



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**“Σύστημα διαχείρισης μετακινήσεων στα ΜΜΜ (Μέσα  
Μαζικής Μετακίνησης) με ειδοποίηση”**



Του φοιτητή

Λαμπρόπουλος Αλέξανδρος

Αρ. Μητρώου: 08/3319

Επιβλέπων καθηγητής

Ιγνάτιος Δεληγιάννης

Θεσσαλονίκη 2015

## ΠΡΟΛΟΓΟΣ

Μετά από ένα εξάμηνο εργασίας, μια εφαρμογή διαχείρισης μετακινήσεων στα ΜΜΜ για το λειτουργικό σύστημα Android έχει ολοκληρωθεί. Παρέχει στο χρήστη διαφορετικές δυνατότητες από άλλες υπάρχουσες εφαρμογές όπως ο καθημερινός προγραμματισμός διαδρομών με το λεωφορείο και η έγκαιρη ειδοποίησή του για την άφιξη αυτού.

Μέσα από αυτό το project μου δόθηκε η ευκαιρία να εξοικειωθώ με τον προγραμματισμό σε Java, την πλατφόρμα Android, την SQLite βάση δεδομένων και τη χρήση υπηρεσιών ιστού (web services) από τη θεωρία στην πράξη. Αυτές οι τεχνολογίες με βοήθησαν να κατανοήσω τον τρόπο ανάπτυξης μιας εφαρμογής καθώς και να την υλοποιήσω. Το μάθημα “Μηχανική Λογισμικού” το οποίο και διδάχτηκα στη σχολή με βοήθησε στη σχεδίαση της εφαρμογής κυρίως μέσω της χρήσης διαγραμμάτων (διάγραμμα ακολουθίας, διάγραμμα κλάσεων, διάγραμμα συστατικών, διάγραμμα παράταξης).

### **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Ιγνάτιο Δεληγιάννη για την υπομονή του, τη συμβολή και τις ιδέες του καθ' όλη τη διάρκεια της πτυχιακής αυτής εργασίας. Τους φίλους μου που με βοήθησαν να τεστάρω την εφαρμογή, μου έδωσαν νέες ιδέες σχεδιασμού και λειτουργικότητας καθώς και όσους συνέβαλαν στην πτυχιακή αυτή εργασία.

## ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ .....	2
ΕΥΧΑΡΙΣΤΙΕΣ.....	3
ΠΕΡΙΕΧΟΜΕΝΑ .....	4
Ευρετήριο σχημάτων .....	7
Ευρετήριο πινάκων .....	8
Ευρετήριο Παραρτημάτων.....	9
Ευρετήριο Εικόνων Οδηγού Χρήσης Λογισμικού .....	9
ΕΙΣΑΓΩΓΗ .....	11
ΚΕΦΑΛΑΙΟ 1 .....	13
Εισαγωγή στο λειτουργικό σύστημα Android.....	13
1.1 ΕΙΣΑΓΩΓΗ .....	13
1.2 Ιστορικά.....	13
1.3 Εφαρμογές .....	15
1.4 Art & Dalvik .....	15
1.5 Αρχιτεκτονική.....	16
1.5.1 Επίπεδο Εφαρμογών (Blue layer - Application) .....	16
1.5.2 Επίπεδο Πλαισίου Εφαρμογών (Blue layer - Application Framework).....	16
1.5.3 Επίπεδο Βιβλιοθηκών (Green layer - Libraries) .....	17
1.5.4 Επίπεδο Εκτέλεσης (Green layer - Android Runtime).....	17
1.5.5 Ο Πυρήνας του Linux (Red Layer) .....	17
1.6 ΕΠΙΛΟΓΟΣ .....	17
ΚΕΦΑΛΑΙΟ 2 .....	18
Βασικά δομικά στοιχεία εφαρμογών Android.....	18
2.1 ΕΙΣΑΓΩΓΗ .....	18
2.2 Δραστηριότητα (Activity) .....	18
2.2.1 Κύκλος ζωής Δραστηριότητας.....	18
2.3 Τεμάχιο (Fragment).....	20
2.3.1 Κύκλος ζωής Τεμαχίου (Fragment) .....	20
2.3.2 Φιλοσοφία σχεδιασμού .....	22
2.4 Υπηρεσία (Service) .....	23
2.4.1 Μορφές Υπηρεσίας (Service) .....	23

2.4.2	Ειδοποίηση μέσω Υπηρεσίας (Service).....	24
2.5	Ειδοποίηση (Notification) .....	25
2.5.1	Συστατικά Ειδοποιήσεων (Notification).....	26
2.6	Προσαρμογέας (Adapter).....	26
2.7	Βάση Δεδομένων (SQLite).....	27
2.7.1	Η κλάση SQLiteOpenHelper .....	27
2.7.2	Αποστολή ερωτημάτων κα διαχείριση αποτελεσμάτων .....	28
2.7.3	Κριτήρια επιλογής και ασφάλεια.....	28
2.8	AndroidManifest .....	28
2.9	ΕΠΙΛΟΓΟΣ .....	29
ΚΕΦΑΛΑΙΟ 3 .....		30
Ασφάλεια στο λειτουργικό Android.....		30
3.1	ΕΙΣΑΓΩΓΗ .....	30
3.2	Χρήση εσωτερικού χώρου αποθήκευσης .....	30
3.3	Χρήση Δικαιωμάτων .....	30
3.4	Αίτηση Δικαιωμάτων.....	31
3.5	Χρήση Υπηρεσιών (Services).....	31
3.6	ΕΠΙΛΟΓΟΣ .....	31
ΚΕΦΑΛΑΙΟ 4 .....		32
Ο κύκλος ζωής μιας εφαρμογής Android.....		32
4.1	ΕΙΣΑΓΩΓΗ .....	32
ΚΕΦΑΛΑΙΟ 5 .....		33
Υπηρεσίες Google και Android APIs.....		33
5.1	ΕΙΣΑΓΩΓΗ .....	33
5.2	Χάρτες Google .....	34
5.3	Τοποθεσία Android (Android location API).....	34
5.4	Geocoder .....	34
5.5	Εγκατάσταση του Google Play Service (Eclipse IDE) .....	34
5.6	Δημιουργία API Project στη κονσόλα Google APIs .....	35
5.7	Απόκτηση Πιστοποιητικού fingerprint SHA1 (Eclipse IDE) .....	35
5.8	Απόκτηση κλειδιού Google Maps API .....	35
5.9	Google Maps Android API v2.....	36
5.9.1	Χρήση χαρτών Google .....	36
5.10	Web Service.....	37

5.10.1	Google Maps API Web Services.....	37
5.10.2	To Google Directions API.....	37
5.10.3	Όρια Χρήσης.....	37
5.10.4	Παράμετροι Αιτήσεων .....	38
5.10.5	Οδηγίες Συγκοινωνίας.....	39
5.11	ΕΠΙΛΟΓΟΣ .....	39
ΚΕΦΑΛΑΙΟ 6 .....		40
Βιβλιοθήκες Υποστήριξης Android.....		40
6.1	ΕΙΣΑΓΩΓΗ .....	40
6.2	v7 Support Libraries .....	40
6.2.1	v7 Appcompat Library .....	40
6.2.2	Εισαγωγή βιβλιοθήκης στο Project.....	41
ΚΕΦΑΛΑΙΟ 7 .....		42
Η Εφαρμογή PlanMyDay .....		42
7.1	ΕΙΣΑΓΩΓΗ .....	42
7.2	Πως λειτουργεί.....	42
7.3	Συστατικά Στοιχεία της Εφαρμογής .....	43
7.3.1	Τεμάχιο (Fragment).....	43
7.3.2	Ειδοποίηση (Notification) .....	43
7.3.3	Προσαρμογέας (Adapter).....	43
7.3.4	Βάση Δεδομένων (SQLite).....	43
7.4	Χρήση Υπηρεσιών ΟΑΣΘ .....	44
7.5	Περιπτώσεις χρήσης .....	44
7.5.1	Βασικά συστατικά ενός διαγράμματος περιπτώσεων χρήσης .....	45
7.5.2	Σχέσεις περιπτώσεων χρήσης .....	45
7.5.3	Διάγραμμα περιπτώσεων χρήσης της εφαρμογής .....	46
7.5.4	Σενάρια περιπτώσεων χρήσης .....	48
7.5.5	Πίνακες σεναρίων Περιπτώσεων Χρήσης.....	49
7.6	Διαγράμματα κλάσεων .....	74
7.6.1	Διάγραμμα κλάσεων της εφαρμογής .....	75
7.7	Διαγράμματα ακολουθίας .....	77
7.7.1	Διαγράμματα ακολουθίας της εφαρμογής.....	77
7.8	Διαγράμματα πακέτων.....	84
7.9	Διαγράμματα συστατικών.....	84

7.9.1	Διάγραμμα συστατικών της εφαρμογής.....	85
7.10	Διαγράμματα παράταξης.....	86
7.10.1	Διάγραμμα παράταξης της εφαρμογής.....	86
7.11	Διάγραμμα Βάσης Δεδομένων.....	87
7.12	Χρήση αποθετηρίου.....	88
7.13	Πρόγραμμα δημιουργίας Βάσης Δεδομένων ΟΑΣΘ.....	88
7.13.1	Πως λειτουργεί.....	88
7.13.2	Βάση Δεδομένων ΟΑΣΘ.....	90
7.14	Το Project της Εφαρμογής.....	92
7.14.1	Πακέτα.....	92
7.15	Εξαγωγή εφαρμογής (Eclipse IDE).....	99
7.16	Εγκατάσταση της Εφαρμογής.....	99
	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	102
	ΑΝΑΦΟΡΕΣ.....	103
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	103
	ΠΑΡΑΡΤΗΜΑΤΑ.....	106
	ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ.....	107

## Ευρετήριο σχημάτων

Σχήμα 1	Κατανομή ώρας που ξοδεύουν οι χρήστες, σε λειτουργικά Android και iOS (9).....	12
Σχήμα 2	Πωλήσεις Λειτουργικών Συστημάτων 2011 – 2014 (8).....	13
Σχήμα 3	Ποσοστό Διανομών Εκδόσεων Android (4).....	14
Σχήμα 4	Στοίβα λογισμικού Android (6).....	16
Σχήμα 5	Κύκλος ζωής Δραστηριότητας (Activity) (1).....	19
Σχήμα 6	Κύκλος ζωής Τεμαχίου (Fragment), καθώς η Δραστηριότητα (Activity) τρέχει (2).....	21
Σχήμα 7	Κύκλος ζωής Υπηρεσίας (Service) (5).....	24
Σχήμα 8	Ειδοποίηση (Notification) στην περιοχή ειδοποιήσεων (3).....	25
Σχήμα 9	Ειδοποίηση στο συρτάρι ειδοποιήσεων.....	25
Σχήμα 10	διάγραμμα περιπτώσεων χρήσης.....	47
Σχήμα 11	διάγραμμα κλάσεων.....	76
Σχήμα 12	διάγραμμα ακολουθίας – Ημερήσιες Διαδρομές.....	78
Σχήμα 13	διάγραμμα ακολουθίας – Αγαπημένες Διαδρομές.....	79
Σχήμα 14	διάγραμμα ακολουθίας – Πρόσφατες Διαδρομές.....	80
Σχήμα 15	διάγραμμα ακολουθίας – Νέα Διαδρομή.....	81
Σχήμα 16	διάγραμμα ακολουθίας – Πλάγιο Μενού.....	82
Σχήμα 17	διάγραμμα ακολουθίας – Create NewRouteActivity.....	83

Σχήμα 18 διάγραμμα ακολουθίας – MakeRoute.....	84
Σχήμα 19 διάγραμμα συστατικών .....	85
Σχήμα 20 διάγραμμα παράταξης.....	86
Σχήμα 21 βάση δεδομένων Διαδρομών του Χρήστη.....	87
Σχήμα 22 Βάση Δεδομένων ΟΑΣΘ .....	90
Σχήμα 23 στιγμιότυπο πίνακα busstop.....	91
Σχήμα 24 στιγμιότυπο πίνακα busline .....	91
Σχήμα 25 δομή του Project .....	92
Σχήμα 26 πόροι εφαρμογής.....	96
Σχήμα 27 αρχείο AndroidManifest.xml.....	99
Σχήμα 28 άνοιγμα αρχείου APK.....	99
Σχήμα 29 αποκλεισμός εγκατάστασης λόγω άγνωστων πηγών.....	100
Σχήμα 30 ενεργοποίηση αγνώστων πηγών .....	100
Σχήμα 31 άδειες εφαρμογής.....	101
Σχήμα 32 αποδοχή ελέγχου Google.....	101

## Ευρετήριο πινάκων

Πίνακας 1 Εκδόσεις Λειτουργικού Συστήματος Android.....	14
Πίνακας 2 Μέθοδοι Δραστηριότητας (Activity).....	20
Πίνακας 3 Μέθοδοι Τεμαχίου (Fragment), αλληλεπίδρασης με τον χρήστη.....	22
Πίνακας 4 Μέθοδοι Τεμαχίου (Fragment), όταν δεν χρησιμοποιείται πλέον .....	22
Πίνακας 5 Μέθοδοι Προσαρμογέα (Adapter) .....	26
Πίνακας 6 Κύρια μέθοδος Βασικού Προσαρμογέα (BaseAdapter) .....	27
Πίνακας 7 Μέθοδοι κλάσης SQLiteOpenHelper .....	28
Πίνακας 8 παράμετροι αιτήσεων.....	38
Πίνακας 9 πεδία αντικειμένου συγκοινωνιών.....	39
Πίνακας 10 κλάσεις της βιβλιοθήκης v7 Appcompat .....	40
Πίνακας 11 Σενάριο χρήσης - Διαχείριση Διαδρομών Ημέρας .....	50
Πίνακας 12 Σενάριο χρήσης - Επιλογή υπάρχουσας διαδρομής.....	51
Πίνακας 13 Σενάριο χρήσης - Προσθήκη/Αφαίρεση στα Αγαπημένα.....	51
Πίνακας 14 Σενάριο χρήσης - Διαγραφή διαδρομής.....	52
Πίνακας 15 Σενάριο χρήσης - Ειδοποίηση διαδρομής .....	53
Πίνακας 16 Σενάριο χρήσης - Επιλογή εβδομαδιαίας επανάληψης.....	55
Πίνακας 17 Σενάριο χρήσης - Επιλογή μηνιαίας επανάληψης.....	56
Πίνακας 18 Σενάριο χρήσης - Δημιουργία νέας διαδρομής.....	57
Πίνακας 19 Σενάριο χρήσης - Επιλογή διαδρομής με σημεία.....	58
Πίνακας 20 Σενάριο χρήσης - Επιλογή βάση τοποθεσίας χρήστη.....	60
Πίνακας 21 Σενάριο χρήσης - Αποθήκευση διαδρομής .....	62
Πίνακας 22 Σενάριο χρήσης - Επιλογή διαδρομής με διευθύνσεις.....	63
Πίνακας 23 Σενάριο χρήσης - Διαχείριση Αγαπημένων Διαδρομών.....	65
Πίνακας 24 Σενάριο χρήσης - Επιλογή αγαπημένης διαδρομής .....	66
Πίνακας 25 Σενάριο χρήσης - Διαγραφή από τα Αγαπημένα.....	67
Πίνακας 26 Σενάριο χρήσης - Εισαγωγή διαδρομής .....	68



Πίνακας 27 Σενάριο χρήσης - Εισαγωγή επιλεγμένης διαδρομής σήμερα .....	68
Πίνακας 28 Σενάριο χρήσης - Εισαγωγή διαδρομής αύριο .....	69
Πίνακας 29 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής με ίδια σημεία εκκίνησης, τερματισμού .....	70
Πίνακας 30 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής σήμερα.....	71
Πίνακας 31 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής αύριο.....	72
Πίνακας 32 Σενάριο χρήσης - Διαχείριση Πρόσφατων Διαδρομών .....	73
Πίνακας 33 Σενάριο χρήσης - Επιλογή πρόσφατης διαδρομής.....	74
Πίνακας 34 συμβολισμός αντικειμένων σε διάγραμμα κλάσεων .....	75
Πίνακας 35 συμβολισμός αντικειμένων σε διάγραμμα κλάσεων .....	77
Πίνακας 36 περιγραφή των κλάσεων του προγράμματος ΟΑΣΘ που δημιουργήθηκε .....	88
Πίνακας 37 περιγραφή των παραμέτρων στο URL της ιστοσελίδας ΟΑΣΘ.....	89
Πίνακας 38 κλάσεις που ανήκουν στο πακέτο Δραστηριοτήτων .....	93
Πίνακας 39 κλάσεις που ανήκουν στο πακέτο Προσαρμογών .....	93
Πίνακας 40 κλάσεις που ανήκουν στο πακέτο Υπηρεσιών.....	94
Πίνακας 41 κλάσεις που ανήκουν στο πακέτο Βάσεων Δεδομένων .....	94
Πίνακας 42 κλάσεις που ανήκουν στο πακέτο Αντικειμένων.....	94
Πίνακας 43 κλάσεις που ανήκουν στο πακέτο του πλάγιου μενού .....	95
Πίνακας 44 κλάσεις που ανήκουν στο πακέτο χρήσιμα.....	95
Πίνακας 45 πόροι εφαρμογής.....	96

## Ευρετήριο Παραρτημάτων

Παράρτημα 1 άδεια χρήσης δεδομένων ΟΑΣΘ.....	106
--	-----

## Ευρετήριο Εικόνων Οδηγού Χρήσης Λογισμικού

Εικόνα 1 άνοιγμα εφαρμογής.....	107
Εικόνα 2 Splash Screen.....	107
Εικόνα 3 Διαχείριση Ημερήσιων Διαδρομών .....	108
Εικόνα 4 Πλάγιο Μενού .....	108
Εικόνα 5 Αγαπημένα .....	109
Εικόνα 6 Πρόσφατα .....	109
Εικόνα 7 διαχείριση πρόσφατης επιλεγμένης διαδρομής .....	110
Εικόνα 8 χρήση σημείων (αφετηρία - τέρμα), για δημιουργία νέας διαδρομής .....	110
Εικόνα 9 διαχείριση επιλεγμένης αγαπημένης διαδρομής.....	111
Εικόνα 10 επιλογή ημέρας.....	111
Εικόνα 11 επιλογή νέας διαδρομής.....	112
Εικόνα 12 επιλογή ώρας άφιξης .....	113
Εικόνα 13 χρήση σημείων στο χάρτη.....	113
Εικόνα 14 χρήση διευθύνσεων .....	114
Εικόνα 15 χρήση τοποθεσίας.....	114

Εικόνα 16 σχεδίαση διαδρομής στο χάρτη.....	115
Εικόνα 17 αποθήκευση διαδρομής .....	115
Εικόνα 18 αποθηκευμένη διαδρομή .....	116
Εικόνα 19 εβδομαδιαία επανάληψη .....	116
Εικόνα 20 μηνιαία επανάληψη .....	117
Εικόνα 21 μενού διαχείρισης επιλεγμένης διαδρομής .....	117
Εικόνα 22 προσθήκη διαδρομής στα αγαπημένα .....	118
Εικόνα 23 διαγραφή επιλεγμένης διαδρομής.....	118
Εικόνα 24 ενεργοποίηση ειδοποίησης επιλεγμένης διαδρομής.....	119
Εικόνα 25 η ειδοποίηση στο συρτάρι ειδοποιήσεων .....	119

## ΕΙΣΑΓΩΓΗ

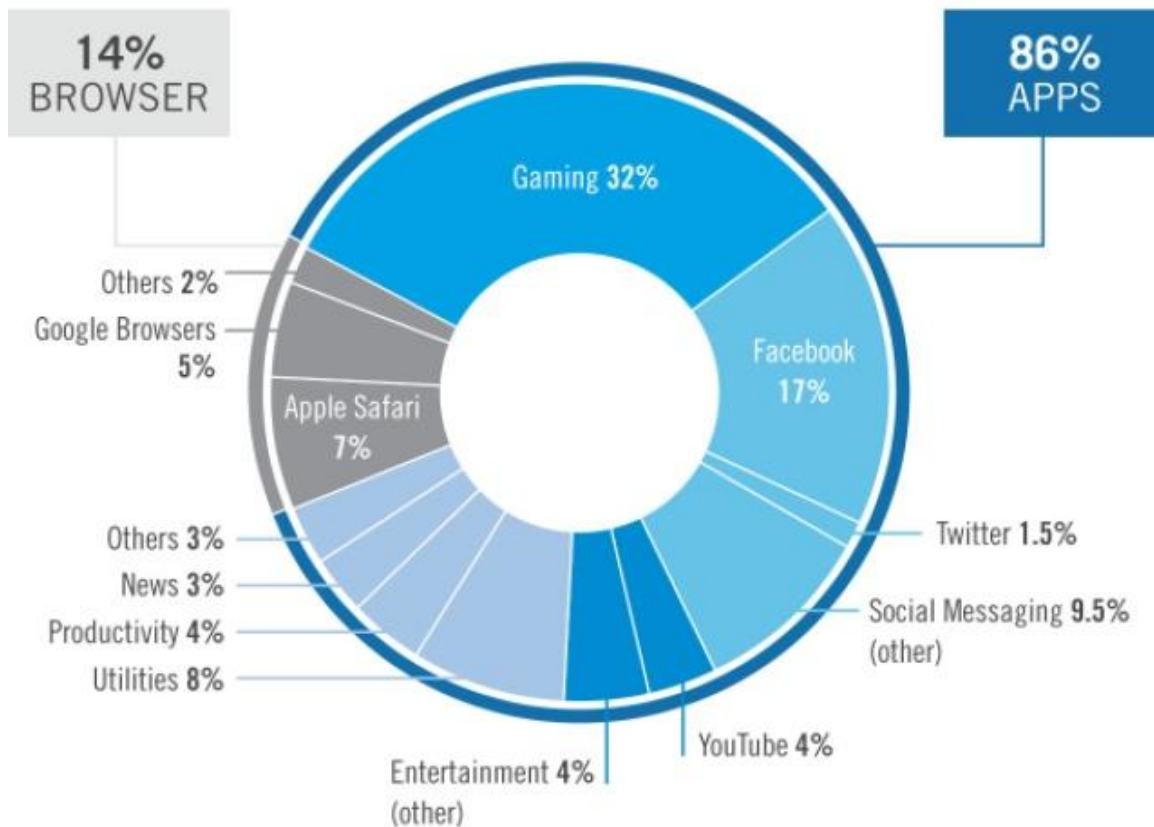
Η επιλογή του θέματος της πτυχιακής εργασίας έγινε λόγω της επιθυμίας μου να μελετήσω και να αποκτήσω εμπειρία πάνω στο λειτουργικό σύστημα Android. Επίσης θεώρησα ότι μία τέτοια εφαρμογή θα μπορούσε να γίνει αναπόσπαστο κομμάτι της καθημερινότητας των πολιτών της Θεσσαλονίκης. Τα λεωφορεία αποτελούν το κυριότερο μέσο μαζικής μετακίνησης και έχουν ενταχθεί στη ζωή όλων. Σημειώνονται καθημερινές μετακινήσεις πολιτών, φοιτητών, εργαζομένων από και προς όλες τις περιοχές της Θεσσαλονίκης, καθώς και από/προς την περιφέρεια της Θεσσαλονίκης η οποία ως επί το πλείστον εξυπηρετείται από τον ΟΑΣΘ. Πιστεύω ότι η εφαρμογή θα διευκολύνει τους πολίτες στον προγραμματισμό των καθημερινών τους διαδρομών.

Ο Οργανισμός Αστικών Συγκοινωνιών Θεσσαλονίκης (ΟΑΣΘ), χρησιμοποιεί καινούριες λειτουργίες για τη διευκόλυνση των πολιτών, όπως ο χρόνος άφιξης των λεωφορείων στις στάσεις και κάποιες ακόμα χρήσιμες υπηρεσίες που βρίσκονται στην ιστοσελίδα του (11). Επίσης έχει δημιουργηθεί μια ιστοσελίδα για κινητά τηλέφωνα (10), που προσφέρει ευκολότερη πρόσβαση στις υπηρεσίες του ΟΑΣΘ μέσω έξυπνων κινητών (smartphones), άλλα και απλούστερων συσκευών κινητής τηλεφωνίας. Δεν έχει δημιουργηθεί ακόμη κάποια επίσημη εφαρμογή σε λειτουργικό σύστημα Android ή iOS.

Υπάρχουν, αρκετές εφαρμογές στο Play Store του λειτουργικού Android, οι οποίες παρέχουν τις ίδιες πληροφορίες με την ιστοσελίδα του ΟΑΣΘ και δίνουν στους χρήστες τη δυνατότητα εύκολα και γρήγορα να εξετάσουν την ώρα άφιξης των επιθυμητών γραμμών χωρίς να χρειαστεί να επισκεφθούν την επίσημη ιστοσελίδα του Οργανισμού.

Η ερώτηση που δημιουργείται εύλογα σε αυτό το σημείο είναι, γιατί να μην επιλέξει ο χρήστης την ιστοσελίδα του ΟΑΣΘ για τα κινητά τηλέφωνα, αντί κάποιας εφαρμογής εφόσον προσφέρουν την ίδια πληροφορία. Μια εύχρηστη και φιλική προς το χρήστη εφαρμογή η οποία σχεδιάστηκε για το λειτουργικό Android είναι προτιμότερη από την επίσκεψη μιας ιστοσελίδας. Πλέον οι εφαρμογές σε λειτουργικό Android και iOS προτιμώνται από τα mobile sites - Άλλωστε είναι της μόδας. Στο Σχήμα 1 φαίνεται πώς κατανέμεται ο χρόνος που ξοδεύουν οι χρήστες σε συσκευές με λειτουργικό Android και iOS. Το 86% του χρόνου που ξοδεύουν οι χρήστες στα κινητά τους τηλέφωνα, βρίσκεται στις εφαρμογές, ενώ μόλις το 14% βρίσκεται στους browser (mobile sites, λοιπές ιστοσελίδες). [32]

## Time Spent on iOS and Android Connected Devices



Source: Flurry Analytics, ComScore, NetMarketShare

Σχήμα 1 Κατανομή ώρας που ξοδεύουν οι χρήστες, σε λειτουργικά Android και iOS (9)

Η εφαρμογή ονομάστηκε ενδεικτικά “PlanMyDay” λόγω των λειτουργιών που παρέχει. Δίνει στο χρήστη διαφορετικές δυνατότητες, που δε δίνονται σε κάποια άλλη υπάρχουσα εφαρμογή στο Play Store του λειτουργικού Android αλλά ούτε και στην επίσημη ιστοσελίδα του ΟΑΣΘ. Κάποιες από αυτές τις δυνατότητες είναι, ο καθημερινός προγραμματισμός της διαδρομής του κάθε χρήστη αλλά και η έγκαιρη ειδοποίηση του για την άφιξη του αντίστοιχου λεωφορείου.

Στα παρακάτω κεφάλαια θα μιλήσουμε γενικά, για το λειτουργικό σύστημα Android και την ασφάλεια αυτού. Θα καλύψουμε σε βάθος τα δομικά συστατικά που αποτελούν τις εφαρμογές Android και θα μιλήσουμε για τον κύκλο ζωής τους. Επίσης γίνεται παρουσίαση κάποιων βασικών βιβλιοθηκών Υποστήριξης, για την καλύτερη σχεδίαση των εφαρμογών, οι οποίες έχουν χρησιμοποιηθεί και στην εφαρμογή “PlanMyDay”. Έπειτα θα περιγράψουμε κάποιες υπηρεσίες ιστού της Google και μερικά χρήσιμα Android APIs και τέλος θα γίνει μια παρουσίαση της εφαρμογής “PlanMyDay” από τον σχεδιασμό της έως το τελικό έργο.

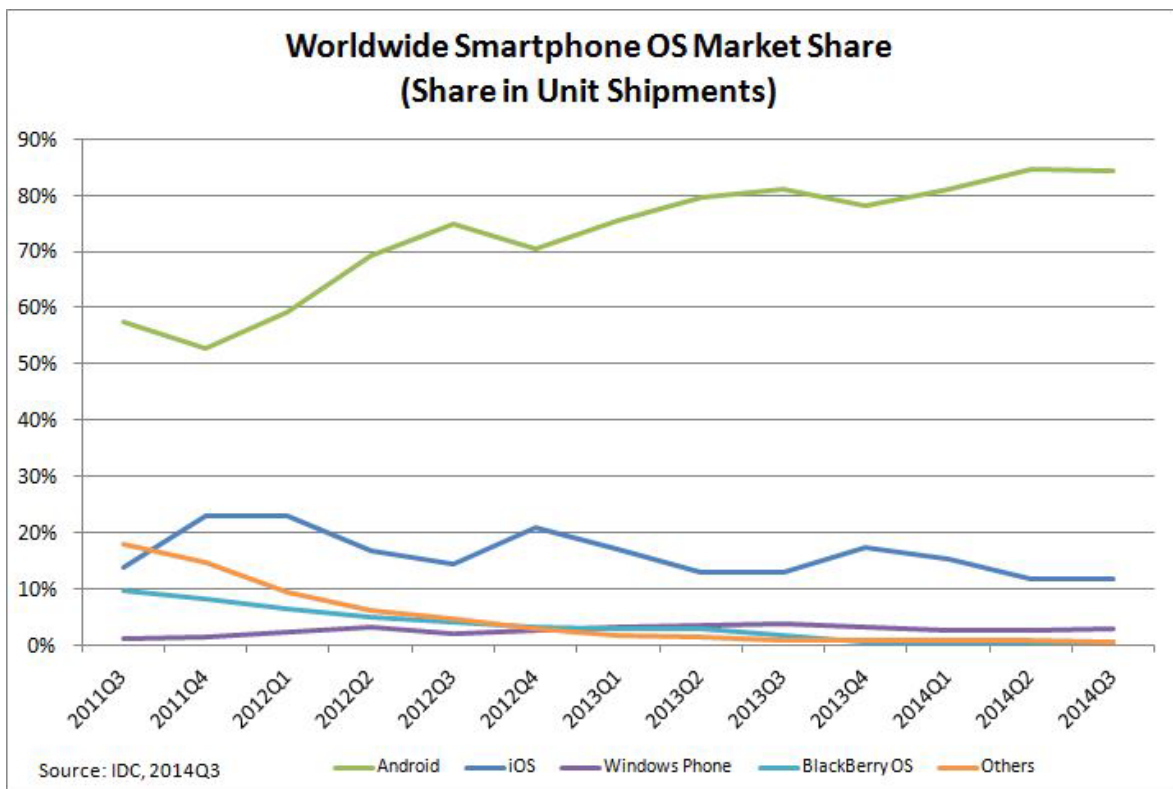
## ΚΕΦΑΛΑΙΟ 1

### Εισαγωγή στο λειτουργικό σύστημα Android

#### 1.1 ΕΙΣΑΓΩΓΗ

Το Android είναι ένα λειτουργικό σύστημα (OS) για κινητά τηλέφωνα, βασισμένο σε ένα Linux πυρήνα (kernel) το οποίο αναπτύσσεται από τη Google. Με μία διεπαφή χρήστη που βασίζεται σε άμεση χειραγώγηση, το Android έχει σχεδιαστεί κυρίως για συσκευές με οθόνες αφής όπως smartphones και υπολογιστές tablet, με εξειδικευμένες διεπαφές χρήστη για τηλεοράσεις (Android TV), αυτοκίνητα (Android Auto) και το ρολόγια χειρός (Android Wear). Το λειτουργικό σύστημα χρησιμοποιεί εισαγωγές με αφή (touch inputs) που αντιστοιχούν με κινήσεις (actions) στο πραγματικό κόσμο όπως σύρσιμο (swiping), πάτημα (tapping) για να χειριστεί τα αντικείμενα στην οθόνη και το εικονικό πληκτρολόγιο. Παρά το γεγονός ότι έχει σχεδιαστεί κυρίως για οθόνες αφής, έχει επίσης χρησιμοποιηθεί σε κονσόλες παιχνιδιών, ψηφιακές κάμερες, κανονικούς υπολογιστές (π.χ. HP Slate 21) και άλλες ηλεκτρονικές συσκευές.

Το Android είναι το πιο ευρέως χρησιμοποιούμενο λειτουργικό σύστημα κινητής τηλεφωνίας και από το 2013 το υψηλότερο σε πωλήσεις λειτουργικών συστημάτων συνολικά, όπως φαίνεται στο Σχήμα 2.

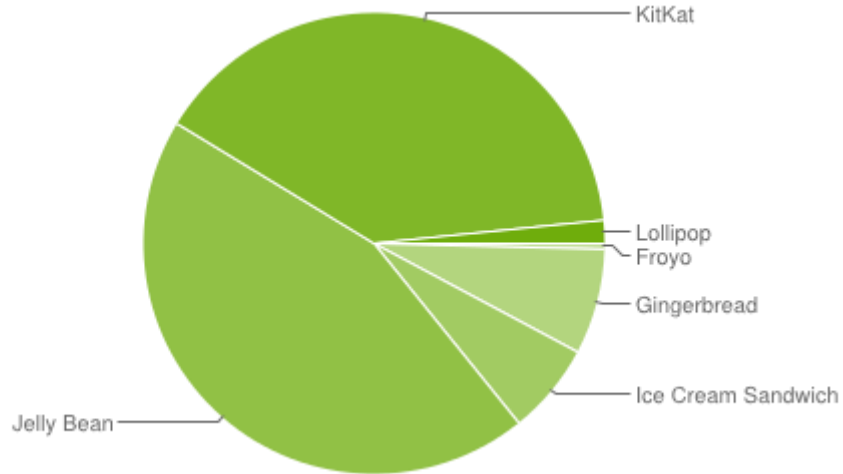


Σχήμα 2 Πωλήσεις Λειτουργικών Συστημάτων 2011 – 2014 (8)

#### 1.2 Ιστορικά

Το λειτουργικό σύστημα Android χρησιμοποιούν παραπάνω από ένα δισεκατομμύριο συσκευές (smartphones, tablets). Όπως αναφέρουν στην επίσημη ιστοσελίδα ([www.android.com](http://www.android.com)) “Εφόσον αυτές οι συσκευές κάνουν τις ζωές μας τόσο γλυκές κάθε έκδοση του Android πήρε το

όνομα της από ένα επιδόρπιο”. Σε χρονολογική σειρά παραθέτονται οι εκδόσεις Android, στον Πίνακα 1 ενώ στο Σχήμα 3 εμφανίζεται το ποσοστό διανομής των εκδόσεων του λειτουργικού συστήματος.[4]



Σχήμα 3 Ποσοστό Διανομών Εκδόσεων Android (4)

Πίνακας 1 Εκδόσεις Λειτουργικού Συστήματος Android

ΕΚΔΟΣΗ	ΟΝΟΜΑ	API	ΣΛΟΓΚΑΝ	ΔΙΑΝΟΜΗ
1.6	Donut		ο κόσμος της πληροφορίας βρίσκεται στα δάχτυλα σου	< 0,1%
2.0	Eclair		Φτιάξε την οθόνη σου όπως ακριβώς τη θέλεις	< 0,1%
2.2	Froyo	8	Η φωνητική πληκτρολόγηση σου επιτρέπει να εισάγεις κείμενο, και οι φωνητικές εντολές σου δίνουν τη δυνατότητα να ελέγχεις το κινητό σου τηλέφωνο μόνο με την ομιλία	0,4%
2.3	Gingerbread	10	Οι νέοι σένσορες κάνουν το Android ιδανικό για παιχνίδια	7,4%
3.0	Honeycomb		Βελτιστοποιημένο για tablets, αυτή η έκδοση ανοίγει νέους ορίζοντες όπου και αν βρίσκεσαι	< 0,1%
4.0	Ice Cream Sandwich	15	Το Android αναπτύσσεται πλήρως με ένα ανανεωμένο σχεδιασμό. Απλό, όμορφο και περισσότερο έξυπνο	6,4%
4.1	Jelly Bean	16 17 18	Το Android είναι γρήγορο και ομαλό με 'βουτυρένια' γραφικά	18,4% 19,8% 6,3%
4.4	Kit Kat	19	Έξυπνο, απλό και αληθινά δικό σας	39,7
5.0	Lollipop	21	Μία νέα γλυκιά έκδοση του Android. Μεταφέρει τα έξυπνα στοιχεία του Android σε μικρές και μεγάλες οθόνες	1,6%

### 1.3 Εφαρμογές

Οι εφαρμογές επεκτείνουν την λειτουργικότητα των συσκευών, γράφονται κυρίως στη γλώσσα προγραμματισμού Java (Java programming language) χρησιμοποιώντας το Android software development πακέτο (SDK).

Υπάρχουν πολλές διαφορές ανάμεσα στο Java API και στο Android API, μία σημαντική είναι ότι το Android δεν χρησιμοποιεί κάποιο Java Virtual Machine αλλά το Dalvik ή το Android Runtime (ART). Αυτές είναι τροποποιημένες Virtual Machines σχεδιασμένες για χρήση σε ενσωματωμένα συστήματα οι οποίες τρέχουν πάνω σε Linux πυρήνες έκδοσης 3.14 (Lollipop).

Το Android SDK περιλαμβάνει ένα ολοκληρωμένο πακέτο από εργαλεία ανάπτυξης, συμπεριλαμβανομένων αποσφαλματωτές (debugger), βιβλιοθήκες λογισμικού, εξομοιωτές (Android emulator), έγγραφα, δείγμα κώδικα και εκπαιδευτικά παραδείγματα.

Το επίσημα υποστηριζόμενο περιβάλλον ανάπτυξης (IDE) ήταν το Eclipse σε συνδυασμό με τα εργαλεία ανάπτυξης του Android (ADT plugin). Η google παράλληλα ανέπτυξε το δικό της περιβάλλον για την ανάπτυξη εφαρμογών σε Android το Android Studio το οποίο μέχρι τους τελευταίους μήνες ήταν σε δοκιμαστικές εκδόσεις (Beta). Λόγω των δοκιμαστικών εκδόσεων του Android Studio, η εφαρμογή της παρούσας πτυχιακής εργασίας έχει αναπτυχθεί στο περιβάλλον Eclipse.

Μια εφαρμογή αποτελείται από κώδικα σε Java σε συνδυασμό με κάποιες απαραίτητες βιβλιοθήκες του Android (π.χ. activities) και xml αρχεία που διαμορφώνουν την εμφάνιση στο χρήστη. Ο συνδυασμός των παραπάνω μετατρέπεται σε αρχείο εφαρμογής με την κατάληξη APK και εγκαθίσταται σε συσκευές Android.

Ουσιαστικά το Android προτρέπει τους προγραμματιστές να δημιουργήσουν εφαρμογές με ένα είδος MVC (model, view, controller) προτύπου. Το User Interface δημιουργείται σε ξεχωριστά XML αρχεία (View). Ο κώδικας (business model) μπορεί να μπει ξεχωριστά από τα Activities (controller) τα οποία με τη σειρά τους συνδυάζουν το κομμάτι του View και Model και τα παρουσιάζουν στον τελικό χρήστη. Παρόλα αυτά δεν ανεφέρεται πουθενά στην επίσημη ιστοσελίδα για τους προγραμματιστές Android η έννοια των MVC ή MVP προτύπων.

### 1.4 Art & Dalvik

Το Android runtime (ART) εμφανίστηκε πρώτη φορά στην έκδοση Kit Kat 4.4 και όχι ως προεπιλεγμένη επιλογή, ενώ ως βασικό περιβάλλον έρχεται με τη έκδοση Lollipop 5.0. Χρησιμοποιείται από εφαρμογές και κάποια services του συστήματος στο Android ως περιβάλλον runtime.

Το Dalvik Virtual Machine από την έκδοση Frogo 2.2 χρησιμοποιεί just-in-time (JIT) compilation για να κάνει compile τα byte του κώδικα κάθε φορά που μια εφαρμογή ξεκινάει. Σε αντίθεση το ART παρουσιάζει τη χρήση του ahead-of-time(AOT) compilation όπου το compile εκτελείται κατά την εγκατάσταση της εφαρμογής.

Έτσι με τη μείωση του συνολικού compile που πρέπει να εκτελείται κατά τη διάρκεια λειτουργίας μιας εφαρμογής, η χρήση του επεξεργαστή της συσκευής μειώνεται και η διάρκεια ζωής της μπαταρίας αυξάνεται. Ταυτόχρονα το ART φέρνει βελτιώσεις στην απόδοση, στο garbage collection, και στην αποσφαλμάτωση εφαρμογών. [7]

## 1.5 Αρχιτεκτονική

Η πλατφόρμα του Android αποτελείται από μια στοίβα λογισμικού. Η στοίβα είναι εμφανής στο Σχήμα 4.



Σχήμα 4 Στοίβα λογισμικού Android (6)

### 1.5.1 Επίπεδο Εφαρμογών (Blue layer - Application)

Στο επίπεδο αυτό περιλαμβάνεται ένα σύνολο από βασικές εφαρμογές οι οποίες έρχονται προ εγκατεστημένες, μερικές από αυτές είναι επαφές, ημερολόγιο, gmail(google email client), hangouts(εφαρμογή της google για sms), φωτογραφική μηχανή, αριθμομηχανή κ.α. Όλες οι εφαρμογές είναι γραμμένες σε Java.

