk.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
```

```
            public void onClick(android.view.View v) {  
                // TODO Auto-generated method stub
```

```
                if(spinner.getSelectedItem().toString().equals("HATPA")){  
                    Intent intent = new Intent(MainActivity.this,Patra.class);  
                    startActivity(intent);  
                }  
            }
```

```
            //for support english language
```

```
            if(spinner.getSelectedItem().toString().equals("PATRA")){  
                Intent intent = new Intent(MainActivity.this,Patra.class);  
                startActivity(intent);  
            }  
        }
```

```

    }
    });
    // activity screen1-screen2
    // activity button Patra

    Button button1 = (Button) findViewById(R.id.buttoninformation);
    button1.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(android.view.View v) {
            // TODO Auto-generated method stub

            Intent intent = new Intent(getApplicationContext(), Information.class);
            startActivity(intent);
        }

    });
}
}
}
}

```

Το αντίστοιχο αρχείο activity\_main.xml που συνοδεύεται από την κλάση MainActivity.java είναι το ακόλουθο με τον αντίστοιχο κώδικα σε xml

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin" tools:context=".MainActivity"
    android:background="@drawable/buslanelarge">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:layout_marginTop="54dp"
        android:src="@drawable/buslanelogo2"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:contentDescription="@string/logobuslane" />

    <Spinner
        android:id="@+id/spinner"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_marginTop="200dp"
        android:entries="@array/select_town"
        android:gravity="center"
        android:paddingBottom="10dp"
        android:paddingTop="10dp"
        android:spinnerMode="dropdown"
        android:layout_alignParentEnd="false"
        android:layout_alignParentStart="false"

```

```
android:textAlignment="center"  
android:visibility="visible"  
android:longClickable="false"  
android:focusable="true"  
android:focusableInTouchMode="true"  
android:clickable="true"  
android:background="@drawable/firstbutton" />
```

<Button

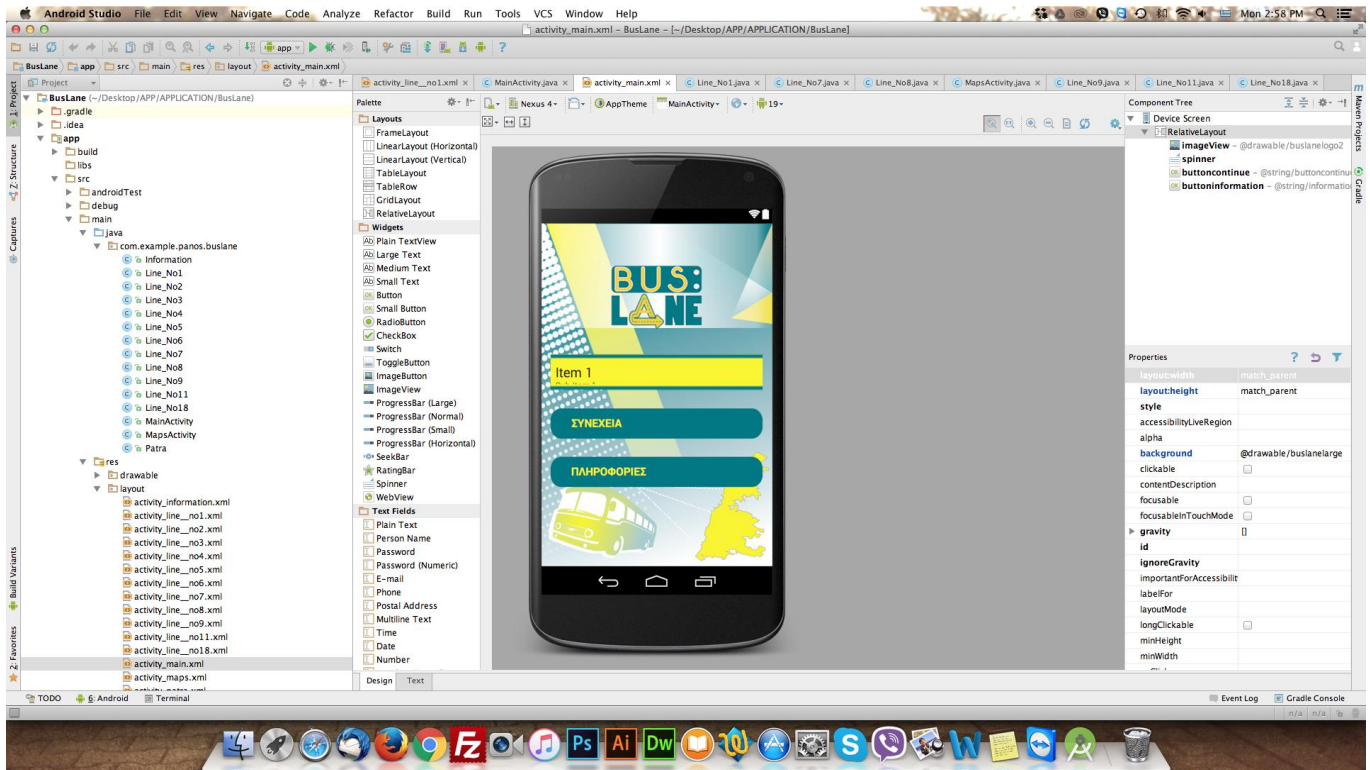
```
android:id="@+id/buttoncontinue"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_below="@+id/spinner"  
android:layout_centerHorizontal="true"  
android:layout_marginTop="30dp"  
android:contentDescription="@string/buttoncontinue"  
android:text="@string/buttoncontinue"  
android:textStyle="bold"  
android:focusable="true"  
android:focusableInTouchMode="true"  
android:textColor="#fafa03"  
android:background="@drawable/buttons"  
android:layout_alignParentStart="false"  
android:paddingEnd="200dp" />
```

<Button

```
android:id="@+id/buttoninformation"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_alignStart="@id/buttoninformation"  
android:layout_below="@+id/buttoncontinue"  
android:layout_marginTop="30dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/ButtonInformations"  
android:text="@string/informations"  
android:textStyle="bold"  
android:textColor="#fafa03"  
android:layout_toEndOf="@id/buttoninformation"  
android:paddingEnd="160dp" />
```

</RelativeLayout>

Με βάση τον κώδικα xml από το αρχείο activity\_main.xml όταν ανοίξουμε την εφαρμογή θα εμφανιστεί το αντίστοιχο user interface όπως φαίνεται και στην παρακάτω εικόνα.



*Εικόνα 45: Σχεδιασμός user interface από το πρόγραμμα Android Studio*



*Εικόνα 46: User Interface της εφαρμογή BusLane και απεικόνιση των κουμπιών «ΣΥΝΕΧΕΙΑ» και «ΠΛΗΡΟΦΟΡΙΕΣ»*

Στη συνέχεια θα περιγράψουμε και θα εξηγήσουμε βήμα βήμα τι κάνει η κάθε κλάση ξεχωριστά.

```
package com.example.panos.buslane;
```

```
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Spinner;
```

```
public class MainActivity extends Activity {  
    public Spinner spinner;
```

Η εφαρμογή έχει την ονομασία buslane και το πακέτο στην οποία τρέχει είναι package com.example.panos.buslane. Είναι συμβατό με τις εκδόσεις Android από 4.4.2 KitKat έως και 5.1.1 Lollipop. Οι βιβλιοθήκες που χρησιμοποιούνται για την εφαρμογή συντάσσονται στην αρχή πριν την κεντρική κλάση με τον ακόλουθο κώδικα «import android.app.Activity;» «import android.content.Intent;» και διάφορα άλλα, ανάλογα με την πολυπλοκότητα της εφαρμογής και του κώδικα.

Στη συνέχεια δημιουργούμε την κλάση MainActivity η οποία επεκτείνει την κλάση Activity που είναι η κεντρική κλάση της εφαρμογής. Στη συνέχεια θέλουμε να εμφανίσουμε ένα πτυσσόμενο μενού (drop down list) ή αλλιώς ένα widget spinner με τα ονόματα των πόλεων που θα επιλέγει ο χρήστης. Στην περίπτωση αυτή έχουμε μόνο την πόλη της Πάτρας. Για να ξεκινήσουμε να φτιάχνουμε το widget αυτό, το πρώτο βήμα που κάνουμε είναι να το δηλώσουμε μέσα στην κεντρική κλάση, με την παρακάτω συνάρτηση

```
public Spinner spinner;
```

Μέσα στην κεντρική κλάση δημιουργούμε μια ιδιωτική (protected) μέθοδο που είναι προσπελάσιμη από υποκλάσεις, την onCreate. Η onCreate(Bundle savedInstanceState) είναι μια συνάρτηση η οποία καλείτε για την προετοιμασία της δραστηριότητας (activity) που θα πραγματοποιηθεί με βάση τον κώδικα που θα γράψουμε. Καλώντας τη συνάρτηση super.onCreate(savedInstanceState) θα εκτελεστεί ο κώδικας που γράψαμε μέσα στο user interface που σχεδιάσαμε με βάση την xml. Στη συγκεκριμένη περίπτωση το αρχείο xml είναι το activity\_main όπως φαίνεται παρακάτω η σύνταξη της μεθόδου και των συναρτήσεών της.



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

Στη συνέχεια μέσα σε αυτή τη μέθοδο θα βάλουμε να εκτελείται ένα widget το οποίο θα εμφανίζει ένα πτυσσόμενο μενού με τις πόλεις που θα επιλέγει ο χρήστης. Στη συγκεκριμένη περίπτωση θα έχει ο χρήστης μόνο την επιλογή της Πάτρας. Συντάσσοντας τη συνάρτηση για το προγραμματισμό του widget που είναι η εξής :  
`spinner = (Spinner)findViewById(R.id.spinner);` στην ουσία δηλώνουμε το widget, που στην προκειμένη περίπτωση είναι η drop down list (spinner) και του δηλώνουμε να βρει μέσα στο αρχείο `activity_main.xml` το στοιχείο με ID `spinner`. Για να είναι λειτουργικό το widget `spinner` όπως και η υπόλοιπη εφαρμογή όλα τα στοιχεία που αναγράφονται μέσα, όπως για παράδειγμα Πόλη, Γραμμές λεωφορείων και άλλα είναι strings και αποθηκεύονται στο σύστημα μας στο εξής paths: `BusLane/app/src/res/values/strings.xml`.

