



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ**  
**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**  
**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Ανθρωποκεντρικός και προσαρμοστικός σχεδιασμός για την  
ανάπτυξη εφαρμογής για συγκοινωνίες με χρήση μηχανικής μάθησης**

**Μιχαήλ Αποστολίδης**  
**A.M. 711171059**

**Εισηγητής: Χρήστος Τρούσσας, Επ. Καθηγητής**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Ανθρωποκεντρικός και προσαρμοστικός σχεδιασμός για την ανάπτυξη εφαρμογής για συγκοινωνίες με χρήση μηχανικής μάθησης**

**Μιχαήλ Αποστολίδης  
Α.Μ. 711171059**

**Εξεταστική Επιτροπή:**

**Χρήστος Τρούσσας, Επ. Καθηγητής**

**Παναγιώτα Τσελέντη, ΕΔΙΠ**

**Ακριβή Κρούσκα, Μεταδιδακτορική Ερευνήτρια**

**Ημερομηνία εξέτασης 14/7/2023**

## ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Μιχαήλ Αποστολίδης του Φωτίου με αριθμό μητρώου 711171059 φοιτητής του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω ότι:

«Βεβαιώνω ότι είμαι συγγραφέας αυτής της Διπλωματικής εργασίας και κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών



## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέποντα καθηγητή μου, Χρήστο Τρούσσα για την καθοδήγηση και υποστήριξη σε όλη τη διαδικασία της έρευνας για την παρούσα διπλωματική εργασία.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένεια και τους φίλους μου για την αμέριστη υποστήριξη, ενθάρρυνση και κατανόησή τους σε όλη τη διάρκεια της ακαδημαϊκής μου διαδρομής.

## Περίληψη

Η διείσδυση της ψηφιοποίησης σε όλους τους τομείς της ζωής και ακόμη και σε όλες τις ηλικιακές ομάδες έχει οδηγήσει στην κατάσταση ότι πλέον εκτελούμε σχεδόν όλες τις εργασίες με κάποια μορφή υποβοήθησης από υπολογιστή. Αυτό σημαίνει ότι οι άνθρωποι, με τις προσωπικές τους εμπειρίες, προσδοκίες, ικανότητες και ανάγκες, διαδραματίζουν ακόμη πιο σημαντικό ρόλο στην ανάπτυξη λύσεων και διεπαφών χρήστη που βασίζονται σε υπολογιστή. Η συνολική εμπειρία χρήστη και η ψηφιακή συμμετοχή έχουν αυξανόμενη οικονομική και κοινωνική σημασία. Αυτή η εργασία διερευνά τον τρόπο με τον οποίο η ποικιλομορφία ή η διαφορετικότητα των ανθρώπων μπορεί να ληφθεί υπόψη στον ανθρωποκεντρικό σχεδιασμό των συστημάτων και των λύσεων τεχνητής νοημοσύνης. Αυτό απαιτεί οι ψηφιακές εφαρμογές και οι διεπαφές χρήστη να γίνουν πιο ευφείς και να μπορούν να προσαρμοστούν στα μεμονωμένα χαρακτηριστικά, τις ικανότητες ή τις καταστάσεις της ζωής μας. Είναι επίσης όλο και πιο σημαντικό οι βασικές τεχνολογίες όπως η τεχνητή νοημοσύνη να λαμβάνουν υπόψη την ποικιλομορφία.