### 1.5.2 Επίπεδο Πλαισίου Εφαρμογών (Blue layer - Application Framework)

Βρίσκεται κάτω από το επίπεδο Εφαρμογών και αποτελείται ένα σύνολο συστημάτων και υπηρεσιών:

- Ένα σύνολο από γραφικά στοιχεία (Views) για την δημιουργία γραφικού περιβάλλοντος συμπεριλαμβανομένων κουτιών επιλογής ώρας (time pickers), λιστών (lists), κουτιών κειμένου (text boxes), κουτιών επιλογής ημερομηνίας(Date pickers), κουμπιών (buttons) κ.α.
- Ένα διαχειριστή περιεχομένου (Content Manager) ο οποίος επιτρέπει στις εφαρμογές την πρόσβαση σε δεδομένα άλλων εφαρμογών ή τον διαμοιρασμό των δικών τους δεδομένων με άλλες εφαρμογές.
- Ένα διαχειριστή πόρων (Resource Manager) για την πρόσβαση στους πόρους όπως εικόνες, xml έγγραφα, μενού.



- Έναν διαχειριστή ειδοποιήσεων (Notification Manager) ο οποίος επιτρέπει την προβολή ειδοποιήσεων στην μπάρα κατάστασης (status bar). Στην εφαρμογή αυτής της πτυχιακής εργασίας έγινε χρήση του Notification Manager, για την ειδοποίηση του χρήστη.
- Έναν διαχειριστή δραστηριοτήτων (Activity Manager) ο οποίος διαχειρίζεται τον κύκλο ζωής των εφαρμογών.

### 1.5.3 Επίπεδο Βιβλιοθηκών (Green layer - Libraries)

Το οποίο περιλαμβάνει ένα σύνολο από βιβλιοθήκες γραμμένες σε C/C++ οι οποίες χρησιμοποιούνται από διάφορα στοιχεία του συστήματος του Android. Οι δυνατότητες που προσφέρουν αυτές οι βιβλιοθήκες είναι προσβάσιμες στους προγραμματιστές δια μέσου του επιπέδου πλαισίου εφαρμογής.

### 1.5.4 Επίπεδο Εκτέλεσης (Green layer - Android Runtime)

Το οποίο αποτελείται από ένα σύνολο από βασικές βιβλιοθήκες και την Dalvik Virtual Machine (< Android 4.4) ή την Android Runtime (ART).

### 1.5.5 Ο Πυρήνας του Linux (Red Layer)

Το Android βασίζεται στον πυρήνα Linux έκδοση 3.14 (Android 5.0) για βασικές υπηρεσίες συστήματος όπως ασφάλεια, διαχείριση μνήμης, διαχείριση διεργασιών, στοίβα δικτύου, και οδηγούς συσκευών. Ο πυρήνας λειτουργεί επίσης ως ένα ενδιάμεσο επίπεδο αφαίρεσης μεταξύ της στοίβας λογισμικού και του υλικού. [3]

## 1.6 ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό έγινε μια εισαγωγή στο Λειτουργικό Σύστημα Android, από τις πρώτες εκδόσεις του μέχρι τη σημερινή και παρουσιάστηκε η αρχιτεκτονική του. Μιλήσαμε για της εφαρμογές που αναπτύσσονται στο λειτουργικό Android γενικά και στο επόμενο κεφάλαιο θα περιγράψουμε σε βάθος κάποια βασικά συστατικά τους.

## ΚΕΦΑΛΑΙΟ 2

### Βασικά δομικά στοιχεία εφαρμογών Android

#### 2.1 ΕΙΣΑΓΩΓΗ

Συστατικά στοιχεία μιας εφαρμογής:

- Δραστηριότητα (Activity)
- Τεμάχιο (Fragment)
- Υπηρεσία (Service)
- Ειδοποίηση (Notification)
- Προσαρμογέας (Adapter)
- Βάση Δεδομένων (SQLite)
- AndroidManifest

Μια εφαρμογή μπορεί να περιέχει κάποια από αυτά τα συστατικά ή και πολλά άλλα. Το βασικότερο είναι η Δραστηριότητα (Activity) το οποίο δεν μπορεί να παραληφθεί για τη δημιουργία μιας εφαρμογής που θέλει να προβάλλει περιεχόμενο στο χρήστη. Παρακάτω γίνεται μια σύντομη περιγραφή του κάθε ενός.

#### 2.2 Δραστηριότητα (Activity)

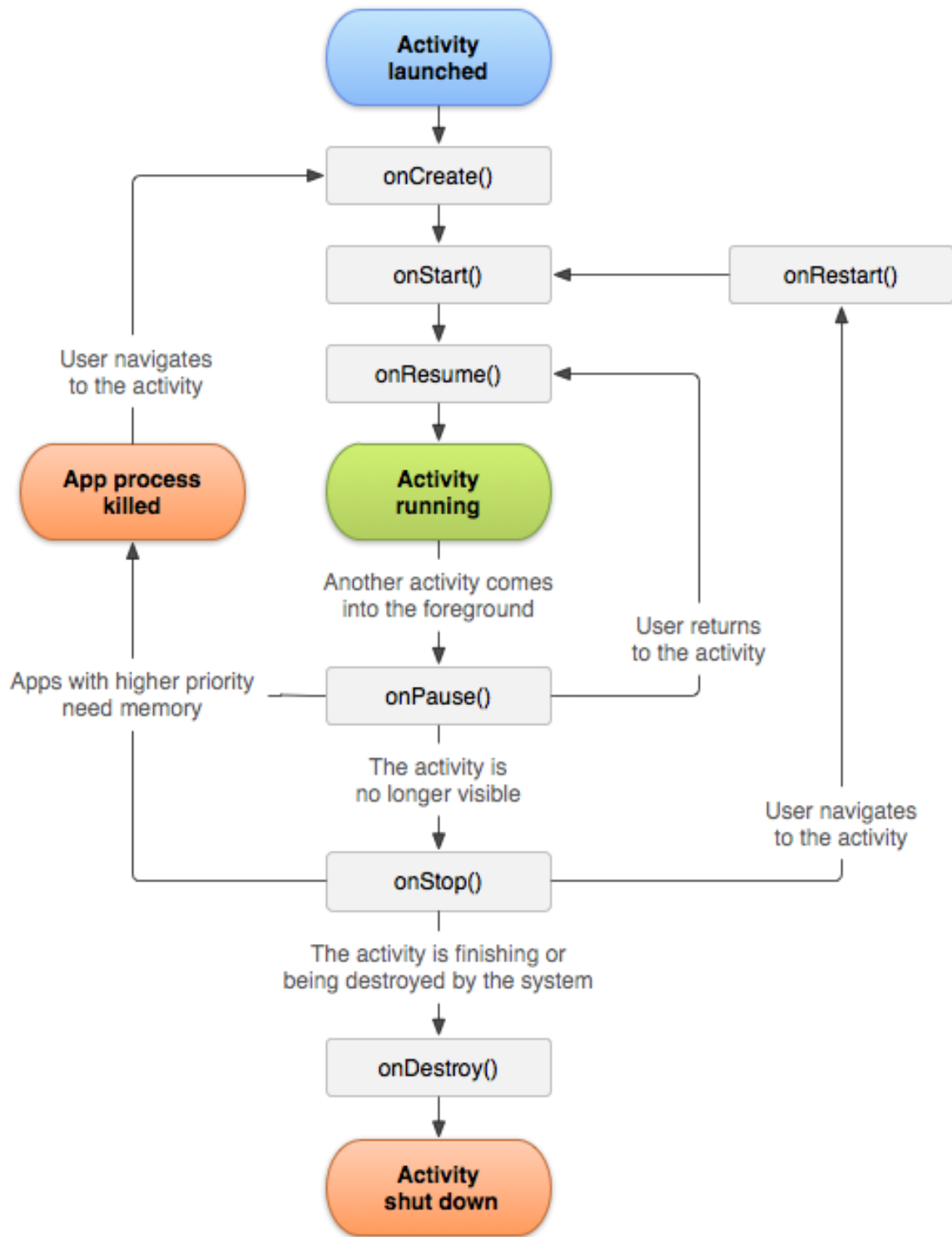
Η Δραστηριότητα (Activity) είναι το πιο βασικό από τα δομικά στοιχεία μιας εφαρμογής. Συνήθως αποτελεί ένα ξεχωριστό παράθυρο σε μία εφαρμογή. Κάθε Δραστηριότητα αναπτύσσεται σε μία κλάση που επεκτείνει (extends) την κλάση Activity και συνοδεύεται από μεθόδους που παρέχει.

Η βασικότερη μέθοδος που παρέχεται μαζί με τη κλάση Activity είναι η “OnCreate”, η οποία εκτελείται μόλις η Δραστηριότητα δημιουργείται. Σε αυτή τη μέθοδο γίνεται χρήση της μεθόδου “setContentView”, όπου ενσωματώνει το περιβάλλον που θα εμφανίσει στο χρήστη (User Interface). [1]

##### 2.2.1 Κύκλος ζωής Δραστηριότητας

Οι Δραστηριότητες σε ένα σύστημα διαχειρίζονται σαν μια στοίβα από δραστηριότητες. Όταν μια καινούργια Δραστηριότητα αρχίζει, τοποθετείται στο πάνω μέρος της στοίβας και γίνεται η Δραστηριότητα που τρέχει αυτή τη στιγμή. Η προηγούμενη Δραστηριότητα πάντα παραμένει πιο χαμηλά στη στοίβα και δεν θα έρθει στο προσκήνιο ξανά μέχρι η νέα Δραστηριότητα να κλείσει.

Το παρακάτω διάγραμμα (Σχήμα 5) δείχνει τα σημαντικά στάδια μιας Δραστηριότητας. Τα ορθογώνια παριστάνουν μεθόδους που μπορούν να ενσωματωθούν και να λειτουργήσουν όταν η Δραστηριότητα κινείται μεταξύ των σταδίων. Τα οβάλ σχήματα αντιπροσωπεύουν τα κύρια στάδια μιας Δραστηριότητας στα οποία μπορεί να βρísκεται. [1]



Σχήμα 5 Κύκλος ζωής Δραστηριότητας (Activity) (1)

Στον Πίνακα 2, περιγράφονται οι μέθοδοι που χρησιμοποιεί μια Δραστηριότητα (Activity).

Πίνακας 2 Μέθοδοι Δραστηριότητας (Activity)

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
onCreate	Η μέθοδος καλείται όταν η Δραστηριότητα δημιουργείται για πρώτη φορά. Ακολουθείται από τη μέθοδο onStart.
onRestart	Καλείται αφότου η Δραστηριότητα έχει σταματήσει. Ακολουθείται από τη μέθοδο onStart.
onStart	Καλείται όταν η Δραστηριότητα γίνεται ορατή στο χρήστη. Ακολουθείται από τη μέθοδο onResume ή onStop αναλόγως το στάδιο που προέρχεται η Δραστηριότητα.
onResume	Καλείται όταν η Δραστηριότητα αλληλεπιδρά με το χρήστη. Σε αυτό το σημείο η Δραστηριότητα είναι τοποθετημένη πρώτη στη στοίβα. Ακολουθείται από τη μέθοδο onPause.
onPause	Καλείται όταν το σύστημα είναι έτοιμο να επαναφέρει την προηγούμενη Δραστηριότητα. Ακολουθείται από τη μέθοδο onResume ή onStop αναλόγως το στάδιο που προέρχεται η Δραστηριότητα.
onStop	Καλείται όταν η δραστηριότητα δεν είναι πλέον ορατή στο χρήστη, επειδή μια άλλη δραστηριότητα επαναφέρεται και καλύπτει τη πρώτη. Ακολουθείται από τη μέθοδο onRestart ή onDestroy αναλόγως το στάδιο που προέρχεται η Δραστηριότητα.
onDestroy	Καλείται αμέσως πριν καταστραφεί η Δραστηριότητα. Αυτό μπορεί να συμβεί είτε γιατί η Δραστηριότητα τελείωσε (Activity.finish) είτε γιατί το σύστημα διαγράφει προσωρινά το στιγμιότυπο από αυτή τη δραστηριότητα για να κερδίσει χώρο.

## 2.3 Τεμάχιο (Fragment)

Τεμάχιο (Fragment) είναι ένα κομμάτι του περιβάλλοντος χρήστη (User Interface) μιας εφαρμογής ή μια συμπεριφορά που μπορεί να τοποθετηθεί σε μια Δραστηριότητα (Activity). Η αλληλεπίδραση με τα Τεμάχια επιτυγχάνεται μέσω του “FragmentManager” που λαμβάνεται από το “Activity.getFragmentManager()” και “Fragment.getFragmentManager()” σε Δραστηριότητα (Activity) και σε Τεμάχιο (Fragment) αντίστοιχα.

Η κλάση του Fragment μπορεί να χρησιμοποιηθεί με πολλούς τρόπους για να επιτευχθεί μια ευρεία ποικιλία αποτελεσμάτων. Στον πυρήνα του, παρουσιάζει μία συγκεκριμένη λειτουργία ή ένα περιβάλλον που τρέχει μέσα σε μία μεγαλύτερη Δραστηριότητα. Το Τεμάχιο (Fragment) είναι στενά δεμένο με τη Δραστηριότητα (Activity) που το εμπεριέχει και δεν μπορεί να χρησιμοποιηθεί ξεχωριστά από αυτήν.

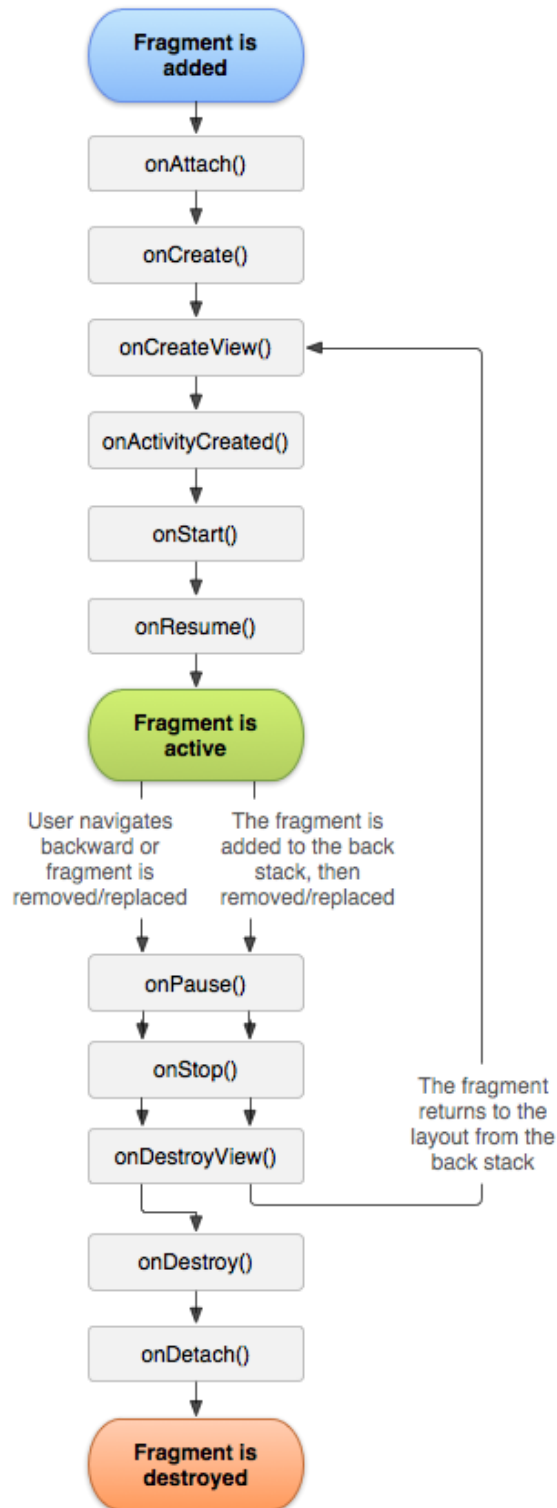
Αν και το Τεμάχιο (Fragment) ορίζει το δικό του κύκλο ζωής, αυτός εξαρτάται από τη Δραστηριότητα. Αυτό σημαίνει ότι, αν η Δραστηριότητα είναι σταματημένη, κανένα από τα Τεμάχια που περιέχει δεν μπορεί να ξεκινήσει. Και αν η Δραστηριότητα καταστραφεί τότε και όλα τα Τεμάχια που περιέχει θα καταστραφούν. [9]

### 2.3.1 Κύκλος ζωής Τεμαχίου (Fragment)

Αν και ο κύκλος ζωής ενός Τεμαχίου (Fragment) είναι δεμένος με τη Δραστηριότητα που ανήκει, έχει το δικό του κύκλο από αυτόν της Δραστηριότητάς του. Περιλαμβάνει τις βασικές μεθόδους του κύκλου ζωής μιας Δραστηριότητας (π.χ. onResume), αλλά και επιπλέον σημαντικές

είναι οι μέθοδοι που σχετίζονται με την αλληλεπίδρασή του, με τη Δραστηριότητα και το περιβάλλον χρήστη. [9]

Στο Σχήμα 6, παρουσιάζεται ο κύκλος ζωής ενός Τεμαχίου ενώ τρέχει η Δραστηριότητα που τα περικλείει.



Σχήμα 6 Κύκλος ζωής Τεμαχίου (Fragment), καθώς η Δραστηριότητα (Activity) τρέχει (2)

Οι μέθοδοι του πυρήνα, που καλούνται για να φέρουν το Τεμάχιο στο στάδιο της επαναφοράς (αλληλεπίδραση με το χρήστη) παρουσιάζονται στον Πίνακα 3.

Πίνακας 3 Μέθοδοι Τεμαχίου (Fragment), αλληλεπίδρασης με τον χρήστη

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
onAttach	Καλείται όταν το Τεμάχιο (Fragment) συσχετίζεται με τη Δραστηριότητά του.
onCreate	καλείται για να κάνει την αρχική δημιουργία του Τεμαχίου (Fragment)
onCreateView	δημιουργεί και επιστρέφει την ιεραρχία διάταξης του περιβάλλοντος που σχετίζεται με το Τεμάχιο (Fragment)
onActivityCreated	ενημερώνει το Τεμάχιο ότι η Δραστηριότητα έχει ολοκληρώσει το στάδιο δημιουργίας της.
onViewStateRestored	ενημερώνει το Τεμάχιο ότι όλα τα αποθηκευμένα στάδια της ιεραρχίας διάταξης περιβάλλοντος έχουν επαναφερθεί.
onStart	κάνει το Τεμάχιο ορατό στο χρήστη (Βασισμένο στο ότι η Δραστηριότητα που ανήκει έχει ξεκινήσει)
onResume	κάνει το Τεμάχιο να αλληλεπιδρά με το χρήστη (Βασισμένο στο ότι η Δραστηριότητα που ανήκει έχει επαναφερθεί)

Εφόσον το Τεμάχιο (Fragment) δεν χρησιμοποιείται πλέον, περνάει από μια αντίστροφη σειρά από κλήσεις (Callbacks). Οι κλήσεις αυτές περιγράφονται στον Πίνακα 4.

Πίνακας 4 Μέθοδοι Τεμαχίου (Fragment), όταν δεν χρησιμοποιείται πλέον

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
onPause	Το Τεμάχιο δεν αλληλεπιδρά άλλο με το χρήστη είτε γιατί η Δραστηριότητα (Activity), που ανήκει, είναι σε παύση, είτε γιατί μια λειτουργία του Τεμαχίου (Fragment) το τροποποιεί στην Δραστηριότητα.
onStop	Το Τεμάχιο (Fragment) δεν είναι πλέον ορατό στο χρήστη είτε γιατί η Δραστηριότητα (Activity), που ανήκει έχει σταματήσει, είτε γιατί μια λειτουργία του Τεμαχίου (Fragment) το τροποποιεί στην Δραστηριότητα.
onDestroyView	επιτρέπει στο Τεμάχιο (Fragment) να καθαρήσει τους πόρους που σχετίζονται με τη διάταξη του περιβάλλοντος χρήστη του.
onDestroy	καλείται για ένα τελικό καθάρισμα του σταδίου που βρίσκεται το Τεμάχιο (Fragment).
onDetach	καλείται αμέσως πριν από την αποκοπή της σχέσης του Τεμαχίου (Fragment) με τη Δραστηριότητα (Activity)

### 2.3.2 Φιλοσοφία σχεδιασμού

Το λειτουργικό σύστημα Android παρουσίασε τα Τεμάχια (Fragments) στο Android 3.0 (Honeycomb), κυρίως για να υποστηρίξει τον δυναμικό και ευέλικτο σχεδιασμό ενός περιβάλλοντος χρήστη (User Interface) σε μεγάλες οθόνες όπως σε Tablet.

Αυτό συνέβη διότι η οθόνη του Tablet είναι πού μεγαλύτερη από αυτή του κινητού τηλεφώνου και υπάρχει αρκετός χώρος για τον συνδυασμό και την ανταλλαγή πληροφοριών μεταξύ των συστατικών του περιβάλλοντος χρήστη. Τα Τεμάχια (Fragments) επιτρέπουν τέτοιο

σχεδιασμό χωρίς να χρειάζεται από τον προγραμματιστή να διαχειρίζεται πολύπλοκες αλλαγές στην ιεραρχία διάταξης του περιβάλλοντος χρήστη.

Διαιρώντας το σχεδιαστικό χώρο μιας Δραστηριότητας (Activity) σε Τεμάχια (Fragments) γίνεται δυνατή η τροποποίηση της εμφάνισης μιας Δραστηριότητας σε πραγματικό χρόνο και η διατήρηση αυτών των αλλαγών σε μια στοίβα που διαχειρίζεται από τη Δραστηριότητα. [9]

## 2.4 Υπηρεσία (Service)

Μια Υπηρεσία (Service) είναι ένα συστατικό εφαρμογής που μπορεί να εκτελέσει μεγάλες σε χρόνο λειτουργίες στο παρασκήνιο χωρίς να παρέχει περιβάλλον χρήστη (User Interface). Κάποιο άλλο συστατικό μιας εφαρμογής μπορεί να ξεκινήσει μία Υπηρεσία (Service) και αυτή να συνεχίσει να τρέχει στο παρασκήνιο ακόμα και αν ο χρήστης μεταβεί σε κάποια άλλη εφαρμογή. Επιπλέον, ένα συστατικό μπορεί να δεσμεύεται σε μια Υπηρεσία για να αλληλοεπιδράσει με αυτή, ακόμα και να εκτελέσει ενδοεπικοινωνία (Interprocess Communication).

Για παράδειγμα, μία Υπηρεσία (Service) μπορεί να χειριστεί συναλλαγές στο δίκτυο, να παίξει μουσική, να εκτελέσει ένα αρχείο ή να αλληλοεπιδράσει με ένα πάροχο περιεχομένου και όλα αυτά από το παρασκήνιο. [19]

### 2.4.1 Μορφές Υπηρεσίας (Service)

Η Υπηρεσία (Service) ουσιαστικά μπορεί να πάρει δύο μορφές:

- έχει ξεκινήσει (Started)

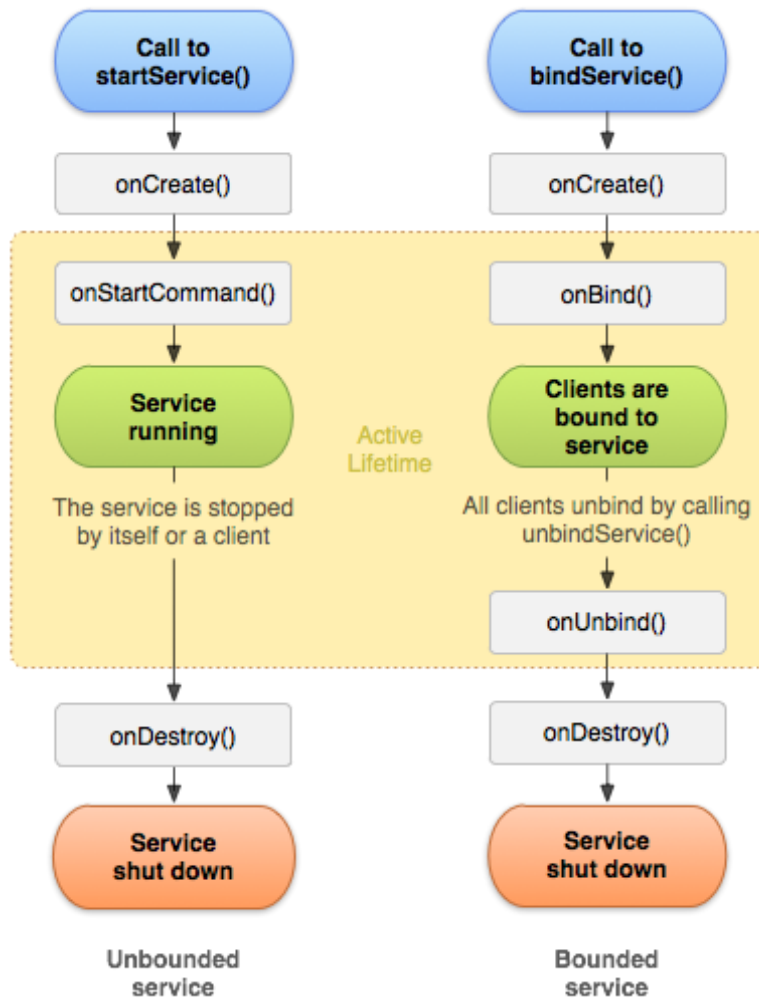
Μια Υπηρεσία “έχει ξεκινήσει” (Started) όταν κάποιο συστατικό εφαρμογής (όπως μια Δραστηριότητα) το εκκινεί καλώντας τη μέθοδο `startService()`. Αφού ξεκινήσει η Υπηρεσία μπορεί να τρέχει στο παρασκήνιο επ’ αόριστον, ακόμα και αν το συστατικό που την εκίνησε έχει καταστραφεί. Συνήθως, μία Υπηρεσία που έχει ξεκινήσει εκτελεί μια ενιαία λειτουργία και δεν επιστρέφει κάποιο αποτέλεσμα σε αυτόν που την κάλεσε. Για παράδειγμα, μπορεί να κατεβάσει ή να ανεβάσει ένα αρχείο στο διαδίκτυο. Όταν η λειτουργία ολοκληρωθεί, η Υπηρεσία πρέπει να σταματήσει τον εαυτό της.

- Συνδεδεμένη (Bound)

Μια Υπηρεσία είναι “Συνδεδεμένη” (Bound) όταν ένα συστατικό εφαρμογής συνδέεται σε αυτή καλώντας τη μέθοδο `bindService()`. Μια συνδεδεμένη Υπηρεσία προσφέρει μία διεπαφή πελάτη - διακομιστή (Client - Server), που επιτρέπει συστατικά να αλληλεπιδρούν με την Υπηρεσία, να στέλνουν αιτήματα, να παίρνουν αποτελέσματα και ακόμα πράξουν διαμέσου διεργασιών με ενδοεπικοινωνία (Interprocess Communication). Μία Συνδεδεμένη Υπηρεσία τρέχει μόνο όσο κάποιο συστατικό εφαρμογής είναι συνδεδεμένο σε αυτήν.

Μια Υπηρεσία μπορεί να δουλέψει και με τους δυο τρόπους. Μπορεί να έχει ξεκινήσει (και να τρέχει επ’ αόριστον) και επίσης να της επιτρέπεται να συνδεθεί. Αυτό έχει να κάνει μόνο με το πως ο προγραμματιστής εφαρμόζει τις μεθόδους:

- `onStartCommand` για να επιτρέψει το συστατικό να ξεκινήσει την Υπηρεσία και
- `onBind` για να το επιτρέψει να συνδεθεί. [19]



Σχήμα 7 Κύκλος ζωής Υπηρεσίας (Service) (5)

Το Σχήμα 7 στα αριστερά παρουσιάζει τον κύκλο ζωής όταν η Υπηρεσία δημιουργείται με τη μέθοδο `startService()`. Ενώ το διάγραμμα δεξιά δείχνει τον κύκλο ζωής όταν η Υπηρεσία δημιουργείται με τη μέθοδο `bindService()`.

#### 2.4.2 Ειδοποίηση μέσω Υπηρεσίας (Service)

Μια Υπηρεσία (Service) που τρέχει, μπορεί να ειδοποιήσει το χρήστη για γεγονότα χρησιμοποιώντας αναδυόμενες ειδοποιήσεις (Toast Notification) ή ειδοποιήσεις στη γραμμή κατάστασης (Status Bar Notifications).

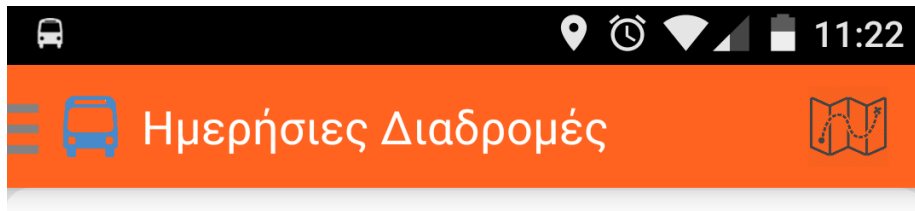
Η αναδυόμενη ειδοποίηση είναι ένα μήνυμα που εμφανίζεται στην επιφάνεια του τρέχοντος παραθύρου για μια στιγμή και μετά εξαφανίζεται, ενώ μια ειδοποίηση στη γραμμή κατάστασης παρέχει ένα εικονίδιο στη γραμμή κατάστασης με ένα μήνυμα, το οποίο μπορεί να επιλεγεί από το χρήστη με σκοπό να δράσει (όπως να ξεκινήσει μια Δραστηριότητα).

Συνήθως, η ειδοποίηση στη γραμμή κατάστασης είναι η καλύτερη τεχνική όταν κάποια δουλειά στο παρασκήνιο έχει ολοκληρωθεί (όπως η λήψη ενός αρχείου που ολοκληρώθηκε) και ο χρήστης μπορεί τώρα να δράσει σε αυτή. Όταν ο χρήστης επιλέξει την ειδοποίηση από το επεκτάσιμο περιβάλλον χρήστη, η ειδοποίηση μπορεί να ξεκινήσει κάποια Δραστηριότητα (όπως να δείξει στο χρήστη το αρχείο που λήφθηκε). [19]

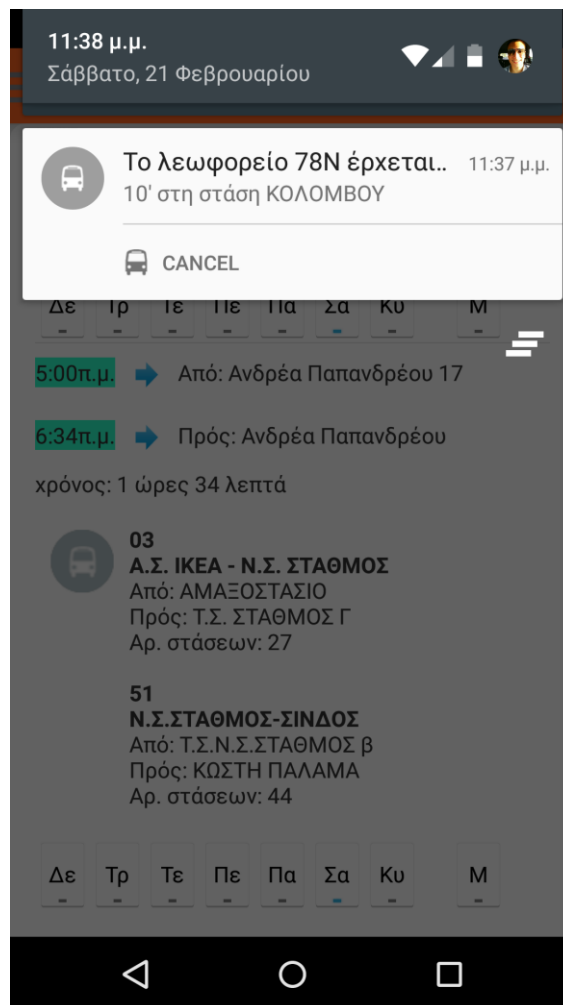


## 2.5 Ειδοποίηση (Notification)

Ειδοποίηση (Notification) είναι ένα μήνυμα που μπορεί να εμφανιστεί στο χρήστη, έξω από το κανονικό περιβάλλον χρήστη (User Interface) της εφαρμογής. Όταν ο προγραμματιστής λείει στο σύστημα να εκπέμψει μία Ειδοποίηση (Notification), αρχικά εμφανίζεται ως ένα εικονίδιο στη περιοχή των ειδοποιήσεων (notification area), Σχήμα 8. Για να δει ο χρήστης τις λεπτομέρειες της Ειδοποίησης, πρέπει να ανοίξει το συρτάρι των ειδοποιήσεων (notification drawer), Σχήμα 9. Η περιοχή αλλά και το συρτάρι ειδοποιήσεων, είναι και τα δυο περιοχές που χειρίζονται από το σύστημα και ο χρήστης μπορεί να τις δει κάθε στιγμή. [15]



Σχήμα 8 Ειδοποίηση (Notification) στην περιοχή ειδοποιήσεων (3)



Σχήμα 9 Ειδοποίηση στο συρτάρι ειδοποιήσεων

### 2.5.1 Συστατικά Ειδοποιήσεων (Notification)

Ένα αντικείμενο ειδοποίησης πρέπει να περιέχει τα ακόλουθα συστατικά:

- ένα μικρό εικονίδιο, που δημιουργείται με τη μέθοδο `setSmallIcon`,
- ένα τίτλο, που δημιουργείται με τη μέθοδο `setContentTitle`,
- λεπτομερείς κείμενο, που δημιουργείται με τη μέθοδο `setContentText`.

Όλες οι υπόλοιπες ρυθμίσεις και περιεχόμενα είναι προαιρετικά.

Παρόλα τα προαιρετικά συστατικά μιας ειδοποίησης θα ήταν πρόπον να προστεθεί τουλάχιστον μια δράση στην Ειδοποίηση. Μια δράση επιτρέπει στο χρήστη να μεταβεί απευθείας από την Ειδοποίηση σε μια Δραστηριότητα (Activity) στην εφαρμογή, όπου μπορεί να κοιτάξει ένα ή περισσότερα γεγονότα ή να κάνει περαιτέρω δουλειά.

Μία Ειδοποίηση μπορεί να παρέχει πολλαπλές δράσεις. Ο προγραμματιστής πρέπει να προσδιορίσει τη δράση που θα ενεργοποιηθεί όταν ο χρήστης πατήσει πάνω στην Ειδοποίηση (Notification) η οποία συνήθως θα ανοίξει μια Δραστηριότητα στην εφαρμογή. Μπορεί επίσης να προσθέσει κουμπιά στην Ειδοποίηση που εκτελούν επιπλέον δράση όπως να αναβάλει ένα ξυπνητήρι ή να απαντήσει άμεσα σε ένα μήνυμα. [15]

### 2.6 Προσαρμογέας (Adapter)

Ένα αντικείμενο Προσαρμογέα (Adapter) δρα σαν γέφυρα μεταξύ μιας όψης Προσαρμογέα (AdapterView) και των δεδομένων που βρίσκονται από κάτω, για αυτήν την όψη. Ο Προσαρμογέας παρέχει πρόσβαση στα στοιχεία των δεδομένων. Είναι επίσης υπεύθυνος για τη δημιουργία της όψης για κάθε στοιχείο του συνόλου δεδομένων του.

Μια συνηθισμένη κλάση κοινής εφαρμογής για έναν Προσαρμογέα (Adapter) είναι η `BaseAdapter`. Μπορεί να χρησιμοποιηθεί και για `ListView` (με την εφαρμογή της ειδικής διεπαφής `ListAdapter`) και για `Spinner` (με την εφαρμογή της ειδικής διεπαφής `SpinnerAdapter`). [2]

Η κλάση `Adapter` παρέχει κάποιες μεθόδους για την εφαρμογή των λειτουργιών της. Μερικές από αυτές περιγράφονται στον Πίνακα 5.

Πίνακας 5 Μέθοδοι Προσαρμογέα (Adapter)

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
<code>getCount</code>	Πόσα στοιχεία περιέχει το σύνολο των δεδομένων, που εκπροσωπούνται από αυτόν τον Προσαρμογέα (Adapter)
<code>getItem</code>	Επιστρέφει το στοιχείο, που σχετίζεται με τη συγκεκριμένη θέση στο σύνολο των δεδομένων.
<code>getItemId</code>	Επιστρέφει τη γραμμή που σχετίζεται με τη συγκεκριμένη θέση στη λίστα.
<code>getView</code>	Επιστρέφει την Όψη (View) που παρουσιάζει τα δεδομένα στη συγκεκριμένη θέση στο σύνολο των δεδομένων.
<code>isEmpty</code>	Επιστρέφει λογική μεταβλητή που ενημερώνει το προγραμματιστή αν ο Προσαρμογέας (Adapter) περιέχει δεδομένα.

Ενώ η κλάση BaseAdapter παρέχει και κάποιες περαιτέρω μεθόδους εκ των οποίων μια βασική περιγράφεται στο Πίνακα 6.

Πίνακας 6 Κύρια μέθοδος Βασικού Προσαρμογέα (BaseAdapter)

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
notifyDataSetChanged	Ειδοποιεί τους παρατηρητές ότι τα δεδομένα κάτω από τον Προσαρμογέα έχουν αλλάξει και κάθε Όψη (View) που αντικατοπτρίζει το σύνολο των δεδομένων πρέπει να ανανεωθεί.

## 2.7 Βάση Δεδομένων (SQLite)

Στο λειτουργικό σύστημα Android τα δεδομένα των εφαρμογών μπορούν να αποθηκευτούν σε βάσεις δεδομένων, αρχεία, ή προτιμήσεις, στον εσωτερικό ή στον εξωτερικό χώρο αποθήκευσης.

Η αποθήκευση των δεδομένων σε Βάσεις Δεδομένων, είναι ιδανική για επαναληπτικά ή δομημένα δεδομένα, όπως στοιχεία επικοινωνίας. Υποθέτοντας ότι ο προγραμματιστής είναι εξοικειωμένος με SQL Βάσεις Δεδομένων γενικά, το Android παρέχει κάποιες κλάσεις που βοηθούν να ξεκινήσει με SQLite Βάσεις Δεδομένων σε αυτό.

Μία από τις βασικές αρχές των SQL Βάσεων Δεδομένων, είναι το σχήμα: μία επίσημη δήλωση του πως είναι οργανωμένη η βάση. Το σχήμα αντικατοπτρίζει τις δηλώσεις SQL που χρησιμοποιούνται για να δημιουργήσουν τη βάση δεδομένων.

Αφού ολοκληρωθεί η δήλωση του σχήματος της βάσης δεδομένων, θα πρέπει να εφαρμοστούν μέθοδοι που δημιουργούν και διατηρούν τη βάση και τους πίνακες.

Όπως και στα αρχεία που αποθηκεύονται στη συσκευή (εσωτερικός χώρος αποθήκευσης), το Android διατηρεί τη βάση δεδομένων σε ιδιωτικό χώρο στο δίσκο που σχετίζεται με την εφαρμογή. Τα δεδομένα εκεί είναι ασφαλή, διότι από προεπιλογή η περιοχή αυτή δεν έχει πρόσβαση από άλλες εφαρμογές. [21]

### 2.7.1 Η κλάση SQLiteOpenHelper

Ένα χρήσιμο σύνολο από APIs βρίσκεται διαθέσιμο στη κλάση SQLiteOpenHelper. Όταν χρησιμοποιείται αυτή η κλάση για να την απόκτηση αναφορών στη βάση δεδομένων, το σύστημα εκτελεί δυνητικά τις μεγάλης διάρκειας λειτουργίες (για τη δημιουργία και την ενημέρωση της βάσης δεδομένων), μόνο όταν χρειάζεται και όχι κατά τη διάρκεια εκκίνησης της εφαρμογής. Το μόνο που χρειάζεται είναι η χρήση της μεθόδου getWritableDatabase ή getReadableDatabase.

Η κλάση SQLiteOpenHelper χρησιμοποιείται ως επέκταση (extends) μιας άλλης και έρχεται μαζί με κάποιες μεθόδους (onCreate, onUpgrade, onOpen, onDowngrade). Οι μέθοδοι αυτοί περιγράφονται στον Πίνακα 7. [21]

Πίνακας 7 Μέθοδοι κλάσης SQLiteOpenHelper

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ
onCreate	Καλείται όταν η βάση δεδομένων δημιουργείται για πρώτη φορά. Σε αυτήν γίνεται η δημιουργία των πινάκων και η αρχικοποίηση τους.
onOpen	Καλείται αφού ανοίξει η σύνδεση με τη βάση δεδομένων.
onUpgrade	Καλείται όταν η βάση δεδομένων πρέπει να αναβαθμιστεί. Η εκτέλεση της μεθόδου πρέπει να γίνει για να διαγράψει πίνακες (drop tables), να προσθέσει πίνακες (add tables), ή να κάνει οτιδήποτε άλλο χρειαστεί για την αναβάθμιση στο καινούριο σχήμα της βάσης.

### 2.7.2 Αποστολή ερωτημάτων κα διαχείριση αποτελεσμάτων

Στη κύρια κλάση ο προγραμματιστής μπορεί να φτιάξει μεθόδους ετοιμάζοντας συγκεκριμένα ερωτήματα (SQL queries) για τη διευκόλυνσή του. Για να μπορέσει ο προγραμματιστής να διαβάσει δεδομένα από τη βάση πρέπει να χρησιμοποιήσει αρχικά τη μέθοδο `query`, όπου περνάει σε αυτή το ερώτημα με τα κριτήρια και τις στήλες που θέλει. Τα αποτελέσματα του ερωτήματος επιστρέφουν σε μορφή `Cursor` αντικειμένου.