Εφόσον δηλώσουμε το widget στον κώδικά μας μέσα στην κλάση `MainActivity` και δηλώσουμε τα στοιχεία που θα συμπεριλαμβάνονται στην drop down list, που είναι οι πόλεις, μέσα στο αρχείο `strings.xml` με τον παρακάτω κώδικα

```
<string-array name="select_town">
    <item>ΕΠΙΛΕΞΤΕ ΠΟΛΗ</item>
    <item>ΠΑΤΡΑ</item>
</string-array>
```

τότε συντάσσουμε μια συνάρτηση η οποία θα δίνει την εντολή παντάσως ο χρήστης πάνω στο πτυσσόμενο μενού (spinner) και αφού επιλέξει την πόλη που επιθυμεί, να πατήσει το κουμπί *συνέχεια* για να προχωρήσει στο επόμενο στάδιο που είναι η λίστα με τις διαθέσιμες γραμμές λεωφορείων. Η συνάρτηση για το κουμπί που θα πατήσει ο χρήστης για να προχωρήσει στο επόμενο βήμα είναι η εξής:

```
Button buttonOk = (Button)findViewById(R.id.buttoncontinue);
buttonOk.setOnClickListener(new View.OnClickListener())
```

Με τη συνάρτηση αυτή δίνουμε εντολή στο σύστημα να βρεί στο αρχείο `activity_main.xml` το στοιχείο με id `buttoncontinue` και στη συνέχεια μέσω της `setOnClickListener` να εκτελέσει την εντολή που του δώσαμε μέσα στη μέθοδο που ακολουθεί.

```
spinner = (Spinner)findViewById(R.id.spinner);
```

```
Button buttonOk = (Button)findViewById(R.id.buttoncontinue);
buttonOk.setOnClickListener(new View.OnClickListener() {
```

```

@Override
public void onClick(android.view.View v) {
    // TODO Auto-generated method stub

    if(spinner.getSelectedItem().toString().equals("ΠΑΤΡΑ")){
        Intent intent = new Intent(MainActivity.this,Patra.class);
        startActivity(intent);
    }
    //for support english language
    if(spinner.getSelectedItem().toString().equals("PATRA")){
        Intent intent = new Intent(MainActivity.this,Patra.class);
        startActivity(intent);
    }
}
});

```

Μέσα στη μέθοδο `onClick` τρέχουν 2 συνθήκες ελέγχου `if`. Η μια συνθήκη υποστηρίζεται για την Ελληνική γλώσσα και η δεύτερη συνθήκη υποστηρίζεται για την Αγγλική γλώσσα. Αναλυτικά έχουμε την πρώτη συνθήκη ελέγχου με τη συνάρτηση

```

if(spinner.getSelectedItem().toString().equals("ΠΑΤΡΑ")){
    Intent intent = new Intent(MainActivity.this,Patra.class);
    startActivity(intent);
}

```

Εάν ο χρήστης επιλέξει από το `spinner`, που αναλύσαμε πιο πάνω το στοιχείο της Πάτρας τότε μέσω του `Intent` θα ζητήσει τη δραστηριότητα της κλάσης Πάτρας (`Patra.class`) και θα μεταβεί στο επόμενο στάδιο που είναι όλες οι διαθέσιμες γραμμές λεωφορείων της πόλης της Πάτρας. Ακριβώς η ίδια συνθήκη ελέγχου ισχύει και στην περίπτωση που είναι στην Αγγλική γλώσσα ή και στην περίπτωση που έχουμε βάλει και άλλες πόλεις στο πτυσσόμενο μενού.

```

Button button1 = (Button) findViewById(R.id.buttoninformation);
button1.setOnClickListener(new View.OnClickListener() {

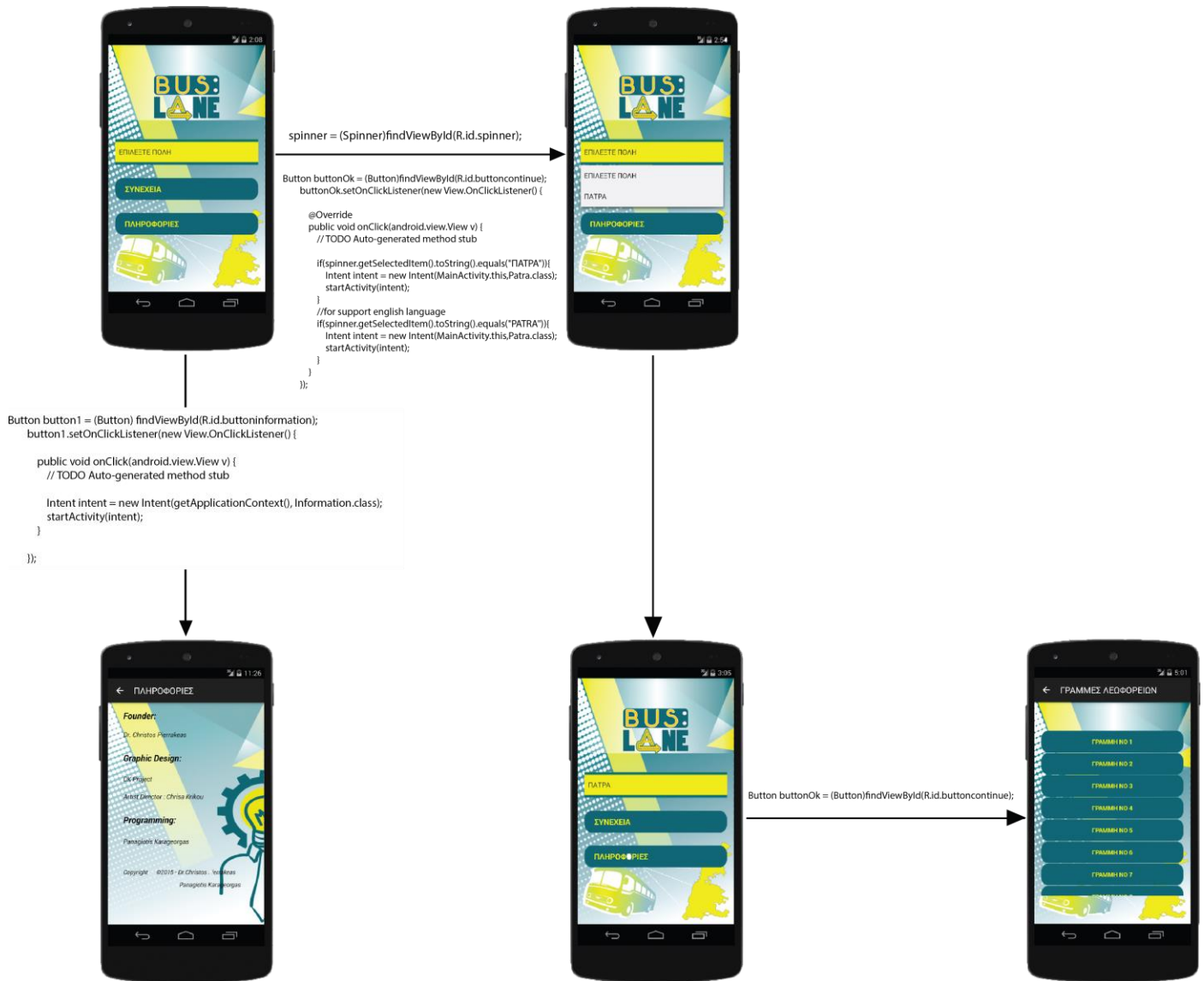
    @Override
    public void onClick(android.view.View v) {
        // TODO Auto-generated method stub

        Intent intent = new Intent(getApplicationContext(), Information.class);
        startActivity(intent);
    }

});
}
}

```

Το ίδιο ακριβώς ισχύει και για το κουμπί (button) που αναφέραμε στη συνάρτηση παραπάνω. Δηλαδή με τη συνάρτηση `findViewById(R.id.buttoninformation)` δίνουμε την εντολή, να βρεί το σύστημα το στοιχείο με id `buttoninformation` στο αρχείο `activity_main.xml` που στην προκειμένη περίπτωση αναφερόμαστε στο κουμπί που θα μας παραπέμψει στις πληροφορίες της εφαρμογής και σε ότι έχει να κάνει με την ομάδα που υλοποίησε την εφαρμογή. Μέσω της μεθόδου `onClick` και του στοιχείου `Intent` ζητάμε να μεταβούμε στην κλάση πληροφορίες (`Information.class`) εφόσον ο χρήστης πατήσει το συγκεκριμένο κουμπί, όπως αναπαριστάνετα στην παρακάτω εικόνα.