Σκοπός της διπλωματικής εργασίας είναι η ανάπτυξη μίας ανθρωποκεντρικής εφαρμογής με χρήση τεχνητής νοημοσύνης για την προσαρμογή της διεπαφής χρήστη. Δημιουργήθηκε μία εφαρμογή κινητών για τα τις συγκοινωνίες της Αθήνας. Η εφαρμογή στοχεύει να καλύψει τις ανάγκες ενός ευρέος φάσματος χρηστών, συμπεριλαμβανομένων εκείνων με διαφορετικό υπόβαθρο, εμπειρίες, ικανότητες και ανάγκες. Με την ενσωμάτωση ευφύων πρακτόρων βασισμένων σε κανόνες, η εφαρμογή στοχεύει βελτίωση της προσβασιμότητας και της ευχρηστίας, εξασφαλίζοντας την ομαλή εμπειρία χρήστη για άτομα με διαφορετικές ανάγκες και προτιμήσεις. Η εργασία ακολουθεί μια προσέγγιση ανάπτυξης με επίκεντρο τον χρήστη, που περιλαμβάνει έρευνα χρηστών, συλλογή σχολίων και δοκιμές σε όλη τη διαδικασία ανάπτυξης. Αξιολογούνται αλγόριθμοι, όπως Naïve Bayes, SGD, KNN, SVM, Random Forest και Decision Tree, στην προσαρμογή της διεπαφής χρήστη με βάση το ιστορικό αλληλεπίδρασής του χρήστη. Ο αλγόριθμος K-Nearest Neighbors (KNN) αναδεικνύεται ως η βέλτιστη επιλογή για την εφαρμογή όπου με την συλλογή στοιχείων κατά την χρήση της εφαρμογής, χρησιμοποιείται για την προσαρμογή της διεπαφής χρήστη και συνεπώς την βελτίωση της χρησιμότητας της εφαρμογής. Το σύστημα, μέσω καταγραφής χαρακτηριστικών στοιχείων κατά την χρήση της εφαρμογής, προσαρμόζεται στις ατομικές ανάγκες, ικανότητες και νοητικά μοντέλα του χρήστη ώστε να παρέχει τη κατάλληλη βοήθεια για την εκτέλεση των λειτουργιών όσο το δυνατόν πιο αποτελεσματικά και, με ευχάριστο ή διασκεδαστικό τρόπο.

## Λέξεις-κλειδιά

Αλληλεπίδραση ανθρώπου-υπολογιστή, ανθρωποκεντρικός σχεδιασμός, διεπαφή χρήστη, android, τεχνητή νοημοσύνη, μηχανική μάθηση, προσβασιμότητα

## Abstract

The spread of digitalization in all areas of life and even across all age groups has led to a situation where we now perform almost all tasks with some form of computer assistance. This means that people, with their personal experiences, expectations, abilities and needs, play an even more important role in the development of computer-based solutions and user interfaces. The overall user experience and digital participation are of increasing economic and social importance. This paper explores how the diversity or difference of people can be taken into account in the human-centric design of AI systems and solutions. This requires digital applications and user interfaces to become more intelligent and adaptable to individual characteristics, abilities or life situations. It is also increasingly important that key technologies such as AI take diversity into account.

The aim of this thesis is the development of a human-centered application using artificial intelligence to customize the user interface. A mobile application was created for the public transport of Athens. The application aims to meet the needs of a wide range of users, including those with different backgrounds, experiences, abilities and needs. By incorporating intelligent rule-based agents, the app aims to improve accessibility and usability, ensuring a smooth user experience for people with different needs and preferences. The work follows a user-centered development approach, including user research, feedback collection and testing throughout the development process. Algorithms such as Naïve Bayes, SGD, KNN, SVM, Random Forest and Decision Tree are evaluated in adapting the user interface based on the user's interaction history. The K-Nearest Neighbors (KNN) algorithm emerges as the optimal choice for the application where by collecting data while using the application, it is used to customize the user interface and thus improve the usability of the application. The system, by recording characteristics during the use of the application, adapts to the individual needs, abilities and mental models of the user in order to provide the appropriate assistance to perform the functions as efficiently as possible and in a pleasant or entertaining way.