Στη συνέχεια ο προγραμματιστής για να δει μια συγκεκριμένη σειρά στο δρομέα (`Cursor`) πρέπει να χρησιμοποιήσει τις μεθόδους μετακίνησής του. Γενικά, πρέπει να ξεκινήσει με τη μέθοδο `moveToFirst`, που μεταφέρει το δρομέα στη πρώτη εγγραφή του αποτελέσματος. Έπειτα για κάθε γραμμή, μπορεί να διαβάσει τη τιμή της στήλης καλώντας τις αντίστοιχες μεθόδους του δρομέα (`getString`, `getLong`, `getInt` κ.τ.λ) [21]

### 2.7.3 Κριτήρια επιλογής και ασφάλεια

Για τη διαγραφή γραμμών από ένα πίνακα, ο προγραμματιστής πρέπει να παρέχει κριτήρια επιλογής τα οποία προσδιορίζουν τις γραμμές. Το API της βάσης δεδομένων του Android παρέχει ένα μηχανισμό για την δημιουργία κριτηρίων επιλογής, τα οποία προστατεύονται ενάντια σε επιθέσεις SQL injection.

Ο μηχανισμός διαχωρίζει τα ερωτήματα επιλογής σε όρους επιλογής και σε επιχειρήματα επιλογής. Οι όροι ορίζουν τις στήλες που θα κοιτάξει το ερώτημα. Τα επιχειρήματα είναι τιμές δεσμευμένες με τους όρους. Επειδή το αποτέλεσμα δεν διαχειρίζεται όπως μια απλή δήλωση SQL, δεν μπορεί να προσβληθεί από SQL injection. [21]

## 2.8 AndroidManifest

Το `AndroidManifest.xml` είναι ένα απαραίτητο αρχείο για κάθε εφαρμογή. Πρέπει να έχει ακριβώς αυτό το όνομα και να βρίσκεται στο κεντρικό φάκελο της εφαρμογής. Στο αρχείο αυτό περιγράφονται καθολικές πληροφορίες της εφαρμογής στο λειτουργικό σύστημα Android, πληροφορίες τις οποίες το σύστημα πρέπει να έχει πριν ξεκινήσει να τρέχει οποιοδήποτε κομμάτι κώδικα της εφαρμογής.

Μέσα σε όλα το αρχείο `AndroidManifest` κάνει τα παρακάτω:

- Ονομάζει τα πακέτα της Java για την εφαρμογή. Πρέπει να χρησιμοποιούνται ως μοναδικά αναγνωριστικά για την εφαρμογή.
- Περιγράφει τα συστατικά της εφαρμογής (Δραστηριότητες (Activities), Υπηρεσίες (Services), πάροχους περιεχομένου κ.τ.λ). Ονομάζει τις κλάσεις που

χρησιμοποιούν καθένα από τα συστατικά και δημοσιεύει τις ικανότητές τους. Αυτές οι δηλώσεις επιτρέπουν το λειτουργικό σύστημα να γνωρίζει ποία συστατικά είναι και κάτω από ποιους περιορισμούς μπορούν να ξεκινήσουν.

- Καθορίζει ποιες διεργασίες θα φιλοξενήσουν τα στοιχεία της εφαρμογής.
- Δηλώνει ποιες άδειες πρέπει να έχει η εφαρμογή για να έχει πρόσβαση σε προστατευόμενα μέρη του API και να αλληλεπιδρά με άλλες εφαρμογές.
- Δηλώνει επίσης τις άδειες που απαιτούνται από άλλους για να αλληλεπιδρά με στοιχεία της εφαρμογής.
- Παραθέτει τις βιβλιοθήκες τις οποίες η εφαρμογή πρέπει να συνδεθεί.
- Δηλώνει το ελάχιστο επίπεδο του Android API που απαιτεί η εφαρμογή. [5]

## 2.9 ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό είδαμε τα βασικά δομικά στοιχεία των εφαρμογών, τον κύκλο ζωής τους και κάποιες βασικές μεθόδους που χρησιμοποιούν. Όπως σε όλες τις εφαρμογές έτσι και σε αυτές πρέπει να μελετήσουμε τρόπους ασφάλειας των ευαίσθητων δεδομένων και των χρηστών. Στο επόμενο κεφάλαιο θα γίνει μια εισαγωγή στους τρόπους αυτούς.

## ΚΕΦΑΛΑΙΟ 3

### Ασφάλεια στο λειτουργικό Android

#### 3.1 ΕΙΣΑΓΩΓΗ

Το λειτουργικό σύστημα Android έχει ενσωματωμένα χαρακτηριστικά ασφάλειας, τα οποία μειώνουν σημαντικά τη συχνότητα των θεμάτων ασφαλείας των εφαρμογών. Το σύστημα είναι σχεδιασμένο έτσι ώστε οι προγραμματιστές να φτιάχνουν εφαρμογές με προεπιλεγμένες άδειες συστήματος και αρχείων, έτσι ώστε να αποφύγουν δύσκολες αποφάσεις για την ασφάλεια.

Κάποια από τα χαρακτηριστικά ασφαλείας του πυρήνα, που βοηθούν τους προγραμματιστές να φτιάξουν εφαρμογές περιλαμβάνουν:

- Το Android Application Sandbox, που απομονώνει τα δεδομένα και τον εκτελέσιμο κώδικα μιας εφαρμογής από άλλες.
- Ένα πλαίσιο εφαρμογής με ισχυρές υλοποιήσεις από συνηθισμένη λειτουργικότητα ασφάλειας, όπως κρυπτογραφία, άδειες και ασφαλής ενδοεπικοινωνία (IPC).
- Τεχνολογίες όπως ASLR, NX, ProPolice, safe\_iop, OpenBSD, dlmalloc, OpenBSD calloc και Linux mmap\_min\_addr μετριάζουν τον κίνδυνο που σχετίζεται με κοινά σφάλματα διαχείρισης μνήμης.
- Ένα κρυπτογραφημένο σύστημα αρχείων, μπορεί να ενεργοποιηθεί για να προστατέψει τα δεδομένα σε περίπτωση που χαθεί ή κλαπεί η συσκευή.
- Δικαιώματα παραχωρούνται στους χρήστες για να περιορίσουν την πρόσβαση τους σε χαρακτηριστικά του συστήματος και σε δεδομένα χρήστη.
- Καθορισμός δικαιωμάτων των εφαρμογών για να ελέγχονται τα δεδομένα τους ανά εφαρμογή. [18]

#### 3.2 Χρήση εσωτερικού χώρου αποθήκευσης

Ως προεπιλογή, τα αρχεία που δημιουργούνται στον εσωτερικό χώρο αποθήκευσης, είναι προσβάσιμα μόνο από την εφαρμογή που τα δημιούργησε. Αυτή η προστασία έχει εφαρμοστεί από το λειτουργικό σύστημα Android και είναι επαρκείς για τις περισσότερες εφαρμογές.

Για τη παροχής επιπρόσθετης προστασίας για ευαίσθητα δεδομένα, οι προγραμματιστές μπορούν να επιλέξουν την κρυπτογράφηση των τοπικών αρχείων χρησιμοποιώντας ένα κλειδί το οποίο δεν είναι άμεσα προσβάσιμο από την εφαρμογή. [18]

#### 3.3 Χρήση Δικαιωμάτων

Επειδή το λειτουργικό σύστημα Android κρύβει τις εφαρμογές του (Sandbox), τη μία από την άλλη, οι εφαρμογές πρέπει ρητά να μοιράζονται τους πόρους και τα δεδομένα τους. Αυτό το πετυχαίνουν δηλώνοντας τα δικαιώματα που χρειάζονται για επιπλέον ικανότητες που δεν παρέχονται από τη βασική λειτουργία του Sandbox, συμπεριλαμβανομένου πρόσβαση χαρακτηριστικά της συσκευής, όπως τη κάμερα. [6]

### 3.4 Αίτηση Δικαιωμάτων

Οι προγραμματιστές που αναπτύσσουν το λειτουργικό σύστημα Android προτείνουν την ελαχιστοποίηση του αριθμού των δικαιωμάτων που ζητούν οι εφαρμογές. Χωρίς να υπάρχει πρόσβαση σε ευαίσθητα δικαιώματα, μειώνεται ο κίνδυνος της ακούσιας κατάχρησης αυτών. Έτσι οι εφαρμογές γίνονται λιγότερο ευάλωτες για τους επιτιθέμενους. Γενικά, αν κάποια άδεια δεν είναι απαραίτητη για την εφαρμογή, δεν θα πρέπει να ζητηθεί. [6]

### 3.5 Χρήση Υπηρεσιών (Services)

Μια Υπηρεσία συχνά χρησιμοποιείται για να προμηθεύσει λειτουργικότητα προς χρήση από άλλες εφαρμογές. Κάθε κλάση Υπηρεσίας πρέπει να έχει τις αντίστοιχες δηλώσεις της στο αρχείο Manifest.

Από προεπιλογή, οι Υπηρεσίες δεν εξάγονται και δεν μπορούν να επικαλεστούν από άλλες εφαρμογές. Ωστόσο, αν μπουν κάποια φίλτρα προθέσεως (intent) στην δήλωση της Υπηρεσίας, τότε εξάγονται από προεπιλογή. Ο προγραμματιστής μπορεί να χειριστεί την εξαγωγή με δηλώσεις στο αρχείο Manifest.

Οι Υπηρεσίες μπορούν ακόμα να προστατευθούν χρησιμοποιώντας χαρακτηριστικό “android:permission” στο αρχείο Manifest. Βάση αυτού, άλλες εφαρμογές θα χρειαστούν αντίστοιχη ειδική δήλωση στο δικό τους Manifest αρχείο, για να μπορέσουν να ξεκινήσουν, να σταματήσουν ή να δεθούν με την Υπηρεσία. [6]

### 3.6 ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό είδαμε τον τρόπο ασφάλειας που εξασφαλίζουν οι εφαρμογές ζητώντας άδειες από τους χρήστες για τη προστασία τους. Στο επόμενο κεφάλαιο θα δούμε τον κύκλο ζωής μιας εφαρμογής Android.

## ΚΕΦΑΛΑΙΟ 4

### Ο κύκλος ζωής μιας εφαρμογής Android

#### 4.1 ΕΙΣΑΓΩΓΗ

Στις περισσότερες περιπτώσεις, κάθε εφαρμογή Android τρέχει σε μία ξεχωριστή Linux διεργασία. Αυτή η διεργασία δημιουργείται για την εφαρμογή, όταν κάποιο κομμάτι κώδικα χρειάζεται να εκτελεστεί και θα συνεχίσει να υπάρχει μέχρις ότου να μην είναι πλέον χρήσιμη και το σύστημα χρειαστεί να αξιοποιήσει τη μνήμη που καταλαμβάνει η διεργασία για άλλες εφαρμογές.

Ένα ασυνήθιστο και θεμελιώδες χαρακτηριστικό του λειτουργικού συστήματος Android, είναι ότι ο κύκλος ζωής μιας διεργασίας εφαρμογής, δεν είναι απευθείας ελεγχόμενος από την ίδια την εφαρμογή. Αντιθέτως, καθορίζεται από το σύστημα, μέσα από ένα συνδυασμό:

- ποια μέρη της εφαρμογής τρέχουν,
- πόσο σημαντικά είναι αυτά στο χρήστη και
- το σύνολο της μνήμης που είναι διαθέσιμο στο σύστημα.

Για να καθοριστεί ποια διεργασία πρέπει να τερματιστεί, σε περίπτωση έλλειψης μνήμης, το λειτουργικό σύστημα τοποθετεί κάθε διεργασία σε μία ιεραρχία. Η ιεραρχία αυτή είναι βασισμένη στα συστατικά που τρέχουν στην κάθε διεργασία και στο στάδιο που βρίσκονται.

Τα επίπεδα της διαδικασίας είναι (βάση σπουδαιότητας):

1. Μία διεργασία στο προσκήνιο (foreground process), είναι απαραίτητη για αυτό που κάνει ο χρήστης αυτή τη στιγμή. Διάφορα συστατικά εφαρμογής μπορούν να φέρουν στο προσκήνιο μια διεργασία με πολλούς τρόπους. Μία διεργασία θεωρείται ότι βρίσκεται στο προσκήνιο σε κάθε μία από τις παρακάτω περιπτώσεις:
  - εκτελεί μια Δραστηριότητα (Activity) στην οθόνη με την οποία ο χρήστης αλληλοεπιδρά (έχει καλεστεί η μέθοδος onResume).
  - υπάρχει έναν Δέκτη Εκπομπών (BroadcastReceiver) που τρέχει.
  - υπάρχει μια Υπηρεσία (Service) η οποία εκτελεί κώδικα (onCreate, onStart, onDestroy).

Θα υπάρξουν μόνο κάποιες διεργασίες στο σύστημα, που θα αναγκαστούν να τερματισθούν σαν τελευταία επιλογή, αν τα επίπεδα μνήμης είναι πάρα πολύ χαμηλά με αποτέλεσμα ούτε η ίδια η διεργασία να μη μπορεί να συνεχίσει να εκτελείται.



Γενικότερα, σε αυτό το σημείο, η συσκευή έχει φθάσει σε ένα όριο της μνήμης, οπότε αυτή η αντιμετώπιση είναι απαραίτητη προκειμένου να συνεχίσει το περιβάλλον χρήστη να αποκρίνεται.

2. Ορατή διεργασία (visible process). Είναι μία διεργασία που περιέχει μια Δραστηριότητα (Activity) ορατή στο χρήστη (on-screen) αλλά όχι στο προσκήνιο (έχει καλεστεί η μέθοδος onPause).
3. Διεργασία υπηρεσίας (service process). Είναι μία διεργασία που περιέχει μια Υπηρεσία (Service), η οποία έχει ξεκινήσει (μέθοδος startService). Αν και αυτές οι διαδικασίες δεν είναι άμεσα ορατές στο χρήστη, κάνουν γενικά πράγματα τα οποία ενδιαφέρουν το χρήστη (κατέβασμα ή ανέβασμα αρχείων από ίντερνετ, μουσική στο παρασκήνιο)
4. Διεργασία παρασκηνίου (background process). Είναι μια διεργασία που περιέχει μια Δραστηριότητα (Activity), η οποία δεν είναι πλέον ορατή στο χρήστη (έχει καλεστεί η μέθοδος onStop). Αυτές οι διεργασίες δεν έχουν άμεσο αντίκτυπο στην εμπειρία του χρήστη. Θα τερματιστούν μετά τα τις προηγούμενες τρεις κατηγορίες διεργασιών, σε περίπτωση χαμηλού επιπέδου μνήμης. Συνήθως υπάρχουν αρκετές από αυτές τις διεργασίες ενεργές και κρατούνται σε λίστα έτσι ώστε να εξασφαλισθεί ότι θα τερματιστεί η λιγότερο πρόσφατη διεργασία που χρησιμοποίησε ο χρήστης.
5. Άδεια διεργασία (empty process). Είναι μια διεργασία η οποία δεν περιέχει κανένα ενεργό στοιχείο εφαρμογής. Ο μόνος λόγος για να κρατηθεί ενεργή μια τέτοια διεργασία, είναι η βελτίωση του χρόνου εκκίνησης κάποιου συστατικού της εφαρμογής της, όταν αυτό πρέπει να εκτελεστεί. [16]

## ΚΕΦΑΛΑΙΟ 5

### Υπηρεσίες Google και Android APIs

#### 5.1 ΕΙΣΑΓΩΓΗ

Το Google προσφέρει μια ποικιλία από υπηρεσίες που βοηθούν το προγραμματιστή να υλοποιήσει ροές εσόδων, να διαχειριστεί διανομές εφαρμογών, να εντοπίσει τη χρήση των εφαρμογών και να ενισχύσει τις εφαρμογές με χαρακτηριστικά όπως χάρτες, είσοδος (google sign-in) και μηνύματα στο σύννεφο (cloud messaging).

Αν και οι υπηρεσίες του Google δεν περιλαμβάνονται στην πλατφόρμα του Android, υποστηρίζονται από τις περισσότερες συσκευές Android. Όταν χρησιμοποιούνται αυτές οι υπηρεσίες, η εφαρμογή μπορεί να διανεμηθεί στο Google Play σε όλες τις συσκευές που τρέχουν έκδοση λειτουργικού Android 2.3 και πάνω. [28]

## 5.2 Χάρτες Google

Οι προγραμματιστές έχουν τη δυνατότητα να συμπεριλάβουν τη δύναμη των Χαρτών της Google, στην εφαρμογή που αναπτύσσουν, με ένα ενσωματωμένο χάρτη. Ο χάρτης μπορεί να προσαρμοστεί με σημάδια (markers) και επικαλύψεις (overlays), ακόμα μπορεί να χειριστεί τη προοπτική του χρήστη, να σχεδιάσει γραμμές, σχήματα και πολλά ακόμα. [12]

## 5.3 Τοποθεσία Android (Android location API)

Περιλαμβάνει κλάσεις πλαισίου API (framework API) του Android, που καθορίζουν υπηρεσίες σχετικές με τοποθεσίες. Σε αυτό υπάρχουν διεπαφές (interfaces) και κλάσεις μέσα στις οποίες είναι:

- LocationListener (διεπαφή η οποία βρίσκει τη τοποθεσία του χρήστη),
- Address (κλάση η οποία αναπαριστά μια διεύθυνση),
- Location (κλάση που αντιπροσωπεύει γεωγραφικές τοποθεσίες),
- LocationManager (παρέχει πρόσβαση σε υπηρεσίες τοποθεσίας του συστήματος),
- Geocoder (διαχειρίζεται γεωγραφικές συντεταγμένες)

Η Google έχει δημιουργήσει ένα δικό της πακέτο που ανήκει στις Υπηρεσίες Google Play το Google Location Services API. Παροτρύνοντας τους προγραμματιστές να το χρησιμοποιούν αντί του android.location API, για χρήση τοποθεσίας στο Android. [11]

## 5.4 Geocoder

Είναι μια κλάση που χειρίζεται τις διαδικασίες Geocoding και αντίστροφου Geocoding. Ως Geocoding ορίζεται η διαδικασία που μετατρέπει διευθύνσεις ή άλλες περιγραφές μιας τοποθεσίας σε συντεταγμένες (γεωγραφικό πλάτος, γεωγραφικό μήκος). Reverse Geocoding είναι η διαδικασία που μετατρέπει συντεταγμένες σε μία μερική διεύθυνση.

Το ποσό των αποτελεσμάτων, σε μία διαδικασία αντίστροφης γεωγραφικά κωδικοποιημένης τοποθεσίας (Reverse Geocoding), μπορεί να διαφέρει. Για παράδειγμα ένα αποτέλεσμα μπορεί να περιέχει την πλήρη διεύθυνση του δρόμου, ενώ ένα άλλο μπορεί να περιέχει μόνο το όνομα της πόλης και τον ταχυδρομικό κώδικα.

Η κλάση Geocoder απαιτεί μια υπηρεσία παρασκηνίου, που δεν περιλαμβάνεται στο πλαίσιο του πυρήνα Android. Οι μέθοδοι ερωτήσεων της κλάσης Geocoder, επιστρέφουν μία άδεια λίστα αν δεν υπάρχει υπηρεσία παρασκηνίου στην πλατφόρμα. [10]

## 5.5 Εγκατάσταση του Google Play Service (Eclipse IDE)

Για την ανάπτυξη μιας εφαρμογής που να χρησιμοποιεί τις υπηρεσίες του Google (Google Play services APIs), χρειάζεται να εγκατασταθεί στο πρότζεκτ το Google Play services.

Για να γίνει διαθέσιμο το Google Play services APIs σε μία εφαρμογή πρέπει να γίνουν τα παρακάτω βήματα:

1. αντιγραφή του πρότζεκτ βιβλιοθήκης <android-sdk>/extras/google/google\_play\_services/libproject/google-play-services\_lib/ στο φάκελο που βρίσκεται το κυρίως πρότζεκτ της εφαρμογής.

2. εισαγωγή (import) του παραπάνω πρότζεκτ βιβλιοθήκης στο χώρο εργασίας του Eclipse.
3. στο πρότζεκτ της εφαρμογής, πρέπει να δημιουργηθεί αναφορά στο Google Play services.
4. Μετά την εισαγωγή της βιβλιοθήκης Google Play service ως εξάρτηση για το πρότζεκτ της εφαρμογής, πρέπει να οριστεί και στο αρχείο Manifest ως παιδί στο στοιχείο <application>:  

```
<meta-data
    android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />
```

Έπειτα από την υλοποίηση των παραπάνω βημάτων, το πρόγραμμα είναι σε θέση να χρησιμοποιήσει τα χαρακτηριστικά του Google Play services APIs. [20]

## 5.6 Δημιουργία API Project στη κονσόλα Google APIs

Για τη δημιουργία νέου Project στο Google APIs πρέπει να ακολουθηθούν τα παρακάτω βήματα:

1. μετάβαση στην ιστοσελίδα google APIs Console (<https://code.google.com/apis/console/?noredirect#project:979363518782>)
2. αν ο χρήστης δεν έχει χρησιμοποιήσει ξανά την κονσόλα Google APIs, του ζητάτε να δημιουργήσει ένα πρότζεκτ.
3. αν ο χρήστης χρησιμοποιεί είδη τη κονσόλα, θα μεταβεί αμέσως σε μια λίστα με όλα τα πρότζεκτ και τις διαθέσιμες υπηρεσίες.
4. ο χρήστης επιλέγει τη κατηγορία Υπηρεσίες (Services) από την αριστερή μπάρα πλοήγησης.
5. από τη λίστα που εμφανίζονται οι υπηρεσίες επιλέγει να ενεργοποιήσει (on) την Google Maps Android API v2.
6. ο χρήστης έπειτα πρέπει να συμφωνήσει με τους όρους.

## 5.7 Απόκτηση Πιστοποιητικού fingerprint SHA1 (Eclipse IDE)

Από το περιβάλλον Eclipse ο χρήστης επιλέγει:

1. Windows
2. Preferences
3. Android
4. Build

το SHA1 fingerprint εμφανίζεται στο αντίστοιχο πεδίο.

## 5.8 Απόκτηση κλειδιού Google Maps API

Αν η εφαρμογή είναι καταχωρημένη με την υπηρεσία Google Maps Android API v2, τότε μπορεί να ζητηθεί ένα API κλειδί (μπορούν να καταχωρηθούν παραπάνω από ένα κλειδιά ανά πρότζεκτ).

1. Μετάβαση στην κονσόλα Google APIs (<https://code.google.com/apis/console/?noredirect#project:979363518782>)

- στη σελίδα με της Υπηρεσίες, γίνεται έλεγχος ότι είναι ενεργό το “Google Maps Android API v2”
- στην αριστερή μπάρα πλοήγησης ο χρήστης επιλέγει “API Access”.
- στη σελίδα που εμφανίζεται επιλέγει “Create New Android Key...”
- στο εμφανιζόμενο παράθυρο διαλόγου, εισάγει το SHA-1 fingerprint (της εφαρμογής του) έπειτα ένα κώμα και μετά το όνομα πακέτου της εφαρμογής του.  
2A:74:FE:74:A1:3D:76:B3:21:2A:46:5A:05:BB:88:A0:8A:  
12:05:14;com.example.planmyday\_ptyxiaki
- η κονσόλα Google APIs εμφανίζει το κλειδί API σαράντα χαρακτήρων  
API key: AlzaSyBxJYTVAsc-uarMBU6i-sSitc1qbm8TwnU

## 5.9 Google Maps Android API v2

Για να γίνει δυνατή η χρήση αυτού του API, πρέπει να δημιουργηθεί το αντίστοιχο κλειδί. Η συνολική διαδικασία για την πρόσθεση ενός Google χάρτη σε μία εφαρμογή γίνεται ακολουθώντας τα παρακάτω βήματα [27]:

- εγκατάσταση του Android SDK
- Κατέβαση και ρύθμιση του Google Play services SDK, που περιλαμβάνει το Google Maps Android APIs.
- δημιουργία κλειδιού. Πρέπει να καταχωρηθεί το πρότζεκτ στην κονσόλα Google APIs και να ληφθεί το αντίστοιχο πιστοποιητικό για την εφαρμογή.
- παραμετροποίηση των ρυθμίσεων στο αρχείο Manifest της εφαρμογής ως παιδί στο στοιχείο <application>:  

```
<meta-data  
    android:name="com.google.android.maps.v2.API_KEY"  
    android:value="AlzaSyBxJYTVAsc-uarMBU6i-sSitc1qbm8TwnU" />
```
- πρόσθεση χάρτη στην εφαρμογή.
- έκδοση της εφαρμογής.

### 5.9.1 Χρήση χαρτών Google

Για τη πρόσβαση στους διακομιστές των χαρτών της Google, απαραίτητη προϋπόθεση είναι η πρόσθεση ενός κλειδιού Maps API στην εφαρμογή. Το κλειδί είναι δωρεάν και μπορεί να χρησιμοποιηθεί για κάθε εφαρμογή που χρησιμοποιεί το Maps API και υποστηρίζει απεριόριστο αριθμό χρηστών.

Ο προγραμματιστής λαμβάνει το κλειδί από τη κονσόλα Google APIs με τη χορήγηση του πιστοποιητικού της εφαρμογής και το όνομα πακέτου της. Έπειτα καταχωρεί το κλειδί στο αρχείο Manifest της εφαρμογής.

Στο αρχείο Manifest θα πρέπει επίσης να υπάρχουν οι παρακάτω άδειες:

- android.permission.Internet, χρησιμοποιείται από το API για να κατεβάσει του χάρτες από τους διακομιστές της Google.
- android.permission.ACCESS\_NETWORK\_STATE, επιτρέπει στο API να ελέγχει τη κατάσταση σύνδεσης με σκοπό να καθορίσει αν μπορεί να κατεβάσει δεδομένα.

- `android.permission.WRITE_EXTERNAL_STORAGE`, επιτρέπει στο API τη προσωρινή αποθήκευση δεδομένων των χαρτών στον αποθηκευτικό χώρο της συσκευής. [27]

## 5.10 Web Service

Όπως ορίζει η IBM, τα *web services* είναι μια τεχνολογία που επιτρέπει στις εφαρμογές να επικοινωνούν μεταξύ τους ανεξαρτήτως πλατφόρμας και γλώσσας προγραμματισμού. Ένα *web service* είναι μια διεπαφή λογισμικού (*software interface*) που περιγράφει μια συλλογή από λειτουργίες οι οποίες μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων XML. Χρησιμοποιεί πρότυπα βασισμένα στη γλώσσα XML για να περιγράψει μία λειτουργία (*operation*) προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή. [35]

### 5.10.1 Google Maps API Web Services

Τα *Google Maps Web Services* είναι μια συλλογή από HTTP διεπαφές. Σκοπό έχουν την παροχή γεωγραφικών δεδομένων για τις εφαρμογές χαρτών της Google. Το *Google Maps API* παρέχει αυτές τις Υπηρεσίες Ιστού σαν διεπαφές για ζητούμενα *Maps API* δεδομένα, σε εξωτερικές υπηρεσίες και τη χρήση τους στις εφαρμογές χαρτών.

Αυτές οι υπηρεσίες έχουν σχεδιαστεί για να χρησιμοποιηθούν σε συνδυασμό με χάρτη (*Google Maps*), σύμφωνα με το *Maps API Terms of Service License Restrictions*.

Οι υπηρεσίες ιστού αυτές χρησιμοποιούν αιτήματα HTTP σε συγκεκριμένες σελίδες (URLs), περνώντας URL παραμέτρους ως δεδομένα στην υπηρεσία. Γενικά, αυτές οι υπηρεσίες επιστρέφουν δεδομένα, στα HTTP αιτήματα, με μορφή JSON ή XML. Αυτά τα δεδομένα μπορούν να αναλυθούν και να επεξεργαστούν από όποια εφαρμογή τα ζήτησε.

Ένα συνήθης αίτημα σε κάποια από τις Υπηρεσίες Ιστού της *Google Maps API* έχει την παρακάτω μορφή:

- <https://maps.googleapis.com/maps/api/service/output?parameters>

Όπου το *service* αναφέρεται σε κάποια συγκεκριμένη υπηρεσία ιστού της Google και το *output* προσδιορίζει τη μορφή της απάντησης (συνήθως *json* ή *xml*). [28]

### 5.10.2 To Google Directions API

Το *Google Directions API* είναι μια υπηρεσία που υπολογίζει οδηγίες ανάμεσα σε τοποθεσίες χρησιμοποιώντας HTTP αιτήματα. Μπορούν να ζητηθούν οδηγίες για διάφορους τρόπους μεταφοράς (*MMM*, οδήγηση, περπάτημα, ποδηλασία). Οι οδηγίες μπορούν να προσδιορίσουν την προέλευση, τους προορισμούς και τα σημεία σαν κείμενο είτε σαν συντεταγμένες (γεωγραφικό πλάτος, γεωγραφικό μήκος). Το *Directions API* μπορεί να επιστρέψει πολλαπλές οδηγίες χρησιμοποιώντας μια σειρά από σημεία.

Αυτή η υπηρεσία έχει σχεδιαστεί γενικά, για τον υπολογισμό οδηγιών για γνωστές εκ των προτέρων διευθύνσεις και την τοποθέτηση τους στο χάρτη μιας εφαρμογής. Αυτή η υπηρεσία δεν έχει σχεδιαστεί για να ανταποκρίνεται σε πραγματικό χρόνο σε κάποια είσοδο του χρήστη. [29]

### 5.10.3 Όρια Χρήσης

Το *Direction API* έχει τα παρακάτω όρια:

- 2,500 αιτήσεις διευθύνσεων ανά 24 ώρες.

- Μέχρι 8 σημεία επιτρέπονται σε κάθε αίτηση. Τα σημεία δεν είναι διαθέσιμα για οδηγίες με MMM.
- 2 αιτήσεις το δευτερόλεπτο

Google Maps API for Work πελάτες:

- 100,000 αιτήσεις διευθύνσεων ανά 24 ώρες.
- 23 σημεία επιτρέπονται σε κάθε αίτηση. Τα σημεία δεν είναι διαθέσιμα για οδηγίες με MMM.
- 10 αιτήσεις το δευτερόλεπτο

Τα Directions API URLs περιορίζονται περίπου στους 2000 χαρακτήρες. Επίσης το Direction API μπορεί να χρησιμοποιηθεί μόνο σε συνδυασμό με ένα Google Map. Πρέπει δηλαδή τα αποτελέσματα να παρουσιάζονται πάνω σε ένα χάρτη της Google. [29]

#### 5.10.4 Παράμετροι Αιτήσεων

Κάποιοι παράμετροι απαιτούνται για την αποστολή αιτήματος, ενώ κάποιοι άλλοι είναι προαιρετικοί. Όπως και σε κάθε URL έτσι και εδώ οι παράμετροι χωρίζονται με χρησιμοποιώντας το χαρακτήρα «&». Στον Πίνακα 8 περιγράφονται κάποιες από τις παραμέτρους.

Πίνακας 8 παράμετροι αιτήσεων

ΠΑΡΑΜΕΤΡΟΣ	ΠΕΡΙΓΡΑΦΗ	ΑΠΑΙΤΟΥΜΕΝΟ
origin	Διεύθυνση αφετηρίας, σε κείμενο ή συντεταγμένες	Ναι
destination	Διεύθυνση προορισμού, σε κείμενο ή συντεταγμένες	Ναι
Mode	Προσδιορίζει τον τρόπο του μεταφορικού μέσου	Όχι
Language	Προσδιορίζει τη γλώσσα στην οποία θα επιστρέψει το αποτέλεσμα	Όχι
Units	Ορίζει τη μονάδα μέτρησης που θα επιστρέψει στο αποτέλεσμα	Όχι
departure_time	Ορίζει την ώρα αναχώρησης	Όχι
arrival_time	Ορίζει την ώρα άφιξης	Όχι

Στο URL μπορεί να μπει ένα από τα δυο, ώρα άφιξης ή ώρα αναχώρησης. Διότι το ένα εξαρτάται από το άλλο. Αν δεν οριστεί κανένα εκ των δυο τότε θεωρείται ώρα αναχώρησης η τρέχουσα ώρα. [29]

### 5.10.5 Οδηγίες Συγκοινωνίας

Αυτή η παράμετρος «Transit» ζητάει οδηγίες μέσω διαδρομές συγκοινωνιών (όπου είναι διαθέσιμες). Όταν ως τρόπος μεταφοράς οριστούν οι συγκοινωνίες, μπορεί να ορισθεί και ένα χρόνος άφιξης ή αναχώρησης, στη ή από τη στάση αντίστοιχα. Αν δεν οριστεί μπαίνει ως προεπιλεγμένη ώρα αναχώρησης η τρέχουσα ώρα.

Οι οδηγίες συγκοινωνίας επιστρέφουν επιπρόσθετη πληροφορία που δεν είναι σχετική με άλλους τρόπους μεταφοράς. Τα πεδία που μπορεί να περιέχει μια απάντηση με ένα αντικείμενο «transit\_details» περιγράφονται στον Πίνακα 9. [29]

Πίνακας 9 πεδία αντικειμένου συγκοινωνιών

ΠΕΔΙΟ	ΠΕΡΙΓΡΑΦΗ
arrival_stop	Περιέχει πληροφορίες για τη στάση αναχώρησης (όνομα, συντεταγμένες)
departure_stop	Περιέχει πληροφορίες για τη στάση άφιξης (όνομα, συντεταγμένες)
arrival_time	Περιέχει πληροφορίες για την ώρα αναχώρησης
departure_time	Περιέχει πληροφορίες για την ώρα άφιξης
num_stops	Περιέχει τον αριθμό στάσεων από τη στάση αναχώρησης μέχρι τη στάση άφιξης
line	Περιέχει πληροφορίες για το λεωφορείο (πλήρες όνομα, κωδικός λεωφορείου)
vehicle	Περιέχει πληροφορίες για τον τύπο του οχήματος (λεωφορείο)

## 5.11 ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό είδαμε κάποιες χρήσιμες υπηρεσίες ιστού της Google αλλά και κάποια Android APIs, που μπορούν να χρησιμοποιήσουν οι προγραμματιστές για να υλοποιήσουν τις επιθυμητές λειτουργίες. Στο επόμενο κεφάλαιο θα μιλήσουμε για κάποιες βιβλιοθήκες υποστήριξης του Android που βοηθούν στη σχεδίαση των εφαρμογών.

## ΚΕΦΑΛΑΙΟ 6

### Βιβλιοθήκες Υποστήριξης Android

#### 6.1 ΕΙΣΑΓΩΓΗ

Το πακέτο Android Support Library, περιέχει διάφορες βιβλιοθήκες που μπορούν να συμπεριληφθούν σε μια εφαρμογή. Κάθε μια από αυτές τις βιβλιοθήκες υποστηρίζει ένα συγκεκριμένο εύρος εκδόσεων Android και ένα συγκεκριμένο σύνολο χαρακτηριστικών.

#### 6.2 v7 Support Libraries

Υπάρχουν πολλές βιβλιοθήκες που σχεδιάστηκαν για να χρησιμοποιούνται με την έκδοση Android 2.1 (API επιπέδου 7) και πάνω. Αυτές οι βιβλιοθήκες παρέχουν συγκεκριμένο σύνολο χαρακτηριστικών και μπορούν να συμπεριληφθούν σε μια εφαρμογή ανεξάρτητα η μια από την άλλη. [22]

##### 6.2.1 v7 Appcompat Library

Αυτή η βιβλιοθήκη, προσθέτει υποστήριξη για σχεδιαστικά πρότυπα του περιβάλλοντος χρήστη της Μπάρα Δράσης (Action Bar). Περιέχει επίσης υποστήριξη για χρήση υλικού σχεδιασμού (material design) στο περιβάλλον χρήστη.

Μερικές βασικές κλάσεις που περιλαμβάνονται στη βιβλιοθήκη v7 Appcompat περιγράφονται στον Πίνακα 10. [22]

Πίνακας 10 κλάσεις της βιβλιοθήκης v7 Appcompat

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
ActionBar	Η κλάση αυτή παρέχει, σχεδιαστικά πρότυπα για τη σχεδίαση περιβάλλοντος χρήστη της Μπάρας Δράσης (Action Bar).
ActionBarActivity	Καλείται αφού ανοίξει η σύνδεση με τη βάση δεδομένων. Προσθέτει μια κλάση Δραστηριότητας που πρέπει να χρησιμοποιείται, ως βάση, για κάθε Δραστηριότητα (Activity) που τη χρησιμοποιεί
ShareActionProvider	Προσθέτει υποστήριξη για μια τυποποιημένη δράση κοινής χρήσης (όπως email, κοινή χρήση σε κοινωνικά δίκτυα) που περιλαμβάνεται στην Μπάρα Δράσης (Action Bar).



### 6.2.2 Εισαγωγή βιβλιοθήκης στο Project

Η v7 Appcompat βιβλιοθήκη βρίσκεται στον κατάλογο `<sdk>/extras/android/support/v7/appcompat/`. Η βιβλιοθήκη περιέχει πόρους για το περιβάλλον χρήστη. Για να συμπεριληφθεί στο πρότζεκτ της εφαρμογής πρέπει:

1. Να υπάρχει το Android Support Library κατεβασμένο από το SDK Manager.
2. Δημιουργία πρότζεκτ βιβλιοθήκης
  - α. επιλογή File > Import
  - β. επιλογή Existing Android Code Into Workspace
  - γ. αναζήτηση του καταλόγου που βρίσκεται το SDK και επιλογή του Support Library φακέλου (π.χ. `<sdk>/extras/android/support/v7/appcompat/`)
  - δ. επιλογή Finish για την εισαγωγή του πρότζεκτ.
  - ε. στο καινούριο πρότζεκτ βιβλιοθήκης (android-support-v7-appcompat), κάτω από το libs/ φάκελο, δεξί κλικ σε κάθε JAR αρχείο και επιλογή Build Path > Add to Build Path.
  - στ. δεξί κλικ στο φάκελο του πρότζεκτ βιβλιοθήκης και επιλογή Build Path > Configure Build Path.
  - ζ. στη καρτέλα Order and Export, επιλογή των JAR αρχείων που μπήκαν πριν στο Build Path, για να είναι διαθέσιμα στα πρότζεκτ που εξαρτώνται σε αυτή τη βιβλιοθήκη.
  - η. αποεπιλογή Android Dependencies.
  - θ. επιλογή OK, για την ολοκλήρωση.

Τώρα υπάρχει το πρότζεκτ βιβλιοθήκης (appcompat) που μπορεί να χρησιμοποιηθεί με ένα ή περισσότερα πρότζεκτ εφαρμογής.

Για να προστεθεί η βιβλιοθήκη σε ένα συγκεκριμένο πρότζεκτ εφαρμογής πρέπει να ακολουθηθούν τα παρακάτω βήματα:

1. στο "Project Explorer", δεξί κλικ στο πρότζεκτ εφαρμογής και επιλογή "Properties"
2. επιλογή κατηγορίας Android
3. επιλογή "Add" στις βιβλιοθήκες
4. επιλογή του πρότζεκτ βιβλιοθήκης > "OK"
5. στο παράθυρο "Properties" > "OK"

## ΚΕΦΑΛΑΙΟ 7

### Η Εφαρμογή PlanMyDay

#### 7.1 ΕΙΣΑΓΩΓΗ

Η εφαρμογή ονομάστηκε ενδεικτικά “PlanMyday” αφού ο κύριος στόχος της είναι να δώσει στους χρήστες τη δυνατότητα να κάνουν το πλάνο της ημέρα τους!

Έχει αναπτυχθεί με το εργαλείο eclipse και αναφέρεται σε Android 4.4.2 και πάνω. Η επιλογή έχει γίνει βάσει των προ απαιτούμενων λειτουργιών και γραφικών για τη καλύτερη λειτουργία της εφαρμογής και το καλύτερο αποτέλεσμα προς τους χρήστες.

Η εφαρμογή έχει σχεδιαστεί με κομμάτια Ενοποιημένης Διαδικασίας (Unified Process ή UP) αλλά και ευέλικτων διαδικασιών ανάπτυξης. Τα στοιχεία που χρησιμοποιήθηκαν από τον Ακραίο Προγραμματισμό (Extreme Programming ή XP), περιγράφονται σε αυτή την παράγραφο. Η ανάπτυξη έγινε επαναληπτικά με επαναλήψεις χρονικής διάρκειας περίπου μιας εβδομάδας. Κάθε εβδομάδα ολοκληρωνόταν ένα μέρος της εφαρμογής και έπειτα χρησιμοποιούταν σενάρια χρήσης για testing. Κατά μεγαλύτερα διαστήματα, περίπου τριών μηνών, η εφαρμογή παρουσιαζόταν στον επιβλέπων καθηγητή κ. Ιγνάτιο Δεληγιάννη, όπου μαζί αξιολογούσαμε την πορεία του έργου σε βάθος χρόνου, αλλά και σχεδιάζαμε νέους στόχους.