**Εικόνα 47: Αναπαράσταση των συναρτήσεων της εφαρμογής**

Στη συνέχεια θα δούμε τον πυρήνα της δραστηριότητας στην οποία εμφανίζονται όλες οι διαθέσιμες γραμμές λεωφορείων, με τον κώδικα στη γλώσσα προγραμματισμού java. Δημιουργούμε μια κλάση Patra.class η οποία απευθύνεται μόνο στις διαθέσιμες γραμμές λεωφορείων σχετικά με την πόλη της Πάτρας.

```

package com.example.panos.buslane;

import android.app.Activity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class Patra extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_patra);

        // activity button Line No1

        Button button3 = (Button) findViewById(R.id.ButtonNo1);
        button3.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub

                Intent intent = new Intent(getApplicationContext(), Line_No1.class);

                intent.putExtra("ButtId", "romantza");
                startActivity(intent);
            }

        });

        // activity button Line No2

        Button button1 = (Button) findViewById(R.id.buttonNo2);
        button1.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub

                Intent intent = new Intent(getApplicationContext(), Line_No2.class);

                intent.putExtra("ButtId", "tarampoura");
                startActivity(intent);

                // intent.putExtra("ButtId", "1");
                // startActivity(intent);
            }

        });
    }
}

```

```
// activity button Line No3
```

```
Button button2 = (Button) findViewById(R.id.ButtonNo3);  
button2.setOnClickListener(new View.OnClickListener() {  
  
    @Override  
    public void onClick(View v) {  
        // TODO Auto-generated method stub  
  
        Intent intent = new Intent(getApplicationContext(), Line_No3.class);  
  
        intent.putExtra("ButtId", "zarouxleika");  
        startActivity(intent);  
    }  
});
```

Όπως ήδη έχουμε αναφέρει, η κάθε κλάση συνοδεύεται από το αντίστοιχο xml αρχείο. Έτσι στην περίπτωση αυτή που η κλάση είναι η Patra.class θα συνοδεύεται από το αρχείο activity\_patra.xml που είναι το user interface της συγκεκριμένης δραστηριότητας. Ο κώδικας σε xml είναι ο παρακάτω:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:paddingLeft="@dimen/activity_horizontal_margin"  
    android:paddingRight="@dimen/activity_horizontal_margin"  
    android:paddingTop="@dimen/activity_vertical_margin"  
    android:paddingBottom="@dimen/activity_vertical_margin"  
    tools:context="com.example.panos.buslane.Patra"  
    android:background="@drawable/buslanelarge">  
  
    <TableRow  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_gravity="center_horizontal"  
        android:id="@+id/tableRow"></TableRow>  
  
    <ScrollView  
        android:layout_width="match_parent"  
        android:layout_height="380dp"  
        android:layout_marginTop="50dp">  
  
        <LinearLayout  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:orientation="vertical">  
  
            <Button  
                android:id="@+id/ButtonNo1"  
                android:layout_width="fill_parent"  
                android:layout_height="50dp"  
                android:layout_marginBottom="1dp"
```

```
android:background="@drawable/buttons"  
android:contentDescription="@string/No1"  
android:textColor="#fafa03"  
android:textStyle="bold|normal"  
android:text="@string/No1" />
```

<Button

```
android:id="@+id/buttonNo2"  
android:layout_width="fill_parent"  
android:layout_height="50dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No2"  
android:textColor="#fafa03"  
android:text="@string/No2"  
android:textStyle="bold" />
```

<Button

```
android:id="@+id/ButtonNo3"  
android:layout_width="fill_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No3"  
android:text="@string/No3"  
android:textColor="#fafa03"  
android:textStyle="bold"  
android:onClick="setContentView" />
```

<Button

```
android:id="@+id/ButtonNo4"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No4"  
android:text="@string/No4"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

<Button

```
android:id="@+id/ButtonNo5"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No5"  
android:text="@string/No5"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

<Button

```
android:id="@+id/ButtonNo6"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"
```

```
android:contentDescription="@string/No6"  
android:text="@string/No6"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

#### <Button

```
android:id="@+id/ButtonNo7"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No7"  
android:text="@string/No7"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

#### <Button

```
android:id="@+id/ButtonNo8"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No8"  
android:text="@string/No8"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

#### <Button

```
android:id="@+id/ButtonNo9"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No9"  
android:text="@string/No9"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

#### <Button

```
android:id="@+id/ButtonNo11"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No11"  
android:text="@string/No11"  
android:textColor="#fafa03"  
android:textStyle="bold" />
```

#### <Button

```
android:id="@+id/ButtonNo18"  
android:layout_width="match_parent"  
android:layout_height="50dp"  
android:layout_marginTop="1dp"  
android:background="@drawable/buttons"  
android:contentDescription="@string/No18"
```



```

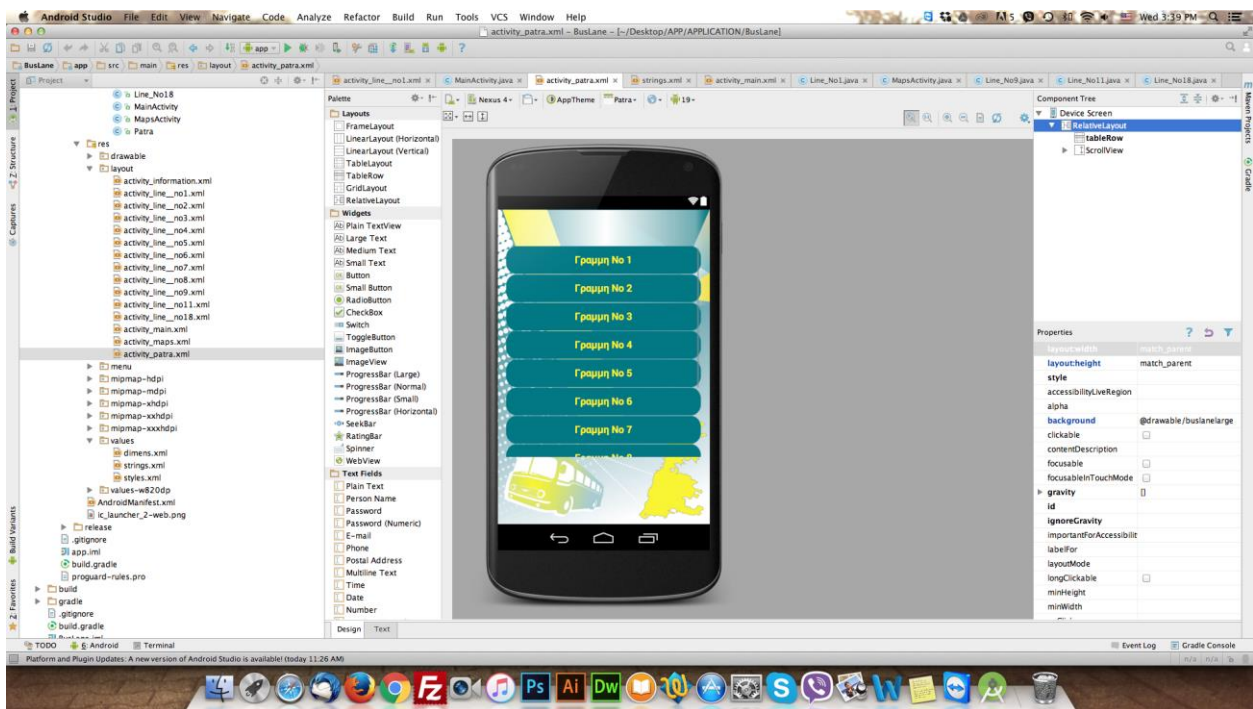
android:text="@string/No18"
android:textColor="#fafa03"
android:textStyle="bold" />
</LinearLayout>

```

```
</ScrollView>
```

```
</RelativeLayout>
```

Με βάση τον παραπάνω κώδικα σε xml ανοίγοντας την εφαρμογή στη δραστηριότητα με τις διαθέσιμες γραμμές λεωφορείων της Πάτρας θα εμφανιστεί το ακόλουθο user interface όπως φαίνεται στην παρακάτω εικόνα.



**Εικόνα 48: User Interface της κλάσης Πάτρας από το πρόγραμμα Android Studio**

Στη συνέχεια θα περιγράψουμε και θα αναλύσουμε βήμα βήμα τον κώδικα της κλάσης Πάτρας (Patra.class).

```
package com.example.panos.buslane;
```

```
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;
```

```
public class Patra extends Activity {
```

Όπως αναφέραμε, εξηγώντας τον κώδικα της πρώτης κλάσης η οποία ήταν η MainActivity.class έτσι και στην περίπτωση αυτή ισχύουν ακριβώς τα ίδια. Στην αρχή της κλάσης κάνουμε εισαγωγή των βιβλιοθηκών που χρησιμοποιούμε μέσα στον κώδικα.

```
@Override
```

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

```
// activity button Line No1
```

```
    Button button3 = (Button) findViewById(R.id.ButtonNo1);  
    button3.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
```

```
        public void onClick(View v) {  
            // TODO Auto-generated method stub
```

```
                Intent intent = new Intent(getApplicationContext(), Line_No1.class);
```

```
                intent.putExtra("ButtId", "romantza");  
                startActivity(intent);
```

```
            }  
        });
```

```
// activity button Line No2
```

```
Button button1 = (Button) findViewById(R.id.buttonNo2);  
button1.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {  
        // TODO Auto-generated method stub
```

```
        Intent intent = new Intent(getApplicationContext(), Line_No2.class);  
        intent.putExtra("ButtId", "tarampoura");  
        startActivity(intent);
```

```
        // intent.putExtra("ButtId", "1");
```

```
        // startActivity(intent);
```

```
    }
```

```
});
```

```
// activity button Line No3
```

```
Button button2 = (Button) findViewById(R.id.ButtonNo3);  
button2.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {  
        // TODO Auto-generated method stub
```

```
        Intent intent = new Intent(getApplicationContext(), Line_No3.class);
```

```
        intent.putExtra("ButtId", "zarouxleika");  
        startActivity(intent);
```

```
    }
```

```
});
```

```
...
```

```
«Υπόλοιπος κώδικας για τις κλάσεις των γραμμών των λεωφορείων»
```

```
...
```

```
}
```

Γράφουμε για την κάθε κλάση ξεχωριστά τη μέθοδο onClick, στην οποία από την κλάση Πάτρα (Patra.class) με user interface το αρχείο activity\_patra.xml ζητάμε από τις κλάσεις μέσω των συναρτήσεων:

```
Intent intent = new Intent(getApplicationContext(), Line_No2.class);
```

```
Intent.putExtra("ButtId" , "tarampoura");  
startActivity(intent);
```

να μας δώσουν τις πληροφορίες της συγκεκριμένης δραστηριότητας που θα κάνει ο χρήστης. Στην προκειμένη περίπτωση σύμφωνα με την παραπάνω συνάρτηση ζητάμε πληροφορίες από την κλάση Γραμμή No1 (Line\_No1.class). Σε κάθε κλάση αντιστοιχεί και ένα string σύμφωνα με την περιοχή εκτέλεσης του δρομολογίου της συγκεκριμένης γραμμής. Για παράδειγμα στη συνάρτηση παραπάνω ζητάμε δεδομένα από την Line\_No1.class που αντιστοιχεί στο string «tarampoura» και εφόσον ο χρήστης πατήσει το συγκεκριμένο κουμπί τότε θα ξεκινήσει η δραστηριότητα που έχουμε ορίσει, η οποία είναι να μεταβεί σε μια δραστηριότητα που θα έχει την επιλογή να δεί τις περιοχές των δρομολογίων καθώς και να επιλέξει την αφετηρία ή τον τερματισμό της γραμμής που θέλει να επιβιβαστεί, όπως φαίνεται στην παρακάτω εικόνα.



*Εικόνα 49: Αναπαράσταση της κλάσης Patra.class με τη δραστηριότητα Line\_No2.class*

Όταν ο χρήστης πατήσει τη διαθέσιμη γραμμή που επιθυμεί να επιβιβαστεί τότε θα μεταβεί σε μια δραστηριότητα στην οποία θα έχει την επιλογή μέσω πτυσσόμενου μενού (drop down list) να επιλέξει το δρομολόγιο του λεωφορείου που θέλει από την αφετηρία ή τον τερματισμό του. Ο αντίστοιχος κώδικας σύμφωνα με τη συγκεκριμένη κλάση σε γλώσσα προγραμματισμού java και σε xml είναι ο ακόλουθος:

```
package com.example.panos.buslane;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.Spinner;

public class Line_No2 extends Activity {
    Spinner spinner;
    ArrayAdapter<CharSequence> adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_line__no2);

        String TAG = "BusLane";
        Log.e(TAG, "Running onCreate!");

        spinner = (Spinner)findViewById(R.id.spinner2);
        adapter = ArrayAdapter.createFromResource(this, R.array.select_destination,
                                                android.R.layout.simple_spinner_item);

        Button buttonOk = (Button)findViewById(R.id.buttons);
        buttonOk.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(android.view.View v) {
                // TODO Auto-generated method stub

                if (spinner.getSelectedItem().toString().equals("Ταραμπούρα - Νέος Δρόμος")) {
                    Intent intent = new Intent(Line_No2.this, MapsActivity.class);
                    startActivity(intent);
                }

                if (spinner.getSelectedItem().toString().equals("Νέος Δρόμος - Ταραμπούρα")) {
                    Intent intent = new Intent(Line_No2.this, MapsActivity.class);
                    startActivity(intent);
                }
            }
        });
    }
}
```

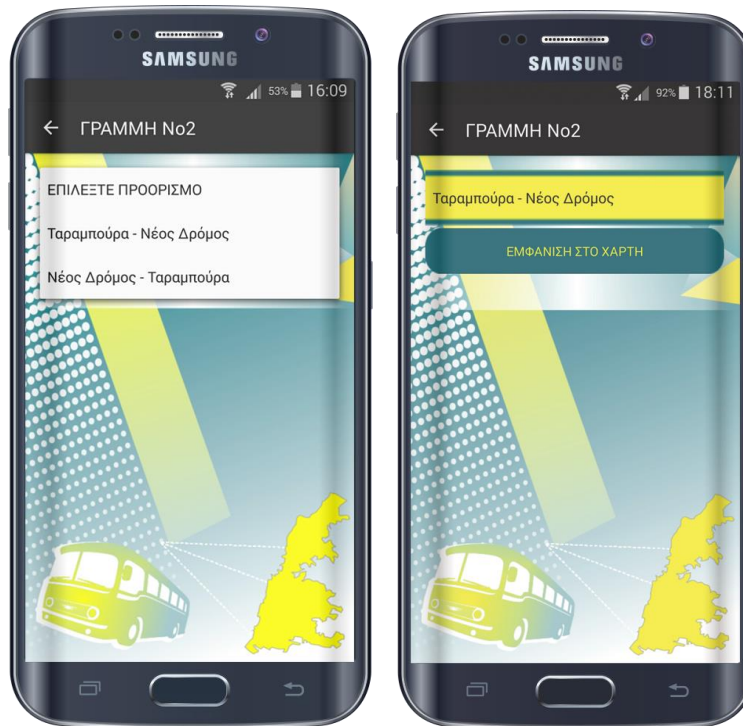
Και το user interface σε xml που συνοδεύεται ο παραπάνω κώδικας για να αλληλεπιδρά ο χρήστης με την εφαρμογή είναι ο ακόλουθος:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context="com.example.panos.buslane.Line_No2"
    android:background="@drawable/buslanelarge">
```

```
<Spinner
    android:id="@+id/spinner2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/firstbutton"
    android:entries="@array/select_destination2"
    android:gravity="center"
    android:paddingBottom="20dp"
    android:paddingTop="20dp"
    android:prompt="@array/select_destination2" />
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="ΕΜΦΑΝΙΣΗ ΣΤΟ ΧΑΡΤΗ"
    android:id="@+id/buttons"
    android:layout_below="@+id/spinner2"
    android:layout_alignEnd="@+id/spinner2"
    android:layout_alignParentStart="true"
    android:layout_marginTop="3dp"
    android:background="@drawable/buttons"
    android:textColor="#fafa03" />
```

```
</RelativeLayout>
```



*Εικόνα 50: Υλοποίηση παραδείγματος για τη γραμμή No2*

Όπως αναφέραμε και πιο πάνω ακριβώς το ίδιο ισχύει και σε αυτή την περίπτωση της κλάσης (Line\_No2.class). Πριν την κλάση Line\_No2 δηλώνουμε τις βιβλιοθήκες στις οποίες θα χρησιμοποιήσουμε κάποια από τα συστατικά τους. Στη συνέχεια δηλώνουμε το πτυσσόμενο μενού που θα χρησιμοποιήσουμε (spinner) το οποίο θέλουμε να το εμφανίσουμε στη δραστηριότητα που θέλουμε να κάνει η συγκεκριμένη κλάση μέσω του ArrayAdapter.

```
package com.example.panos.buslane;
```

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.Spinner;
```

```
public class Line_No2 extends Activity {
    Spinner spinner;
    ArrayAdapter<CharSequence> adapter;
```

Το ArrayAdapter για να λειτουργήσει και να εμφανίσει σωστά τις λίστες (ListView) στο user interface της δραστηριότητας πρέπει να δηλωθεί στο αρχείο strings.xml σαν array όπως φαίνεται στο παράδειγμα παρακάτω.

```
<string name="title_activity_line_no2">Line No2</string>
<string-array name="select_destination2">
  <item>ΕΠΙΛΕΞΤΕ ΠΡΟΟΡΙΣΜΟ</item>
  <item>Ταραμπούρα - Νέος Δρόμος</item>
  <item>Νέος Δρόμος - Ταραμπούρα</item>
</string-array>
```

Στη συνέχεια εφόσον δηλώσουμε τη λίστα στην οποία θα εμφανίζονται στο spinner μέσω του ArrayAdapter τα στοιχεία που δηλώσαμε στο αρχείο strings.xml θα συντάξουμε τη συνάρτηση του spinner και στον κώδικα java για να είναι λειτουργικό.

**@Override**

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_line_no2);
```

```
String TAG = "BusLane";
Log.e(TAG, "Running onCreate!");
```

```
spinner = (Spinner)findViewById(R.id.spinner2);
adapter = ArrayAdapter.createFromResource(this, R.array.select_destination,
                                         android.R.layout.simple_spinner_item);
```

Όπως και στις προηγούμενες περιπτώσεις έτσι και εδώ δημιουργούμε μια μέθοδο onCreate η οποία θα εκτελείται για το user interface activity\_line\_no2.xml. Μέσα στο συγκεκριμένο xml αρχείο υπάρχει ένα πτυσσόμενο μενού (spinner) που του δίνουμε με βάση τη συνάρτηση, spinner = (Spinner)findViewById(R.id,spinner2); να βρεί τη συγκεκριμένη drop down list με το id spinner2. Στη συνέχεια μέσω του Adapter δίνουμε την εντολή στον κατασκευαστή να βρεί στο αρχείο strings.xml τη λίστα με τα στοιχεία που ορίσαμε για να εμφανιστούν στο συγκεκριμένο πτυσσόμενο μενού.

```
Button buttonOk = (Button)findViewById(R.id.buttons);
buttonOk.setOnClickListener(new View.OnClickListener() {
```

**@Override**

```
public void onClick(android.view.View v) {
    // TODO Auto-generated method stub
```

```
if (spinner.getSelectedItem().toString().equals("Ταραμπούρα - Νέος Δρόμος")) {
    Intent intent = new Intent(Line_No2.this, MapsActivity.class);
    startActivity(intent);
}
```