## Keywords

Human-computer interaction, human-centered design, user interface, android, artificial intelligence, machine learning, accessibility

## Πίνακας περιεχομένων

1	Εισαγωγή.....	1
2	Θεωρητικό υπόβαθρο.....	4
2.1	Αλληλεπίδραση ανθρώπου - υπολογιστή .....	4
2.2	Διεπαφές χρήστη .....	4
2.2.1	Μέθοδοι τεχνητής νοημοσύνης στα IUI .....	6
2.3	Προσαρμοστικές διεπαφές χρήστη .....	7
2.4	Ανθρωποκεντρικός σχεδιασμός.....	9
2.5	Προσβασιμότητα .....	12
3	Μεθοδολογία.....	14
3.1	Στάδια πρώιμης έρευνας .....	15
3.2	Παρόμοιες εφαρμογές συγκοινωνιών.....	16
3.2.1	Εφαρμογές συγκοινωνίας.....	16
3.2.2	Εφαρμογή OASA Telematics .....	17
3.2.3	Εφαρμογή Moovit.....	19
3.2.4	Εφαρμογή Citymapper.....	22
3.3	Περιβάλλον Υλοποίησης.....	24
4	Αρχιτεκτονική και Ανάπτυξη Εφαρμογής .....	26
4.1	Αρχιτεκτονική και σχεδιασμός.....	26
4.2	Υλοποίηση και Ανάπτυξη.....	32
4.2.1	Χαρακτηριστικά εφαρμογής.....	32
4.2.2	Διεπαφή χρήστη.....	33
4.2.3	Ενσωμάτωση Python SDK .....	40
4.2.4	Ανάπτυξη Front-End.....	41
4.2.5	Προσαρμογή μέσω μηχανικής μάθησης .....	47
4.2.6	Ανάπτυξη Back-End.....	53
4.2.7	Ευφυείς πράκτορες για προσαρμοστικό UI.....	58
5	Αξιολόγηση εφαρμογής.....	61
6	Συμπεράσματα.....	71
6.1	Περιορισμοί .....	72
6.2	Μελλοντικές επεκτάσεις.....	73
7	Βιβλιογραφία .....	76

## Πίνακας Εικόνων

Εικόνα 1 (Sonntag, 2017)	5
Εικόνα 3 Ανθρωποκεντρικός σχεδιασμός	9
Εικόνα 4 Ανθρωποκεντρικό Μοντέλο σχεδιασμού	10
Εικόνα 5 Διεπαφή χρήστη εφαρμογής OASA Telematics	19
Εικόνα 6 Στιγμιότυπα UI Moovit (Inc., About Moovit, 2023)	21
Εικόνα 7 Στιγμιότυπα εφαρμογής Citymapper	23
Εικόνα 8 Λογότυπο Android Studio (Google, Android Studio IDE, 2022)	24
Εικόνα 9 MainActivity Layout (Σκούρο θέμα)	33
Εικόνα 10 MainActivity Layout (Ανοιχτό θέμα)	33
Εικόνα 11 Στιγμιότυπο λίστας γραμμών λεωφορείων	34
Εικόνα 12 Στιγμιότυπο εμφάνισης στάσεων και δρομολογίων γραμμής	35
Εικόνα 13 Στιγμιότυπα εμφάνισης θέσης λεωφορείου στον χάρτη	36
Εικόνα 14 Εμφάνιση στάσης και λεωφ.	37
Εικόνα 15 Προσθήκη στάσης στα αγαπημένα	37
Εικόνα 16 Λειτουργία ειδοποίησης άφιξης λεωφ.	37
Εικόνα 17 Εμφάνιση αγαπημένων στάσεων	38
Εικόνα 18 Εμφάνιση αγαπημένων γραμμών	38
Εικόνα 19 Εμφάνιση ρυθμίσεων	39
Εικόνα 20 Στιγμιότυπο εγκατάστασης βιβλιοθηκών	40
Εικόνα 21 XML αρχεία που χρησιμοποιούνται στην εφαρμογή	42
Εικόνα 22 Διαφοροποίηση πυκνοτήτων	43
Εικόνα 23 Παράδειγμα API response	55
Εικόνα 24 Διάγραμμα βάσης δεδομένων	58