Κατά τον αρχικό σχεδιασμό της εφαρμογής, οι απαιτήσεις διατυπώθηκαν με τη μορφή περιπτώσεων χρήσης και περιεγράφηκαν αναλυτικά. Κατά τη διάρκεια της ανάπτυξης όμως, διατυπώθηκαν και νέες απαιτήσεις μέσω διηγήσεων από τους χρήστες, χωρίς περαιτέρω ανάλυση. Επίσης, κατασκευάστηκαν οπτικά μοντέλα σε UML κυρίως στη σχεδίαση της εφαρμογής. [33] [37]

#### 7.2 Πως λειτουργεί

Η Πτυχιακή Εργασία και κατ' επέκταση η εφαρμογή "PlanMyDay", όπως φαίνεται και από τον τίτλο της, παραπέμπει στην ειδοποίησή του χρήστη για κάποιες προγραμματισμένες διαδρομές. Έχει σχεδιαστεί έτσι ώστε ο χρήστης να μπορεί να προγραμματίσει την καθημερινότητα του ενώ μετακινείται με Μέσα Μαζικής Μεταφοράς.

Εκμεταλλεύομενος μιας Υπηρεσίας Ιστού της Google που περιγράψαμε στο Κεφάλαιο 5.10 και των πληροφοριών από την ιστοσελίδα του ΟΑΣΘ κατάφερα να αναπτύξω μια εφαρμογή σε λειτουργικό σύστημα Android. Η εφαρμογή αυτή, συνδυάζει τις πληροφορίες των διαδρομών από την Υπηρεσία Ιστού της Google και τις μετατρέπει σε δεδομένα του ΟΑΣΘ. Έπειτα, επιστρέφει τη ζητούμενη πληροφορία στο χρήστη μέσω ειδοποίησης.

## 7.3 Συστατικά Στοιχεία της Εφαρμογής

Η παρούσα εφαρμογή περιέχει τα παρακάτω συστατικά στοιχεία:

- Δραστηριότητα (Activity)
- Τεμάχιο (Fragment)
- Υπηρεσία (Service)
- Ειδοποίηση (Notification)
- Προσαρμογέας (Adapter)
- Βάση Δεδομένων (SQLite)
- AndroidManifest

Τα στοιχεία αυτά περιεγράφηκαν στο Κεφάλαιο 2. Εδώ θα γίνουν κάποιες συγκεκριμένες αναφορές για τη χρήση τους στην εφαρμογή της Πτυχιακής Εργασίας.

### 7.3.1 Τεμάχιο (Fragment)

Στη παρούσα Πτυχιακή Εργασία, Τεμάχια (Fragments) χρησιμοποιήθηκαν για τη δημιουργία ενός μενού διαφανειών. Σε μία Δραστηριότητα (Activity) εναλλάσσονται τρία Τεμάχια (Fragments) που αντιστοιχούν σε αντίστοιχες κατηγορίες (Ημερήσια Διαχείριση, Αγαπημένα, Πρόσφατα). Επίσης Τεμάχια χρησιμοποιήθηκαν στους επιλογής ημερομηνίας και ώρας.

### 7.3.2 Ειδοποίηση (Notification)

Όπως προαναφέρθηκε στο Κεφάλαιο 2, μια Υπηρεσία (Service) μπορεί να εκκινήσει μια Ειδοποίηση (Notification). Στην Εφαρμογή της παρούσας Πτυχιακής Εργασίας, γίνεται χρήση Ειδοποίησης (Notification) μέσω μιας Υπηρεσίας (Service). Η Υπηρεσία δουλεύει επ'αόριστον και ανανεώνει την Ειδοποίηση κάθε ένα λεπτό. Η Ειδοποίηση περιέχει τα απαραίτητα συστατικά (τίτλο, κείμενο, δράση) και επίσης χρησιμοποιεί ένα κουμπί για την ακύρωση της Υπηρεσίας.

### 7.3.3 Προσαρμογέας (Adapter)

Στην εφαρμογή της Πτυχιακής Εργασίας γίνεται χρήση τριών Adapter, ένας για κάθε κατηγορία (Ημερήσια Διαχείριση, Αγαπημένα, Πρόσφατα) και ενός ακόμα για τη παρουσίαση στοιχείων στο μενού. Οι τρεις πρώτοι παίρνουν τα δεδομένα τους από τη Βάση Δεδομένων ενώ ο τελευταίος από μια λίστα. Κάθε ένας από τους Προσαρμογείς χρησιμοποιεί διαφορετικό περιβάλλον για τη παρουσίαση των δεδομένων στο χρήστη.

### 7.3.4 Βάση Δεδομένων (SQLite)

Στην εφαρμογή της Πτυχιακής Εργασίας γίνεται χρήση δύο βάσεων δεδομένων, μία εξωτερική (ΟΑΣΘ) και μια εσωτερική (user). Και οι δύο είναι υλοποιημένες με τη βοήθεια της κλάσης SQLiteOpenHelper και χρησιμοποιούν τις βασικές λειτουργίες της (create, update, delete).

Η βάση δεδομένων που βρίσκεται σε εξωτερικό αρχείο χρειάζεται να αντιγραφεί (copyDataBase) πρώτα από το φάκελο που υπάρχει στο σύστημα και έπειτα να ανοίξει μια σύνδεση προς αυτή (openDatabase). Έχει έτοιμους τους πίνακες και τα δεδομένα σε αυτούς.

Αντίθετα η βάση δεδομένων του χρήστη είναι εσωτερική και δημιουργείται κάθε φορά που η εγκαθίσταται η εφαρμογή σε μια συσκευή. Οι πίνακες δημιουργούνται με τη δημιουργία

της βάσης (onCreate), ενώ τα δεδομένα αποθηκεύονται σε πραγματικό χρόνο κατά τη διάρκεια εκτέλεσης της εφαρμογής όταν αυτό χρειαστεί.

Και στις δυο περιπτώσεις υπάρχουν μέθοδοι επιλογής με τα απαραίτητα κριτήρια, για τη χρήση των δεδομένων όπου και όταν αυτά χρειαστούν.

## 7.4 Χρήση Υπηρεσιών ΟΑΣΘ

Ο Οργανισμός Αστικών Συγκοινωνιών Θεσσαλονίκης (ΟΑΣΘ), παρέχει στους χρήστες κάποιες χρήσιμες πληροφορίες, όπως τα δρομολόγια των λεωφορείων, την άφιξη αυτών σε κάποια στάση. Αυτές οι πληροφορίες είναι διαθέσιμες μέσω μιας ιστοσελίδας του οργανισμού (10) για κινητά τηλέφωνα (mobile site).

Ο οργανισμός ενώ έχει αναπτύξει χρήσιμες λειτουργίες για τους χρήστες του μέσω της ιστοσελίδας του, δε δίνει τη δυνατότητα στους προγραμματιστές να εκμεταλλευτούν τις πληροφορίες του για επεξεργασία ή αναμετάδοση, αν δεν έχουν λάβει την απαραίτητη άδεια.

Ως αποτέλεσμα αυτού χρειάστηκε άδεια (Παράρτημα 1) από τον ΟΑΣΘ για τη δημιουργία της εφαρμογής. Την άδεια μας χορήγησε ο οργανισμός μετά από αίτηση η οποία περιλάμβανε αποδεικτικά στοιχεία από τον επιβλέπων καθηγητή κ. Ιγνάτιο Δεληγιάννη και ισχύει μόνο για εκπαιδευτικούς σκοπούς.

Παρόλα αυτά δεν υπήρχε κάποια περαιτέρω Υπηρεσία Ιστού (Web Service), API ή κάποια βάση δεδομένων παρεχόμενη από τον οργανισμό. Αυτό είχε ως αποτέλεσμα την υλοποίηση ενός προγράμματος το οποίο δημιουργεί αυτόματα μια βάση δεδομένων 35.000 εγγραφών περίπου με τις ονομασίες των οχημάτων του ΟΑΣΘ, τους κωδικούς των διαδρομών των λεωφορείων, των στάσεων κοκ. Για το πρόγραμμα θα μιλήσουμε σε επόμενο κεφάλαιο αναλυτικά.

Έτσι κατέχοντας την απαραίτητη πληροφορία έμενε η παρουσίαση της στο χρήστη μέσα από ένα εύχρηστο και φιλικό περιβάλλον, το οποίο προσέφερε το λειτουργικό σύστημα Android.

## 7.5 Περιπτώσεις χρήσης

Ένας τρόπος τόσο για την περιγραφή των απαιτήσεων όσο και για την επιβεβαίωση ότι οι μηχανικοί λογισμικού κατανοούν με τον ίδιο τρόπο τη λειτουργία του υπό ανάπτυξη λογισμικού είναι η διατύπωση σεναρίων χρήσης. Μέσω αυτών έχουμε τον τελικό χρήστη του συστήματος ή μια εξωτερική του συστήματος συσκευή να αλληλοεπιδρά με το σύστημα, με σκοπό να λάβει μια υπηρεσία. Η υπηρεσία αυτή μπορεί να είναι η πληροφόρηση για το αποτέλεσμα μιας διεργασίας ή η εκτέλεση μιας πράξης, όπως για παράδειγμα στην περίπτωση μας, να λάβει την ώρα άφιξης του λεωφορείου στη στάση για να φτάσει στο τελικό προορισμό του. Το σενάριο ξεκινά με ένα ερέθισμα του τελικού χρήστη στο σύστημα, την εκτέλεση μιας σειράς πράξεων από το σύστημα και την ανακοίνωση του αποτελέσματος στον τελικό χρήστη.

Ο τρόπος περιγραφής της λειτουργικότητας του λογισμικού βάσει σεναρίων συστηματοποιείται με τις περιπτώσεις χρήσης. Οι περιπτώσεις χρήσης (use cases), είναι μια διαδομένη μέθοδος για την αποτύπωση των λειτουργικών απαιτήσεων του λογισμικού.

Το σύνολο των περιπτώσεων χρήσης περιγράφει τη λειτουργικότητα που παρέχεται από το σύστημα. Κάθε περίπτωση χρήσης παρέχει μια υπηρεσία σε κάποιο τελικό χρήστη του λογισμικού. Οι περιπτώσεις χρήσης τεκμηριώνονται σε φυσική γλώσσα ακολουθώντας κάποιο προκαθορισμένο πρότυπο. Παράλληλα, η UML παρέχει τα διαγράμματα περιπτώσεων χρήσης (use case diagrams) για την οπτική απεικόνιση της λειτουργικότητας του συστήματος και του περιβάλλοντός του. [36]

### 7.5.1 Βασικά συστατικά ενός διαγράμματος περιπτώσεων χρήσης

Actor είναι μια οντότητα εκτός του συστήματος που αλληλοεπιδρά με αυτό. Ένας actor αντιπροσωπεύει συνήθως κάποιο χρήστη του συστήματος, αλλά μπορεί να είναι και ένας οργανισμός ή ακόμη και ένα άλλο σύστημα. Στην περίπτωση μας έχουμε τρεις Actor :

- ένα χρήστη, που είναι ο Βασικός Actor της συσκευής που χρησιμοποιεί την εφαρμογή,
- τον οργανισμό αστικών συγκοινωνιών ΟΑΣΘ, που μπορεί να θεωρηθεί ως σύστημα (10) από όπου παίρνουμε πληροφορίες για τα λεωφορεία.
- το web service της google που είναι και αυτο ένα σύστημα και μας παρέχει πληροφορίες για τις διαδρομές.

Η περίπτωση χρήσης συμβολίζεται με έλλειψη και η γραμμή που συνδέει τον actor με την περίπτωση χρήσης υποδηλώνει την αλληλεπίδραση τους. Η γραμμή είναι ο συμβολισμός της συσχέτισης (association) της UML και θεωρούμε ότι δηλώνει μια σχέση επικοινωνίας μεταξύ του actor και της περίπτωσης χρήσης. [36]

### 7.5.2 Σχέσεις περιπτώσεων χρήσης

Οι απαιτήσεις ενός συστήματος λογισμικού περιγράφονται με αρκετές περιπτώσεις χρήσης. Εφόσον οι περιπτώσεις χρήσης περιγράφουν μία ολοκληρωμένη λειτουργικότητα περιμένουμε ότι κάθε περίπτωση χρήσης είναι ανεξάρτητη από τις υπόλοιπες. Ορισμένες φορές όμως κάποιες περιπτώσεις χρήσης μπορεί να σχετίζονται. Οι διαφορετικές σχέσεις μεταξύ των περιπτώσεων χρήσης μπορεί να είναι σχέσεις συμπερίληψης, επέκτασης ή γενίκευσης.

#### Συμπερίληψη

Σε αρκετές περιπτώσεις τα βήματα που εκτελούνται σε μια περίπτωση χρήσης μπορεί να επαναλαμβάνονται στην εκτέλεση βημάτων κάποιας άλλης. Προκειμένου να αποφύγουμε την επανάληψη των βημάτων, μπορούμε να εισάγουμε τη σχέση συμπερίληψης (include) μεταξύ δύο περιπτώσεων χρήσης όπου μία περίπτωση χρήσης συμπεριλαμβάνει, στις ροές των βημάτων της, βήματα μιας δεύτερης περίπτωσης χρήσης.

#### Επέκταση

Μία δεύτερη σχέση περιπτώσεων χρήσης είναι η σχέση της επέκτασης. Υποδηλώνει ότι μία περίπτωση χρήσης επεκτείνει τη λειτουργικότητα μιας βασικής περίπτωσης χρήσης, χωρίς όμως η βασική περίπτωση χρήσης να το γνωρίζει.

Όταν χρησιμοποιούμε τη συμπερίληψη, έχουμε σαφή αναφορά της συμπεριλαμβανόμενης περίπτωσης χρήσης στο κείμενο που περιγράφει τη βασική. Αντίθετα, με τη σχέση της επέκτασης η λειτουργικότητα της βασικής περίπτωσης χρήσης επεκτείνεται, χωρίς η ίδια να το γνωρίζει. Ένα τελευταίο σημαντικό σημείο διαφοροποίησης της επέκτασης από τη

συμπερίληψη είναι ότι η σ βασική περίπτωση χρήσης μπορεί να νοηθεί ανεξάρτητα από τις επεκτάσεις της.

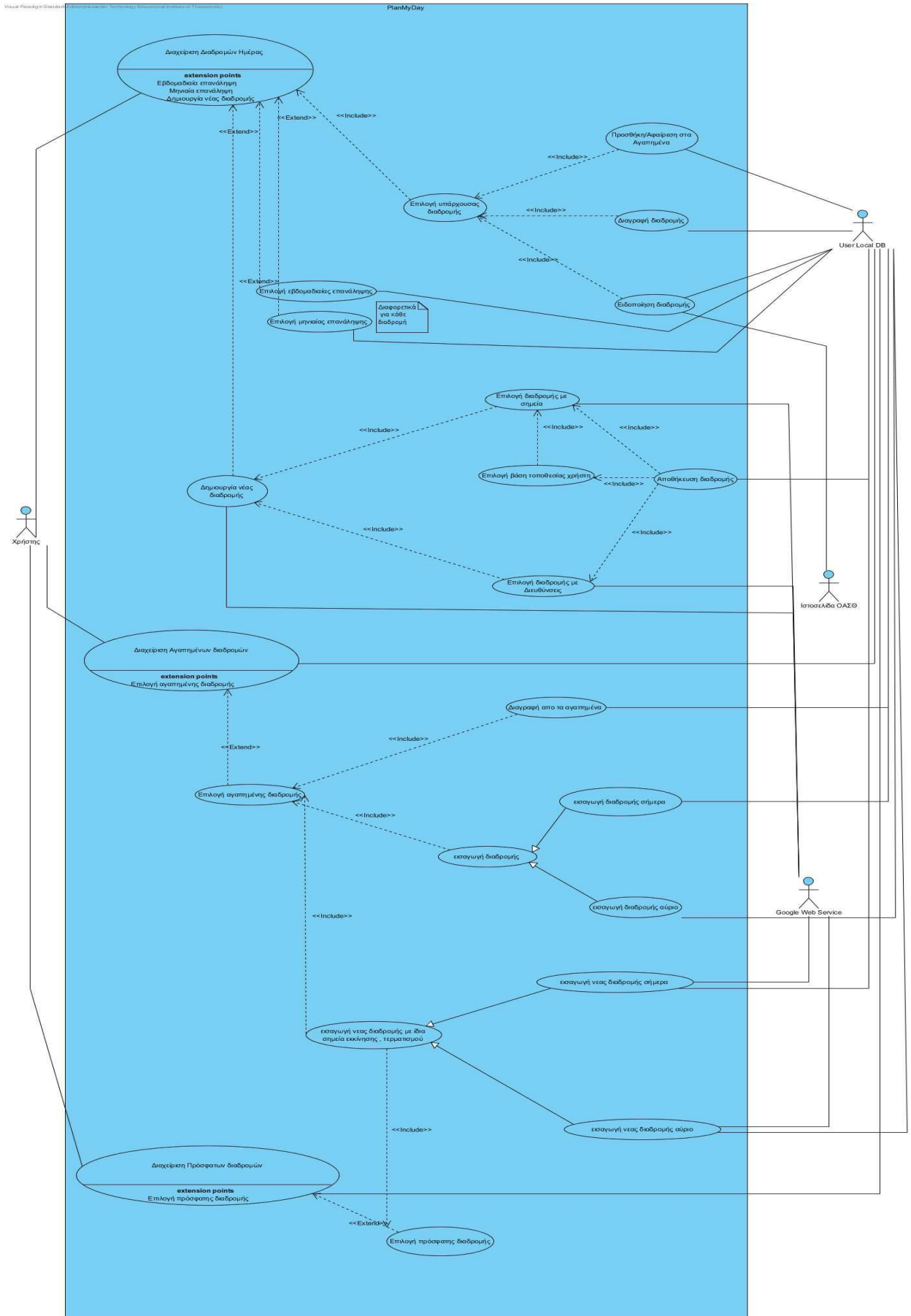
### *Γενίκευση*

Η γενίκευση σημαίνει ότι μια περίπτωση χρήσης κληρονομεί τη συμπεριφορά της περίπτωσης χρήσης την οποία και εξειδικεύει. Η σχέση της γενίκευσης είναι χρήσιμη, όταν οι δύο περιπτώσεις χρήσης έχουν παρόμοια χρήση. Όλα τα βήματα της γενικής περίπτωσης χρήσης θα πρέπει(της βασικής ροής και των εναλλακτικών ροών) θα πρέπει να υπάρχουν και στην περίπτωση χρήσης που εξειδικεύουν. Τα γενικά βήματα της γενικής περίπτωσης χρήσης εξειδικεύονται σε πιο συγκεκριμένα βήματα, ανάλογα με τη χρήση της περίπτωσης χρήσης που εξειδικεύει. Παράλληλα η ειδική περίπτωση χρήσης μπορεί να εισάγει και νέα βήματα. [36]

### **7.5.3 Διάγραμμα περιπτώσεων χρήσης της εφαρμογής**

Το Σχήμα 10, παρουσιάζει το διάγραμμα περιπτώσεων χρήσης της εφαρμογής. Το διάγραμμα έχει σχεδιαστεί στο Visual Paradigm. [34]

Πτυχιακή εργασία του φοιτητή Λαμπρόπουλου Αλέξανδρου



Σχήμα 10 διάγραμμα περιπτώσεων χρήσης

#### 7.5.4 Σενάρια περιπτώσεων χρήσης

Ένα σενάριο (ή στιγμιότυπο περίπτωσης χρήσης) είναι μια ακολουθία ενεργειών και αλληλεπιδράσεων actors και συστήματος. Μια περίπτωση χρήσης μπορεί να θεωρηθεί ως ένα σύνολο πιθανών σεναρίων που εξυπηρετούν ένα συγκεκριμένο στόχο του πρωτεύοντος actor και είναι πιθανό να εκτελεστούν, όταν ο πρωτεύον actor εκκινεί την περίπτωση χρήσης. Οι ροές των βημάτων σε μια περίπτωση χρήσης χωρίζονται σε δυο κατηγορίες :

- Η πρώτη κατηγορία είναι η βασική ροή (basic flow) η οποία περιγράφει το κύριο σενάριο και είναι μια τυπική ροή των βημάτων με επιτυχή κατάληξη.
- Η δεύτερη κατηγορία , είναι οι εναλλακτικές ροές (alternative flows) που είναι εναλλακτικές επιτυχημένες ή αποτυχημένες ροές εκτέλεσης της περίπτωσης χρήσης.

Ο τίτλος κάθε περίπτωσης χρήσης αναδεικνύει το στόχο του πρωτεύοντος actor .Ο πρωτεύων actor είναι αυτός του οποίου εξυπηρετείται ο στόχος και ο οποίος κατά κανόνα εκκινεί την περίπτωση χρήσης.

#### Προϋποθέσεις

Οι προϋποθέσεις (preconditions) περιγράφουν τι θα πρέπει να έχει εξασφαλίσει το σύστημα , έτσι ώστε να είναι δυνατή η εκκίνηση μιας περίπτωσης χρήσης. Θεωρείται ότι μια προϋπόθεση ισχύει , χωρίς να εξετάζεται από την περίπτωση χρήσης.

Σε αυτήν την εφαρμογή μερικές προϋποθέσεις είναι:

- η σύνδεση στο ίντερνετ,
- άδεια για σύνδεση στο ίντερνετ από την εφαρμογή,
- άδεια για χρήση τοποθεσίας μέσω gps,
- άδεια για εγγραφή στη συσκευή (για τη βάση δεδομένων)
- ελάχιστο SDK Version (έκδοση Android) 19.

Σε κάποιες περιπτώσεις , οι προϋποθέσεις είναι το αποτέλεσμα ολοκλήρωσης κάποιας άλλης περίπτωσης χρήσης.

#### Βασική Ροή

Η βασική ροή αποτελείται από μια ακολουθιακή ροή βημάτων που καταλήγει στην επιτυχή ολοκλήρωση της περίπτωσης χρήσης. Τα βήματα περιγράφονται με απλές καταφατικές και σύντομες προτάσεις όπου διαχωρίζονται σαφώς το τι κάνει ο πρωτεύον actor και τι το σύστημα. Τα βήματα αριθμούνται ακολουθιακά , επειδή βοηθούν στη σύνδεση των εναλλακτικών σεναρίων.

#### Εναλλακτικές Ροές

Οι εναλλακτικές ροές εκφράζουν τις συνθήκες ελέγχου και τις διακλαδώσεις στη ροή των βημάτων του κυρίου σεναρίου, όπως επίσης και εσφαλμένη χρήση του λογισμικού από τον πρωτεύοντα actor. Μαζί με τις ροές του κυρίου σεναρίου σχηματίζουν το σύνολο των σεναρίων της περίπτωσης χρήσης.



### *Μετά - συνθήκες*

Μια μετά συνθήκη (postcondition) είναι μια συνθήκη η οποία θα πρέπει να ισχύει μετά την ολοκλήρωση της περίπτωσης χρήσης. Οι μετασυνθήκες αναφέρονται είτε στην επιτυχή ολοκλήρωση της περίπτωσης χρήσης είτε σε αποτυχημένη ολοκλήρωση.

### *Εναύσματα*

Άλλη μια ενότητα που μπορεί να εισαχθεί σε μια περίπτωση χρήσης είναι το έναυσμα(trigger), δηλαδή το γεγονός που την ενεργοποιεί. Εναλλακτικά το έναυσμα που ενεργοποιεί μια περίπτωση χρήσης μπορεί να είναι το πρώτο βήμα της βασικής ροής.

### *Εξαιρέσεις*

Μία επιλογή που έχουμε στην περιγραφή των περιπτώσεων χρήσης είναι ο διαχωρισμός των επιτυχημένων και αποτυχημένων εναλλακτικών ροών. Μπορούμε να εισαγουμε μια νέα ενότητα με τίτλο εξαιρέσεις (excerptions) που θα συμπεριλαμβάνει ξεχωριστά τις αποτυχημένες ροές.

### *Ειδικές απαιτήσεις*

Οι περιπτώσεις χρήσης είναι η βάση για την περιγραφή των λειτουργικών απαιτήσεων του λογισμικού. Μία περίπτωση χρήσης μπορεί επιπλέον να συνοδεύεται από μη λειτουργικές απαιτήσεις ή άλλους περιορισμούς. Οι μη λειτουργικές απαιτήσεις που αφορούν μία συγκεκριμένη περίπτωση χρήσης αναφέρονται σε ξεχωριστή ενότητα των ειδικών απαιτήσεων (special requirements). Οι ειδικές απαιτήσεις μπορεί να συμπληρώνουν την εικόνα των μη λειτουργικών απαιτήσεων που καταγράφονται για το σύνολο του λογισμικού.

### *Δευτερεύοντες actors*

Όταν υπάρχουν δευτερεύοντες actors σε μία περίπτωση χρήσης, τότε αυτοί μπορούν να εμφανίζονται αμέσως μετά τον πρωτεύοντα.

### *Σημεία επέκτασης*

Οι αναφορές για την επέκταση της συμπεριφοράς της, γίνεται σε μία ξεχωριστή ενότητα της περιγραφής της περίπτωσης χρήσης που είναι τα σημεία επέκτασης (extension points). Τα σημεία επέκτασης αναφέρονται στα σημεία των ροών της βασικής περίπτωσης χρήσης, όπου γίνεται η επέκταση της συμπεριφοράς της. [36]

## **7.5.5 Πίνακες σεναρίων Περιπτώσεων Χρήσης**

Στο Κεφάλαιο αυτό περιγράφονται τα σενάρια των περιπτώσεων χρήσης (ροή γεγονότων, εναλλακτικές ροές, χρήστες, μετά – συνθήκες κ.α.). Οι πίνακες έχουν σχεδιαστεί αρχικά με το εργαλείο Visual Paradigm. [36]

Πίνακας 11 Σενάριο χρήσης - Διαχείριση Διαδρομών Ημέρας

<b>ID Περίπτωσης 1</b>																
<b>Χρήσης</b>																
<b>Περίπτωση Χρήσης Γονιού</b>																
<b>Βασικός Actor</b>	Χρήστης															
<b>Δευτερεύον Actor</b>																
<b>Σύντομη περιγραφή</b>	εμφάνιση των διαδρομών της επιλεγμένης ημέρας, στο χρήστη . Δίνεται η δυνατότητα στο χρήστη να επιλέξει ημερομηνία, να δημιουργήσει νέα διαδρομή και να διαχειριστεί τις υπάρχουσες.															
<b>Συνθήκες που πρέπει να ισχύουν</b>																
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th><b>Actor Input</b></th> <th><b>System Response</b></th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης ανοίγει το πρόγραμμα</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>εμφανίζει ένα "splash screen" με το λογότυπο της εφαρμογής</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης περιμένει κάποια δευτερόλεπτα</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>εμφανίζει την ημερήσια διαχείριση διαδρομών</td> </tr> </tbody> </table>		<b>Actor Input</b>	<b>System Response</b>	<b>1</b>	ο χρήστης ανοίγει το πρόγραμμα		<b>2</b>		εμφανίζει ένα "splash screen" με το λογότυπο της εφαρμογής	<b>3</b>	ο χρήστης περιμένει κάποια δευτερόλεπτα		<b>4</b>		εμφανίζει την ημερήσια διαχείριση διαδρομών
	<b>Actor Input</b>	<b>System Response</b>														
<b>1</b>	ο χρήστης ανοίγει το πρόγραμμα															
<b>2</b>		εμφανίζει ένα "splash screen" με το λογότυπο της εφαρμογής														
<b>3</b>	ο χρήστης περιμένει κάποια δευτερόλεπτα															
<b>4</b>		εμφανίζει την ημερήσια διαχείριση διαδρομών														
<b>Μετά - συνθήκες</b>																
<b>Εναλλακτικές ροές και εξαιρέσεις</b>																

Πίνακας 12 Σενάριο χρήσης - Επιλογή υπάρχουσας διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>2</b>									
<b>Περίπτωση Χρήσης Γονιού</b>	1									
<b>Βασικός Actor</b>	Χρήστης									
<b>Δευτερεύον Actor</b>										
<b>Σύντομη περιγραφή</b>	Ο χρήστης επιλέγει διαδρομή από τη λίστα και μπορεί να την επεξεργαστεί.									
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>πρέπει να υπάρχει διαθέσιμη διαδρομή στην επιλεγμένη ημέρα</li> </ul>									
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>ο χρήστης επιλέγει μια διαδρομή από τη λίστα , πατώντας παρατεταμένα πάνω σε αυτή (longclick)</td> <td></td> </tr> <tr> <td>2</td> <td></td> <td>αλλάζει το μενού (top menu) και εμφανίζει ένα καινούριο, για την διαχείριση των διαδρομών (διαγραφή , προσθήκη στα αγαπημένα , ειδοποίηση)</td> </tr> </tbody> </table>		Actor Input	System Response	1	ο χρήστης επιλέγει μια διαδρομή από τη λίστα , πατώντας παρατεταμένα πάνω σε αυτή (longclick)		2		αλλάζει το μενού (top menu) και εμφανίζει ένα καινούριο, για την διαχείριση των διαδρομών (διαγραφή , προσθήκη στα αγαπημένα , ειδοποίηση)
	Actor Input	System Response								
1	ο χρήστης επιλέγει μια διαδρομή από τη λίστα , πατώντας παρατεταμένα πάνω σε αυτή (longclick)									
2		αλλάζει το μενού (top menu) και εμφανίζει ένα καινούριο, για την διαχείριση των διαδρομών (διαγραφή , προσθήκη στα αγαπημένα , ειδοποίηση)								
<b>Μετά - συνθήκες</b>										
<b>Εναλλακτικές ροές και εξαιρέσεις</b>										

Πίνακας 13 Σενάριο χρήσης - Προσθήκη/Αφαίρεση στα Αγαπημένα

<b>ID Περίπτωσης Χρήσης</b>	<b>3</b>
<b>Περίπτωση Χρήσης Γονιού</b>	2
<b>Βασικός Actor</b>	Χρήστης
<b>Δευτερεύον Actor</b>	
<b>Σύντομη περιγραφή</b>	Προσθήκη ή αφαίρεση της επιλεγμένης διαδρομής από τα "Αγαπημένα"
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή υπάρχουσας διαδρομής" .</li> <li>Πρέπει να υπάρχει άδεια για εγγραφή μνήμη.</li> </ul>

Ροή των γεγονότων	Actor Input		System Response
	1	ο χρήστης πατάει το εικονίδιο με το "γκρι αστέρι" (προσθήκη στα αγαπημένα) από το μενού	
2			επικοινωνεί με τη βάση, αποθηκεύει την επιλεγμένη διαδρομή στον αντίστοιχο πίνακα με τα "Αγαπημένα". Εμφανίζει ανάλογο μήνυμα στο χρήστη.  Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας".
<b>Μετά - συνθήκες</b>			
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει το εικονίδιο με το "κίτρινο αστέρι"(αφαίρεση από τα αγαπημένα)</p> <ol style="list-style-type: none"> <li>το σύστημα επικοινωνεί με τη βάση δεδομένων και διαγράφει την επιλεγμένη διαδρομή από τον πίνακα με τα "Αγαπημένα" εμφανίζοντας το αντίστοιχο μήνυμα</li> <li>το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ol> <p>2.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα</p> <ol style="list-style-type: none"> <li>το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ol>		

Πίνακας 14 Σενάριο χρήσης - Διαγραφή διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	4		
<b>Περίπτωση Χρήσης Γονιού</b>	2		
<b>Βασικός Actor</b>	Χρήστης, user local db		
<b>Δευτερεύον Actor</b>			
<b>Σύντομη περιγραφή</b>	διαγραφή επιλεγμένης διαδρομής από τη βάση δεδομένων		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή υπάρχουσας διαδρομής" .</li> <li>Πρέπει να υπάρχει άδεια για εγγραφή στη μνήμη.</li> </ul>		
Ροή των γεγονότων	Actor Input		System Response
	1	ο χρήστης επιλέγει το εικονίδιο με το	

	"κάδο" από το μενού	
	2	επικοινωνεί με τη βάση δεδομένων του χρήστη και διαγράφει την επιλεγμένη διαδρομή από τον αντίστοιχο πίνακα ("Διαδρομή"). Εμφανίζεται μήνυμα στο χρήστη. Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας".
<b>Μετά - συνθήκες</b>		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα  1. το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"	

Πίνακας 15 Σενάριο χρήσης - Ειδοποίηση διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	5	
<b>Περίπτωση Χρήσης Γονιού</b>	2	
<b>Βασικός Actor</b>	Χρήστης	
<b>Δευτερεύον Actor</b>	user local db, ιστοσελίδα ΟΑΣΘ	
<b>Σύντομη περιγραφή</b>	ο χρήστης επιλέγει να ειδοποιηθεί από το σύστημα, για την ώρα άφιξης του λεωφορείου στη στάση.	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Σύνδεση στο ίντερνετ( δεδομένα ή wifi)</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή υπάρχουσας διαδρομής" .</li> <li>• Πρέπει να υπάρχει άδεια για εγγραφή στη βάση δεδομένων και άδεια για επικοινωνία μέσω ίντερνετ.</li> </ul>	
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>
	1 ο χρήστης πατάει το εικονίδιο με τη "καμπάνα" (ειδοποίηση διαδρομής)	
	2	επικοινωνεί με τη βάση των διαδρομών και έπειτα χρησιμοποιεί αυτά τα δεδομένα στην ιστοσελίδα του ΟΑΣΘ. Λαμβάνει πίσω το αποτέλεσμα και το παρουσιάζει στο χρήστη μέσω ειδοποίησης, 10 λεπτά πριν την ώρα αναχώρησης. Εμφανίζει στο χρήστη ανάλογο μήνυμα. Επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"
	3 ο χρήστης πατάει πάνω στην	

	αναδυόμενη ειδοποίηση	
	4	μεταφέρεται στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"
<b>Μετά - συνθήκες</b>	Η ειδοποίηση θα εμφανίζεται ακόμα και αν η εφαρμογή κλείσει.	
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει το εικονίδιο με τη "καμπάνα" (τερματισμός ειδοποίησης)</p> <ol style="list-style-type: none"> <li>το σύστημα τερματίζει τη λειτουργία της ειδοποίησης.</li> <li>το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ol> <p>1.β. ο χρήστης πατάει το εικονίδιο με τη "καμπάνα" (ειδοποίηση) σε μια άλλη διαδρομή.</p> <ol style="list-style-type: none"> <li>το σύστημα τερματίζει τη λειτουργία της ειδοποίησης.</li> <li>το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ol> <p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα</p> <ol style="list-style-type: none"> <li>το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ol> <p>3.α. ο χρήστης πατάει "ακύρωση" στην αναδυόμενη ειδοποίηση</p> <ol style="list-style-type: none"> <li>το σύστημα τερματίζει τη λειτουργία της ειδοποίησης.</li> </ol>	

Πίνακας 16 Σενάριο χρήσης - Επιλογή εβδομαδιαίας επανάληψης

<b>ID Περίπτωσης Χρήσης</b>	<b>6</b>	
<b>Περίπτωση Χρήσης Γονιού</b>	1	
<b>Βασικός Actor</b>	Χρήστης	
<b>Δευτερεύον Actor</b>	user local db	
<b>Σύντομη περιγραφή</b>	ο χρήστης επιλέγει σε ποιες μέρες της εβδομάδας θα επαναληφθεί η διαδρομή.	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Πρέπει να υπάρχει τουλάχιστον μια διαδρομή καταχωρημένη σε αυτή την ημέρα.</li> <li>• Πρέπει να υπάρχει άδεια για εγγραφή στη μνήμη.</li> </ul>	
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>
	<ol style="list-style-type: none"> <li>1 ο χρήστης πατάει ένα από τα κουμπιά που αντιστοιχούν στις ημέρες της τρέχουσας εβδομάδας(ΔΕ, ΤΡ, ΤΕ, ΠΕ, ΠΑ, ΣΑ, ΚΥ) και είναι ανενεργό (γκρί χρώμα).</li> </ol>	
	<ol style="list-style-type: none"> <li>2</li> </ol>	Επικοινωνεί με τη βάση και επαναλαμβάνει την αντίστοιχη διαδρομή (στην οποία ανήκουν τα κουμπιά), στην επιλεγμένη ημέρα της εβδομάδας. Το κουμπί που πατήθηκε ενεργοποιείται (μπλε χρώμα).
<b>Μετά - συνθήκες</b>	<ul style="list-style-type: none"> <li>• Κάθε κουμπί που είναι ενεργό (μπλε χρώμα) σημαίνει ότι υπάρχει επανάληψη της διαδρομής στην αντίστοιχη ημέρα.</li> </ul>	
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α ο χρήστης πατάει ένα από τα κουμπιά (ΔΕ, ΤΡ, ΤΕ, ΠΕ, ΠΑ, ΣΑ, ΚΥ) που είναι ενεργό (μπλε χρώμα)</p> <ol style="list-style-type: none"> <li>1. το σύστημα επικοινωνεί με τη βάση και διαγράφει τη διαδρομή από την ημέρα που έχει επιλέξει ο χρήστης. Το κουμπί απενεργοποιείται (γκρι χρώμα).</li> </ol> <p>1.β. ο χρήστης πατάει το κουμπί της τρέχουσας ημέρας της εβδομάδας που είναι ενεργό(μπλε χρώμα).</p> <ol style="list-style-type: none"> <li>1. το σύστημα δεν διαγράφει τη διαδρομή της τρέχουσας ημέρας μέσω της εβδομαδιαίας επανάληψης.</li> </ol>	

Πίνακας 17 Σενάριο χρήσης - Επιλογή μηνιαίας επανάληψης

<b>ID Περίπτωσης Χρήσης</b>	<b>7</b>	
<b>Περίπτωση Χρήσης Γονιού</b>	1	
<b>Βασικός Actor</b>	Χρήστης	
<b>Δευτερεύον Actor</b>	user local db	
<b>Σύντομη περιγραφή</b>	ο χρήστης επιλέγει να επαναληφθεί η διαδρομή , της τρέχουσας ημέρας και των ημερών που είναι επιλεγμένες στην εβδομαδιαία επανάληψη, σε όλο τον τρέχων μήνα.	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Πρέπει να υπάρχει τουλάχιστον μια διαδρομή καταχωρημένη σε αυτή την ημέρα.</li> <li>• Πρέπει να υπάρχει άδεια για εγγραφή στη μνήμη.</li> </ul>	
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>
	<ol style="list-style-type: none"> <li>1 ο χρήστης πατάει το κουμπί της μηνιαίας επανάληψης (M) το οποίο βρίσκεται στη διαδρομή και είναι ανενεργό (γκρι χρώμα)</li> <li>2</li> </ol>	
		επικοινωνεί με τη βάση δεδομένων και επαναλαμβάνει τη διαδρομή της τρέχουσας ημέρας και των ημερών που είναι επιλεγμένες στην εβδομαδιαία επανάληψη, σε όλο τον τρέχων μήνα. Το κουμπί (M) ενεργοποιείται (μπλε χρώμα), τα κουμπιά της εβδομαδιαίας επανάληψης κλειδώνουν.
<b>Μετά - συνθήκες</b>	<ul style="list-style-type: none"> <li>• όταν το κουμπί της μηνιαίας επανάληψης (M) είναι ενεργό (μπλε χρώμα) σημαίνει ότι οι επιλεγμένες ημέρες από την εβδομαδιαία επανάληψη επαναλαμβάνονται σε όλο το μήνα κάθε εβδομάδα.</li> </ul>	
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει το κουμπί της μηνιαίας επανάληψης (M), που βρίσκεται στη διαδρομή και είναι ενεργό.</p> <ol style="list-style-type: none"> <li>1. το σύστημα διαγράφει όλες τις διαδρομές του μήνα που επαναλαμβάνονταν, εκτός αυτών της τρέχουσας εβδομάδας. Το κουμπί (M) απενεργοποιείται (γκρι χρώμα), τα κουμπιά της εβδομαδιαίας επανάληψης στη συγκεκριμένη διαδρομή ξεκλειδώνονται.</li> </ol>	



Πίνακας 18 Σενάριο χρήσης - Δημιουργία νέας διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>8</b>		
<b>Περίπτωση Χρήσης Γονιού</b>	1		
<b>Βασικός Actor</b>	Χρήστης		
<b>Δευτερεύον Actor</b>	Google Web Service, user local db		
<b>Σύντομη περιγραφή</b>	ο χρήστης δημιουργεί μια νέα διαδρομή και καταχωρείται στην επιλεγμένη ημέρα.		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Σύνδεση στο ίντερνετ( δεδομένα ή wifi)</li> <li>• Άδεια για επικοινωνία μέσω ίντερνετ.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li> </ul>		
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>	
	<b>1</b>	ο χρήστης πατάει το εικονίδιο με το "χάρτη" από το μενού.	
	<b>2</b>		εμφανίζεται ένα νέο παράθυρο, το οποίο περιλαμβάνει ένα χάρτη (google maps), δύο κουμπιά ("αφετηρία", "τέρμα") και ένα πεδίο επιλογής ώρας
	<b>3</b>	ο χρήστης πατάει πάνω στο πεδίο επιλογής της ώρας	
	<b>4</b>		εμφανίζεται ένα νέο αναδυόμενο παράθυρο για την επιλογή της ώρας
	<b>5</b>	ο χρήστης επιλέγει την ώρα που θέλει να φτάσει στο προορισμό του	
	<b>6</b>		το σύστημα εμφανίζει την επιλεγμένη ώρα στο αντίστοιχο πεδίο.
<b>Μετά - συνθήκες</b>	<ul style="list-style-type: none"> <li>• Αν η ώρα αναχώρησης είναι πριν από την τρέχουσα πραγματική ώρα, τότε το σύστημα ορίζει ως ώρα αναχώρησης τη τρέχουσα ώρα.</li> </ul>		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α.ο χρήστης πατάει "πίσω"</p> <p>1. το σύστημα επιστρέφει στην περίπτωση χρήσης "Διαχείριση Διαδρομών"</p>		