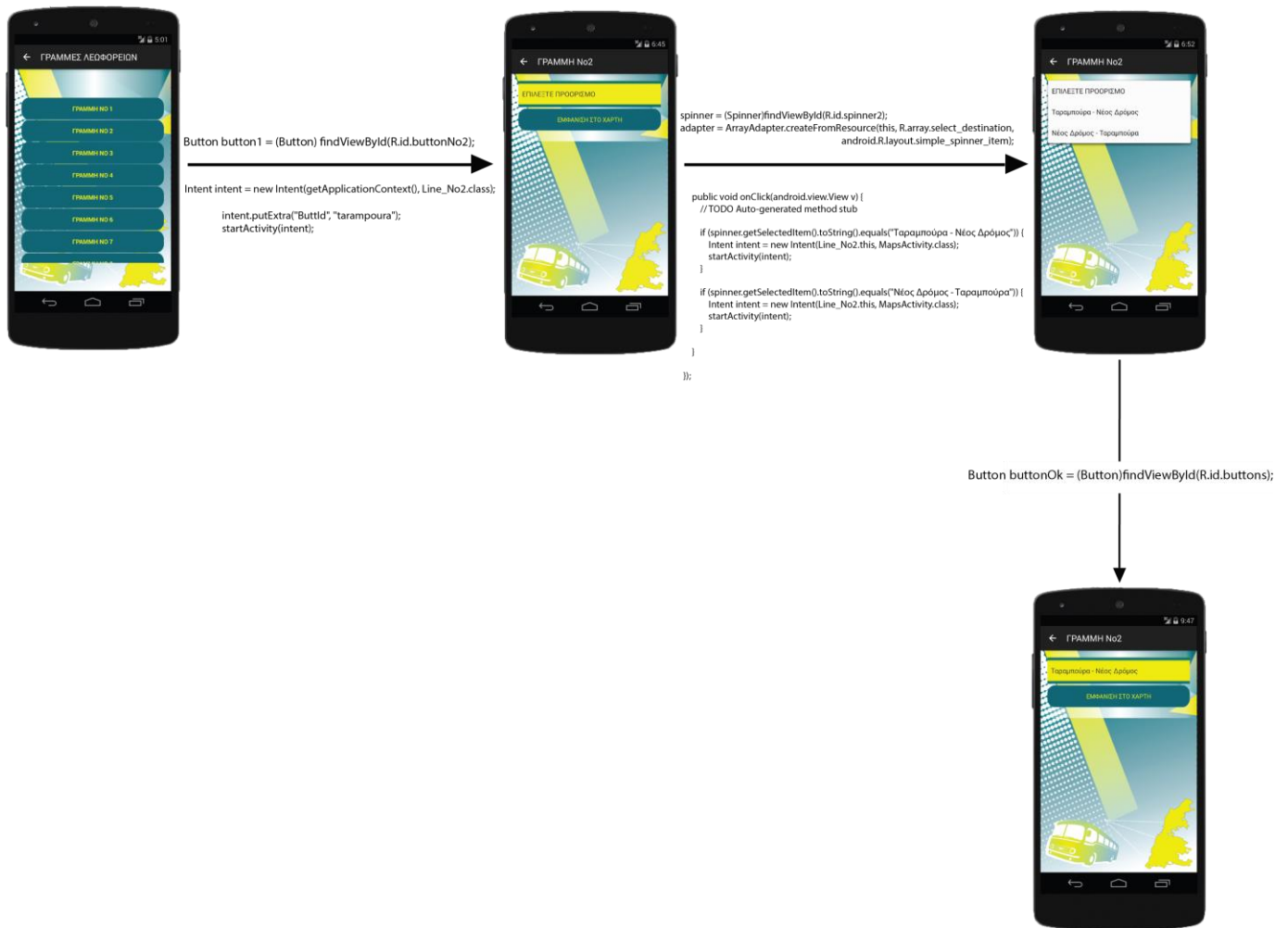


```

    if (spinner.getSelectedItem().toString().equals("Νέος Δρόμος - Ταραμπούρα")) {
        Intent intent = new Intent(Line_No2.this, MapsActivity.class);
        startActivity(intent);
    }
}
});
}
}
}

```

Εαν ο χρήστης επιλέξει από την drop down list (spinner) το string «Ταραμπούρα – Νέος Δρόμος» και πατήσει το κουμπί «ΣΥΝΕΧΕΙΑ» σύμφωνα με τη συνάρτηση, `Button buttonOk = (Button)findViewById(R.id.buttons);` τότε μέσω του Intent, η δραστηριότητα θα ζητήσει πληροφορίες από την υπάρχουσα κλάση `Line_No2.class` στην κλάση με το χάρτη (`MapsActivity.class`) για να εμφανιστεί το στίγμα του χρήστη αλλά και του λεωφορείου, όπως φαίνεται στην παρακάτω εικόνα.



*Εικόνα 51: Αναπαράσταση των συναρτήσεων για την Γραμμή No2*

## 7.3 Περιγραφή Βάσεων Δεδομένων

Η εφαρμογή για να λειτουργήσει σωστά χρησιμοποιεί μια βάση δεδομένων για την αποθήκευση των δεδομένων των αστικών λεωφορείων. Στην περίπτωση αυτή έχουμε έναν πίνακα `location` με τρεις ιδιότητες.

1. `busID` με ένα μοναδικό κλειδί για κάθε λεωφορείο.
2. `Latitude` για την πρώτη συντεταγμένη του λεωφορείου.
3. `Longitude` για τη δεύτερη συντεταγμένη του λεωφορείου.
4. `Waiting time` για το χρόνο αναμονής του λεωφορείου.

Η βάση δεδομένων χρησιμοποιείται για την καταγραφή και αποθήκευση των δεδομένων καθώς η αποθήκευση σε `xml` αρχεία δεν επιτρέπεται από το λογισμικό, καθώς τα `xml` αρχεία είναι βασικά συστατικά της δομής των εφαρμογών όπως είδαμε παραπάνω και για αποφυγή κακόβουλων παρεμβάσεων έχει απενεργοποιηθεί αυτή η δυνατότητα. Η βάση δεδομένων χρησιμοποιείται επίσης για την προσωρινή αποθήκευση των χρηστών σε μια υπηρεσία ώστε να είναι άμεσα διαθέσιμα σε περίπτωση ανάγκης επικοινωνίας ή ανταλλαγής αρχείων ή στην περίπτωσή μας για την εμφάνιση στο χάρτη της τοποθεσίας τους.

## 7.4 Εμφάνιση χρηστών στο χάρτη

Στη συνέχεια όταν ο χρήστης επιλέξει τον προορισμό που επιθυμεί να επιβιβαστεί στο λεωφορείο της γραμμής τότε πρέπει να πατήσει το κουμπί *συνέχεια*, όπως αναλύσαμε και πιο πάνω. Πατώντας το συγκεκριμένο κουμπί θα μεταβεί σε μια δραστηριότητα που θα έχει την ικανότητα να βρεί το στίγμα το δικό του αλλά και του λεωφορείου έτσι ώστε ο υπολογισμός από τις συντεταγμένες του χρήστη να είναι σωστός. Αναλυτικά έχουμε την κλάση `χάρτης` (`MapsActivity.class`) με το αντίστοιχο `xml` αρχείο για την εμφάνιση του χάρτη στη συσκευή του χρήστη.

```

package com.example.panos.buslane;

import android.os.Bundle;
import android.support.v4.app.FragmentActivity;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MapsActivity extends FragmentActivity {

    private GoogleMap mMap; // Might be null if Google Play services APK is not available.

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        setUpMapIfNeeded();
    }

    @Override
    protected void onResume() {
        super.onResume();
        setUpMapIfNeeded();
    }

    private void setUpMapIfNeeded() {
        // Do a null check to confirm that we have not already instantiated the map.
        if (mMap == null) {
            // Try to obtain the map from the SupportMapFragment.
            mMap = ((SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.map))
                .getMap();
            // Check if we were successful in obtaining the map.
            if (mMap != null) {
                setUpMap();
            }
        }
    }

    private void setUpMap() {

        LatLng greece = new LatLng(37.984062,23.728113);
        mMap.addMarker(new MarkerOptions().position(greece).title("Marker in Greece"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(greece));
        mMap.setMyLocationEnabled(true);

    }
}

```

Το αντίστοιχο xml αρχεία για την εμφάνιση του χάρτη στην οθόνη είναι το ακόλουθο:

```
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" android:id="@+id/map"
    tools:context="com.example.panos.buslane.MainActivity"
    android:name="com.google.android.gms.maps.SupportMapFragment" />
```

Εδώ πρέπει να τονίσουμε πως για να λειτουργήσει και να εμφανιστεί ο χάρτης στην οθόνη πρέπει να έχουμε πάρει από την Google ένα κλειδί API key που αφορά Google Maps, το οποίο μπορούμε να το πάρουμε δωρεάν, εφόσον δημιουργήσουμε ένα λογαριασμό στο Google Developers και συνδεθούμε με κάποιο Gmail λογαριασμό. Εφόσον ενεργοποιήσουμε το κλειδί και το εντάξουμε στο αρχείο google\_maps\_api.xml θα έχει την ακόλουθη μορφή:

```
<resources>
    <string name="google_maps_key" translatable="false"
        templateMergeStrategy="preserve">
        YOUR_KEY_HERE
    </string>
</resources>
```

Ενεργοποιώντας τη συνάρτηση `mMap.setMyLocationEnabled(true)` ενεργοποιούμαι και το κουμπί της τοποθεσίας στην οποία βρισκόμαστε. Με τον τρόπο αυτό ο χάρτης εμφανίζεται στην οθόνη του χρήστη και του παρέχετε η δυνατότητα να δει ακριβώς το στίγμα το δικό του, αλλά και του λεωφορείου της επιλογής του όπως φαίνεται παρακάτω στην εικόνα.