## Πίνακας Γραφημάτων

Γράφημα 1 Απεικόνιση φύλου	61
Γράφημα 2 Απεικόνιση Εύρους Ηλικίας	62
Γράφημα 3 Απεικόνιση επαγγέλματος	62
Γράφημα 4 Απεικόνιση τεχνολογικής εμπειρίας	63
Γράφημα 5 Απεικόνιση τεχνολογίας στο επάγγελμα	63
Γράφημα 6 Εξοικείωση με το θέμα	64
Γράφημα 7 Πλοήγηση και κατανόηση λειτουργιών	65
Γράφημα 8 Αξιολόγηση UI	65
Γράφημα 9 Αξιολόγηση προσαρμοστικότητας UI	66
Γράφημα 10 Αξιολόγηση συνολικής εμπειρίας χρήστη	67
Γράφημα 11 Αξιολόγηση λειτουργιών	67
Γράφημα 12 Αξιολόγηση στόχων και απαιτήσεων	68
Γράφημα 13 Απεικόνιση προτεινόμενων νέων χρηστών	68
Γράφημα 14 Προτάσεις βελτίωσης	69



## Πίνακας Διαγραμμάτων

Διάγραμμα 1 Διάγραμμα ροής εφαρμογής αρχικής οθόνης	26
Διάγραμμα 2 Διάγραμμα ροής γραμμών λεωφορείου	27
Διάγραμμα 3 Διάγραμμα ροής θέσης λεωφορείων	28
Διάγραμμα 4 Διάγραμμα ροής κοντινών στάσεων	28
Διάγραμμα 5 Διάγραμμα ροής επιλογής αγαπημένων	29
Διάγραμμα 6 Διάγραμμα ροής ρυθμίσεων	29
Διάγραμμα 7 Διάγραμμα ροής εμφάνισης στάσης	30
Διάγραμμα 8 Διάγραμμα ροής ευφυή πράκτορα	31

## Πίνακας Παράρτημα Κώδικα

Κώδικας 1 MainActivity	83
Κώδικας 2 Haptic και sound feedback	83
Κώδικας 3 Ενεργοποίηση voice-to-text δραστηριότητας	84
Κώδικας 7 UsageTip κλάση	84
Κώδικας 4 Συλλογή μετρικών input 1/2	85
Κώδικας 5 Συλλογή μετρικών input 2/2	86
Κώδικας 6 Λειτουργία voice-to-speech	87
Κώδικας 8 Εισαγωγή στοιχείων στον πίνακα user_interactions	87
Κώδικας 9 Διάλογος με τα tips	88
Κώδικας 10 OASAService κλάση	89
Κώδικας 11 Ανάκτηση στοιχείων γραμμών από API	90
Κώδικας 12 Ανίχνευσης ζούμ και κάλεσμα reflexAgent	91
Κώδικας 13 Επεξεργασία κατάστασης από δεδομένα χρήστη	91
Κώδικας 14 Εφαρμογή ενεργειών	92
Κώδικας 15 Χάρτης από ενέργειες ανά κανόνα	92
Κώδικας 16 Έλεγχος άδειας για επεξεργασία εξωτερικού χώρου αποθήκευσης	93
Κώδικας 17 Έλεγχος άδειας τοποθεσίας κινητού	93
Κώδικας 18 Καταγραφή στοιχείων χρήσης και χρήση ML μοντέλου	94
Κώδικας 22 Εισαγωγής προτεινόμενης γραμμής στην αρχή της λίστας	94
Κώδικας 19 Αξιολόγηση μοντέλων	95
Κώδικας 20 Συνάρτηση εκπαίδευσης και εκτίμησης μοντέλων	95
Κώδικας 21 Αποθήκευση στοιχείων πίνακα user_interactions σε csv αρχείο	96