Πίνακας 19 Σενάριο χρήσης - Επιλογή διαδρομής με σημεία

<b>ID Περίπτωσης Χρήσης</b>	<b>9</b>		
<b>Περίπτωση Χρήσης Γονιού</b>	8		
<b>Βασικός Actor</b>	Χρήστης		
<b>Δευτερεύον Actor</b>	google web service		
<b>Σύντομη περιγραφή</b>	ο χρήστης πατάει παρατεταμένα στο χάρτη για να ορίσει τα σημεία "αφετηρία" και "τέρμα".		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Σύνδεση στο ίντερνετ( δεδομένα ή wifi)</li> <li>• Άδεια για επικοινωνία μέσω ίντερνετ.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Δημιουργία νέας διαδρομής"</li> </ul>		
<b>Ροή των γεγονότων</b>		Actor Input	
		System Response	
	<b>1</b>	ο χρήστης πατάει το κουμπί "αφετηρία"	
	<b>2</b>		το σύστημα είναι έτοιμο να δεχτεί ένα σημείο πάνω στο χάρτη ως αφετηρία.
	<b>3</b>	ο χρήστης πατάει παρατεταμένα πάνω στο χάρτη	
	<b>4</b>		το σημείο έχει επιλεχθεί και μία "κόκκινη πινέζα" προστίθεται σε αυτό.
	<b>5</b>	ο χρήστης πατάει το κουμπί "τέρμα"	
	<b>6</b>		το σύστημα είναι έτοιμο να δεχτεί το επόμενο σημείο πάνω στο χάρτη ως τέρμα.
	<b>7</b>	ο χρήστης πατάει παρατεταμένα πάνω στο χάρτη	
	<b>8</b>		το σημείο που επιλέχθηκε εμφανίζεται με μια "κόκκινη πινέζα" στο χάρτη. Εφόσον και τα δυο σημεία είναι ορατά στο χάρτη τα κουμπιά απενεργοποιούνται και ενεργοποιείται ένα κουμπί "Διαδρομή"
	<b>9</b>	ο χρήστης πατάει το κουμπί "Διαδρομή"	
<b>10</b>		το σύστημα επικοινωνεί με το google web service, και σχεδιάζει τη διαδρομή μεταξύ των δύο σημείων. Το κουμπί "Διαδρομή" απενεργοποιείται ενώ ενεργοποιούνται τα κουμπιά "αφετηρία" και "τέρμα"	
<b>Μετά - συνθήκες</b>			
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	*Σε οποιαδήποτε στιγμή ο χρήστης πατάει το "πίσω"		

	<ol style="list-style-type: none"><li>1. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας"</li></ol> <p>1.α. ο χρήστης πατάει πρώτα το κουμπί "τέρμα"</p> <ol style="list-style-type: none"><li>1. το σύστημα δεν περιμένει το χρήστη να επιλέξει σημείο στο χάρτη.</li></ol> <p>5.α. ο χρήστης ξανά πατάει το κουμπί "αφετηρία".</p> <ol style="list-style-type: none"><li>1. το σύστημα σβήνει το σημείο αφετηρίας από το χάρτη.</li><li>2. το σύστημα περιμένει να δεχτεί το σημείο "αφετηρία" πάνω στο χάρτη.</li></ol> <p>8.α. ο χρήστης ξανά πατάει το κουμπί "αφετηρία".</p> <ol style="list-style-type: none"><li>1. το σύστημα διαγράφει τα σημεία "αφετηρία" και "τέρμα" από το χάρτη.</li><li>2. το σύστημα περιμένει να δεχτεί το σημείο "αφετηρία" πάνω στο χάρτη.</li></ol> <p>8. β. ο χρήστης ξανά πατάει το κουμπί "τέρμα".</p> <ol style="list-style-type: none"><li>1. το σύστημα διαγράφει το σημείο "τέρμα" από το χάρτη.</li><li>2. το σύστημα περιμένει να δεχτεί το σημείο "τέρμα" πάνω στο χάρτη.</li></ol> <p>10.α. ο χρήστης πατάει το κουμπί "αφετηρία".</p> <ol style="list-style-type: none"><li>1. το σύστημα διαγράφει όλα τα σημεία και τη διαδρομή από το χάρτη.</li><li>2. το σύστημα περιμένει να δεχτεί το σημείο "αφετηρία" πάνω στο χάρτη.</li></ol> <p>10.β. ο χρήστης ξανά πατάει το κουμπί "τέρμα".</p> <ol style="list-style-type: none"><li>1. το σύστημα διαγράφει όλα τα σημεία και τη διαδρομή από το χάρτη.</li></ol> <p>10.γ. τα σημεία στο χάρτη δεν αντιστοιχούν σε διαδρομή με MMM που εξυπηρετούνται από τον ΟΑΣΘ.</p> <ol style="list-style-type: none"><li>1. το σύστημα παρουσιάζει σχετικό μήνυμα στο χρήστη</li></ol>
--	---

Πίνακας 20 Σενάριο χρήσης - Επιλογή βάση τοποθεσίας χρήστη

<b>ID Περίπτωσης</b>	<b>10</b>																																	
<b>Χρήσης</b>																																		
<b>Περίπτωση Χρήσης Γονιού</b>	9																																	
<b>Βασικός Actor</b>	Χρήστης																																	
<b>Δευτερεύον Actor</b>	google web service																																	
<b>Σύντομη περιγραφή</b>	Ο χρήστης μπορεί να επιλέξει την τοποθεσία του, ως αφετηρία της διαδρομής																																	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Σύνδεση στο ίντερνετ( δεδομένα ή wifi)</li> <li>• GPS</li> <li>• Άδεια για επικοινωνία μέσω ίντερνετ και χρήση gps.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Δημιουργία νέας διαδρομής"</li> </ul>																																	
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει από το μενού το εικονίδιο με τη "μπλε πινέζα"</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>το σύστημα βρίσκει τη τοποθεσία του χρήστη</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης πατάει το κουμπί "αφετηρία"</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>μια "κόκκινη πινέζα" εμφανίζεται πάνω στην τοποθεσία του χρήστη</td> </tr> <tr> <td><b>5</b></td> <td>ο χρήστης πατάει το κουμπί "τέρμα"</td> <td></td> </tr> <tr> <td><b>6</b></td> <td></td> <td>το σύστημα περιμένει να δεχτεί το σημείο για το "τέρμα"</td> </tr> <tr> <td><b>7</b></td> <td>ο χρήστης πατάει παρατεταμένα πάνω στο χάρτη</td> <td></td> </tr> <tr> <td><b>8</b></td> <td></td> <td>μια ακόμα "κόκκινη πινέζα" εμφανίζεται Εφόσον υπάρχουν και τα δύο σημεία, ενεργοποιείται το κουμπί "Διαδρομή"</td> </tr> <tr> <td><b>9</b></td> <td>ο χρήστης πατάει το κουμπί "Διαδρομή"</td> <td></td> </tr> <tr> <td><b>10</b></td> <td></td> <td>το σύστημα επικοινωνεί με το google ws και σχεδιάζει στο χάρτη τη διαδρομή.</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει από το μενού το εικονίδιο με τη "μπλε πινέζα"		<b>2</b>		το σύστημα βρίσκει τη τοποθεσία του χρήστη	<b>3</b>	ο χρήστης πατάει το κουμπί "αφετηρία"		<b>4</b>		μια "κόκκινη πινέζα" εμφανίζεται πάνω στην τοποθεσία του χρήστη	<b>5</b>	ο χρήστης πατάει το κουμπί "τέρμα"		<b>6</b>		το σύστημα περιμένει να δεχτεί το σημείο για το "τέρμα"	<b>7</b>	ο χρήστης πατάει παρατεταμένα πάνω στο χάρτη		<b>8</b>		μια ακόμα "κόκκινη πινέζα" εμφανίζεται Εφόσον υπάρχουν και τα δύο σημεία, ενεργοποιείται το κουμπί "Διαδρομή"	<b>9</b>	ο χρήστης πατάει το κουμπί "Διαδρομή"		<b>10</b>		το σύστημα επικοινωνεί με το google ws και σχεδιάζει στο χάρτη τη διαδρομή.
	Actor Input	System Response																																
<b>1</b>	ο χρήστης πατάει από το μενού το εικονίδιο με τη "μπλε πινέζα"																																	
<b>2</b>		το σύστημα βρίσκει τη τοποθεσία του χρήστη																																
<b>3</b>	ο χρήστης πατάει το κουμπί "αφετηρία"																																	
<b>4</b>		μια "κόκκινη πινέζα" εμφανίζεται πάνω στην τοποθεσία του χρήστη																																
<b>5</b>	ο χρήστης πατάει το κουμπί "τέρμα"																																	
<b>6</b>		το σύστημα περιμένει να δεχτεί το σημείο για το "τέρμα"																																
<b>7</b>	ο χρήστης πατάει παρατεταμένα πάνω στο χάρτη																																	
<b>8</b>		μια ακόμα "κόκκινη πινέζα" εμφανίζεται Εφόσον υπάρχουν και τα δύο σημεία, ενεργοποιείται το κουμπί "Διαδρομή"																																
<b>9</b>	ο χρήστης πατάει το κουμπί "Διαδρομή"																																	
<b>10</b>		το σύστημα επικοινωνεί με το google ws και σχεδιάζει στο χάρτη τη διαδρομή.																																
<b>Μετά - συνθήκες</b>																																		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>*Οποιαδήποτε στιγμή ο χρήστης ξανά πατήσει το εικονίδιο με τη "μπλε πινέζα" (τοποθεσία) στο μενού</p> <ol style="list-style-type: none"> <li>1. η λειτουργία της τοποθεσίας απενεργοποιείται.</li> <li>2. όλα τα σημεία στο χάρτη και η διαδρομή σβήνονται.</li> </ol>																																	

<p>1.α. ο χρήστης πατάει οπουδήποτε πάνω στο χάρτη.</p> <ol style="list-style-type: none"><li>το σύστημα προσθέτει μια "κόκκινη πινέζα" πάνω από την τοποθεσία του χρήστη.</li></ol> <p>1.β. η τοποθεσία δεν είναι ενεργοποιημένη από το λειτουργικό σύστημα</p> <ol style="list-style-type: none"><li>το σύστημα ανοίγει ένα νέο παράθυρο με τις ρυθμίσεις συστήματος, για ενεργοποίηση της τοποθεσίας.</li><li>το σύστημα επιστρέφει στη περίπτωση χρήσης "Επιλογή βάση τοποθεσίας χρήστη"</li></ol> <p>3.α. το σύστημα αργεί να βρει τη τοποθεσία του χρήστη ενώ αυτός πατάει το κουμπί "αφετηρία"</p> <ol style="list-style-type: none"><li>το σύστημα εμφανίζει στο χρήστη ανάλογο μήνυμα.</li></ol> <p>9.α. ο χρήστης πατάει ξανά το κουμπί "τέρμα"</p> <ol style="list-style-type: none"><li>το σύστημα σβήνει το σημείο "τέρμα" από το χάρτη και περιμένει νέο σημείο.</li></ol> <p>10.α. ο χρήστης πατάει ξανά το κουμπί "τέρμα"</p> <ol style="list-style-type: none"><li>το σύστημα σβήνει τα σημεία και τη διαδρομή από το χάρτη.</li></ol> <p>10.β. τα σημεία στο χάρτη δεν αντιστοιχούν σε διαδρομή με MMM που εξυπηρετούνται από τον ΟΑΣΘ.</p> <ol style="list-style-type: none"><li>το σύστημα παρουσιάζει σχετικό μήνυμα στο χρήστη.</li></ol>
---

Πίνακας 21 Σενάριο χρήσης - Αποθήκευση διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>12</b>	
<b>Περίπτωση Χρήσης Γονιού</b>	10	
<b>Βασικός Actor</b>	Χρήστης	
<b>Δευτερεύον Actor</b>	User Local DB	
<b>Σύντομη περιγραφή</b>	αποθήκευση της διαδρομής	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Άδεια εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί μια από τις ακόλουθες περιπτώσεις χρήσης "Επιλογή διαδρομής με σημεία", "Επιλογή διαδρομής βάση τοποθεσίας χρήστη" ή "Επιλογή διαδρομής με Διευθύνσεις"</li> </ul>	
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>
	<ol style="list-style-type: none"> <li>1 ο χρήστης πατάει το εικονίδιο "αποθήκευση" του μενού</li> <li>2</li> </ol>	το σύστημα επικοινωνεί με τη βάση δεδομένων του χρήστη και αποθηκεύει τη διαδρομή στον αντίστοιχο πίνακα. Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας", έχοντας πρόσθεση τη νέα διαδρομή.
<b>Μετά - συνθήκες</b>		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει αποθήκευση χωρίς να έχει επιλεγμένη διαδρομή.</p> <p>1. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας" χωρίς να έχει προσθέσει κάποια νέα εφαρμογή.</p>	

Πίνακας 22 Σενάριο χρήσης - Επιλογή διαδρομής με διευθύνσεις

<b>ID Περίπτωσης Χρήσης</b>	<b>11</b>		
<b>Περίπτωση Χρήσης Γονιού</b>	8		
<b>Βασικός Actor</b>	Χρήστης		
<b>Δευτερεύον Actor</b>	Google Web Service		
<b>Σύντομη περιγραφή</b>	επιλογή των σημείων "αφετηρία", "τέλος", με εισαγωγή διευθύνσεων		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Σύνδεση στο ίντερνετ( δεδομένα ή wifi)</li> <li>• Άδεια για επικοινωνία μέσω ίντερνετ.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Δημιουργία νέας διαδρομής"</li> </ul>		
<b>Ροή των γεγονότων</b>		System Response	
	<b>1</b>	ο χρήστης επιλέγει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού	
	<b>2</b>		εμφανίζεται ένα υπό - μενού
	<b>3</b>	ο χρήστης επιλέγει "αναζήτηση μέσω διευθύνσεων"	
	<b>4</b>		τα κουμπιά "αφετηρία", "τέρμα", τοποθεσία, απενεργοποιούνται. Εμφανίζονται στο χάρτη δυο πεδία για εισαγωγή διευθύνσεων.
	<b>5</b>	ο χρήστης πληκτρολογεί τη διεύθυνση της επιλογής του, στο πεδίο "Από" και πατάει το "έντερ"	
	<b>6</b>		το σύστημα με τη βοήθεια της κλάσης Geocoder (Android Location API) και εμφανίζει μια λίστα κάτω από το πεδίο με τις προτεινόμενες διευθύνσεις.
	<b>7</b>	ο χρήστης επιλέγει τη διεύθυνση που επιθυμεί	
	<b>8</b>		η λίστα με τις προτεινόμενες επιλογές εξαφανίζεται. Η διεύθυνση μετατρέπεται σε σημείο πάνω στο χάρτη.
	<b>9</b>	ο χρήστης πληκτρολογεί τη διεύθυνση της επιλογής του, στο πεδίο "Προς" και πατάει "έντερ"	
	<b>10</b>		το σύστημα επικοινωνεί με το google ws και εμφανίζει μια λίστα κάτω από το

		αντίστοιχο πεδίο με τις προτεινόμενες διευθύνσεις.
	<b>11</b>	ο χρήστης επιλέγει τη διεύθυνση που επιθυμεί
	<b>12</b>	η λίστα με τις προτεινόμενες επιλογές εξαφανίζεται. Η διεύθυνση μετατρέπεται σε σημείο πάνω στο χάρτη. Ένα κουμπι "Διαδρομή" εμφανίζεται.
	<b>13</b>	ο χρήστης πατάει το κουμπι "Διαδρομή"
	<b>14</b>	το σύστημα επικοινωνεί με το google ws και σχεδιάζει τη διαδρομή ανάμεσα στα δύο σημεία.
<b>Μετά - συνθήκες</b>		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>*Σε οποιαδήποτε στιγμή αν ο χρήστης επιλέξει από το υπό - μενού ξανά "αναζήτηση μέσω διευθύνσεων"</p> <p>1. το σύστημα διαγράφει τα σημεία πάνω από το χάρτη και τη διαδρομή.</p> <p>7.α. ο χρήστης δεν βρίσκει τη διεύθυνση που έψαχνε στη λίστα, τότε σβήνει και ξανά πληκτρολογεί με περισσότερες λεπτομέρειες τη διεύθυνση και πατάει "έντερ"</p> <p>1. το σύστημα επικοινωνεί με το google ws και εμφανίζει μια λίστα κάτω από το πεδίο με τις προτεινόμενες διευθύνσεις.</p> <p>11.α. ο χρήστης δεν βρίσκει τη διεύθυνση που έψαχνε στη λίστα, τότε σβήνει και ξανά πληκτρολογεί με περισσότερες λεπτομέρειες τη διεύθυνση και πατάει "έντερ"</p> <p>1. το σύστημα επικοινωνεί με το google ws και εμφανίζει μια λίστα κάτω από το πεδίο με τις προτεινόμενες διευθύνσεις.</p> <p>13.α. τα σημεία στο χάρτη δεν αντιστοιχούν σε διαδρομή με MMM που εξυπηρετούνται από τον ΟΑΣΘ.</p> <p>1. το σύστημα παρουσιάζει σχετικό μήνυμα στο χρήστη.</p>	



Πίνακας 23 Σενάριο χρήσης - Διαχείριση Αγαπημένων Διαδρομών

<b>ID Περίπτωσης 13</b>																
<b>Χρήσης</b>																
<b>Περίπτωση Χρήσης Γονιού</b>																
<b>Βασικός Actor</b>	Χρήστης															
<b>Δευτερεύον Actor</b>	user local db															
<b>Σύντομη περιγραφή</b>	Ο χρήστης μπορεί να διαχειριστεί τις διαδρομές που έχει αποθηκεύσει στα "Αγαπημένα"															
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Άδεια εγγραφής στη μνήμη.</li> </ul>															
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>το κυρίως μενού εμφανίζεται</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης επιλέγει από το μενού τη κατηγορία "Αγαπημένα"</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"		<b>2</b>		το κυρίως μενού εμφανίζεται	<b>3</b>	ο χρήστης επιλέγει από το μενού τη κατηγορία "Αγαπημένα"		<b>4</b>		το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.
	Actor Input	System Response														
<b>1</b>	ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"															
<b>2</b>		το κυρίως μενού εμφανίζεται														
<b>3</b>	ο χρήστης επιλέγει από το μενού τη κατηγορία "Αγαπημένα"															
<b>4</b>		το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.														
<b>Μετά - συνθήκες</b>																
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α ο χρήστης σέρνει το δάχτυλο του από τα αριστερά προς τα δεξιά</p> <ol style="list-style-type: none"> <li>το σύστημα εμφανίζει το κυρίως μενού.</li> </ol> <p>3.α. ο χρήστης πατάει το κουμπί του κυρίως μενού ξανά.</p> <ol style="list-style-type: none"> <li>Το σύστημα κλείνει το μενού.</li> <li>Το σύστημα επιστρέφει στη τελευταία περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας", "Διαχείριση Αγαπημένων Διαδρομών" ή "Διαχείριση Πρόσφατων Διαδρομών".</li> </ol> <p>3.β. ο χρήστης πατάει σε κάποιο σημείο εκτός του κυρίως μενού.</p> <ol style="list-style-type: none"> <li>Το σύστημα κλείνει το μενού.</li> <li>Το σύστημα επιστρέφει στη τελευταία περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας", "Διαχείριση Αγαπημένων Διαδρομών" ή "Διαχείριση Πρόσφατων Διαδρομών".</li> </ol> <p>3.γ ο χρήστης επιλέγει κάποια άλλη κατηγορία.</p>															

	<p>1. το σύστημα τον μεταφέρει στην αντίστοιχη κατηγορία.</p> <p>3.δ. ο χρήστης σέρνει το δάχτυλο του από τα δεξιά προς τα αριστερά</p> <p>1. το σύστημα κλείνει το κυρίως μενού.</p>
--	---

Πίνακας 24 Σενάριο χρήσης - Επιλογή αγαπημένης διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>14</b>									
<b>Περίπτωση Χρήσης Γονιού</b>	13									
<b>Βασικός Actor</b>	Χρήστης									
<b>Δευτερεύον Actor</b>										
<b>Σύντομη περιγραφή</b>	ο χρήστης επιλέγει μια διαδρομή από τη λίστα "Αγαπημένα"									
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Να υπάρχει τουλάχιστον μία διαδρομή αποθηκευμένη στα "Αγαπημένα"</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ul>									
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (διαγραφή, εισαγωγή σήμερα, εισαγωγή αύριο, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή		<b>2</b>		το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (διαγραφή, εισαγωγή σήμερα, εισαγωγή αύριο, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)
	Actor Input	System Response								
<b>1</b>	ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή									
<b>2</b>		το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (διαγραφή, εισαγωγή σήμερα, εισαγωγή αύριο, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)								
<b>Μετά - συνθήκες</b>										
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>2.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <p>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</p> <p>2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</p>									

Πίνακας 25 Σενάριο χρήσης - Διαγραφή από τα Αγαπημένα

ID	<b>15</b>	
Περίπτωσης Χρήσης		
Περίπτωση Χρήσης Γονιού	14	
Βασικός Actor	Χρήστης	
Δευτερεύον Actor	User Local DB	
Σύντομη περιγραφή	διαγράφει τη διαδρομή από τα "Αγαπημένα"	
Συνθήκες που πρέπει να ισχύουν	<ul style="list-style-type: none"> <li>• Άδεια εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>	
Ροή των γεγονότων	Actor Input	System Response
	<b>1</b> ο χρήστης πατάει το εικονίδιο "κάδος" (διαγραφή) από το μενού	
	<b>2</b>	επικοινωνεί με τη βάση δεδομένων, διαγράφει την επιλεγμένη διαδρομή από τα "Αγαπημένα" και εμφανίζει στο χρήστη αντίστοιχο μήνυμα. Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"
Μετά - συνθήκες		
Εναλλακτικές ροές και εξαιρέσεις	1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα. <ol style="list-style-type: none"> <li>το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</li> <li>το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ol>	

Πίνακας 26 Σενάριο χρήσης - Εισαγωγή διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>16</b>									
<b>Περίπτωση Χρήσης Γονιού</b>	14									
<b>Βασικός Actor</b>	Χρήστης									
<b>Δευτερεύον Actor</b>	User Local DB									
<b>Σύνομη περιγραφή</b>	εισάγει την επιλεγμένη διαδρομή									
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Άδεια εγγραφής στη μνήμη.</li> <li>Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>									
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" από το μενού</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>εμφανίζεται το υπο - μενού</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" από το μενού		<b>2</b>		εμφανίζεται το υπο - μενού
	Actor Input	System Response								
<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" από το μενού									
<b>2</b>		εμφανίζεται το υπο - μενού								
<b>Μετά - συνθήκες</b>										
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <ol style="list-style-type: none"> <li>το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</li> <li>το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ol>									

Πίνακας 27 Σενάριο χρήσης - Εισαγωγή επιλεγμένης διαδρομής σήμερα

<b>ID Περίπτωσης Χρήσης</b>	<b>17</b>						
<b>Περίπτωση Χρήσης Γονιού</b>	εισαγωγή διαδρομής						
<b>Βασικός Actor</b>	Χρήστης						
<b>Δευτερεύον Actor</b>	User Local DB						
<b>Σύνομη περιγραφή</b>	εισάγει την επιλεγμένη διαδρομή σήμερα						
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Άδεια εγγραφής στη μνήμη.</li> <li>Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>						
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>		Actor Input	System Response			
	Actor Input	System Response					

	1	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" από το μενού	
	2		εμφανίζεται το υπο - μενού
	3	ο χρήστης επιλέγει "εισαγωγή διαδρομής σήμερα"	
	4		το σύστημα επικοινωνεί με τη βάση και εισάγει την επιλεγμένη διαδρομή στη τρέχουσα ημέρα. Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"
<b>Μετά - συνθήκες</b>			
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <p>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού. 2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</p>		

Πίνακας 28 Σενάριο χρήσης - Εισαγωγή διαδρομής αύριο

**ID Περίπτωσης** 18  
**Χρήσης**

<b>Περίπτωση Χρήσης Γονιού</b>	εισαγωγή διαδρομής	
<b>Βασικός Actor</b>	Χρήστης	
<b>Δευτερεύον Actor</b>	User Local DB	
<b>Σύντομη περιγραφή</b>	εισάγει την επιλεγμένη διαδρομή αύριο	
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Άδεια εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>	
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>
	1	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" από το μενού
	2	εμφανίζεται το υπο - μενού

	3	ο χρήστης επιλέγει "εισαγωγή διαδρομής αύριο"	
	4		το σύστημα επικοινωνεί με τη βάση και εισάγει την επιλεγμένη διαδρομή στην επόμενη ημέρα. Επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"
<b>Μετά - συνθήκες</b>			
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <p>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού. 2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</p>		

Πίνακας 29 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής με ίδια σημεία εκκίνησης, τερματισμού

ID Περίπτωσης 19  
Χρήσης

<b>Περίπτωση Χρήσης Γονιού</b>			
<b>Βασικός Actor</b>	Χρήστης		
<b>Δευτερεύον Actor</b>	User Local DB, Google Web Service		
<b>Σύντομη περιγραφή</b>	εισάγει νέα διαδρομή με ίδια σημεία "αφετηρία" - "τέρμα" με την επιλεγμένη.		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Να έχει σύνδεση στο ίντερνετ( δεδομένα, wifi)</li> <li>• Άδεια χρήσης ίντερνετ και εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>		
<b>Ροή των γεγονότων</b>	<b>Actor Input</b>	<b>System Response</b>	
	1	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού	
	2		εμφανίζεται το υπο - μενού
<b>Μετά - συνθήκες</b>			
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <p>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού. 2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</p>		

Πίνακας 30 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής σήμερα

<b>ID Περίπτωσης Χρήσης</b>	<b>20</b>																		
<b>Περίπτωση Χρήσης Γονιού</b>	εισαγωγή νέας διαδρομής με ίδια σημεία εκκίνησης , τερματισμού																		
<b>Βασικός Actor</b>	χρήστης																		
<b>Δευτερεύον Actor</b>	User Local DB , google web service																		
<b>Σύντομη περιγραφή</b>	εισάγει νέα διαδρομή με ίδια σημεία "αφετηρίας" - "τέρμα" με την επιλεγμένη, σήμερα.																		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Να έχει σύνδεση στο ίντερνετ( δεδομένα, wifi)</li> <li>• Άδεια χρήσης ίντερνετ και εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>																		
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>εμφανίζεται το υπο - μενού</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής σήμερα"</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.</td> </tr> <tr> <td><b>5</b></td> <td>ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"</td> <td></td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού		<b>2</b>		εμφανίζεται το υπο - μενού	<b>3</b>	ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής σήμερα"		<b>4</b>		το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.	<b>5</b>	ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"	
	Actor Input	System Response																	
<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού																		
<b>2</b>		εμφανίζεται το υπο - μενού																	
<b>3</b>	ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής σήμερα"																		
<b>4</b>		το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.																	
<b>5</b>	ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"																		
<b>Μετά - συνθήκες</b>	<ul style="list-style-type: none"> <li>• η διαδρομή θα καταχωρηθεί στην τρέχουσα ημέρα.</li> </ul>																		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <ol style="list-style-type: none"> <li>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</li> <li>2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ol>																		

Πίνακας 31 Σενάριο χρήσης - Εισαγωγή νέας διαδρομής αύριο

<b>ID Περίπτωσης Χρήσης</b>	<b>21</b>																		
<b>Περίπτωση Χρήσης Γονιού</b>	εισαγωγή νέας διαδρομής με ίδια σημεία εκκίνησης , τερματισμού																		
<b>Βασικός Actor</b>	Χρήστης																		
<b>Δευτερεύον Actor</b>	User Local DB , google web service																		
<b>Σύντομη περιγραφή</b>	εισάγει νέα διαδρομή με ίδια σημεία "αφετηρίας" - "τέρμα" με την επιλεγμένη, αύριο.																		
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Να έχει σύνδεση στο ίντερνετ( δεδομένα, wifi)</li> <li>• Άδεια χρήσης ίντερνετ και εγγραφής στη μνήμη.</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Επιλογή Αγαπημένης Διαδρομής"</li> </ul>																		
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>εμφανίζεται το υπο - μενού</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής αύριο"</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.</td> </tr> <tr> <td><b>5</b></td> <td>ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"</td> <td></td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού		<b>2</b>		εμφανίζεται το υπο - μενού	<b>3</b>	ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής αύριο"		<b>4</b>		το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.	<b>5</b>	ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"	
	Actor Input	System Response																	
<b>1</b>	ο χρήστης πατάει το εικονίδιο "τρεις κάθετες τελείες" (ρυθμίσεις) από το μενού																		
<b>2</b>		εμφανίζεται το υπο - μενού																	
<b>3</b>	ο χρήστης επιλέγει "εισαγωγή νέας διαδρομής αύριο"																		
<b>4</b>		το σύστημα επικοινωνεί με τη βάση και το google web service. Ανοίγει το παράθυρο "Νέα Διαδρομή" και τα σημεία "αφετηρία"- "τέρμα", εμφανίζονται στο χάρτη.																	
<b>5</b>	ο χρήστης ακολουθεί τα βήματα της περίπτωσης χρήσης "Νέα Διαδρομή"																		
<b>Μετά - συνθήκες</b>	<ul style="list-style-type: none"> <li>• η διαδρομή θα καταχωρηθεί στην επόμενη ημέρα.</li> </ul>																		
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>1.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <ol style="list-style-type: none"> <li>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</li> <li>2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ol>																		



Πίνακας 32 Σενάριο χρήσης - Διαχείριση Πρόσφατων Διαδρομών

<b>ID Περίπτωσης</b>	<b>22</b>															
<b>Χρήσης</b>																
<b>Περίπτωση Χρήσης Γονιού</b>																
<b>Βασικός Actor</b>	Χρήστης															
<b>Δευτερεύον Actor</b>	User Local DB															
<b>Σύντομη περιγραφή</b>	Διαχείριση τελευταίων δέκα διαδρομών.															
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>Άδεια εγγραφής στη μνήμη.</li> </ul>															
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>το κυρίως μενού εμφανίζεται</td> </tr> <tr> <td><b>3</b></td> <td>ο χρήστης επιλέγει από το μενού τη κατηγορία "Πρόσφατα"</td> <td></td> </tr> <tr> <td><b>4</b></td> <td></td> <td>το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"		<b>2</b>		το κυρίως μενού εμφανίζεται	<b>3</b>	ο χρήστης επιλέγει από το μενού τη κατηγορία "Πρόσφατα"		<b>4</b>		το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.
	Actor Input	System Response														
<b>1</b>	ο χρήστης ανοίγει το κυρίως μενού πατώντας το εικονίδιο "τρεις οριζόντιες παύλες"															
<b>2</b>		το κυρίως μενού εμφανίζεται														
<b>3</b>	ο χρήστης επιλέγει από το μενού τη κατηγορία "Πρόσφατα"															
<b>4</b>		το σύστημα επικοινωνεί με τη βάση δεδομένων και εμφανίζει όλες τις διαδρομές που είναι αποθηκευμένες στον αντίστοιχο πίνακα.														
<b>Μετά - συνθήκες</b>																
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>3.α. ο χρήστης πατάει το κουμπί του κυρίως μενού ξανά.</p> <ol style="list-style-type: none"> <li>1. Το σύστημα κλείνει το μενού.</li> <li>2. Το σύστημα επιστρέφει στη τελευταία περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας", "Διαχείριση Αγαπημένων Διαδρομών" ή "Διαχείριση Πρόσφατων Διαδρομών".</li> </ol> <p>3.β. ο χρήστης πατάει σε κάποιο σημείο εκτός του κυρίως μενού.</p> <ol style="list-style-type: none"> <li>1. Το σύστημα κλείνει το μενού.</li> <li>2. Το σύστημα επιστρέφει στη τελευταία περίπτωση χρήσης "Διαχείριση Διαδρομών Ημέρας", "Διαχείριση Αγαπημένων Διαδρομών" ή "Διαχείριση Πρόσφατων Διαδρομών".</li> </ol> <p>3.γ. ο χρήστης επιλέγει κάποια άλλη κατηγορία.</p> <ol style="list-style-type: none"> <li>1. το σύστημα τον μεταφέρει στην αντίστοιχη κατηγορία.</li> </ol>															

Πίνακας 33 Σενάριο χρήσης - Επιλογή πρόσφατης διαδρομής

<b>ID Περίπτωσης Χρήσης</b>	<b>23</b>									
<b>Περίπτωση Χρήσης Γονιού</b>	22									
<b>Βασικός Actor</b>	Χρήστης									
<b>Δευτερεύον Actor</b>										
<b>Σύντομη περιγραφή</b>	ο χρήστης επιλέγει μια διαδρομή από τη λίστα "Πρόσφατα"									
<b>Συνθήκες που πρέπει να ισχύουν</b>	<ul style="list-style-type: none"> <li>• Να υπάρχει τουλάχιστον μία διαδρομή αποθηκευμένη στα "Αγαπημένα"</li> <li>• Να έχει ολοκληρωθεί η περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ul>									
<b>Ροή των γεγονότων</b>	<table border="1"> <thead> <tr> <th></th> <th>Actor Input</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td><b>1</b></td> <td>ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή</td> <td></td> </tr> <tr> <td><b>2</b></td> <td></td> <td>το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)</td> </tr> </tbody> </table>		Actor Input	System Response	<b>1</b>	ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή		<b>2</b>		το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)
	Actor Input	System Response								
<b>1</b>	ο χρήστης πατάει παρατεταμένα πάνω σε μία διαδρομή									
<b>2</b>		το σύστημα επιλέγει τη διαδρομή και εμφανίζει ένα νέο μενού (χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή σήμερα, χρήση σημείων "αφετηρία" - "τέρμα" για εισαγωγή αύριο)								
<b>Μετά - συνθήκες</b>										
<b>Εναλλακτικές ροές και εξαιρέσεις</b>	<p>2.α. ο χρήστης πατάει πάνω σε μια διαδρομή από τη λίστα.</p> <ol style="list-style-type: none"> <li>1. το σύστημα αποεπιλέγει τη διαδρομή και απενεργοποιεί το μενού.</li> <li>2. το σύστημα επιστρέφει στη περίπτωση χρήσης "Διαχείριση Αγαπημένων Διαδρομών"</li> </ol>									

## 7.6 Διαγράμματα κλάσεων

Τα διαγράμματα κλάσεων (class diagrams) είναι ίσως τα σημαντικότερα διαγράμματα της UML. Χρησιμοποιούνται καθ' όλη τη διάρκεια του κύκλου ζωής του λογισμικού, από την ανάλυση έως και τον προγραμματισμό και τη συντήρηση. Χρησιμοποιούνται στην ανάλυση για την ανάδειξη των σημαντικότερων εννοιών του προβλήματος, όπως επίσης και στη σχεδίαση για τη λεπτομερέστερη προδιαγραφή των βασικότερων μονάδων του αντικειμενοστραφούς προγραμματισμού που είναι οι κλάσεις του λογισμικού. [36]

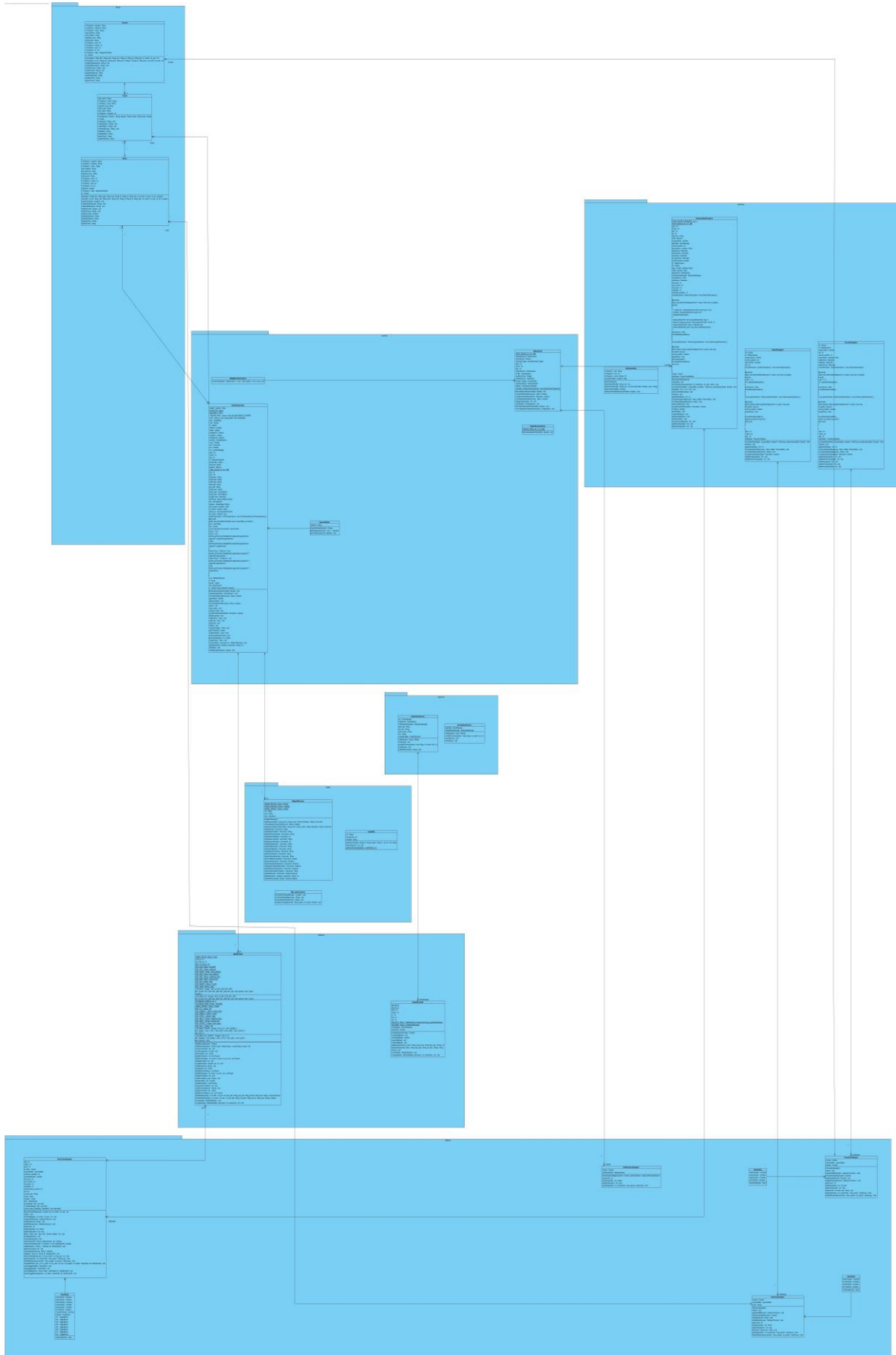
Στον Πίνακα 34 περιγράφεται ο συμβολισμός των αντικειμένων σε ένα διάγραμμα κλάσεων.

Πίνακας 34 συμβολισμός αντικειμένων σε διάγραμμα κλάσεων

ΑΝΤΙΚΕΙΜΕΝΟ	ΣΥΜΒΟΛΙΣΜΟΣ
Κλάσεις	Ο συμβολισμός της UML για μία κλάση είναι ένα απλό παραλληλόγραμμο με τρία τμήματα. Στο πάνω τμήμα αναγράφεται το όνομα της κλάσης, στο μεσαίο τμήμα οι ιδιότητες της (attributes) και στο τρίτο οι λειτουργίες της (operations).
Ιδιότητες	Οι ιδιότητες μίας κλάσης είναι τα δεδομένα που περιέχουν τα αντικείμενα της κλάσης και παραπέμπουν στα πεδία (fields) μιας κλάσης στη Java.
Λειτουργίες	Οι λειτουργίες καθορίζουν τη συμπεριφορά των αντικειμένων της κλάσης και παραπέμπουν στις μεθόδους (methods) μιας κλάσης της Java.
Συσχετίσεις	αναφορά στις συσχετίσεις που χρησιμοποιεί στο διάγραμμα κλάσεων μου.

### 7.6.1 Διάγραμμα κλάσεων της εφαρμογής

Το διάγραμμα κλάσεων περιέχει και τα πακέτα στα οποία ανήκουν οι κλάσεις, έτσι υπερκαλύπτει και το διάγραμμα πακέτων που θα δούμε παρακάτω (Σχήμα 11). Το διάγραμμα έχει σχεδιαστεί στο Visual Paradigm. [24]



Σχήμα 11 διάγραμμα κλάσεων

## 7.7 Διαγράμματα ακολουθίας

Τα διαγράμματα ακολουθίας (sequence diagrams) είναι διαγράμματα που εμφανίζουν την επικοινωνία αντικειμένων για την επίτευξη κάποιας συμπεριφοράς. Σε ένα διάγραμμα ακολουθίας τα αντικείμενα τοποθετούνται οριζόντια.

Κάτω από τα αντικείμενα σχεδιάζεται η γραμμή ζωής (lifeline) που αναπαριστά τη διάρκεια ζωής των αντικειμένων και σχεδιάζεται ως κατακόρυφη διακεκομμένη γραμμή κάτω από τα αντικείμενα.

Τα μηνύματα είναι απλά βέλη με συμπαγή άκρη. Τα άκρα τους τοποθετούνται στις γραμμές ζωής με το βέλος να δείχνει προς τον παραλήπτη του μηνύματος.