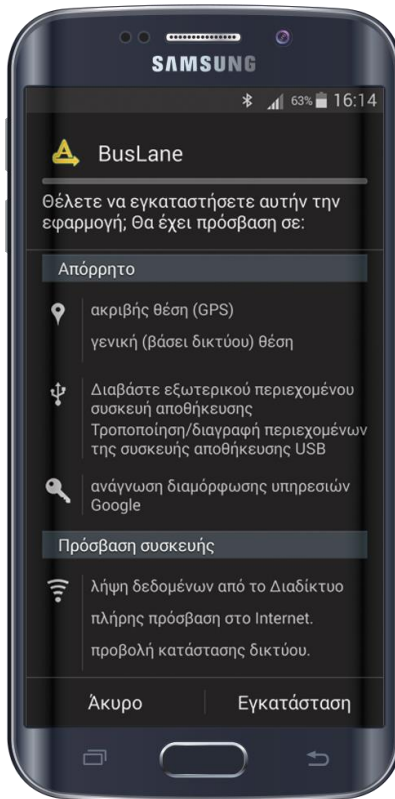
*Εικόνα 52: Αναπαράσταση συνάρτησης και εμφάνιση του χάρτη.*

## 7.5 Εφαρμογή σε smartphones

Τα smartphones για να εγκαταστήσουν μια εφαρμογή διαβάζουν ένα συγκεκριμένο αρχείο το οποίο είναι το `AndroidManifest.xml`. Το συγκεκριμένο αρχείο δηλώνει βασικές πληροφορίες σχετικά με την εφαρμογή και το σύστημα Android, που πρέπει να έχει το σύστημα για να τρέξει τον οποιοδήποτε κώδικα της εφαρμογής. Το συγκεκριμένο αρχείο δηλώνει επίσης το όνομα του πακέτου εφαρμογής της Java που χρησιμεύει ως μοναδικό αναγνωριστικό για την εφαρμογή.

Επιπλέον περιγράφει τα στοιχεία μιας αίτησης, που περιλαμβάνουν τις δραστηριότητες, τις υπηρεσίες και τους παρόχους των περιεχομένων που συνθέτουν την εφαρμογή, όπως για παράδειγμα τα `Intent`. Οι δηλώσεις αυτές ενημερώνουν το σύστημα Android από τα συστατικά και τις συνθήκες υπό τις οποίες μπορεί να ξεκινήσει.

Δηλώνει τα δικαιώματα που η αίτηση πρέπει να έχει προκειμένου να έχουν πρόσβαση σε API για να αλληλεπιδρούν με άλλες εφαρμογές και δηλώνει και τα δικαιώματα που πρέπει να έχουν οι άλλοι ώστε να αλληλεπιδρούν με τα στοιχεία της εφαρμογής, όπως για παράδειγμα φαίνεται στην παρακάτω εικόνα και τέλος δηλώνει το ελάχιστο επίπεδο του Android API που απαιτεί η εφαρμογή.



**Εικόνα 53: Εγκατάσταση εφαρμογής BusLane και δήλωση δικαιωμάτων**

Παρακάτω εμφανίζεται ο κώδικας που είναι στο αρχείο AndroidManifest.xml, που είναι απαραίτητο να υπάρχει σε μια εφαρμογή έτσι ώστε να δουλέψει σωστά.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.panos.buslane" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />
    <!--
    The ACCESS_COARSE/FINE_LOCATION permissions are not required to use
    Google Maps Android API v2, but are recommended.
    -->
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher_2"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
```

```

    android:name=".MainActivity"
    android:label="@string/app_name" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity
    android:name=".Patra"
    android:label="ΓΡΑΜΜΕΣ ΛΕΩΦΟΡΕΙΩΝ"
    android:parentActivityName=".MainActivity"
    android:screenOrientation="portrait" >
    <meta-data
        android:name="android.support.PARENT_ACTIVITY"
        android:value=".MainActivity" />
</activity>
<activity
    android:name=".Line_No1"
    android:label="ΓΡΑΜΜΗ No1"
    android:parentActivityName=".Patra"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name=".Line_No2"
    android:label="ΓΡΑΜΜΗ No2"
    android:parentActivityName=".Patra"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name=".Line_No3"
    android:label="ΓΡΑΜΜΗ No3"
    android:parentActivityName=".Patra"
    android:screenOrientation="portrait" >
</activity>

```

... «Τα υπόλοιπα activity για τις γραμμές τις υπόλοιπες»

```

<meta-data
    android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />
<meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="@string/google_maps_key" />
<activity
    android:name=".MapsActivity"
    android:label="@string/title_activity_maps"
    android:parentActivityName=".Patra"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name=".Information"
    android:label="@string/title_activity_information"
    android:parentActivityName=".MainActivity"
    android:screenOrientation="portrait" >
    <meta-data

```



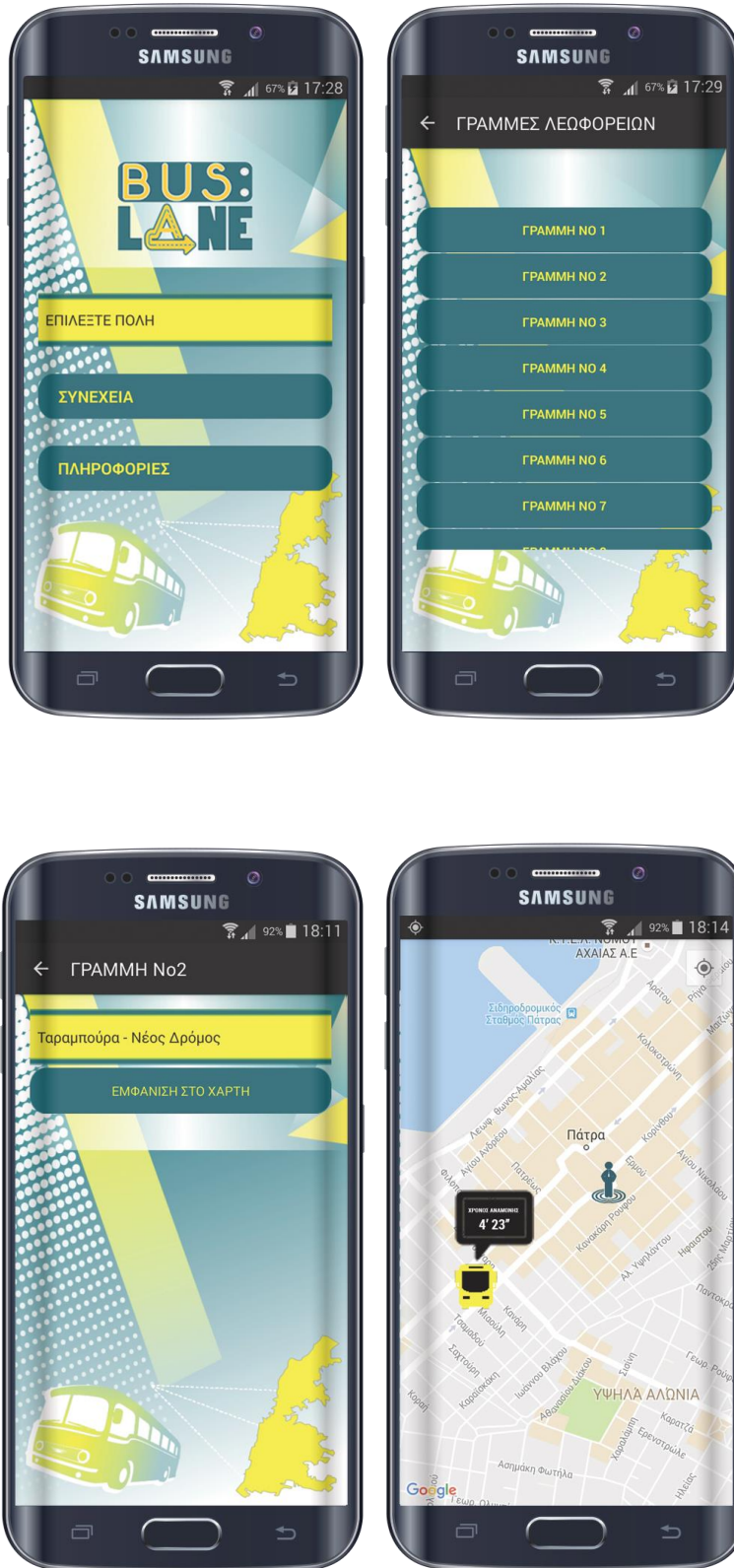
```
        android:name="android.support.PARENT_ACTIVITY"
        android:value=".MainActivity" />
    </activity>
</application>