## Πίνακας Παράρτημα Στιγμιότυπων Εκτέλεσης

Στιγμιότυπο 1 Εμφάνιση οδηγιών χρήσης	97
Στιγμιότυπο 2 Ενεργοποίηση θέματος υψηλής αντίθεσης	97
Στιγμιότυπο 3 Ενεργοποίηση γραμματοσειράς φιλικής προς δυσλεκτικούς	98
Στιγμιότυπο 4 Λειτουργία φωνητικών εντολών	98
Στιγμιότυπο 5 Λειτουργία μεγεθυντικού φακού	98

## Αλφαβητικό Ευρετήριο Όρων

<b>Συντομογραφία</b>	<b>Αγγλικός όρος</b>	<b>Ελληνικός όρος</b>
<b>AI</b>	Artificial Intelligence	Τεχνητή Νοημοσύνη
<b>AUI</b>	Adaptive User Interface	Προσαρμοστική Διεπαφή Χρήστη
<b>HCI</b>	Human–computer Interaction	Αλληλεπίδραση Ανθρώπου Υπολογιστή
<b>IUI</b>	Intelligent User Interface	Ευφυής Διεπαφή Χρήστη
<b>ML</b>	Machine Learning	Μηχανική Μάθηση
<b>UI</b>	User Interface	Διεπαφή Χρήστη
<b>API</b>	Application Programming Interface	Διεπαφή Προγραμματισμού Εφαρμογών
<b>DB</b>	Database	Βάση Δεδομένων

# 1 Εισαγωγή

Το παρόν κεφάλαιο πραγματεύεται την επισκόπηση του βασικού πλαισίου της θεματικής περιοχής, των βασικών όρων που χρησιμοποιούνται σε αυτή τη διπλωματική εργασία, του χώρου προβλημάτων και ως εκ τούτου, τα ερευνητικά ερωτήματα, πριν από τη λεπτομερή ανάλυση των ερευνητικών στόχων, του πεδίου εφαρμογής και της περιγραφής του έργου.

Η εξέλιξη του λογισμικού υπολογιστών αφορά μια συνεχή διαδικασία που με την πάροδο του χρόνου έχει γνωρίσει καινοτομίες στη χρηστικότητα και τη φιλικότητα προς τον χρήστη. Από τα αρχικά στάδια, το λογισμικό χρησιμοποιήθηκε κυρίως από μηχανικούς και επιστήμονες με έναν προκαθορισμένο τρόπο για συγκεκριμένες εργασίες, ενώ δεν δόθηκε ιδιαίτερη προσοχή στη διεπαφή χρήστη (User Interface, UI). Το λογισμικό που αναπτύχθηκε για τους πρώτους υπολογιστές ήταν εξαιρετικά εξειδικευμένο και μη φιλικό προς το χρήστη χωρίς εκτεταμένη εκπαίδευση για την λειτουργία τους.

Ωστόσο, καθώς οι υπολογιστές έγιναν πιο διαδεδομένοι στην καθημερινή ζωή, έγινε εμφανής η ανάγκη για το λογισμικό να είναι πιο προσιτό σε ένα ευρύτερο φάσμα χρηστών. Η άνοδος των προσωπικών υπολογιστών στη δεκαετία του 1980 οδήγησε στην ανάπτυξη γραφικών διεπαφών χρήστη (Graphical User Interfaces, GUIs) σχεδιασμένων για μεμονωμένους χρήστες. Αυτό οδήγησε στην ανάπτυξη ενός λογισμικού που να είναι ιδιαίτερα προσαρμοσμένο στον χρήστη για να καλύπτονται συγκεκριμένες ανάγκες και προτιμήσεις του, καθιστώντας το πιο αποτελεσματικό. Ο στόχος αυτών των διεπαφών είναι να καταστήσουν την τεχνολογία πιο προσιτή και ευκολότερη στη χρήση για όλους, συμπεριλαμβανομένων των ατόμων με αναπηρία.