Προκειμένου να είναι εμφανή τα αντικείμενα που συμμετέχουν σε ένα κύκλο αποστολής μηνυμάτων, τοποθετούνται μπάρες προδιαγραφής εκτέλεσης (execution specification) πάνω στις γραμμές ζωής. Οι μπάρες προδιαγραφής εκτέλεσης δείχνουν τη ροή ελέγχου στην ανταλλαγή μηνυμάτων σε αντικείμενα που έχουν ενεργοποιηθεί. Ένα ενεργοποιημένο αντικείμενο είτε εκτελεί μία δική του λειτουργία είτε περιμένει την επιστροφή, όταν αποστέλλει ένα μήνυμα σε κάποιο άλλο αντικείμενο. Η μπάρα προδιαγραφής εκτέλεσης εκτείνεται προς τα κάτω, όταν το αντικείμενο περιμένει την επιστροφή από κάποιο μήνυμα που έχει αποστείλει. [36]

Στον Πίνακα 35 περιγράφεται ο συμβολισμός μερικών βασικών λειτουργιών των κλάσεων σε ένα διάγραμμα ακολουθίας.

Πίνακας 35 συμβολισμός αντικειμένων σε διάγραμμα κλάσεων

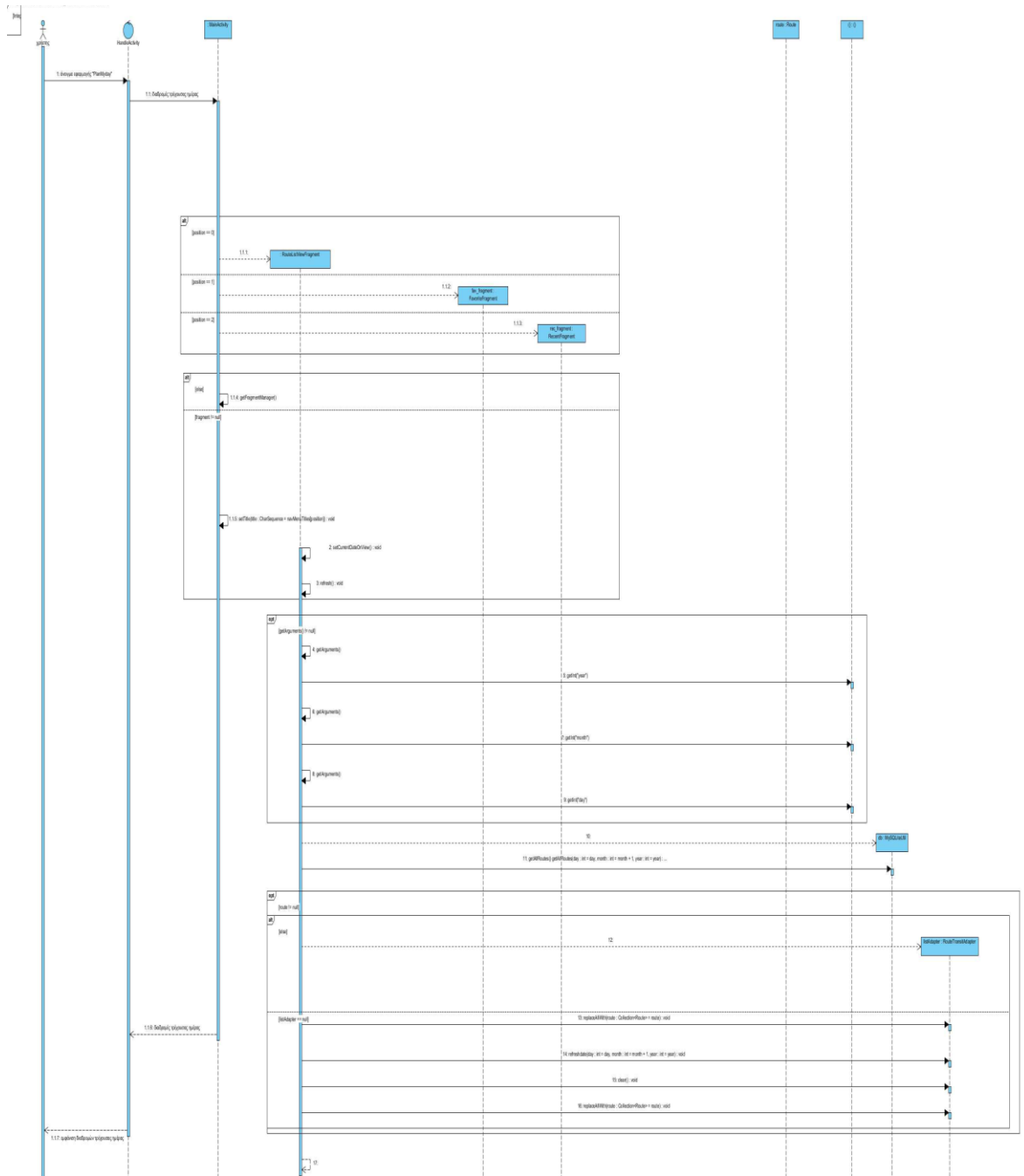
ΛΕΙΤΟΥΡΓΙΕΣ	ΣΥΜΒΟΛΙΣΜΟΣ
Επανάληψη	Η επανάληψη απεικονίζεται με τη χρήση πλαισίων που περικλείουν τη ροή των μηνυμάτων που ανήκει στην επανάληψη. Το πλαίσιο ονομάζεται με την ένδειξη loop η οποία συνοδεύεται από τη συνθήκη της επανάληψης μέσα σε αγκύλες.
Μήνυμα υπό συνθήκη	Η αποστολή μηνυμάτων όταν αληθεύει μια συνθήκη γίνεται με τη χρήση πλαισίων. Η ένδειξη στο πλαίσιο είναι opt η οποία συνοδεύεται με τη συνθήκη. Με παρόμοιο τρόπο αντιμετωπίζεται η αποστολή αμοιβαίως αποκλειόμενων μηνυμάτων. Η ένδειξη στο πλαίσιο γίνεται alt και τα δύο εναλλακτικά μονοπάτια διαχωρίζονται με μια διακεκομμένη γραμμή.
Δημιουργία αντικειμένου	Οι λειτουργίες καθορίζουν τη συμπεριφορά των αντικειμένων της κλάσης και παραπέμπουν στις μεθόδους (methods) μιας κλάσης της Java.

### 7.7.1 Διαγράμματα ακολουθίας της εφαρμογής

Το διάγραμμα ακολουθίας της εφαρμογής, έχει χωριστεί σε μικρότερα και το καθένα περιγράφει μια βασική λειτουργία. Αυτό έγινε με σκοπό τη διευκόλυνση των αναγνωστών. Από τα διαγράμματα επίσης, έχουν αφαιρεθεί μέθοδοι του συστήματος Android που σχετίζονται με κλάσεις γραφικών. Όλα τα διαγράμματα έχουν υλοποιηθεί στο Visual Paradigm. [31]

#### Λειτουργία – Ημερήσιες Διαδρομές

Στο Σχήμα 12, παρουσιάζεται το τρόπο που ο χρήστης αλληλοεπιδρά με τη λειτουργία “Ημερήσιες Διαδρομές”, σε ένα διάγραμμα ακολουθίας.

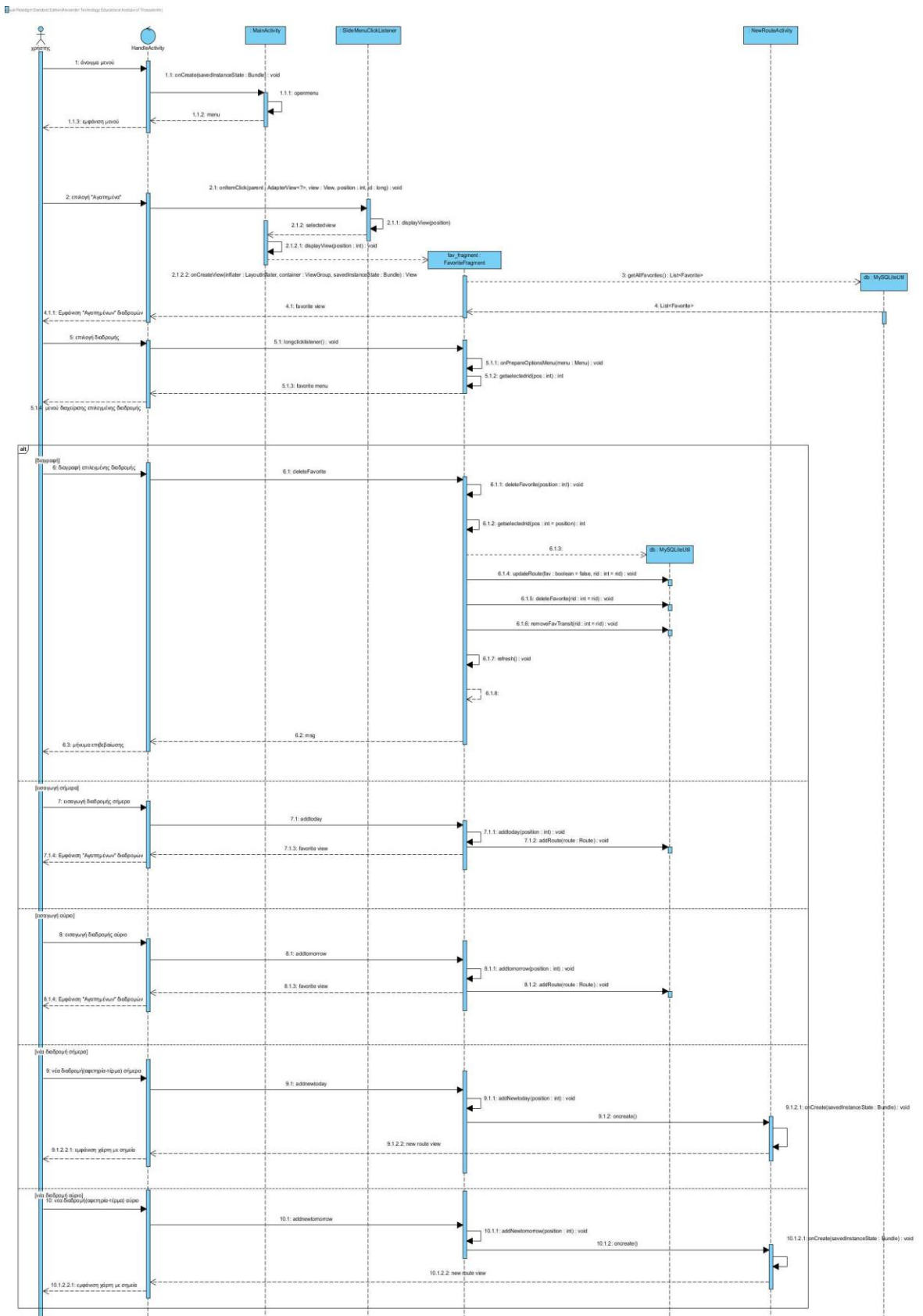


Σχήμα 12 διάγραμμα ακολουθίας – Ημερήσιες Διαδρομές

**Λειτουργία – Αγαπημένες Διαδρομές**

Στο Σχήμα 13, περιγράφεται η λειτουργία "Αγαπημένες Διαδρομές" μέσα από ένα διάγραμμα ακολουθίας.

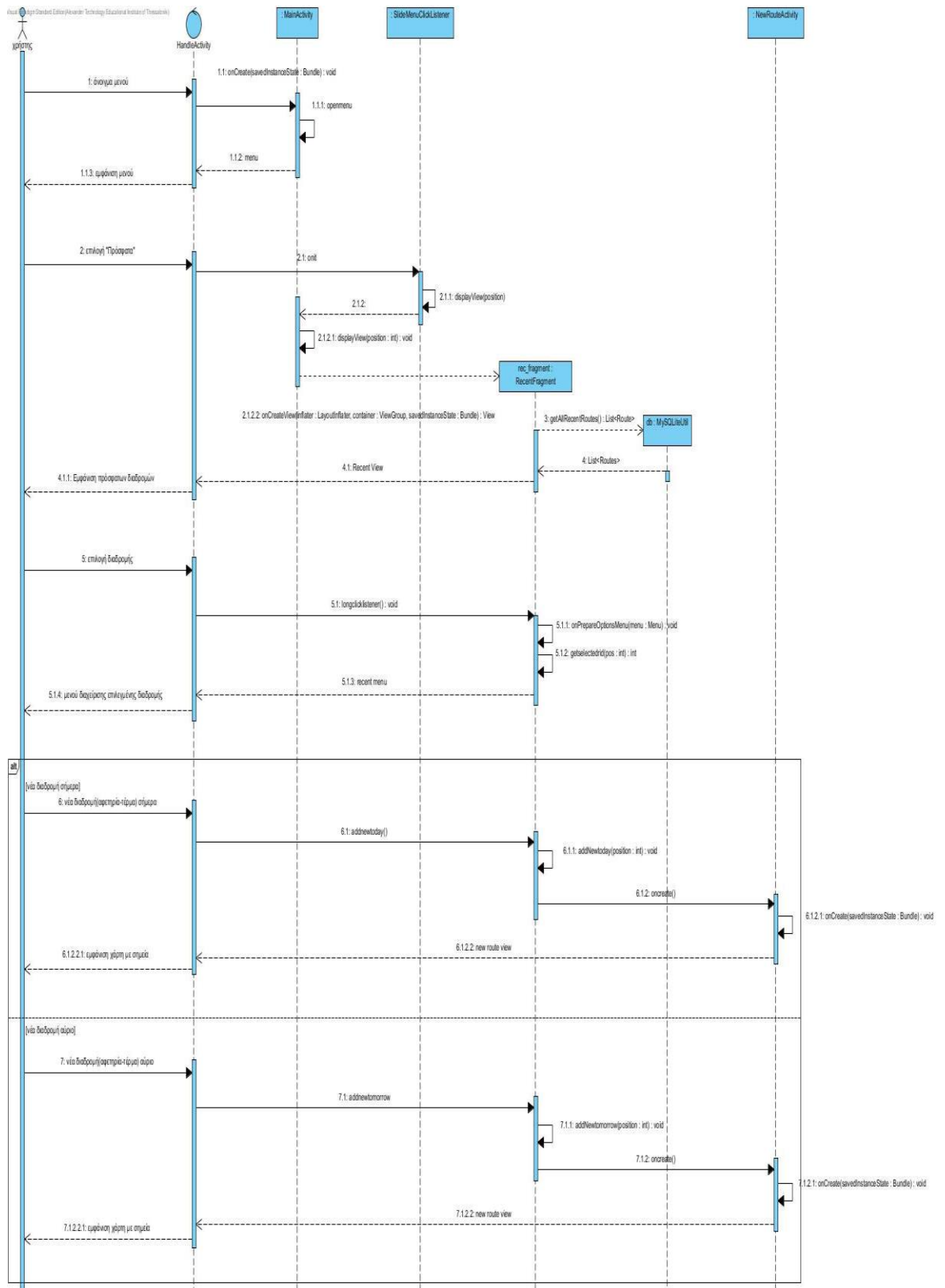
# Πτυχιακή εργασία του φοιτητή Λαμπρόπουλου Αλέξανδρου



Σχήμα 13 διάγραμμα ακολουθίας – Αγαπημένες Διαδρομές

Λειτουργία – Πρόσφατες Διαδρομές

Στο Σχήμα 14, παρουσιάζεται η αλληλεπίδραση του χρήστη με τη λειτουργία των Πρόσφατων Διαδρομών, σε ένα διάγραμμα ακολουθίας.

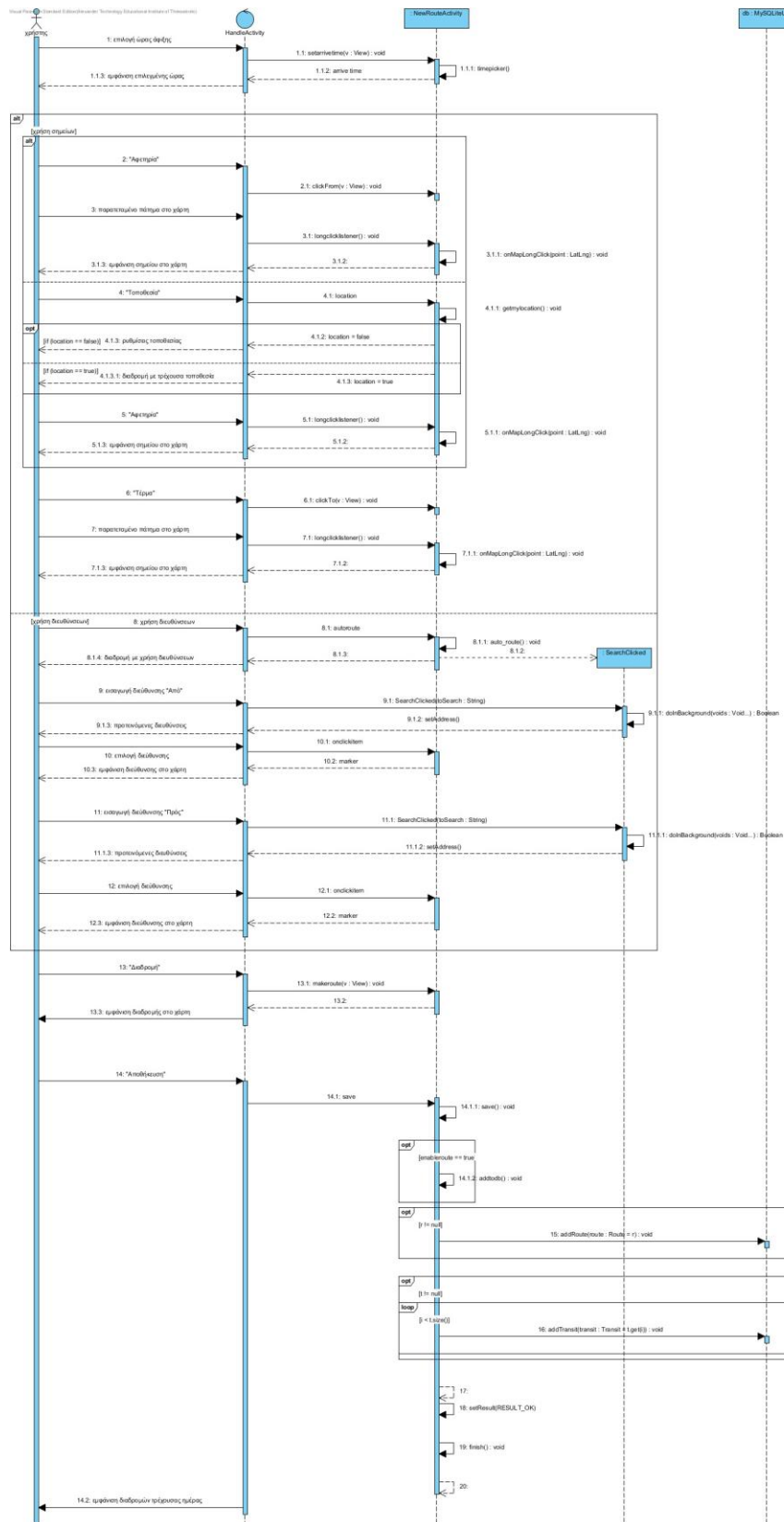


Σχήμα 14 διάγραμμα ακολουθίας – Πρόσφατες Διαδρομές



Λειτουργία – Νέα Διαδρομή

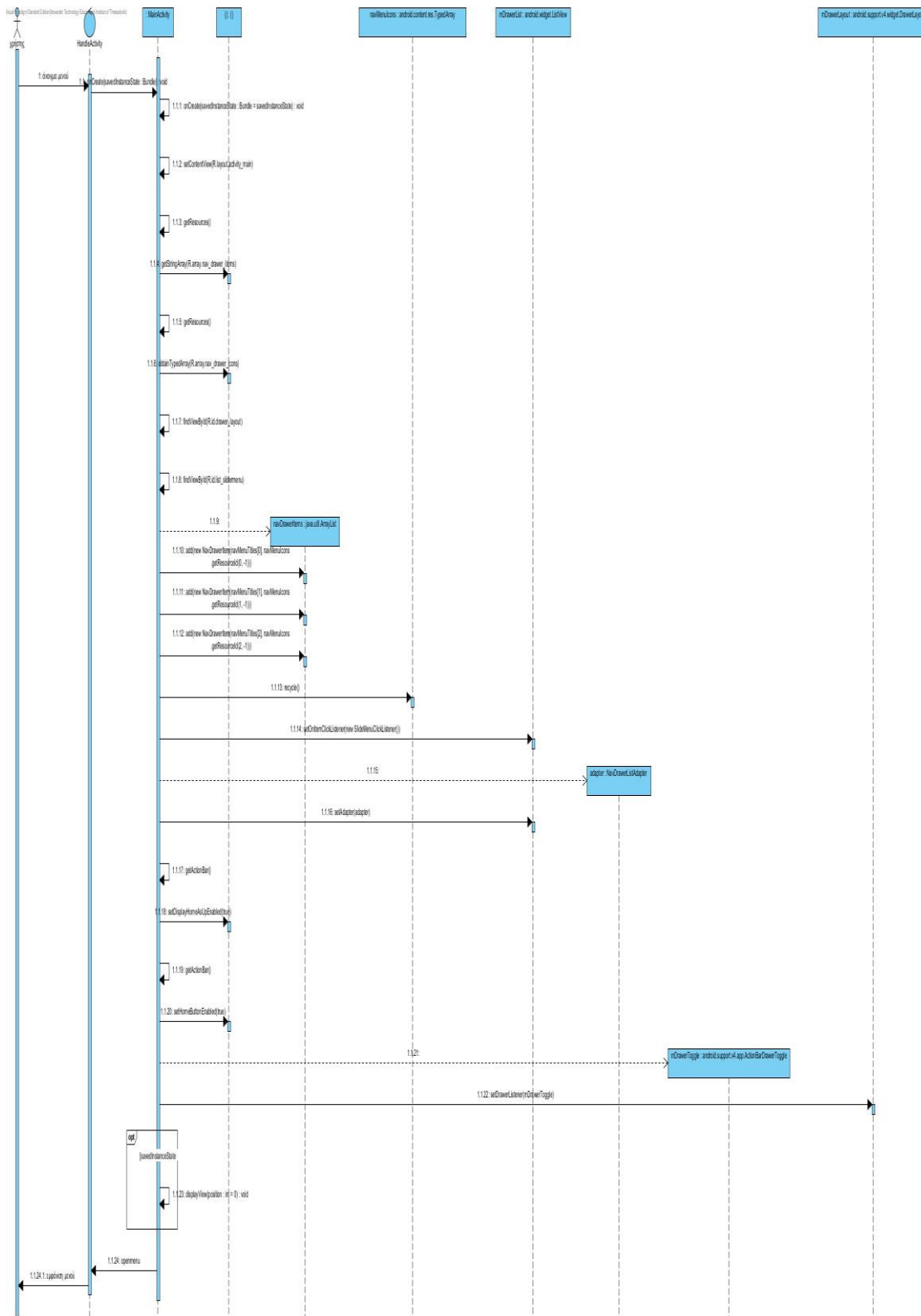
Στο διάγραμμα ακολουθίας που φαίνεται στο Σχήμα 15, περιγράφεται η διαδικασία δημιουργίας μιας νέας διαδρομής από τον χρήστη.



Σχήμα 15 διάγραμμα ακολουθίας – Νέα Διαδρομή

Λειτουργία – Μενού

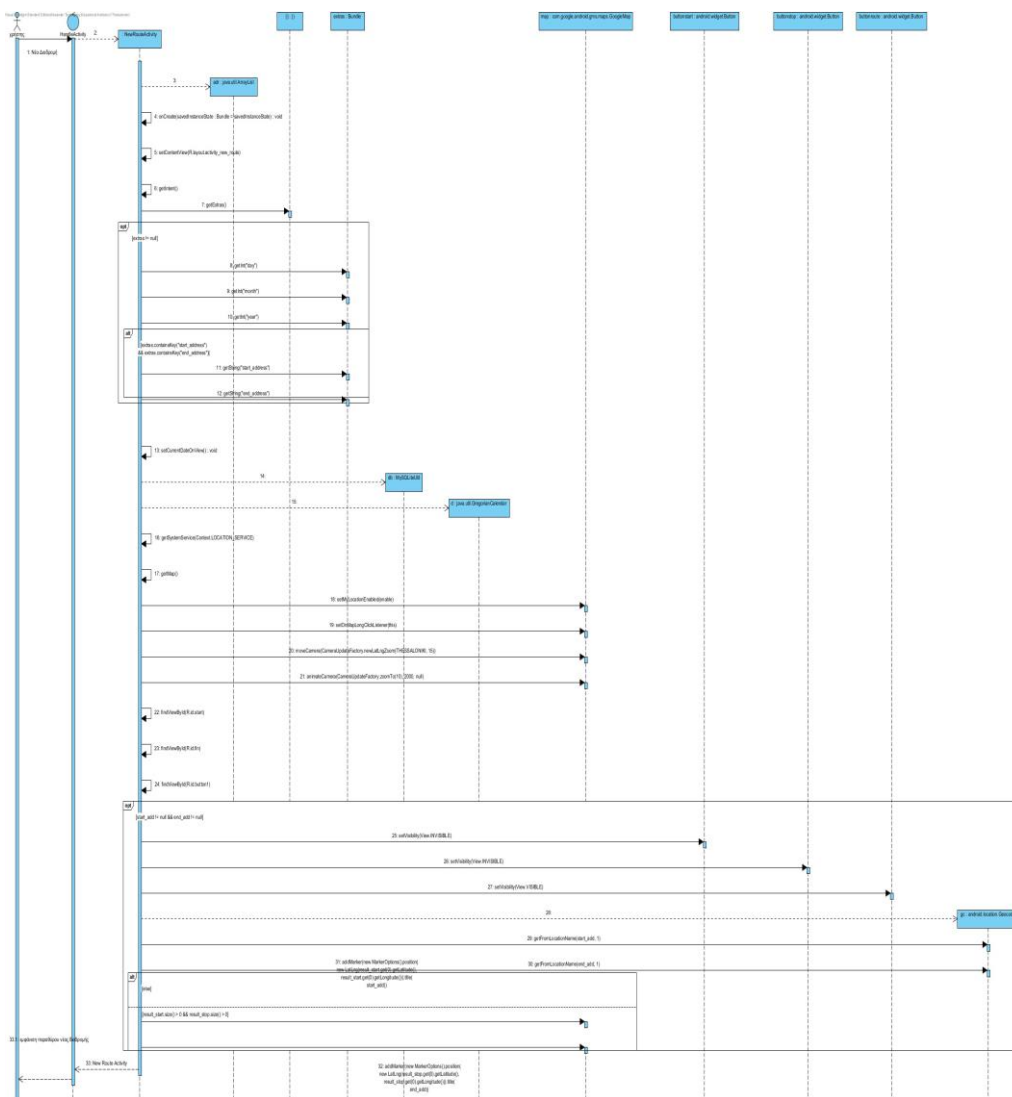
Στο Σχήμα 16, παρουσιάζεται το διάγραμμα ακολουθίας στη περίπτωση που ο χρήστης αλληλοεπιδρά με το πλάγιο μενού (slidemenu). [13]



Σχήμα 16 διάγραμμα ακολουθίας – Πλάγιο Μενού

### Μέθοδος – Create NewRouteActivity

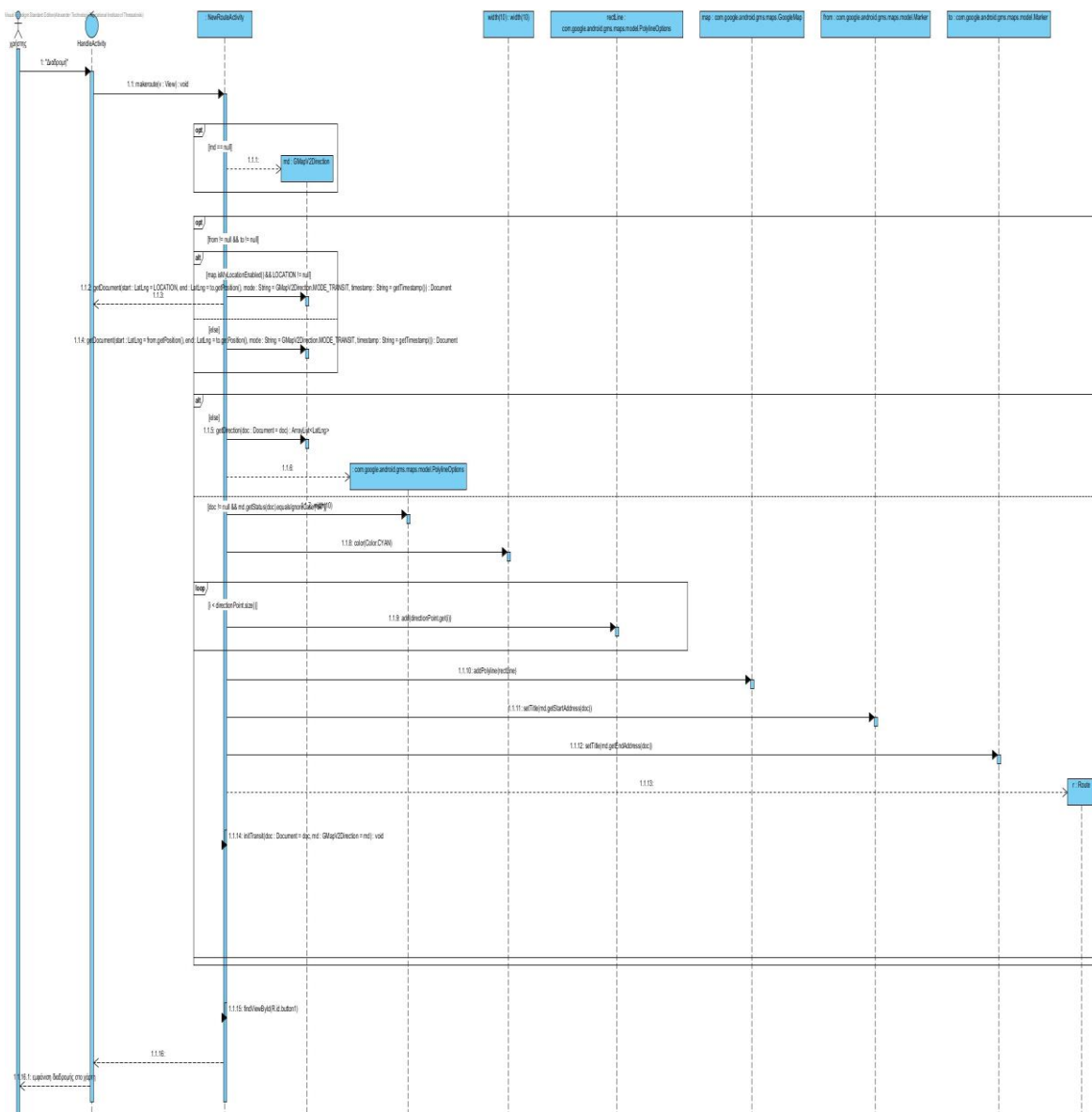
Σε αυτό το σημείο περιγράφεται μέσο ενός διαγράμματος ακολουθίας, πως δημιουργείται μια Δραστηριότητα (Activity) στο Android. Στην περίπτωση μας παρουσιάζουμε τη δημιουργία της Δραστηριότητας NewRouteActivity. Στο Σχήμα 17, φαίνεται ένα μέρος της δημιουργίας μιας Δραστηριότητας (NewRouteActivity).



Σχήμα 17 διάγραμμα ακολουθίας – Create NewRouteActivity

### Μέθοδος – MakeRoute

Το διάγραμμα ακολουθίας “MakeRoute” που φαίνεται στο Σχήμα 18, παρουσιάζει τη μέθοδο σχεδίασης διαδρομής στο χάρτη (Google Maps). Η μέθοδος αυτή χρησιμοποιεί δεδομένα από την Υπηρεσία Ιστού της Google για Οδηγίες (Directions), που περιγράψαμε σε προηγούμενο κεφάλαιο.



Σχήμα 18 διάγραμμα ακολουθίας – MakeRoute

## 7.8 Διαγράμματα πακέτων

Τα πακέτα της UML είναι ένα μηχανισμός γενικού σκοπού για την οργάνωση και ομαδοποίηση των στοιχείων μοντελοποίησης. Τα πακέτα της UML παραπέμπουν σε μεγάλο βαθμό στα πακέτα της Java και έχουν δύο χρήσεις:

- την οργάνωση των μοντέλων που δημιουργούνται με τη UML,
- την απεικόνιση της οργάνωσης των κλάσεων του λογισμικού σε πακέτα λογισμικού του κώδικα. [36]

Το διάγραμμα πακέτων στη περίπτωση της εφαρμογής αυτής, είναι το ίδιο με το διάγραμμα κλάσεων στο Σχήμα 10. [30]

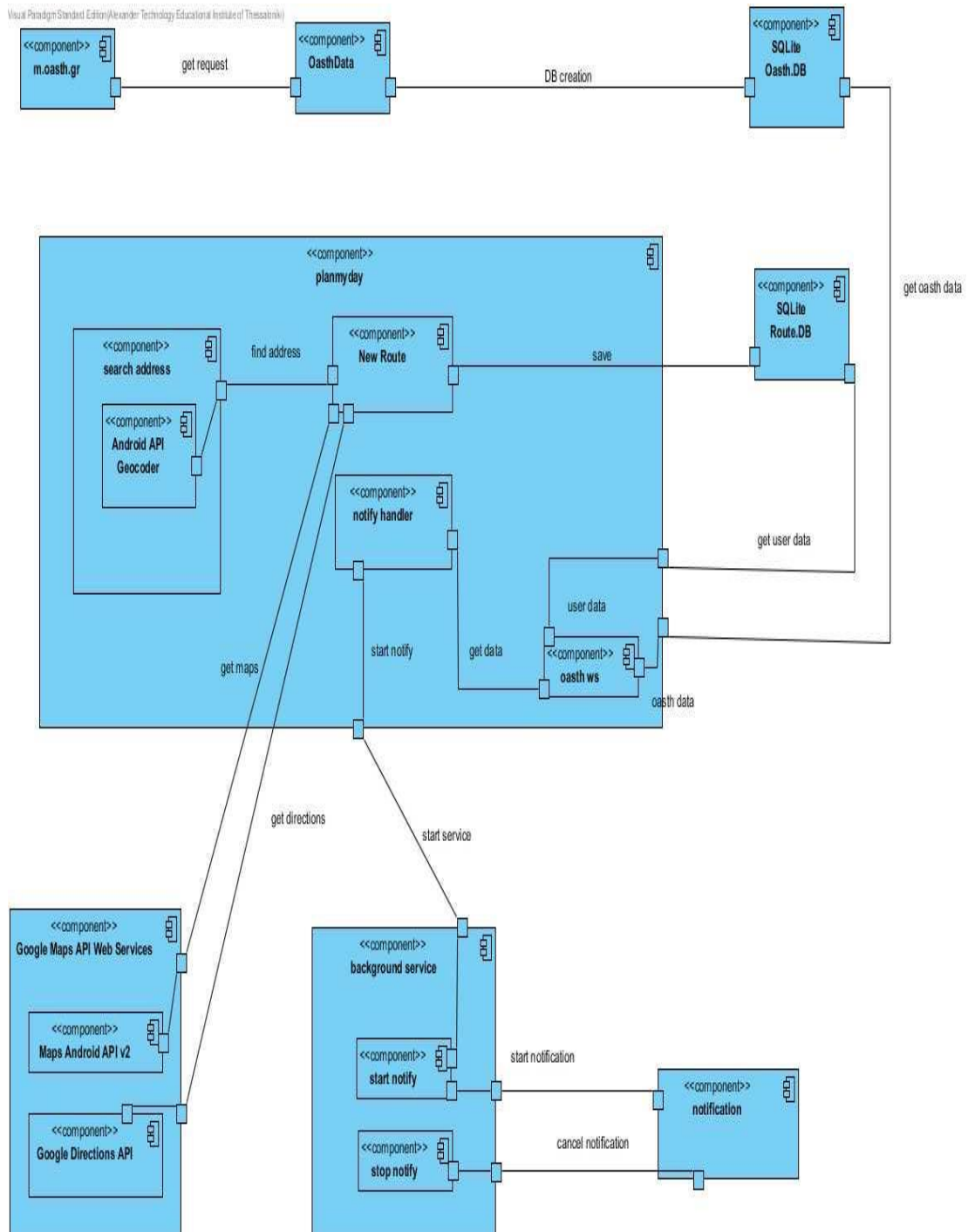
## 7.9 Διαγράμματα συστατικών

Η UML παρέχει τα διαγράμματα συστατικών (component diagrams) για να επιδείξει την ύπαρξη και χρήση των συστατικών στο λογισμικό. Τα διαγράμματα συστατικών δεν ασχολούνται

με τη φυσική υπόσταση των συστατικών και δεν δείχνουν το πως εντάσσονται σε φυσικό επίπεδο. Η φυσική υπόσταση των συστατικών είναι τα προϊόντα (artifacts) τα οποία εμφανίζονται σε διαγράμματα παράταξης. Τα διαγράμματα συστατικών εμφανίζουν τη λογική τους υπόσταση ή την υπόσταση που παίρνουν στο χρόνο εκτέλεσης του λογισμικού. [36]

### 7.9.1 Διάγραμμα συστατικών της εφαρμογής

Στο διάγραμμα συστατικών της παρούσας εφαρμογής, περιλαμβάνεται και το πρόγραμμα «OasthData» ως συστατικό (component). Στο Σχήμα 19 παρουσιάζεται το διάγραμμα συστατικών της εφαρμογής. Το διάγραμμα έχει σχεδιαστεί στο Visual Paradigm. [25]



Σχήμα 19 διάγραμμα συστατικών

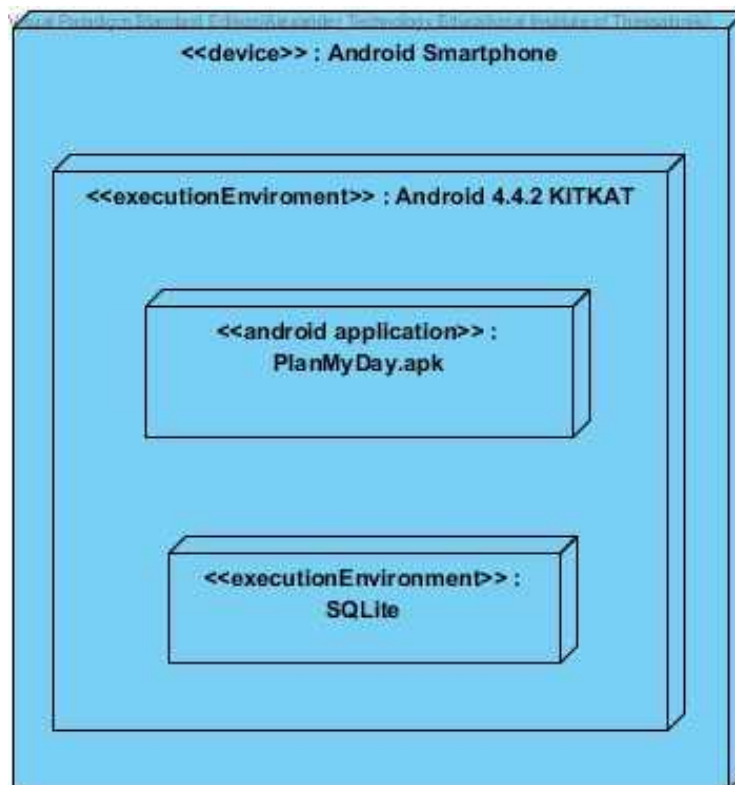
## 7.10 Διαγράμματα παράταξης

Τα διαγράμματα πακέτων παρέχουν μια εικόνα της λογικής οργάνωσης του οργανισμού σε μονάδες προγράμματος. Τα διαγράμματα παράταξης (deployment diagrams) παρουσιάζουν το πραγματικό περιβάλλον στο οποίο λειτουργεί το λογισμικό. Ένα διάγραμμα παράταξης παρουσιάζει συσκευές του υλικού και τις φυσικές μονάδες λογισμικού που διανέμονται στο υλικό. Τα διαγράμματα παράταξης είναι ένα μέσο για την τεκμηρίωση της φυσικής αρχιτεκτονικής του συστήματος.