</manifest>
```

Σχετικά με τη δήλωση των δικαιωμάτων υπάρχει περιορισμός που επιβάλλεται για την προστασία κρίσιμων δεδομένων και του κώδικα που θα μπορούσαν να χρησιμοποιηθούν για κακόβουλες λειτουργίες. Ένα χαρακτηριστικό προστατεύεται από μια μόνο άδεια. Εάν η αίτηση πρέπει να έχει πρόσβαση σε μια λειτουργία που προστατεύεται από την άδεια, θα πρέπει να δηλώσει ότι απαιτεί την άδεια με ένα <uses-permission>. Όταν η εφαρμογή έχει εγκατασταθεί στη συσκευή, το πρόγραμμα εγκατάστασης καθορίζει εάν θα χορηγήσει την αιτούμενη άδεια από τον έλεγχο που δηλώνουν τα πιστοποιητικά της εφαρμογής. Εάν χορηγείται η άδεια, η εφαρμογή είναι σε θέση να χρησιμοποιήσει τα προστατευόμενα χαρακτηριστικά. Στην περίπτωση που δε χορηγηθεί άδεια τότε η εφαρμογή δε θα έχει πρόσβαση σε αυτά τα στοιχεία. Για παράδειγμα μερικές δραστηριότητες που καλούνται να πάρουν άδεια για να έχουν πρόσβαση στο διαδίκτυο, να έχουν πρόσβαση σε χάρτη καθώς και να έχουν πρόσβαση και στην τοποθεσία του χρήστη, συντάσσονται ως εξής:

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />

<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```



**Εικόνα 54: Εφαρμογή BusLane**

### 8. Σύνοψη

Στο κεφάλαιο αυτό θα αναφέρουμε τα συμπεράσματα που προέκυψαν από την υλοποίηση της εφαρμογής καθώς και τις μελλοντικές επεκτάσεις της, με νέες ιδέες και υπηρεσίες που θα βοηθήσουν τους χρήστες της στην καλύτερη εξυπηρέτηση τους και στη λιγότερη καθημερινή ταλαιπωρία τους.

#### 8.1 Συμπεράσματα

Στη συγκεκριμένη πτυχιακή εργασία ασχοληθήκαμε με την ανάπτυξη της εφαρμογής BusLane πάνω στη βάση της πλατφόρμας, Android. Από την αρχή ξεκινήσαμε αναλύοντας σε θεωρητικό επίπεδο την εξέλιξη της τεχνολογίας. Από τα παλιά κινητά τηλέφωνα που προλάβσαμε έως τα πιο σύγχρονα smartphones και tablets. Στη συνέχεια περάσαμε σε ένα επίπεδο πιο χαμηλά για να δούμε διάφορες τεχνολογίες εντοπισμού θέσης από τα παλιά κινητά τηλέφωνα μέχρι τις πιο σύγχρονες συσκευές. Πριν προχωρήσουμε στην εφαρμογή αναλύσαμε την αρχιτεκτονική του λογισμικού Android και από ποια στάδια αποτελείται και στη συνέχεια ξεκινήσαμε να αναλύουμε την εφαρμογή σε θεωρητικό επίπεδο πριν προχωρήσουμε στον προγραμματισμό της και στην υλοποίησή της.

Καταγράψαμε όλες τις προδιαγραφές της, για το τι θα κάνει. Αναλύσαμε με διαγράμματα κλάσεων και περιπτώσεων χρήσης την εφαρμογή και όλα τα στάδιά της. Προχωρώντας σε ακόμα πιο χαμηλό επίπεδο περιγράψαμε το μοντέλο με το οποίο θα υλοποιούταν η εφαρμογή και καθώς συμφωνήσαμε για τις προδιαγραφές και τα διαγράμματα προχωρήσαμε στην υλοποίηση της εφαρμογής καθώς και στον προγραμματισμό της.

Αναπτύχθηκε το κομμάτι της εφαρμογής από την πλευρά του χρήστη και παρέμεινε σε θεωρητικό στάδιο το κομμάτι από την πλευρά του λεωφορείου. Δώσαμε βαρύτητα στην κατασκευή της από την πλευρά του χρήστη κάνοντας την εφαρμογή απλή, λειτουργική και με ένα πολύ όμορφο γραφικό περιβάλλον συνδυάζοντας χρώματα και σχέδια που δεν κουράζουν το μάτι για το χρήστη, κάνοντάς την εξαιρετικά απλή και μοντέρνα. Καταφέραμε να την κάνουμε συμβατή από την έκδοση android 4.4.2 έως και την τελευταία version android lollipop 5x.

## 8.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ ΜΕ ΝΕΕΣ ΥΠΗΡΕΣΙΕΣ

Η εφαρμογή BusLane δημιουργήθηκε για την καλύτερη εξυπηρέτηση των χρηστών που χρησιμοποιούν αστικά λεωφορεία. Είναι μια εφαρμογή που βοηθάει τους πολίτες να έχουν λιγότερη ταλαιπωρία στα μέσα μαζικής μεταφοράς. Έχει δημιουργηθεί για την πόλη της Πάτρας αλλά η βάση του είναι έτοιμη για να δεχτεί οποιαδήποτε πόλη της Ελλάδας. Μελλοντικός σκοπός της συγκεκριμένης εφαρμογής είναι ένα αυτοματοποιημένο σύστημα και δίκτυο σε πανελλαδικό επίπεδο, καθώς θα υπάρχουν όλες οι πόλεις και τα αντίστοιχα δρομολόγια τους και ο χρήστης θα μπορεί να βρίσκει το λεωφορείο που επιθυμεί για να επιβιβαστεί, σε όποια πόλη επιθυμεί στην περιοχή που βρίσκεται.

Επίσης μια νέα υπηρεσία που θα μπορούσε να συνοδεύσει την παρούσα εφαρμογή είναι, η αγορά ηλεκτρονικού εισιτηρίου. Με τη νέα αυτή υπηρεσία ο χρήστης θα έχει τη δυνατότητα να αγοράσει το εισιτήριό του, ανάλογα με το αν είναι φοιτητής, απλός πολίτης, πολίτεκνος ή οτιδήποτε άλλο, έτσι ώστε να υπάρχει και η αντίστοιχη έκπτωση. Με τα ψηφιακά εισιτήρια θα γλυτώσει ο χρήστης την ταλαιπωρία να ψάξει να βρει κάποιο διαθέσιμο περίπτερο ή σταθμό για να τα προμηθευτεί. Θα υπάρχει η δυνατότητα μέσω μιας δευτερεύουσας πλατφόρμας να αγοράζει διαδικτυακά αλλά και ψηφιακά το εισιτήριό του και να το έχει αποθηκευμένο στο smartphone του σαν ένα αρχείο ή σαν μια εικόνα που θα έχει το αντίστοιχο barcode. Όταν ο χρήστης – επιβάτης, επιβιβαστεί στο λεωφορείο τότε θα πρέπει να ανοίξει την εφαρμογή και να σαρώσει η ειδική συσκευή (scanner) που θα έχει το λεωφορείο το συγκεκριμένο barcode, έτσι ώστε να πιστοποιηθεί πως όντως ο χρήστης το έχει αγοράσει και έχει ελεγχθεί από το αντίστοιχο μηχάνημα. Το συγκεκριμένο μηχάνημα θα αποθηκεύει την ώρα και την ημερομηνία που «χτύπησε» ο χρήστης το εισιτήριό του έτσι ώστε να μην υπάρχουν παρατυπίες. Με το τρόπο αυτό ο χρήστης δε θα έχει το δικαίωμα να χρησιμοποιήσει ξανά το εισιτήριό του γιατί θα το έχει χρησιμοποιήσει ήδη μια φορά.

Με τον τρόπο αυτό δημιουργείτε ένα αυτοματοποιημένο σύστημα σε πανελλαδικό επίπεδο που οι επιβάτες – χρήστες – πολίτες θα έχουν τη δυνατότητα με μια μόνο εφαρμογή να επιλέγουν την πόλη και τα δρομολόγια που θέλουν να επιβιβαστούν να βλέπουν τα διαθέσιμα δρομολόγια και πληρώνουν το εισιτήριό τους ηλεκτρονικά.

# Παράρτημα

Εικόνα 1. DELL X50 PDA και τεχνικά χαρακτηριστικά .....	8
Εικόνα 2. Acer n35 GPS PDA .....	8
Εικόνα 3. Κάρτα SIM (Subscriber Identity Module) .....	11
Εικόνα 4. Κινητό τηλέφωνο NOKIA 3310 .....	13
Εικόνα 5. Κινητό τηλέφωνο Sony Ericsson W600 .....	13
Εικόνα 6. Android Lollipop launcher 5.0 (2014) .....	15
Εικόνα 7. iOS 8.3 launcher (2015) .....	16
Εικόνα 8. Windows Phone 8.1 launcher (2014) .....	17
Εικόνα 9. BlackBerry launcher .....	18
Εικόνα 10. iPhone 6 & iPhone 6 plus .....	22
Εικόνα 11. Samsung Galaxy S6 Edge .....	22
Εικόνα 12. Nokia Lumia 730 .....	23
Εικόνα 13. User Interface σε tablet με λειτουργικό σύστημα Android .....	29
Εικόνα 14. User Interface σε iPad με λειτουργικό σύστημα iOS 8 .....	30
Εικόνα 15. User Interface σε tablet με λειτουργικό σύστημα Windows 8 .....	31
Εικόνα 16. iPad mini .....	32
Εικόνα 17. Samsung Galaxy Tab 3 .....	32
Εικόνα 18. Αναπαράσταση σχεδίου του συστήματος GPS .....	35
Εικόνα 19. Συσκευή GPS .....	36
Εικόνα 20. GSM (Global System for Mobile communications) .....	39
Εικόνα 21. Αρχιτεκτονικό διάγραμμα του Android .....	43
Εικόνα 22. Εγκατάσταση Android SDK με Android Studio .....	48
Εικόνα 23. Άνοιγμα emulator στο Android Studio .....	48
Εικόνα 24. Σύμβολο κλάσης .....	56
Εικόνα 25. Πάνω από μια συσχετίσεις μεταξύ των κλάσεων .....	58
Εικόνα 26. Σύμβολο γενίκευσης .....	59
Εικόνα 27. Σύμβολο συναρμολόγησης .....	59
Εικόνα 28. Σύμβολο σύνθεσης .....	59
Εικόνα 29. Σύμβολο περίπτωσης χρήσης .....	60
Εικόνα 30. Σύμβολο χειριστή .....	61
Εικόνα 31. Πλαίσιο συστήματος διαγράμματος περιπτώσεων χρήσης .....	61
Εικόνα 32. Σχέση χειριστή με περιπτώσεις χρήσης .....	62
Εικόνα 33. Σχέση συμπερίληψης μεταξύ περιπτώσεων χρήσης .....	63
Εικόνα 34. Σχέση επέκτασης μεταξύ περιπτώσεων χρήσης .....	64
Εικόνα 35. Αναπαράσταση σε screenshot στο άνοιγμα της εφαρμογής .....	67
Εικόνα 36. Διάγραμμα κλάσεων .....	68