Τα τελευταία χρόνια, η χρήση φορητών συσκευών έχει αυξηθεί εκθετικά. Σύμφωνα με τα στοιχεία του (Statista Search Department, 2022) η πλειοψηφία των ανθρώπων παγκοσμίως είναι κάτοχοι έξυπνων φορητών συσκευών που χρησιμοποιούν καθημερινά. Οι φορητές συσκευές έχουν γίνει αναπόσπαστο μέρος της καθημερινής ζωής, με πολλούς ανθρώπους να βασίζονται σε αυτά για επικοινωνία, ψυχαγωγία και πρόσβαση σε πληροφορίες. Ως εκ τούτου, η ενσωμάτωση αυτών στην καθημερινότητα έχει οδηγήσει σε μια αλλαγή στον τρόπο που αλληλεπιδρούμε με την τεχνολογία. Οι διεπαφές χρήστη, χρησιμοποιώντας νέες τεχνολογίες, έχουν εξελιχθεί ώστε να διευκολύνουν τους χρήστες να πλοηγούνται και να βρίσκουν τις πληροφορίες που χρειάζονται με ένα πιο διαισθητικό και φιλικό τρόπο.

Ωστόσο, παρά τα τεράστια άλματα στον σχεδιασμό της διεπαφής χρήστη, οι φορητές συσκευές έχουν δημιουργήσει νέες προκλήσεις όσον αφορά την προσβασιμότητα. Οι φυσικοί περιορισμοί αυτών των συσκευών έχουν σημαντικό αντίκτυπο στο μέγεθος οθόνης και στους διαθέσιμους τρόπους

αλληλεπίδρασης, τέτοιοι περιορισμοί και άλλοι έχουν δημιουργήσει ορισμένες προκλήσεις χρησιμότητας των εφαρμογών κινητών συσκευών. Για παράδειγμα, άτομα με προβλήματα όρασης είναι πιθανό να εμφανίσουν δυσκολία στην ανάγνωση κειμένου που είναι πολύ μικρό ή που έχει χαμηλή αντίθεση. Ορισμένα άτομα με αναπηρία μπορεί να δυσκολευτούν να πλοηγηθούν σε περίπλοκες οθόνες αφής ή να αντιμετωπίσουν δυσκολίες στην αντίληψη του οπτικού περιεχομένου λόγω προβλημάτων όρασης, ενώ τα άτομα με κινητικές αναπηρίες μπορεί να εμφανίζουν δυσκολίες στη χρήση μικρών κουμπιών και διεπαφών αφής που είναι κοινά στις φορητές συσκευές.

Η αυξανόμενη εξάρτηση από εφαρμογές των έξυπνων φορητών συσκευών για βασικές υπηρεσίες, επικοινωνία και διάδοση πληροφοριών μπορεί να αποκλείσει ακούσια άτομα με αναπηρία, εάν οι εφαρμογές δεν έχουν σχεδιαστεί με γνώμονα την προσβασιμότητα. Συνεπώς, είναι σημαντική η στροφή προς την ανάπτυξη προσαρμοστικών, ανθρωποκεντρικών διεπαφών χρήστη. Αυτό είναι δυνατό χάρη στην πρόοδο της τεχνητής νοημοσύνης και της μηχανικής μάθησης που επιτρέπουν στους υπολογιστές να μαθαίνουν και να προσαρμόζονται στη συμπεριφορά του χρήστη με την πάροδο του χρόνου.