Το βασικό στοιχείο ενός διαγράμματος παράταξης είναι οι κόμβοι (nodes). Η πρώτη κατηγορία των κόμβων είναι οι συσκευές (device) που αναπαριστούν φυσικές μονάδες επεξεργασίας που αντιστοιχούν στο υλικό. Η δεύτερη κατηγορία των κόμβων είναι τα περιβάλλοντα εκτέλεσης (execution environments). Περιβάλλοντα εκτέλεσης είναι λογισμικό συστημάτων, όπως λειτουργικά συστήματα, συστήματα διαχείρισης βάσεων δεδομένων κ.α. [36]

### 7.10.1 Διάγραμμα παράταξης της εφαρμογής

Στο Σχήμα 20, παρουσιάζεται το διάγραμμα παράταξης της παρούσας εφαρμογής. Το διάγραμμα έχει σχεδιαστεί στο Visual Paradigm. [26]

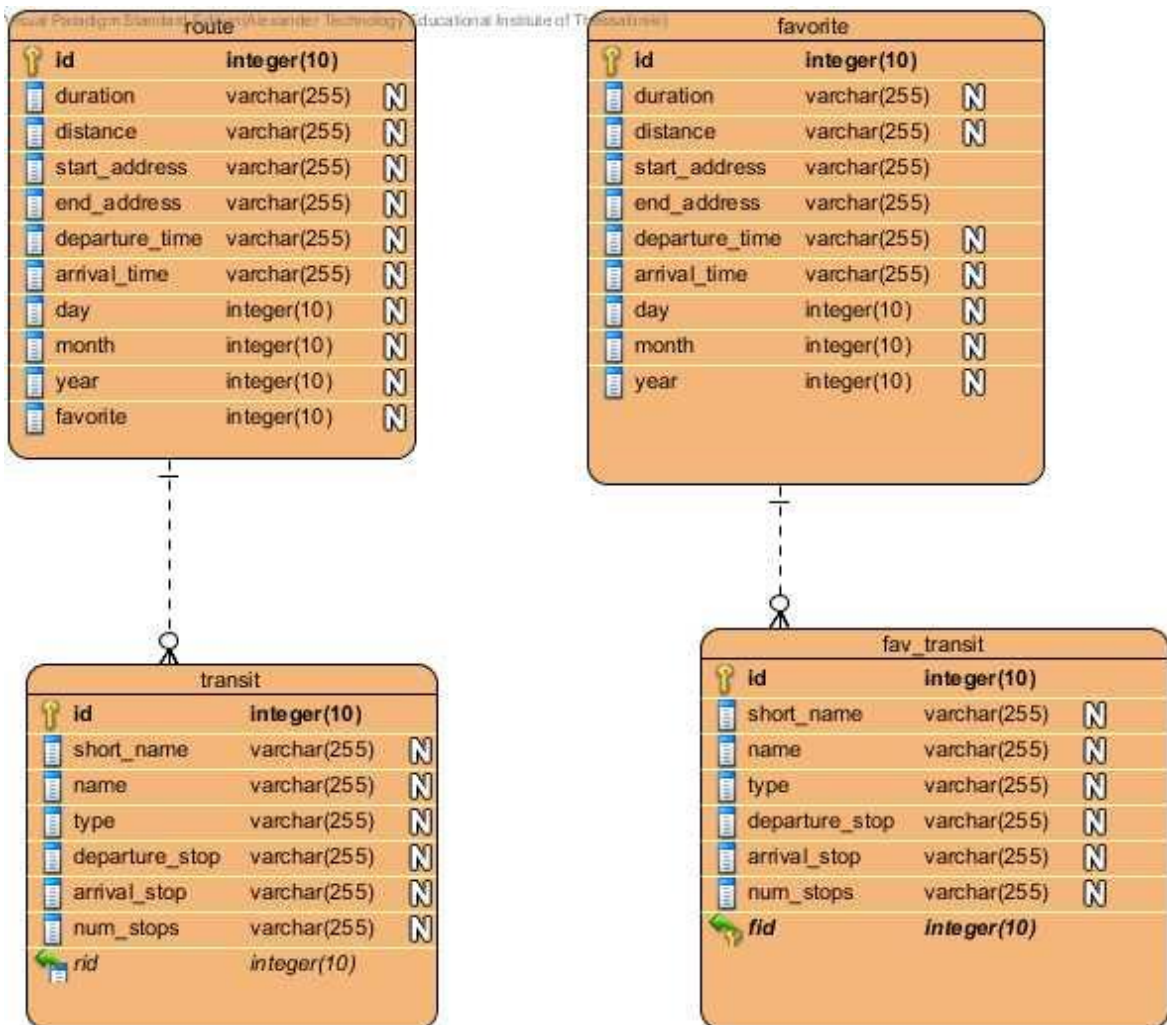


Σχήμα 20 διάγραμμα παράταξης

### 7.11 Διάγραμμα Βάσης Δεδομένων

Στο Σχήμα 21 παρουσιάζεται το σχήμα της βάσης δεδομένων του χρήστη. Η βάση αυτή αποτελείται από τους πίνακες “route”, “favorite”, “transit”, “fav\_transit”. Δημιουργείται δυναμικά, όταν ο χρήστης αποθηκεύει μια διαδρομή σε ένα πίνακα, σε αντίθεση με την βάση δεδομένων του ΟΑΣΘ που είναι σταθερή και δεν μεταβάλλεται. Για τη βάση του ΟΑΣΘ θα μιλήσουμε σε παρακάτω κεφάλαιο.

Ο πίνακας “route” περιγράφει μια διαδρομή και αποτελείται από ένα ή περισσότερα (σχέση ένα προς πολλά) “transit” (μεταφορικά μέσα). Η ίδια σχέση ισχύει ανάμεσα στον πίνακα “favorite” και “fav\_transit”, μόνο που εδώ περιγράφονται οι αγαπημένες διαδρομές του χρήστη. Το διάγραμμα έχει υλοποιηθεί με το εργαλείο Visual Paradigm.



Σχήμα 21 βάση δεδομένων Διαδρομών του Χρήστη

## 7.12 Χρήση αποθετηρίου

Για τον έλεγχο εκδόσεων και την διαχείριση του κώδικα της παρούσας Πτυχιακής Εργασίας, έγινε χρήση του Bitbucket Repository. (7)

Το Bitbucket είναι μια υπηρεσία φιλοξενίας πρότζεκτ που χρησιμοποιούν Mercurial ή GIT σύστημα. Προσφέρει λογαριασμό δωρεάν (μέχρι πέντε χρήστες) ενώ ο λογαριασμός μπορεί να περιέχει απεριόριστο αριθμό ιδιωτικών αποθετηρίων. Ο περιορισμός των χρηστών μπορεί να αλλάξει σε περίπτωση που οι χρήστες θα προσκαλέσουν και άλλους στην υπηρεσία. Τα ιδιωτικά αποθετήρια δεν είναι προσβάσιμα από άλλους χρήστες χωρίς την έγκριση του ιδιοκτήτη.

Ο χρήστης έχει τη δυνατότητα να χρησιμοποιήσει κάποιες εφαρμογές πελάτες του Bitbucket (SourceTree), για να ανεβάσει το κώδικα στην πλατφόρμα εκτός από το GIT σε γραμμή εντολών. Επίσης η υπηρεσία δίνει τη δυνατότητα στο χρήστη να επιλέξει πρότζεκτ τύπου Android.

## 7.13 Πρόγραμμα δημιουργίας Βάσης Δεδομένων ΟΑΣΘ

Η υλοποίηση του προγράμματος έγινε σε Android για περαιτέρω τριβή με το αντικείμενο της Πτυχιακής Εργασίας. Σκοπός του προγράμματος είναι η δημιουργία μίας βάσης δεδομένων (SQLite) με δεδομένα όλα τα στοιχεία του ΟΑΣΘ.

Το πρόγραμμα αποτελείται από πέντε κλάσεις οι οποίες περιγράφονται στον Πίνακα 36.

Πίνακας 36 περιγραφή των κλάσεων του προγράμματος ΟΑΣΘ που δημιουργήθηκε

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
GetOasthDataSQLite	Επεκτείνει την κλάση SQLiteOpenHelper. Περιέχει όλες τις μεθόδους δημιουργίας της βάσης δεδομένων
Oasth_busline	Αντικείμενο που περιέχει τα πεδία (id, short_name, name) που αναφέρονται στη γραμμή του λεωφορείου (busline)
Oasth_busroute	Αντικείμενο που περιέχει τα πεδία (lid, sid, sorder, direction, rc) που αναφέρονται στη διαδρομή του λεωφορείου (busroute)
Oasth_busstop	Αντικείμενο που περιέχει τα πεδία (id, name) που αναφέρονται στη στάση του λεωφορείου (busstop)
OasthDataActivity	Δραστηριότητα που χρησιμοποιεί στιγμιότυπα από τις παραπάνω κλάσεις

### 7.13.1 Πως λειτουργεί

Μετά από μελέτη των παραμέτρων που περιγράφονται στον Πίνακα 14, σχεδιάστηκε ένα πρόγραμμα που δημιουργεί μια βάση δεδομένων με όλες τις παραμέτρους αυτές, για κάθε λεωφορείο, κάθε στάση και κάθε κατεύθυνση.

Οι πρώτες δοκιμές έγιναν με τη βοήθεια του restclient, πρόσθετου σε Firefox και Chrome. Μέσα από το restclient, στάλθηκαν ερωτήματα στο παραπάνω URL για την επεξεργασία των απαντήσεων που βοήθησαν στη λογική του προγράμματος.

Το πρόγραμμα πηγαίνει σε κάθε επίπεδο της σελίδας του ΟΑΣΘ και αποθηκεύει τη χρήσιμη πληροφορία. Αυτό γίνεται στέλνοντας HTTP ερωτήματα, στην ιστοσελίδα για κινητά του ΟΑΣΘ (10). Τα ερωτήματα έχουν την μορφή:



- `url = "http://m.oasth.gr/index.php?md=3&sn=3&start=" + sid + "&sorder="+ sorder + "&rc="+ rc + "&line=" + lid + "&dir=" + direction;`

Οι παράμετροι που χρησιμοποιούνται στο URL περιγράφονται στον Πίνακα 37.

Πίνακας 37 περιγραφή των παραμέτρων στο URL της ιστοσελίδας ΟΑΣΘ

ΠΑΡΑΜΕΤΡΟΣ	ΠΕΡΙΓΡΑΦΗ
md	3 <sup>ο</sup> επίπεδο/επιλογή στην ιστοσελίδα
sn	3 <sup>ο</sup> επίπεδο/επιλογή στην ιστοσελίδα
start	Κωδικός στάσης
sorder	Αρίθμηση στάσης κατά αύξοντα αριθμό, σε μετάβαση και επιβίβαση
rc	Διαφορετικό ανά κατεύθυνση
line	Κωδικός της γραμμής του λεωφορείου
dir	Κατεύθυνση λεωφορείου (1=μετάβαση, 2=επιστροφή)

Λαμβάνει την ζητούμενη σελίδα την οποία “κόβει” σε κάποια βασικά σημεία για να πάρει τα δεδομένα που χρειάζεται, τα οποία έπειτα αποθηκεύει στη βάση δεδομένων με τη βοήθεια της κλάσης `GetOasthDataSQLite`.

Τα δεδομένα αυτά χρησιμοποιούνται ξανά στην ιστοσελίδα του ΟΑΣΘ για τα κινητά (10). Αυτή τη φορά για να λάβουν ως απάντηση την ώρα άφιξης του ζητούμενου λεωφορείου στη στάση. Στέλνονται HTTP ερωτήματα στην ιστοσελίδα με δεδομένα της βάσης ως παραμέτρους:

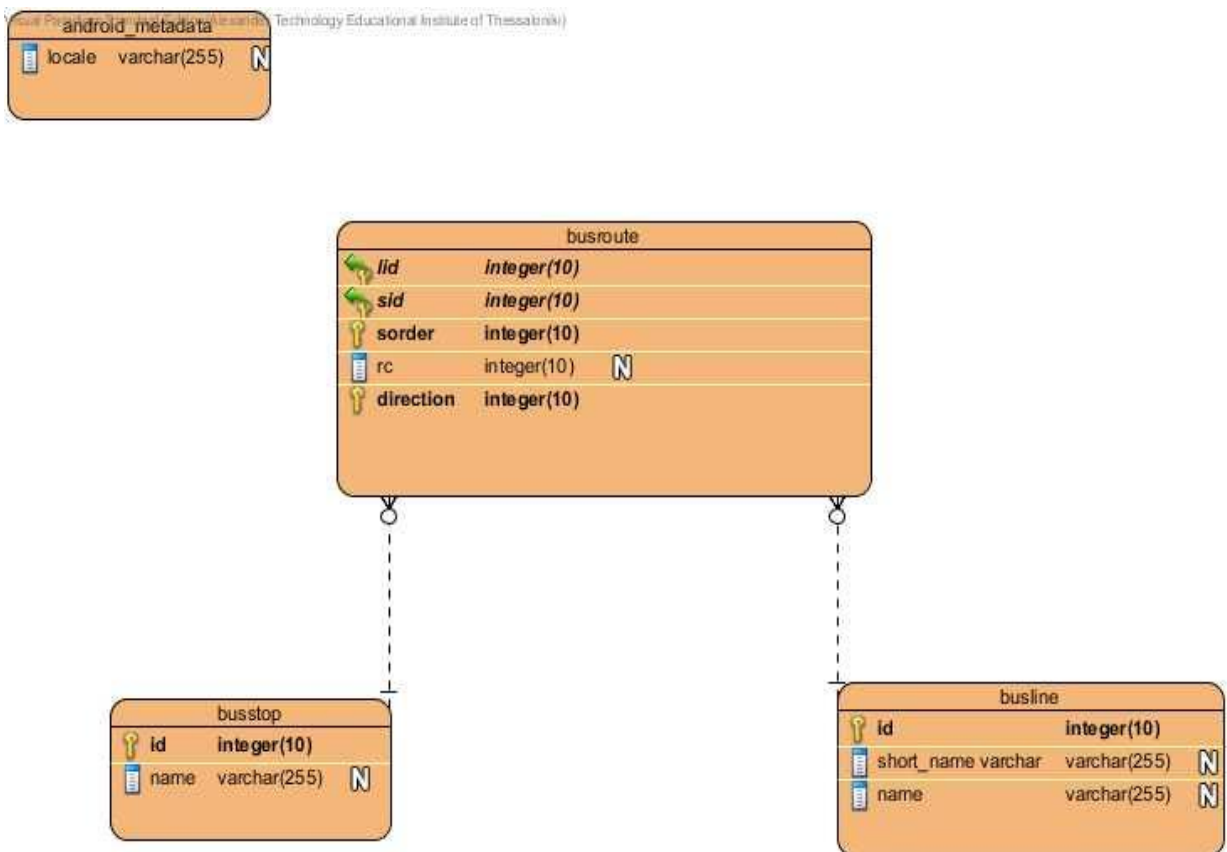
- `url = "http://m.oasth.gr/index.php?md=3&sn=3&start=" + sid + "&sorder="+ sorder + "&rc=" + rc + "&line=" + lid + "&dir=" + dir;`

Οι παράμετροι που χρησιμοποιούνται στο URL περιγράφονται στον Πίνακα 37.

### 7.13.2 Βάση Δεδομένων ΟΑΣΘ

Το σχήμα της Βάσης Δεδομένων που δημιουργεί το πρόγραμμα παρουσιάζεται στο Σχήμα 22. Η βάση αποτελείται από τρεις πίνακες, “busroute”, “busline” και “busstop”. Η βάση παραμένει σταθερή και δεν μεταβάλλεται κατά τη διάρκεια χρήσης της εφαρμογής “PlanMyDay” από τον χρήστη.

Ο πίνακας “busroute” περιγράφει μια διαδρομή με λεωφορεία και αποτελείται από πολλά “busline” (γραμμές λεωφορείων) και πολλά “busstop” (στάσεις λεωφορείων). Το διάγραμμα έχει υλοποιηθεί με το εργαλείο Visual Paradigm.



Σχήμα 22 Βάση Δεδομένων ΟΑΣΘ

Στο Σχήμα 23 και 24, παρουσιάζονται κάποια στιγμιότυπα πινάκων, της Βάσης Δεδομένων του ΟΑΣΘ. Η διαχείριση της βάσης δεδομένων έγινε με τη βοήθεια του SQLite Manager, που είναι πρόσθετο στο πρόγραμμα περιήγησης ιστού Firefox.

Πτυχιακή εργασία του φοιτητή Λαμπρόπουλου Αλέξανδρου

TABLE

id	name
833	Τ.Σ. ΠΥΛΛΙΑΣ
834	Τ.Σ. ΠΑΝΟΡΑΜΑ Ν.751
837	Τ.Σ. ΟΔΟΥ ΜΠΟΣΝΑΚΗ (ΕΠΤΑΠΥΡΓΙΟ)
838	ΑΝΑΤΟΛΙΚΟΣ ΣΤΑΘΜΟΣ ΙΚΕΑ Α
839	Τ.Σ. ΜΗΧΑΝΙΩΝΑ
840	Τ.Σ. ΕΠΑΝΟΜΗΣ ΓΡ77
841	Τ.Σ. ΚΕΡΑΣΙΑΣ
842	Τ.Σ. ΣΙΝΔΟΥ
843	ΑΕΡΟΔΡΟΜΙΟ ΜΑΚΕΔΟΝΙΑ - ΑΦΙΞΕΙΣ
875	ΚΥΒΕΛΕΙΑ
876	ΙΠΠΟΚΡΑΤΕΙΟ
877	ΔΗΜΗΤΡΙΟΥ ΜΗΤΡΟΠΟΥΛΟΥ
878	ΒΡΥΣΑΚΙ
879	25ΗΣ ΜΑΡΤΙΟΥ
881	ΘΕΑΓΕΝΕΙΟ
882	Τ.Σ. ΑΓΓΕΛΟΧΩΡΙΟΥ
884	ΑΓΙΑΣ ΣΟΦΙΑΣ
885	ΙΑΣΩΝΙΔΟΥ
886	ΚΑΜΑΡΑ
887	ΑΓΓΕΛΑΚΗ
888	ΜΟΥΣΕΙΑ
889	ΣΤΡΑΤΗΓΕΙΟ - ΜΟΥΣΕΙΟ ΒΥΖΑΝΤΙΝΟΥ ΠΟΛΙΤΙΣΜΟΥ

<< < 1 to 100 of 3600 > >>

Σχήμα 23 στιγμιότυπο πίνακα busstop

TABLE

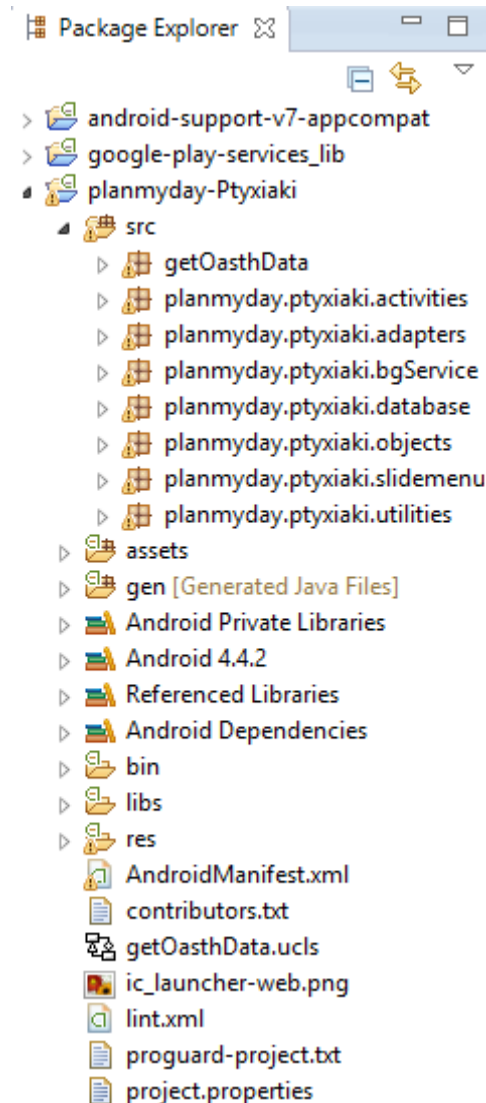
id	short_name	name
434	01	CARREFOUR - ΝΟΣ. ΠΑΠΑΓΕΩΡΓΙΟΥ - Κ.Τ.Ε.Λ.
450	01A	CARREFOUR - ΝΟΣ. ΠΑΠΑΓΕΩΡΓΙΟΥ - Κ.Τ.Ε.Λ. ΜΕΣ.
67	02	Α.Σ. ΙΚΕΑ - Ν.Σ. ΣΤΑΘΜΟΣ ΜΕΣΩ ΕΓΝΑΤΙΑΣ
469	02A	Α.Σ. ΙΚΕΑ - Ν.Σ. ΣΤΑΘΜΟΣ ΜΕΣΩ ΑΓΡΟΚΤΗΜΑΤΟΣ
68	03	Α.Σ. ΙΚΕΑ - Ν.Σ. ΣΤΑΘΜΟΣ
473	04A	ΧΑΡΙΛΑΟΥ - Κ.Τ.Ε.Λ. ΧΑΛΚΙΔΙΚΗΣ - ΚΑΛΑΜΑΡΙΑ
69	04B	Κ.Τ.Ε.Λ. ΧΑΛΚΙΔΙΚΗΣ - ΧΑΡΙΛΑΟΥ - ΚΑΛΑΜΑΡΙΑ
537	04E	ΚΑΛΑΜΑΡΙΑ - Κ.Τ.Ε.Λ. ΧΑΛΚΙΔΙΚΗΣ - ΓΕΡΜΑΝΙΚΗ Σ
74	05	Ν.ΚΡΗΝΗ - ΒΕΝΙΖΕΛΟΥ
297	05A	ΝΕΑ ΚΡΗΝΗ - ΒΕΝΙΖΕΛΟΥ - ΣΧΟΛΗ ΔΙΚΑΣΤΩΝ
466	05B	ΚΑΛΑΜΑΡΙΑ - ΒΕΝΙΖΕΛΟΥ
11	06	ΚΑΛΑΜΑΡΙΑ-ΒΕΝΙΖΕΛΟΥ
72	07	ΑΓ. ΙΩΑΝΝΗΣ-ΠΑΝΕΠΙΣΤΗΜΙΟ
73	08	ΙΚΕΑ-ΚΤΕΛ ΜΕΣΩ ΕΓΝΑΤΙΑΣ
70	09	Ν.Σ. ΣΤΑΘΜΟΣ-ΛΑΧΑΝΑΓΟΡΑ
197	09A	Ν.Σ. ΣΤΑΘΜΟΣ - ΜΕΣΑ ΑΠΟ ΤΗ ΛΑΧΑΝΑΓΟΡΑ
539	09B	Ν.Σ. ΣΤΑΘΜΟΣ - ΛΑΧΑΝΑΓΟΡΑ Κ.Τ.Ε.Ο.
540	09E	Ν.Σ. ΣΤΑΘΜΟΣ - Κ.Τ.Ε.Ο. ΜΕΣΑ ΑΠΟ ΤΗ ΛΑΧΑΝΑΓ.
63	10	ΧΑΡΙΛΑΟΥ - Ν.Σ. ΣΤΑΘΜΟΣ
47	11	ΠΥΛΛΙΑ - Ν.Σ. ΣΤΑΘΜΟΣ
538	11B	ΠΥΛΛΙΑ - Ν.Σ. ΣΤΑΘΜΟΣ ΞΕΝΩΝΑΣ ΘΕΑΓΕΝΕΙΟΥ
71	12	ΚΤΕΛ ΚΑΤΟ ΤΟΥΜΠΑ

<< < 1 to 100 of 239 > >>

Σχήμα 24 στιγμιότυπο πίνακα busline

## 7.14 Το Project της Εφαρμογής

Η εφαρμογή έχει υλοποιηθεί στο περιβάλλον Eclipse IDE σε συνδυασμό με το πρόσθετο (plugin) Android Development Tools (ADT). Το πρότζεκτ αποτελείται από κλάσεις, αρχεία xml, εικόνες, βιβλιοθήκες και μια Βάση Δεδομένων. Η δομή παρουσιάζεται στο Σχήμα 25.



Σχήμα 25 δομή του Project

### 7.14.1 Πακέτα

Τα πακέτα είναι ένας τρόπος να οργανωθούν συλλογές κλάσεων Java σε βιβλιοθήκες. Με αυτό το τρόπο μπορεί να αποφευχθεί η σύγκρουση ονομάτων των κλάσεων. Ένα πακέτο μπορεί να περιέχει οποιοδήποτε αριθμό κλάσεων που σχετίζονται μεταξύ τους με βάση το σκοπό τους.

Η Java χρησιμοποιεί πακέτα για να σχηματίσει βιβλιοθήκες κλάσεων. Ένα πακέτο είναι μια ομάδα κλάσεων που έχουν τοποθετηθεί σε ένα κατάλογο. Τα πακέτα μπορούν να χρησιμοποιηθούν από κάθε πρόγραμμα - κλάση που περιλαμβάνει μια εντολή εισαγωγής (Import) που αναφέρεται στο αντίστοιχο πακέτο.

Οι κλάσεις στο πρότζεκτ της εφαρμογής, είναι χωρισμένες σε πακέτα ανάλογα με το σκοπό και την κληρονομικότητα τους και περιγράφονται παρακάτω.

### Δραστηριότητες (Activities)

Στο πακέτο αυτό περιέχονται όλες οι κλάσεις που επεκτείνουν (extends) την κλάση Activity. Είναι Δραστηριότητες (Activities) με τις οποίες αλληλοεπιδρά ο χρήστης. Στον Πίνακα 38 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 38 κλάσεις που ανήκουν στο πακέτο Δραστηριοτήτων

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
MainActivity	Είναι η κεντρική Δραστηριότητα που περικλείει τα Τεμάχια (Fragments) της εφαρμογής. Ο χρήστης βρίσκεται σε αυτή αμέσως μετά το Splash Screen.
NewRouteActivity	Στην Δραστηριότητα αυτή περιέχεται ο χάρτης της Google και όλες οι λειτουργίες για την δημιουργία νέας διαδρομής.
SplashScreenActivity	Η Δραστηριότητα αποτελείται από ένα Splash Screen που εμφανίζεται μόλις ο χρήστης εκτελέσει την εφαρμογή. Περιέχει το λογότυπο της εφαρμογής.

### Προσαρμογείς (Adapters)

Το πακέτο αυτό περιέχει τις κλάσεις που επεκτείνουν (extends) την κλάση BaseAdapter. Είναι Προσαρμογείς (Adapters) που βρίσκονται ενδιάμεσα από τις λίστες (listview) και τα δεδομένα τους. Είναι υπεύθυνοι και για την εμφάνιση των δεδομένων στις λίστες. Στον Πίνακα 39 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 39 κλάσεις που ανήκουν στο πακέτο Προσαρμογέων

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
FavoriteListAdapter	Η κλάση αυτή προσαρμόζει τα δεδομένα από τον πίνακα Αγαπημένων Διαδρομών, της βάσης δεδομένων και περιέχει μεθόδους για την εμφάνιση τους σε λίστα.
NavDrawerListAdapter	Η κλάση προσαρμόζει τα στοιχεία των κατηγοριών και τα παρουσιάζει στο πλάγιο μενού (slidemenu) [14]
RecentListAdapter	Η κλάση αυτή προσαρμόζει δεδομένα από τον πίνακα Διαδρομών, της Βάσης Δεδομένων και περιέχει μεθόδους για την εμφάνιση τους σε λίστα.
RouteTransitListAdapter	Η κλάση αυτή προσαρμόζει τα δεδομένα του πίνακα Διαδρομών και Μεταφορικών Μέσων, από τη Βάση Δεδομένων και περιέχει μεθόδους για την εμφάνιση τους σε λίστα.

### Υπηρεσίες (Services)

Στο πακέτο αυτό περιέχονται όλες οι κλάσεις που επεκτείνουν (extends) την κλάση Service. Είναι Υπηρεσίες (Services) παρασκηνίου, υπεύθυνες για τις ειδοποιήσεις στο χρήστη. Στον Πίνακα 40 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 40 κλάσεις που ανήκουν στο πακέτο Υπηρεσιών

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
NotificationService	Η κλάση αυτή περιέχει όλες τις λειτουργίες για τη δημιουργία μιας Υπηρεσίας (Service) παρασκηνίου και τη δημιουργία Ειδοποίησης (Notification) στο χρήστη.
CancelNotification	Η κλάση αυτή περιέχει όλες τις λειτουργίες για τον τερματισμό της Υπηρεσίας (Service) που έχει δημιουργηθεί και την ακύρωση των Ειδοποιήσεων προς το χρήστη.

### Βάσεις Δεδομένων (Databases)

Το πακέτο αυτό περιέχει τις κλάσεις που επεκτείνουν (extends) την κλάση SQLiteOpenHelper. Οι κλάσεις μέσα σε αυτό το πακέτο είναι υπεύθυνες για την δημιουργία βάσεων δεδομένων (SQLite) και τη διαχείριση των δεδομένων τους (SQL ερωτήματα). Στον Πίνακα 41 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 41 κλάσεις που ανήκουν στο πακέτο Βάσεων Δεδομένων

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
MySQLiteUtil	Σε αυτή τη κλάση περιέχονται όλες οι μέθοδοι δημιουργίας της Βάσης Δεδομένων Διαδρομών του χρήστη (SQLite) αλλά και οι μέθοδοι διαχείρισης των δεδομένων της (SQL ερωτήματα).
OasthSQLiteUtil	Η κλάση αυτή περιέχει όλες τις λειτουργίες για την αντιγραφή και σύνδεση της Βάσης Δεδομένων του ΟΑΣΘ. Επίσης περιέχει όλες τις απαραίτητες μεθόδους για τη διαχείριση των δεδομένων αυτών.

### Αντικείμενα (Objects)

Σε αυτό το πακέτο περιλαμβάνονται κλάσεις που δημιουργήθηκαν με σκοπό τη χρήση τους ως αντικείμενα. Στον Πίνακα 42 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 42 κλάσεις που ανήκουν στο πακέτο Αντικειμένων

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
Favorite	Κλάση για τη δημιουργία αντικειμένου Αγαπημένης Διαδρομής. Περιέχει όλες τις απαραίτητες μεθόδους (Get, Set) και δομητές.
Route	Κλάση για τη δημιουργία αντικειμένου Διαδρομής. Περιέχει όλες τις απαραίτητες μεθόδους (Get, Set) και δομητές.
Transit	Κλάση για τη δημιουργία αντικειμένου Μεταφορικού Μέσου. Περιέχει όλες τις απαραίτητες μεθόδους (Get, Set) και δομητές.

### Πλάγιο Μενού(Slidemenu)

Στο πακέτο αυτό περιέχονται όλες οι κλάσεις που επεκτείνουν (extends) την κλάση Fragment αλλά και αυτές που είναι υπεύθυνες για τη δημιουργία του πλάγιου μενού (slide menu). Στον Πίνακα 43 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 43 κλάσεις που ανήκουν στο πακέτο του πλάγιου μενού

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
DatePickerFragment	Η κλάση αυτή επεκτείνει (extends) τη κλάση Fragment. Περιέχει ένα Τεμάχιο επιλογής ημερομηνίας.
FavoriteFragment	Η κλάση αυτή επεκτείνει (extends) τη κλάση Fragment. Περιέχει μια λίστα που παρουσιάζει τις Αγαπημένες Διαδρομές, τις οποίες παρέχει ο αντίστοιχος Προσαρμογέας (Adaptor). Εμφανίζεται όταν ο χρήστης επιλέξει από το πλάγιο μενού την αντίστοιχη κατηγορία.
NavDrawerItem	Η κλάση αυτή περιέχει τις λειτουργίες που είναι υπεύθυνες για την εμφάνιση (τίτλοι, εικονίδια) του πλάγιου μενού (Slide Menu). [14]
RecentFragment	Η κλάση αυτή επεκτείνει (extends) τη κλάση Fragment. Περιέχει μια λίστα που παρουσιάζει τις Πρόσφατες Διαδρομές, τις οποίες παρέχει ο αντίστοιχος Προσαρμογέας (Adaptor). Εμφανίζεται όταν ο χρήστης επιλέξει από το πλάγιο μενού την αντίστοιχη κατηγορία.
RouteListViewFragment	Η κλάση αυτή επεκτείνει (extends) τη κλάση Fragment. Περιέχει μια λίστα που παρουσιάζει τις Διαδρομές και τα Μέσα Μεταφοράς, τις οποίες παρέχει ο αντίστοιχος Προσαρμογέας (Adaptor). Εμφανίζεται όταν ο χρήστης επιλέξει από το πλάγιο μενού την αντίστοιχη κατηγορία.

### Χρήσιμα (Utilities)

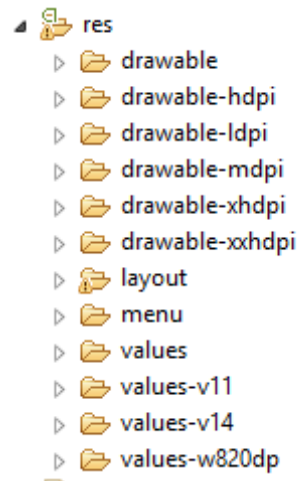
Το πακέτο αυτό περιέχει κλάσεις, οι οποίες υλοποιούν λειτουργίες που χρησιμοποιεί το πρόγραμμα καθ' όλη τη διάρκεια λειτουργίας του. Στον Πίνακα 44 παρουσιάζονται όλες οι κλάσεις που ανήκουν σε αυτό το πακέτο.

Πίνακας 44 κλάσεις που ανήκουν στο πακέτο χρήσιμα

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
GMapV2Directions	Η κλάση αυτή περιέχει όλες τις μεθόδους για την αποστολή αιτήσεων (requests) στο Google APIs και μεθόδους για τη διαχείριση των αποτελεσμάτων (xml) που λαμβάνει.
MyLocationListener	Η κλάση αυτή εφαρμόζει τη διεπαφή (interface) LocationListener. Παρέχει μεθόδους για την λήψη των συντεταγμένων του χρήστη, όσο αυτός μετακινείται.
OasthWS	Σε αυτή τη κλάση, περιέχονται οι μέθοδοι που επικοινωνούν με την ιστοσελίδα του ΟΑΣΘ.

### Πόροι της εφαρμογής

Η εφαρμογή στηρίζεται επίσης σε πόρους (layout, menu, values, drawables), οι οποίοι είναι υπεύθυνοι για τη σχεδίαση των περιβαλλόντων διεπαφής χρήστη. Η δομή φαίνεται στο Σχήμα 26.



Σχήμα 26 πόροι εφαρμογής

Το περιεχόμενο της δομής πόρων, περιγράφεται στον Πίνακα 45.

Πίνακας 45 πόροι εφαρμογής

ΚΛΑΣΗ	ΠΕΡΙΓΡΑΦΗ
drawable	Σε αυτό το φάκελο υπάρχουν xml αρχεία σχετικά με τη σχεδίαση του πλάγιου μενού (slide menu).
drawable-hdpi	Σε αυτό το φάκελο υπάρχουν οι εικόνες που χρησιμοποιεί η εφαρμογή. Αναφέρεται σε συσκευές με μεγάλη ανάλυση οθόνης (~240dpi).
drawable-ldpi	Σε αυτό το φάκελο υπάρχουν οι εικόνες που χρησιμοποιεί η εφαρμογή. Αναφέρεται σε συσκευές με μικρή ανάλυση οθόνης (~120dpi).
drawable-mdpi	Σε αυτό το φάκελο υπάρχουν οι εικόνες που χρησιμοποιεί η εφαρμογή. Αναφέρεται σε συσκευές με μεσαία ανάλυση οθόνης (~160dpi).
drawable-xhdpi	Σε αυτό το φάκελο υπάρχουν οι εικόνες που χρησιμοποιεί η εφαρμογή. Αναφέρεται σε συσκευές με έξτρα μεγάλη ανάλυση οθόνης (~320dpi).
drawable-xxhdpi	Σε αυτό το φάκελο υπάρχουν οι εικόνες που χρησιμοποιεί η εφαρμογή. Αναφέρεται σε συσκευές με έξτρα - έξτρα μεγάλη ανάλυση οθόνης (~640dpi).
layout	Σε αυτό το φάκελο υπάρχουν τα xml αρχεία που είναι υπεύθυνα για το σχεδιασμό και την εμφάνιση των συστατικών της εφαρμογής (Δραστηριότητες, λίστες, Τεμάχια).
menu	Σε αυτό το φάκελο τοποθετούνται τα xml αρχεία που αφορούν τη σχεδίαση των μενού σε κάθε Δραστηριότητα (Activity) ή Τεμάχιο (Fragment).
values	Σε αυτό το φάκελο μπαίνουν αρχεία xml στα οποία δηλώνονται κάποιες μεταβλητές (ονόματα, χρώματα, θέματα, διαστάσεις). Αυτά μπορούν να χρησιμοποιηθούν από άλλα αρχεία xml των παραπάνω κατηγοριών (layout, menu).



### *AndroidManifest.xml*

Όπως έχουμε αναφέρει κάθε Android εφαρμογή είναι απαραίτητο να έχει ένα αρχείο Manifest. Στο Σχήμα 27 παρουσιάζεται το αρχείο AndroidManifest.xml της εφαρμογής.

Στο αρχείο αυτό βλέπουμε τις άδειες που ζητάει η εφαρμογή από τον χρήστη όταν εγκατασταθεί και την ελάχιστη έκδοση λειτουργικού που χρειάζεται για να λειτουργήσει. Επίσης φαίνονται αναλυτικά όλες οι Δραστηριότητες (Activities) που έχει η εφαρμογή και επιλέγεται η ιεραρχία του (πρώτη, σχέσεις μεταξύ δραστηριοτήτων). Δηλώνονται ακόμα, οι Υπηρεσίες (Services) και το κλειδί των χαρτών της Google.

## Πτυχιική εργασία του φοιτητή Λαμπρόπουλου Αλέξανδρου

```
planmyday-Ptyxiaki Manifest
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.planmyday_ptyxiaki"
4     android:versionCode="1"
5     android:versionName="1.0" >
6
7     <uses-sdk
8         android:minSdkVersion="19"
9         android:targetSdkVersion="19" />
10
11     <uses-feature
12         android:glEsVersion="0x00020000"
13         android:required="true" />
14
15     <permission
16         android:name="om.example.planmyday_ptyxiaki.permission.MAPS_RECEIVE"
17         android:protectionLevel="signature" />
18
19     <uses-permission android:name="com.example.planmyday_ptyxiaki.permission.MAPS_RECEIVE" />
20     <uses-permission android:name="android.permission.INTERNET" />
21     <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
22     <uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES" />
23     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
24     <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
25     <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
26     <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
27
28     <application
29         android:allowBackup="true"
30         android:icon="@drawable/ic_launcher"
31         android:label="@string/app_name"
32         android:theme="@style/CustomActionBarTheme" android:logo="@drawable/bus_blue_icon">
33         <activity
34             android:name="planmyday.ptyxiaki.activities.NewRouteActivity"
35             android:configChanges="orientation|screenSize"
36             android:label="@string/title_activity_map_pane"
37             android:parentActivityName="planmyday.ptyxiaki.activities.RouteListViewActivity" >
38         </activity>
39
40         <meta-data
41             android:name="com.google.android.gms.version"
42             android:value="@integer/google_play_services_version" />
43         <meta-data
44             android:name="com.google.android.maps.v2.API_KEY"
45             android:value="AIzaSyBxJYTVAsc-uaxMBU6i-sSitc1qbm8TvnU" />
46
47         <activity
48             android:name="planmyday.ptyxiaki.activities.RouteListViewActivity"
49             android:configChanges="orientation|screenSize"
50             android:label="@string/title_activity_route_list_view"
51             android:parentActivityName="planmyday.ptyxiaki.activities.LoginActivity" >
52         </activity>
53         <activity
54             android:name="planmyday.ptyxiaki.activities.TransitListViewActivity"
55             android:configChanges="orientation|screenSize"
56             android:label="@string/title_activity_transit_list_view"
57             android:parentActivityName="planmyday.ptyxiaki.activities.RouteListViewActivity" >
58             <meta-data
59                 android:name="android.support.PARENT_ACTIVITY"
60                 android:value="planmyday.ptyxiaki.activities.RouteListViewActivity" />
61         </activity>
62         <activity
63             android:name="getOasthData.OasthDataActivity"
64             android:configChanges="orientation|screenSize"
65             android:label="@string/title_activity_oasth_data" >
66         </activity>
67
68         <service
69             android:name="planmyday.ptyxiaki.bgService.NotificationService"
70             android:enabled="true" />
71         <service
72             android:name="planmyday.ptyxiaki.bgService.CancelAlarmService"
73             android:enabled="true" />
74
75         <activity
76             android:name="planmyday.ptyxiaki.activities.MainActivity"
77             android:configChanges="orientation|screenSize"
78             android:label="@string/title_activity_main" >
79         </activity>
80         <activity
81             android:name="planmyday.ptyxiaki.activities.SplashScreenActivity"
82             android:label="@string/app_name"
83             android:screenOrientation="portrait"
84             android:theme="@android:style/Theme.Translucent" >
85             <intent-filter>
86                 <action android:name="android.intent.action.MAIN" />
87
88                 <category android:name="android.intent.category.LAUNCHER" />
89             </intent-filter>
90         </activity>
91     </application>
92 </manifest>
```

Σχήμα 27 αρχείο AndroidManifest.xml

## 7.15 Εξαγωγή εφαρμογής (Eclipse IDE)

Για να την εξαγωγή της εφαρμογής Android μέσω του Eclipse, πρέπει να ακολουθηθούν τα παρακάτω βήματα:

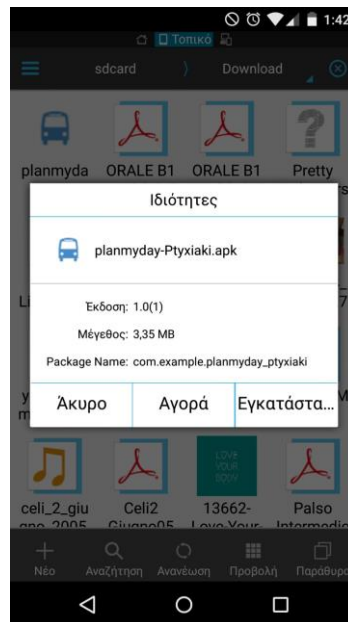
1. άνοιγμα της καρτέλας “File”
2. επιλογή “Export”
3. από το εμφανιζόμενο παράθυρο Android και Export Android Application
4. επιλογή βασικού πρότζεκτ
5. δημιουργία ή χρήση υπάρχοντος “keystore”
6. εισαγωγή στοιχείων και εξαγωγή

## 7.16 Εγκατάσταση της Εφαρμογής

Η εφαρμογή είναι σχεδιασμένη για συσκευές με λειτουργικό Android έκδοσης KIT KAT 4.4.2 και πάνω. Η εφαρμογή μπορεί να εγκατασταθεί μέσω του Eclipse, μόλις τρέξει στη συσκευή που θέλουμε. Ακόμα μπορεί να εγκατασταθεί μέσω του αρχείου APK που δημιουργήσαμε προηγουμένως.