Εικόνα 37. Διάγραμμα περίπτωσης χρήσης .....	69
Εικόνα 38. Αναπαράσταση σε screenshot, η εφαρμογή BusLane τη στιγμή που ο χρήστης επιλέγει πόλη .....	70
Εικόνα 39. Αναπαράσταση σε screenshot τη στιγμή που ο χρήστης επιλέξει γραμμή λεωφορείου .....	71
Εικόνα 40. Αναπαράσταση σε screenshot τη στιγμή που ο χρήστης επιλέγει προορισμό .....	72

Εικόνα 41. Αναπαράσταση σε screenshot τη στιγμή που ο χρήστης περιμένει το λεοφορείο της επιλογής του .....	73
Εικόνα 42. Το μοντέλο της πρωτοτυποποίησης .....	74
Εικόνα 43. Screenshot του Android Studio .....	77
Εικόνα 44. Screenshot του προγράμματος Android Studio .....	77
Εικόνα 45. Σχεδιασμός user interface από το πρόγραμμα Android Studio .....	81
Εικόνα 46. User Interface της εφαρμογής BusLane και απεικόνιση των κουμπιών «ΣΥΝΕΧΕΙΑ» και «ΠΛΗΡΟΦΟΡΙΕΣ» .....	81
Εικόνα 47. Αναπαράσταση των συναρτήσεων της εφαρμογής .....	86
Εικόνα 48. User Interface της κλάσης Πάτρας από το πρόγραμμα Android Studio .....	91
Εικόνα 49. Αναπαράσταση της κλάσης Patra.class με τη δραστηριότητα Line_No2.class .....	94
Εικόνα 50. Υλοποίηση παραδείγματος για τη γραμμή Νο2 .....	97
Εικόνα 51. Αναπαράσταση των συναρτήσεων για τη Γραμμή Νο2 .....	100
Εικόνα 52. Αναπαράσταση συνάρτησης και εμφάνισης του χρήστη .....	104
Εικόνα 53. Εγκατάσταση εφαρμογής BusLane και δήλωση δικαιωμάτων .....	105
Εικόνα 54. Εφαρμογή BusLane .....	108

# Βιβλιογραφία

---

- [1] Personal Digital Assistans (PDA)  
<http://searchmobilecomputing.techtarget.com/definition/personal-digital-assistant>
- [2] Personal Digital Assistans (PDA)  
[https://en.wikipedia.org/wiki/Personal\\_digital\\_assistant](https://en.wikipedia.org/wiki/Personal_digital_assistant)
- [3] Κινητό Τηλέφωνο  
[https://el.wikipedia.org/wiki/Κινητό\\_τηλέφωνο](https://el.wikipedia.org/wiki/Κινητό_τηλέφωνο)
- [4] Radiolinja Company  
<https://en.wikipedia.org/wiki/Radiolinja>
- [5] GSM (Global System for Mobile Communications)  
<http://en.wikipedia.org/wiki/GSM>
- [6] Code Division Multiple Access (CDMA)  
[https://en.wikipedia.org/wiki/Code\\_division\\_multiple\\_access](https://en.wikipedia.org/wiki/Code_division_multiple_access)
- [7] Removable User Identity Module (R-UIM)  
[https://en.wikipedia.org/wiki/Removable\\_User\\_Identity\\_Module](https://en.wikipedia.org/wiki/Removable_User_Identity_Module)
- [8] M-PESA Mobile Banking  
<https://en.wikipedia.org/wiki/M-Pesa>
- [9] Mobile Phone  
[https://en.wikipedia.org/wiki/Mobile\\_phone](https://en.wikipedia.org/wiki/Mobile_phone)
- [10] Smartphone  
<http://el.wikipedia.org/wiki/Smartphone>
- [11] Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)  
[https://el.wikipedia.org/wiki/Κεντρική\\_Μονάδα\\_Επεξεργασίας](https://el.wikipedia.org/wiki/Κεντρική_Μονάδα_Επεξεργασίας)
- [12] NPD Group  
[https://en.wikipedia.org/wiki/The\\_NPD\\_Group](https://en.wikipedia.org/wiki/The_NPD_Group)
- [13] Microsoft Metro (Design Language)  
[https://en.wikipedia.org/wiki/Metro\\_\(design\\_language\)](https://en.wikipedia.org/wiki/Metro_(design_language))
- [14] Smartphone  
<https://simple.wikipedia.org/wiki/Smartphone>

- [15] Υπολογιστής Ταμπλέτα  
[https://el.wikipedia.org/wiki/Υπολογιστής\\_ταμπλέτα](https://el.wikipedia.org/wiki/Υπολογιστής_ταμπλέτα)
- [16] Telautograph  
<https://en.wikipedia.org/wiki/Telautograph>
- [17] EPOC (Operating system)  
[http://en.wikipedia.org/wiki/EPOC\\_\(operating\\_system\)](http://en.wikipedia.org/wiki/EPOC_(operating_system))
- [18] Resistive Touchscreen  
[http://en.wikipedia.org/wiki/Resistive\\_touchscreen](http://en.wikipedia.org/wiki/Resistive_touchscreen)
- [19] Capacitive Sensing  
[https://en.wikipedia.org/wiki/Capacitive\\_sensing](https://en.wikipedia.org/wiki/Capacitive_sensing)
- [20] ISO/IEC 14443  
[https://en.wikipedia.org/wiki/ISO/IEC\\_14443](https://en.wikipedia.org/wiki/ISO/IEC_14443)
- [21] JooJoo Tablet  
<http://en.wikipedia.org/wiki/JooJoo>
- [22] Intel Atom (CPU)  
[http://en.wikipedia.org/wiki/Intel\\_Atom\\_\(CPU\)](http://en.wikipedia.org/wiki/Intel_Atom_(CPU))
- [23] Windows for Pen Computing  
[https://en.wikipedia.org/wiki/Windows\\_for\\_Pen\\_Computing](https://en.wikipedia.org/wiki/Windows_for_Pen_Computing)
- [24] Tablet Computer  
[http://en.wikipedia.org/wiki/Tablet\\_computer](http://en.wikipedia.org/wiki/Tablet_computer)
- [25] Software platform Maemo  
<https://en.wikipedia.org/wiki/Maemo>
- [26] Λειτουργικό σύστημα MeeGo  
<https://el.wikipedia.org/wiki/MeeGo>
- [27] Μέσα επικοινωνίας δι' εντοπισμού  
[https://el.wikipedia.org/wiki/Μέσα\\_επικοινωνίας\\_δι'\\_εντοπισμού](https://el.wikipedia.org/wiki/Μέσα_επικοινωνίας_δι'_εντοπισμού)
- [28] Global Positioning System (GPS)  
[https://el.wikipedia.org/wiki/Global\\_Positioning\\_System](https://el.wikipedia.org/wiki/Global_Positioning_System)



- [29] Global System for Mobile Communications  
[https://el.wikipedia.org/wiki/Global\\_System\\_for\\_Mobile\\_Communications](https://el.wikipedia.org/wiki/Global_System_for_Mobile_Communications)
- [30] Location based service  
[https://en.wikipedia.org/wiki/Location-based\\_service](https://en.wikipedia.org/wiki/Location-based_service)
- [31] Bluetooth  
<https://el.wikipedia.org/wiki/Bluetooth>
- [32] Android operating system  
[https://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [33] ΣΤΥΛΙΑΝΟΣ ΒΟΥΚΑΤΑΣ, “Το λειτουργικό σύστημα Android και η υλοποίησ εφαρμογής με το Google SDK”  
[http://xsomaras.somweb.gr/wp-content/uploads/2011\\_voukatas.pdf](http://xsomaras.somweb.gr/wp-content/uploads/2011_voukatas.pdf)
- [34] Android Application Componets  
[http://www.tutorialspoint.com/android/android\\_application\\_components.htm](http://www.tutorialspoint.com/android/android_application_components.htm)
- [35] Broadcast Receiver  
<http://stackoverflow.com/questions/5296987/what-is-broadcastreceiver-and-when-we-use-it>
- [36] Android Content Provider  
<http://stackoverflow.com/tags/android-contentprovider/info>
- [37] ΜΟΥΤΖΟΥΡΙΔΗΣ ΓΕΩΡΓΙΟΣ, “Ανάπτυξη mobile εφαρμογών και services στην πλατφόρμα Android”  
[http://eureka.lib.teithe.gr:8080/bitstream/handle/10184/7306/Moutzouridis\\_Georgios.pdf?sequence=3](http://eureka.lib.teithe.gr:8080/bitstream/handle/10184/7306/Moutzouridis_Georgios.pdf?sequence=3)
- [38] Β. Γερογιάννης, Γ. Κακαρόντζας, Α. Καμέας, Γ. Σταμέλος, Π. Φιτσιλής,  
"Αντικειμενοστρεφής Ανάπτυξη Λογισμικού με τη UML"
- [39] Ian Sommerville, "Βασικές αρχές Τεχνολογίας Λογισμικού" 2009
- [40] Android Intents  
<http://www.vogella.com/tutorials/AndroidIntent/article.html>
- [41] Using an ArrayAdapter with ListView  
[https://www.github.com/codepath/android\\_guides/wiki/Using-an-ArrayAdapter-with-ListView](https://www.github.com/codepath/android_guides/wiki/Using-an-ArrayAdapter-with-ListView)
- [42] App Manifest  
<https://developer.android.com/guide/topics/manifest/manifest-intro.html>

[43] Widget Spinner

<https://developer.android.com/reference/android/widget/Spinner.html>

[44] Difference between BaseAdapter and ArrayAdapter

<http://stackoverflow.com/questions/16796182/whats-the-difference-between-baseadapter-and-arrayadapter>

[45] Google Maps API for Android

<https://developers.google.com/maps/documentation/android-api/>

[46] ΓΕΩΡΓΙΟΣ ΠΑΝΟΥ ΠΑΠΑΔΗΜΗΤΡΙΟΥ “Ανάπτυξη Υπηρεσιών με βάση τη θέση του χρήστη, με το λογισμικό Android” , Αθήνα Φεβρουάριος 2010