Στη παρούσα διπλωματική εργασία έχουν διερευνηθεί και οργανωθεί οι βασικές έννοιες του ανθρωποκεντρικού σχεδιασμού λογισμικών καθώς και της προσαρμοστικής διεπαφής χρήστη με σκοπό την βελτίωση της ευχρηστίας και την παροχή κατάλληλων βοηθειών για την εκτέλεση των λειτουργιών ενός λογισμικού. Σχεδιάστηκε και αναπτύχθηκε μια εφαρμογή Android για το σύστημα λεωφορείων της Αθήνας με ανθρωποκεντρική προσέγγιση που εξασφαλίζει την ευχάριστη και διαδραστική εμπειρία του εκάστοτε χρήστη. Κατά την χρήση του συστήματος οι χρήστες έχουν την δυνατότητα να πληροφορηθούν σε πραγματικό χρόνο για τα δρομολόγια, τις αφίξεις και αναχωρήσεις των λεωφορείων και τρόλεϊ της πόλης με έναν προσαρμοστικό και προσβάσιμο τρόπο. Ο σχεδιασμός της διεπαφής χρήστη (UI) της ανθρωποκεντρικής εφαρμογής πρέπει να είναι προσαρμοστικός, λαμβάνοντας υπόψη το ποικίλο φάσμα των χρηστών και τα διαφορετικά επίπεδα τεχνολογικής επάρκειάς τους. Επίσης, είναι αναγκαία η χρήση διαισθητικής πλοήγησης, καθαρών εικονιδίων και εύκολα κατανοητών ετικετών που καθιστούν την εφαρμογή πιο προσίτη σε όλους τους χρήστες. Επιπλέον, η ενσωμάτωση μιας οπτικά ελκυστικής και συνεπούς γλώσσας σχεδιασμού σε όλη την εφαρμογή ενισχύει την αφοσίωση και την ικανοποίηση των χρηστών.

Οι λειτουργίες προσβασιμότητας αποτελούσαν κορυφαία προτεραιότητα κατά την ανάπτυξη αυτής της εφαρμογής. Η εφαρμογή παρέχει υποστήριξη για συσκευές ανάγνωσης οθόνης και φωνητικές εντολές που βοηθούν τους χρήστες με προβλήματα όρασης ή κινητικές αναπηρίες. Η λειτουργία μετατροπής κειμένου σε ομιλία μπορεί να βοηθήσει τους χρήστες να πλοηγηθούν στην εφαρμογή, η χρήση

λειτουργιών υψηλής αντίθεσης, αύξηση μεγεθών γραμματοσειράς οι προσαρμόσιμοι συνδυασμοί χρωμάτων βελτιώνουν την αναγνωσιμότητα για άτομα με προβλήματα όρασης.

## 2 Θεωρητικό υπόβαθρο

### 2.1 Αλληλεπίδραση ανθρώπου - υπολογιστή

Η αλληλεπίδραση ανθρώπου-υπολογιστή (Human Computer Interaction, HCI) αναφέρεται στο πεδίο που μελετά το σχεδιασμό, την αξιολόγηση και τη υλοποίηση διαδραστικών υπολογιστικών συστημάτων για την ανθρώπινη χρήση. Στόχος της αλληλεπίδρασης ανθρώπου-υπολογιστή είναι ο σχεδιασμός διεπαφών χρήστη που είναι διαισθητικές, εύκολες στην πλοήγηση και παρέχουν σαφή ανατροφοδότηση σχετικά με την κατάσταση του συστήματος. Ο τομέας του HCI έχει αναπτυχθεί σημαντικά τα τελευταία χρόνια, λόγω της αυξανόμενης επικράτησης της τεχνολογίας στην καθημερινότητα των ανθρώπων (Roser, Ritchie, & Mathieu, 2023). Λαμβάνοντας υπόψιν, τις ανάγκες και τις επιθυμίες των χρηστών, ο διαδραστικός σχεδιασμός προσφέρει στον χρήστη μάθηση, προστασία, απόδοση, αποδοτικότητα και ικανοποίηση, πάντα με γνώμονα τον σχεδιασμό των υπολογιστικών συστημάτων. Κρίσιμο ρόλο στον τομέα του HCI κατέχουν οι διεπαφές χρήστη καθώς αποτελούν το σημείο αλληλεπίδρασης μεταξύ του ανθρώπου και ενός υπολογιστή ή λογισμικού.