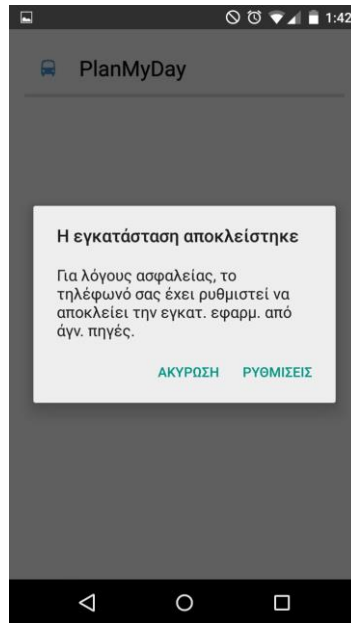
Ο χρήστης επιλέγει το αρχείο APK και μπορεί να δει τις άδειες πρόσβασης που ζητάει η συγκεκριμένη εφαρμογή, έπειτα επιλέγει να την εγκαταστήσει ή όχι. Παρακάτω παρουσιάζεται η διαδικασία εγκατάστασης τους αρχείου APK της εφαρμογής.

Ο χρήστης ανοίγει το αρχείο APK για εγκατάσταση της εφαρμογής. Σχήμα 28



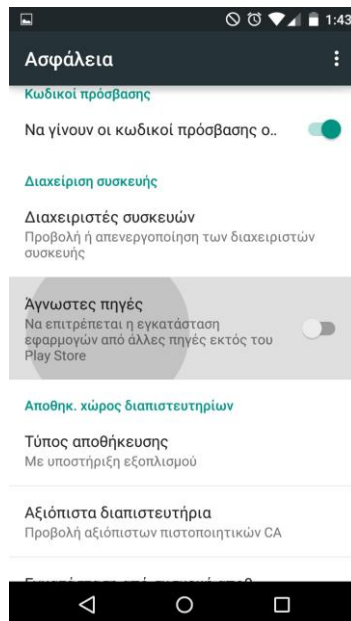
Σχήμα 28 άνοιγμα αρχείου APK

Στο Σχήμα 29, εμφανίζεται ένα μήνυμα σε περίπτωση που ο χρήστης δεν έχει ενεργοποιημένες τις άγνωστες πηγές για εγκατάσταση εφαρμογών. Αν η εφαρμογή δεν είναι κατεβασμένη από το Google Play Store τότε πρέπει να ενεργοποιηθεί η ρύθμιση αυτή για να γίνει επιτυχής η εγκατάσταση.



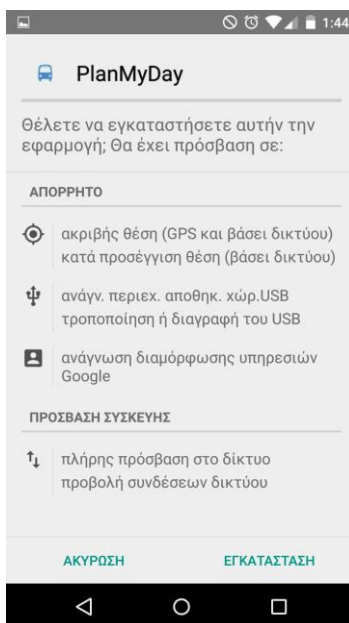
Σχήμα 29 αποκλεισμός εγκατάστασης λόγω άγνωστων πηγών

Μόλις ο χρήστης επιλέξει ΡΥΘΜΙΣΕΙΣ, θα βρεθεί στις ρυθμίσεις για την ενεργοποίηση των αγνώστων πηγών Σχήμα 30.



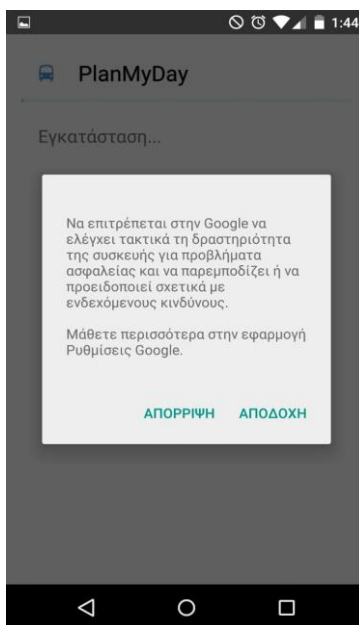
Σχήμα 30 ενεργοποίηση αγνώστων πηγών

Έπειτα ο χρήστης μπορεί να συνεχίσει κανονικά την εγκατάσταση της εφαρμογής και να δει τις άδειες που χρειάζεται για να λειτουργήσει όπως φαίνεται στο Σχήμα 31.



Σχήμα 31 άδειες εφαρμογής

Αν συμφωνήσει να παρέχει αυτές τις άδειες στην εφαρμογή τότε η εγκατάσταση ολοκληρώνεται. Τέλος, λόγω του Google Maps που χρησιμοποιείται ο χρήστης πρέπει να επιλέξει αποδοχή ελέγχου όπως φαίνεται στο Σχήμα 32.



Σχήμα 32 αποδοχή ελέγχου Google

## ΣΥΜΠΕΡΑΣΜΑΤΑ

Ο κόσμος της πληροφορικής συνεχώς εξελίσσεται. Με την ολοκλήρωση της Πτυχιακής Εργασίας, έχω κατανοήσει τα βασικά στοιχεία του λειτουργικού Android και το πώς μπορεί να δημιουργηθεί μια εφαρμογή.

Η εφαρμογή ημερήσιας διαχείρισης διαδρομών “PlanMyDay” καθώς και όλες οι απαραίτητες λειτουργίες της είναι επιτυχείς μετά από πολλές ώρες testing. Σε αυτές περιλαμβάνονται: ο καθημερινός προγραμματισμός των διαδρομών, η εβδομαδιαία επανάληψη, η μηνιαία επανάληψη και η ειδοποίηση των χρηστών.

Η σχεδίαση από το μηδέν μίας εφαρμογής σε Android λειτουργικό με τη χρήση UML διαγραμμάτων ήταν μία πρόκληση για εμένα. Η εβδομαδιαία μου επικοινωνία με τους χρήστες με βοήθησε να εξάγω τις σωστές απαιτήσεις και περιπτώσεις χρήσης για την υλοποίηση της εφαρμογής.

Όπως είναι αναμενόμενο προέκυψαν αρκετά προβλήματα κατά τη διάρκεια υλοποίησης του project. Μερικές φορές ένα μικρό λάθος με έκανε να ξοδέψω αρκετό χρόνο για να το βρω διορθώσω. Κάποια άλλα προβλήματα με έκαναν να σκεφτώ νέους τρόπους και τεχνάσματα για την λύση τους.

Θα ήθελα στο μέλλον να εφαρμόσω στην εφαρμογή “PlanMyDay” λειτουργικότητα και σχεδιασμό (material design) από τη νέα έκδοση του Android (Lollipop 5.0), το οποίο θα δώσει στο χρήστη ένα περιβάλλον πιο εύχρηστο και ευχάριστο. Για την επαγγελματική της ολοκλήρωση θα ήταν καλή η χρήση ενός server για την αποθήκευση των βάσεων δεδομένων αλλά και για την υλοποίηση περίπλοκων λειτουργιών σε αυτόν. Τέλος θα ήθελα να επεκτείνω την εφαρμογή για χρήση και στην υπόλοιπη Ελλάδα.

Το Τμήμα Μηχανικών Πληροφορικής, παρέχει καλή γνώση στους φοιτητές του ως προς τον προγραμματισμό και το σχεδιασμό εφαρμογών. Πέρα όμως από τη βασική γνώση σε πολλούς τομείς της πληροφορικής, ο φοιτητής πρέπει να αποκτήσει περισσότερη επαφή με δική του πρωτοβουλία, να μπει σε project, να δουλέψει πάνω σε νέα αντικείμενα για να αναπτύξει τη γνώση που του έχει δώσει το τμήμα.

Η Πτυχιακή Εργασία και η Πρακτική άσκηση δίνουν τη δυνατότητα στους φοιτητές να δουλέψουν σε πραγματικά project και τους προετοιμάζει για την αγορά εργασίας.

## ΑΝΑΦΟΡΕΣ

- (1) Android Developers, Activities, <http://developer.android.com/guide/components/activities.html>
- (2) Android Developers, Fragments, <http://developer.android.com/guide/components/fragments.html>
- (3) Android Developers, Notification, <http://developer.android.com/guide/topics/ui/notifiers/notifications.html>
- (4) Android Developers, Platform Versions, <https://developer.android.com/about/dashboards/index.html>
- (5) Android Developers, Service, <http://developer.android.com/reference/android/app/Service.html>
- (6) Android Developers, system architecture, <http://developer.android.com/images/system-architecture.jpg>
- (7) Atlassian Bitbucket, <https://bitbucket.org/>
- (8) IDC Analyze the Future, Smartphone OS Market Share, Q3 2014, <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- (9) Simon Khalaf, 1 Απριλίου 2014, Apps Solidify Leadership Six Years into the Mobile Revolution, <http://www.flurry.com/bid/109749/Apps-Solidify-Leadership-Six-Years-into-the-Mobile-Revolution>
- (10) Οργανισμός Αστικών Συγκοινωνιών Θεσσαλονίκης, ιστοσελίδα για κινητά, <http://m.oasth.gr>
- (11) Οργανισμός Αστικών Συγκοινωνιών Θεσσαλονίκης, κεντρική ιστοσελίδα, <http://www.oasth.gr>

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Android Developers, Activities, <http://developer.android.com/guide/components/activities.html>
- [2] Android Developers, Adapter, <http://developer.android.com/reference/android/widget/Adapter.html>
- [3] Android Developers, Android Interfaces, <https://source.android.com/devices/index.html>
- [4] Android Developers, Android, the world's most popular mobile platform, <http://developer.android.com/about/index.html>
- [5] Android Developers, App Manifest, <http://developer.android.com/guide/topics/manifest/manifest-intro.html>
- [6] Android Developers, Application Security, <https://source.android.com/devices/tech/security/overview/app-security.html>
- [7] Android Developers, Art and Dalvik, <http://source.android.com/devices/tech/dalvik/>
- [8] Android Developers, Drawable Resources, <http://developer.android.com/guide/topics/resources/drawable-resource.html>
- [9] Android Developers, Fragments, <http://developer.android.com/guide/components/fragments.html>
- [10] Android Developers, Geocoder, <http://developer.android.com/reference/android/location/Geocoder.html>

- [11]Android Developers, Location APIs, <https://developer.android.com/google/play-services/location.html>
- [12]Android Developers, Marker, <http://developer.android.com/reference/com/google/android/gms/maps/model/Marker.html>
- [13]Android Developers, Menus, <http://developer.android.com/guide/topics/ui/menus.html>
- [14]Android Developers, Navigation Drawer, <https://developer.android.com/design/patterns/navigation-drawer.html>
- [15]Android Developers, Notifications, <http://developer.android.com/guide/topics/ui/notifiers/notifications.html>
- [16]Android Developers, Processes and Threads, <http://developer.android.com/guide/components/processes-and-threads.html>
- [17]Android Developers, Scheduling Repeating Alarms, <https://developer.android.com/training/scheduling/alarms.html>
- [18]Android Developers, Security, <https://source.android.com/devices/tech/security/>
- [19]Android Developers, Service, <http://developer.android.com/reference/android/app/Service.html>
- [20]Android Developers, Setting Up Google Play Services, <http://developer.android.com/google/play-services/setup.html>
- [21]Android Developers, SQLite, <http://developer.android.com/reference/android/database/sqlite/package-summary.html>
- [22]Android Developers, Support Library Features, <https://developer.android.com/tools/support-library/features.html>
- [23]Android, The Android Story, <http://www.android.com/history/>
- [24]Class diagram, <http://www.visual-paradigm.com/VPGallery/diagrams/Class.html>
- [25]Component diagram <http://www.visual-paradigm.com/VPGallery/diagrams/Component.html>
- [26]Deployment diagram, <http://www.visual-paradigm.com/VPGallery/diagrams/Deployment.html>
- [27]Google Developers, Google Maps Android API v2, [https://developers.google.com/maps/documentation/android/start?hl=EL#install\\_and\\_configure\\_the\\_google\\_play\\_services\\_sdk](https://developers.google.com/maps/documentation/android/start?hl=EL#install_and_configure_the_google_play_services_sdk)
- [28]Google Developers, Google Maps API Web Services, <https://developers.google.com/maps/documentation/webservices/>
- [29]Google Developers, The Google Directions API, <https://developers.google.com/maps/documentation/directions/>
- [30]Package diagram, <http://www.visual-paradigm.com/VPGallery/diagrams/Package.html>
- [31]Sequence diagram, <http://www.visual-paradigm.com/VPGallery/diagrams/Sequence.html>
- [32]Simon Khalaf, 1 Απριλίου 2014, Apps Solidify Leadership Six Years into the Mobile Revolution, <http://www.flurry.com/bid/109749/Apps-Solidify-Leadership-Six-Years-into-the-Mobile-Revolution>
- [33]Sommerville, Ian (2009), Βασικές Αρχές Τεχνολογίας Λογισμικού, μτφρ: Τσιλογιάννης, Δ., επιμελ: Άλβας, Τ. και Αρκουδέας, Π. και Καρτσακλής, Δ., εκδ.: Κλειδάριθμος, Αθήνα, Ελλάδα.



[34]Use case diagram, <http://www.visual-paradigm.com/VPGallery/diagrams/UseCase.html>

[35]Wikipedia, Web sevice, [http://en.wikipedia.org/wiki/Web\\_service](http://en.wikipedia.org/wiki/Web_service)

[36]Γιακουμάκης, Μ. και Διαμαντίδης, Ν. (2009), Τεχνολογία Λογισμικού, εκδ.: Σταμούλη Α.Ε., Αθήνα, Ελλάδα.

[37]Χατζηγεωργίου Αλέξανδρος, Ν. (2005), Αντικειμενοστρεφής σχεδίαση UML, αρχές, πρότυπα και ευρετικοί κανόνες, εκδ.: Κλειδάριθμος, Αθήνα, Ελλάδα.

ΠΑΡΑΡΤΗΜΑΤΑ



ΟΡΓΑΝΙΣΜΟΣ ΑΣΤΙΚΩΝ ΣΥΓΚΟΙΝΩΝΙΩΝ ΘΕΣΣΑΛΟΝΙΚΗΣ

Π ρ ο ς  
Τον κ. Αλέξανδρο Λαμπρόπουλο  
Παριβάλλη 30  
Τ.Κ. 546 42 - Θεσσαλονίκη

ΑΡΙΘΜ. ΠΡΩΤ.: 9681  
ΠΛΗΡΟΦΟΡΙΕΣ: Ζ. Ζαχαρνά  
ΤΗΛΕΦΩΝΟ: 2310 981191  
ΘΕΣΣΑΛΟΝΙΚΗ: 17-10-2014

**Θ έ μ α :** «Απάντηση σε αίτηση»  
**Σ χ ε τ :** Η με αριθμ. γεν. πρωτ. 9130/7-10-2014 αίτησή σας.

Αγαπητέ κ. Λαμπρόπουλε,

Η δημιουργία και λειτουργία της ηλεκτρονικής ιστοσελίδας του Οργανισμού μας, δεν αποτελεί παρά ένα ακόμη μέσο για την πλήρη, συνεχή και υπεύθυνη ενημέρωση του επιβατικού κοινού.

Μπορείτε κάλλιστα να αντλήσετε όλα τα απαραίτητα στοιχεία που σας χρειάζονται για την εργασία σας, με μόνη προϋπόθεση ότι ο σκοπός θα είναι να χρησιμοποιηθούν αποκλειστικά και μόνο για να σας βοηθήσουν να φέρετε σε πέρας το συγκεκριμένο πόνημά σας.

Καλή επιτυχία στη προσπάθειά σας.

Με εκτίμηση



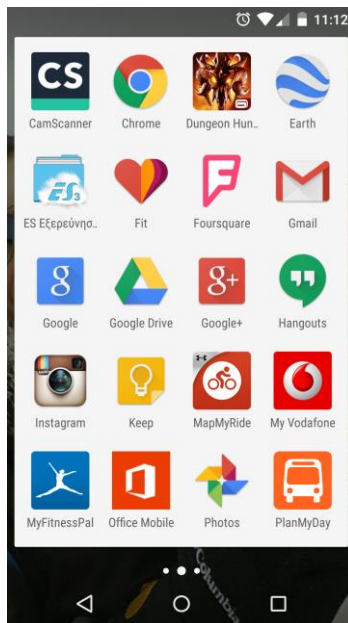
ΧΡΗΣΤΟΣ ΑΓΓ. ΣΤΕΦΑΝΙΔΗΣ  
Πρόεδρος Δ.Σ. Ο.Α.Σ.Θ.

Παράρτημα 1 άδεια χρήσης δεδομένων ΟΑΣΘ

## ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

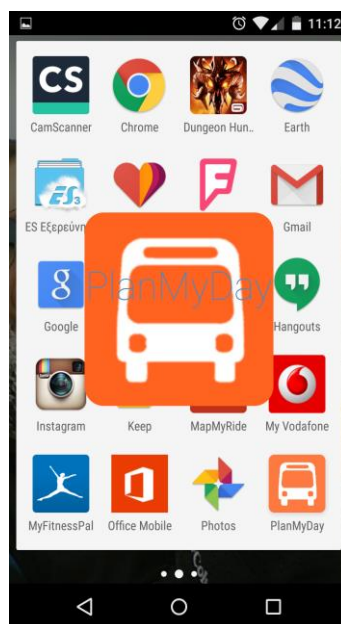
Σε αυτό το σημείο της Πτυχιακής Εργασίας, παρουσιάζεται ένας οδηγός χρήσης της εφαρμογής “PlanMyDay”, για ένα χρήστη. Περιγράφονται όλες οι βασικές χρήσεις της εφαρμογής.

Ο χρήστης μετά την εγκατάσταση της εφαρμογής που περιεγράφηκε σε προηγούμενο κεφάλαιο, είναι έτοιμος να ξεκινήσει την εφαρμογή “PlanMyDay”. Μπορεί να τη βρει στο κεντρικό μενού της Android συσκευής του, Εικόνα 1.



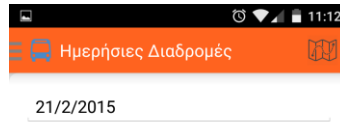
Εικόνα 1 άνοιγμα εφαρμογής

Εμφανίζεται ένα “splash screen” (Εικόνα 2), με το λογότυπο της εφαρμογής, για κάποια δευτερόλεπτα.



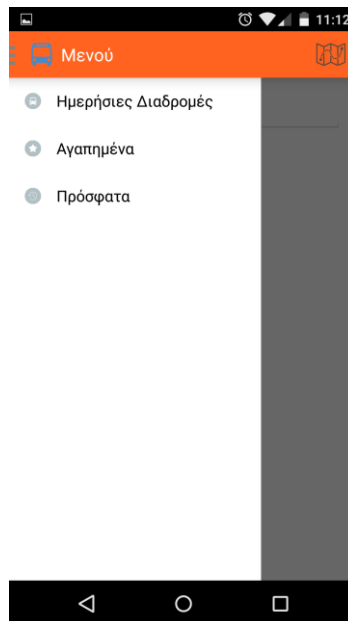
Εικόνα 2 Splash Screen

Έπειτα ο χρήστης μεταφέρεται στην εφαρμογή, στην οποία εμφανίζεται η “Διαχείριση Ημερήσιων Διαδρομών”, Εικόνα 3.



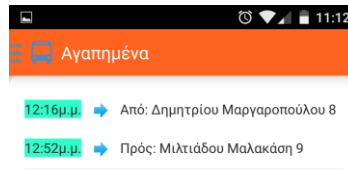
Εικόνα 3 Διαχείριση Ημερήσιων Διαδρομών

Ο χρήστης σέρνοντας (slide) το δάχτυλο του, προς τα δεξιά εμφανίζει το κεντρικό μενού, στο οποίο υπάρχουν κατηγορίες που μπορεί να επιλέξει (“Διαχείριση Ημερήσιων Διαδρομών”, “Αγαπημένα”, “Πρόσφατα”), Εικόνα 4.

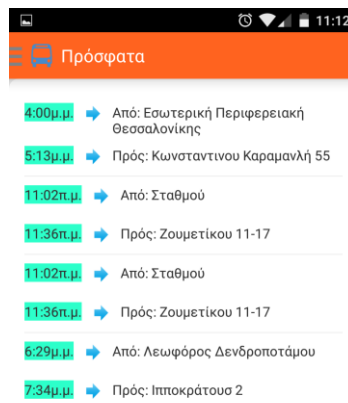


Εικόνα 4 Πλάγιο Μενού

Ο χρήστης μπορεί να επιλέξει από τις εμφανιζόμενες κατηγορίες όποια επιθυμεί. Στην Εικόνα 5, έχει επιλέξει “Αγαπημένα” ενώ στην Εικόνα 6, έχει επιλέξει “Πρόσφατα”.

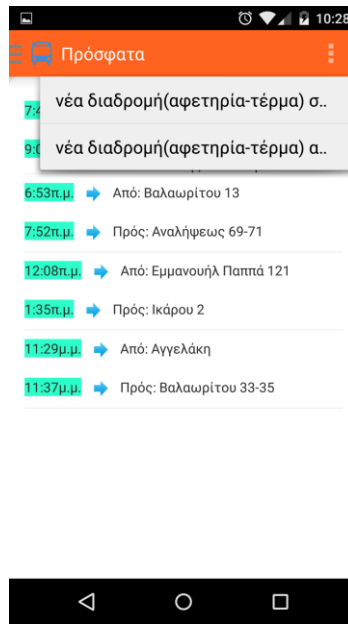


Εικόνα 5 Αγαπημένα

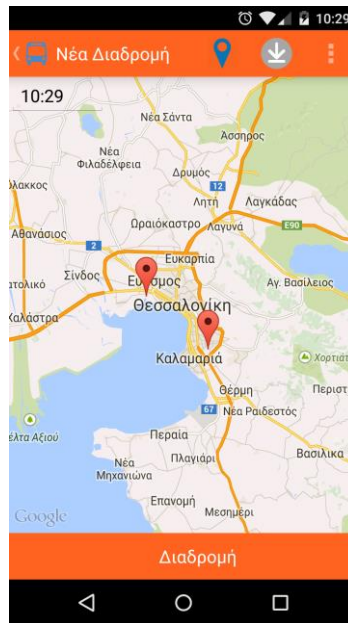


Εικόνα 6 Πρόσφατα

Ο Χρήστης επιλέγει από τη κατηγορία “Πρόσφατα” μια διαδρομή και μπορεί να την προσθέσει (σήμερα ή αύριο), χρησιμοποιώντας τα ίδια σημεία εκκίνησης – τερματισμού σε μια νέα διαδρομή (Εικόνα 7, 8).



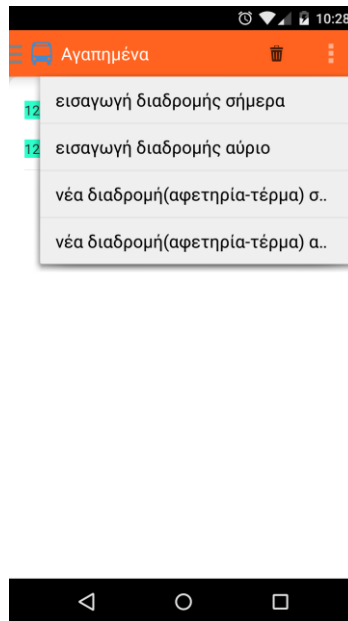
Εικόνα 7 διαχείριση πρόσφατης επιλεγμένης διαδρομής



Εικόνα 8 χρήση σημείων (αφετηρία - τέρμα), για δημιουργία νέας διαδρομής

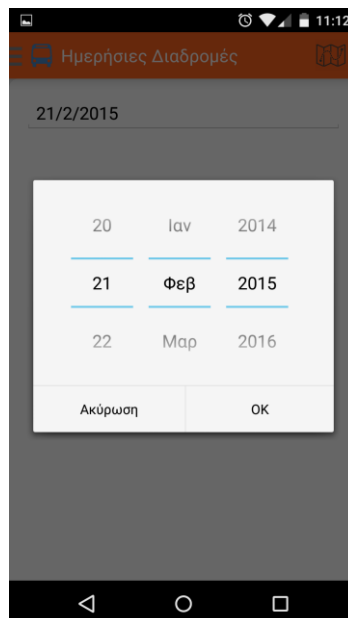
Ο χρήστης από τη κατηγορίας “Αγαπημένα” έχει τη δυνατότητα επιλέγοντας μια διαδρομή (Εικόνα 9), να:

- Την αφαιρέσει από τα αγαπημένα
- Να την προσθέσει αυτούσια (σήμερα ή αύριο)
- Να χρησιμοποιήσει τα σημεία της (αφετηρία - τέρμα), σε μια νέα διαδρομή σήμερα ή αύριο

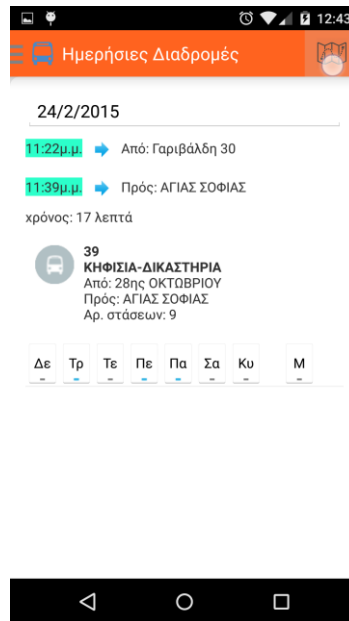


Εικόνα 9 διαχείριση επιλεγμένης αγαπημένης διαδρομής

Ο χρήστης από την κατηγορία “Διαχείριση Ημερησίων Διαδρομών” μπορεί να επιλέξει ημέρα (Εικόνα 10) και έπειτα το εικονίδιο “Νέα Διαδρομή”, Εικόνα 11.



Εικόνα 10 επιλογή ημέρας

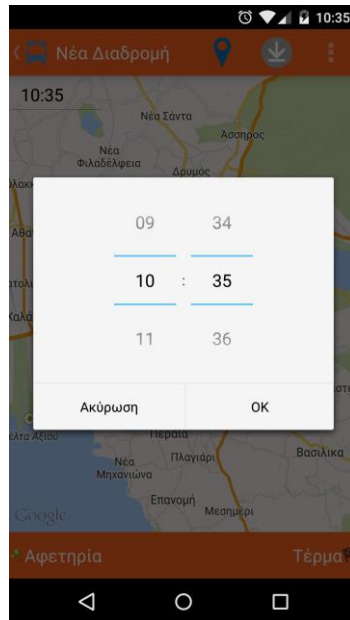


Εικόνα 11 επιλογή νέας διαδρομής

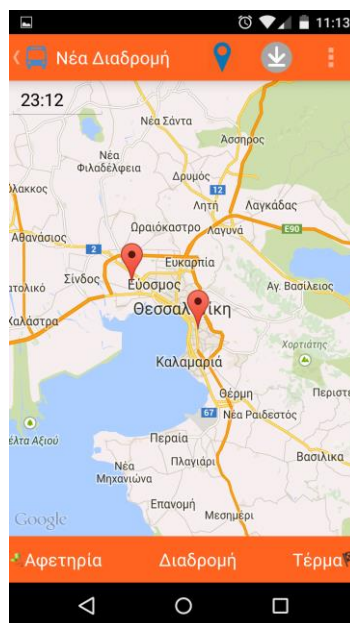
Για να δημιουργήσει μια νέα διαδρομή ο χρήστης πρέπει πρώτα να επιλέξει ώρα άφιξης (Εικόνα 12) και έπειτα έχει την επιλογή να:

- Χρησιμοποιήσει σημεία πάνω στον χάρτη (πάτημα του αντίστοιχου κουμπιού πρώτα και έπειτα παρατεταμένο πάτημα στον χάρτη), Εικόνα 13
- Χρησιμοποιήσει διευθύνσεις, Εικόνα 14
- Χρησιμοποιήσει τη τοποθεσία του για σημείο έναρξης, Εικόνα 15

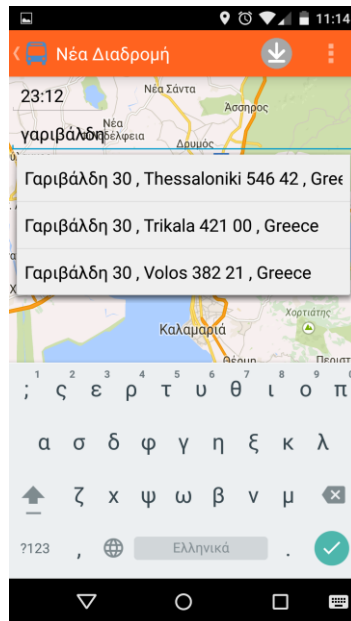




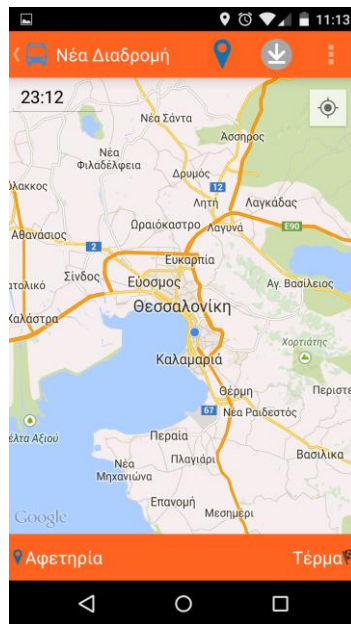
Εικόνα 12 επιλογή ώρας άφιξης



Εικόνα 13 χρήση σημείων στο χάρτη

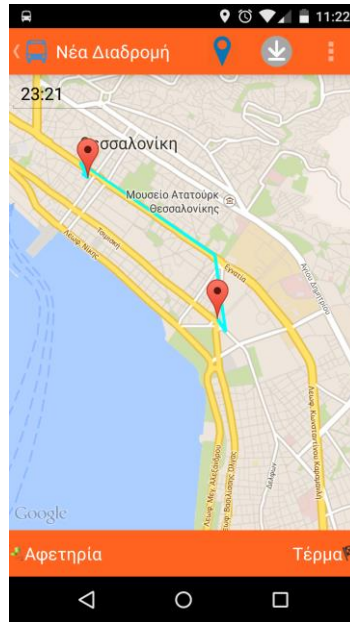


Εικόνα 14 χρήση διευθύνσεων



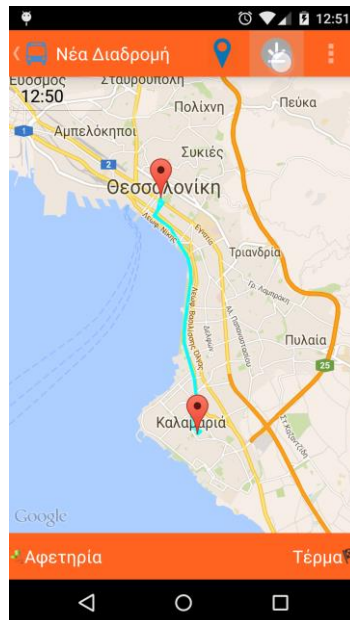
Εικόνα 15 χρήση τοποθεσίας

Έπειτα από την επιλογή αφετηρίας και τέρμα, ο χρήστης επιλέγει "Διαδρομή", για να σχεδιαστεί η αντίστοιχη διαδρομή στο χάρτη, Εικόνα 16.



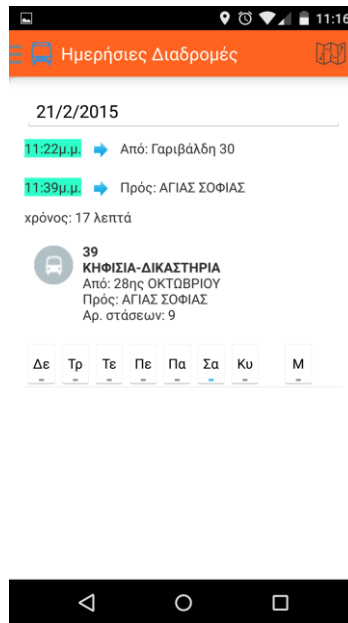
Εικόνα 16 σχεδίαση διαδρομής στο χάρτη

Τέλος ο χρήστης πατάει το εικονίδιο “Αποθήκευση” και η διαδρομή του αποθηκεύεται, Εικόνα 17.



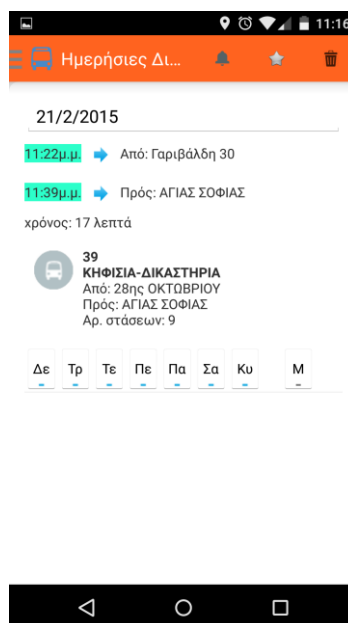
Εικόνα 17 αποθήκευση διαδρομής

Τώρα στην “Διαχείριση Ημερησίων Διαδρομών”, ο χρήστης βλέπει τις προγραμματισμένες του διαδρομές για την επιλεγμένη ημέρα (ώρα αναχώρησης, ώρα άφιξης, στάσεις και διευθύνσεις άφιξης και αναχώρησης), Εικόνα 18.

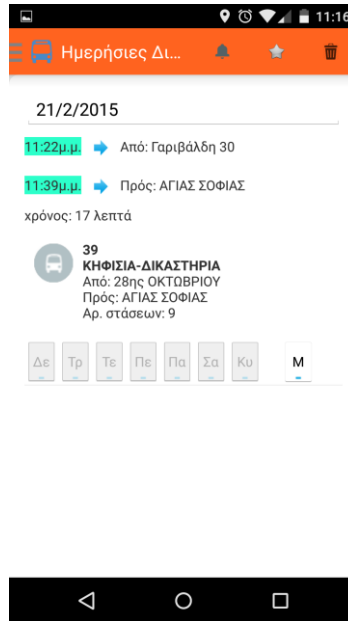


Εικόνα 18 αποθηκευμένη διαδρομή

Ο χρήστης μπορεί να επιλέξει εβδομαδιαία επανάληψη (Εικόνα 19) ή/και μηνιαία επανάληψη (Εικόνα 20) για κάθε διαδρομή ξεχωριστά. Μόλις ο χρήστης ενεργοποιήσει την μηνιαία επανάληψη τότε ο συνδυασμός των επιλεγμένων ημερών (εβδομαδιαία επανάληψη), θα επαναληφθεί σε όλο το μήνα.



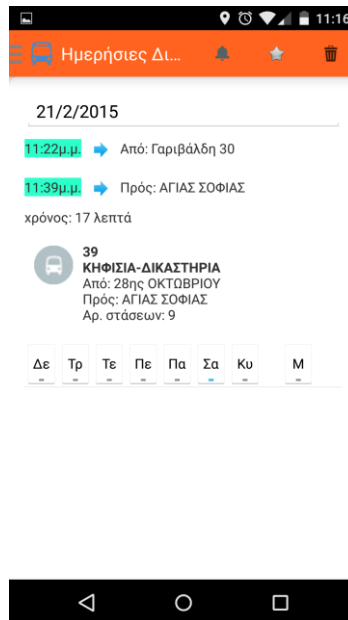
Εικόνα 19 εβδομαδιαία επανάληψη



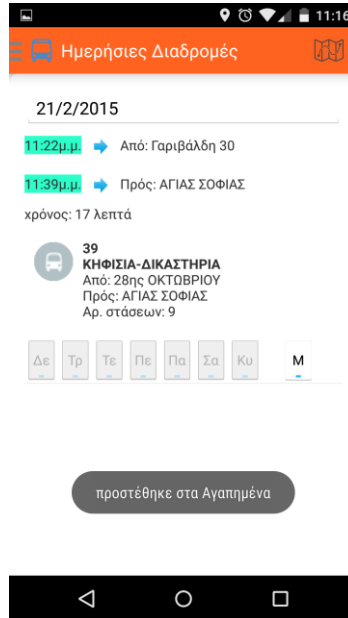
Εικόνα 20 μηνιαία επανάληψη

Επιλέγοντας μια διαδρομή (παρατεταμένο πάτημα στη διαδρομή από τη λίστα), εμφανίζεται ένα μενού (Εικόνα 21) από το οποίο μπορεί να:

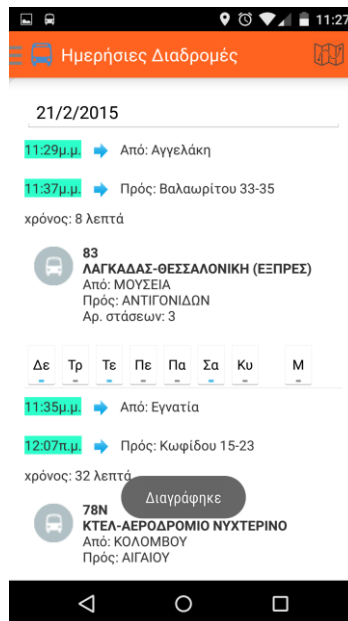
- Προσθέσει στα αγαπημένα την διαδρομή (Εικόνα 22),
- Διαγράψει την διαδρομή (Εικόνα 23),
- Ενεργοποιήσει την ειδοποίηση για την διαδρομή (Εικόνα 24)



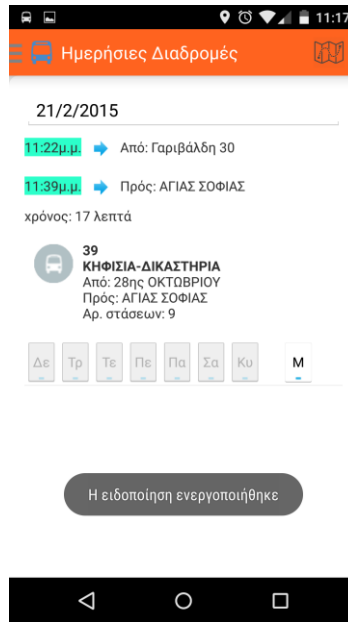
Εικόνα 21 μενού διαχείρισης επιλεγμένης διαδρομής



Εικόνα 22 προσθήκη διαδρομής στα αγαπημένα



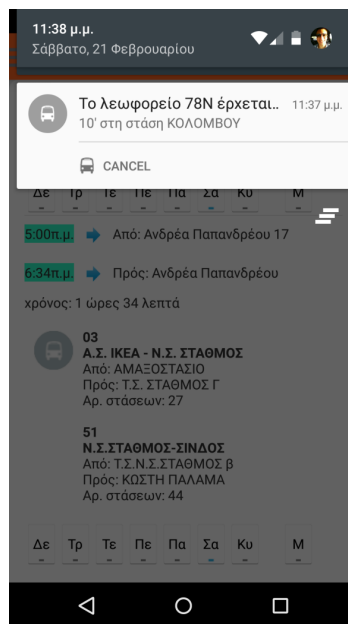
Εικόνα 23 διαγραφή επιλεγμένης διαδρομής



Εικόνα 24 ενεργοποίηση ειδοποίησης επιλεγμένης διαδρομής

Η ειδοποίηση λειτουργεί για μια διαδρομή κάθε φορά. Σκοπό έχει την ενημέρωση του χρήστη για την ώρα άφιξης του λεωφορείου στην στάση (της διαδρομής που επιλέχθηκε). Αν η ώρα αναχώρησης έχει ξεπεράσει την τρέχον ώρα, τότε η ειδοποίηση θα αρχίσει άμεσα, ειδώλλως θα ξεκινήσει δέκα λεπτά πριν την ώρα αναχώρησης. Η ειδοποίηση ανανεώνεται κάθε ένα λεπτό.

Η ειδοποίηση λειτουργεί ακόμα και αν ο χρήστης κλείσει την εφαρμογή. Ακυρώνεται μόνο αν πατήσει το κουμπί "Ακύρωση" (Εικόνα 25) από την ειδοποίηση στο συρτάρι ειδοποιήσεων ή αν ο χρήστης επιλέξει ειδοποίηση για μια άλλη διαδρομή. Αν ο χρήστης πατήσει πάνω στην ειδοποίηση, από το συρτάρι ειδοποιήσεων, τότε ανοίγει η εφαρμογή.



Εικόνα 25 η ειδοποίηση στο συρτάρι ειδοποιήσεων