Οι ερευνητές, οι οποίοι έχουν ασχοληθεί με το πεδίο του HCI (Siricharoen, 2018), μελετούν τον τρόπο με τον οποίο οι άνθρωποι αλληλεπιδρούν μεταξύ τους μέσω υπολογιστικών συστημάτων, όπως σε διαδικτυακές κοινότητες και μέσα κοινωνικής δικτύωσης, για να ενημερώσουν το σχεδιασμό διεπαφών που διευκολύνουν την επικοινωνία και τη συνεργασία. Διάφορες προσεγγίσεις, όπως η έρευνα χρηστών, οι δοκιμές ευχρηστίας ή ο σχεδιασμός με επίκεντρο τον χρήστη, χρησιμοποιούνται για την αξιολόγηση του σχεδιασμού και της λειτουργικότητας μιας διεπαφής χρήστη, καθώς και για την ικανοποίηση των αναγκών και των προσδοκιών των χρηστών. Στην συνέχεια, χρησιμοποιώντας τις πληροφορίες που συλλέγονται από αυτές τις τεχνικές, για τη βελτίωση του σχεδιασμού και της λειτουργίας τους, δημιουργούν πιο φιλικές και αποτελεσματικές διεπαφές χρήστη.

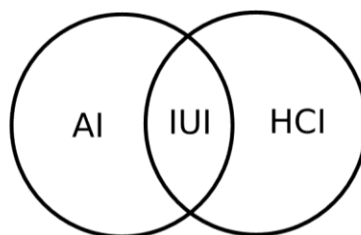
### 2.2 Διεπαφές χρήστη

Η διεπαφή χρήστη (UI) αποτελεί το στοιχείο διασύνδεσης του ανθρώπου και υπολογιστή ή λογισμικού. Ο τρόπος με τον οποίο ο χρήστης αλληλεπιδρά και ελέγχει το λογισμικό ή το υπολογιστικό σύστημα περιλαμβάνει στοιχεία όπως κουμπιά, μενού, εικονίδια και άλλες οπτικές ενδείξεις. Στόχος μιας διεπαφής χρήστη ενός λογισμικού είναι να διευκολύνει όσο το δυνατόν περισσότερο τον χρήστη να εκτελεί τις λειτουργίες του παρέχοντας σαφή ανατροφοδότηση σχετικά με την κατάσταση του

συστήματος και τυχόν σφάλματα που μπορεί να προκύψουν. Μια καλή διεπαφή χρήστη είναι διαισθητική, εύκολη στην πλοήγηση και συνεπής στο σχεδιασμό και τη συμπεριφορά της.

Ωστόσο, η άνοδος και η ενσωμάτωση νέων τεχνολογιών στην καθημερινή ζωή, όπως οι κινητές συσκευές και το Διαδίκτυο των Πραγμάτων (IoT), έφερε νέα χαρακτηριστικά και δυνατότητες που μπορούν να κάνουν τις διεπαφές χρήστη πιο περίπλοκες και πιο δύσκολες στην χρήση. Ως αποτέλεσμα, το πεδίο έχει στραφεί προς τη μελέτη νέων τύπων διεπαφών, όπως οι φυσικές διεπαφές χρήστη (Natural User Interfaces - NUIs) και οι ευφυείς διεπαφές χρήστη (Intelligent User Interfaces - IUI). Οι φυσικές διεπαφές χρήστη σχεδιάζονται για την αλληλεπίδραση συστημάτων με φυσικές μορφές εισαγωγής, όπως χειρονομίες, ομιλία και αφή, αντί για παραδοσιακές μορφές εισαγωγής, όπως πληκτρολόγιο και ποντίκι. Μερικές περιπτώσεις εφαρμογής NUI περιλαμβάνουν οθόνες αφής, φωνητικές εντολές και αναγνώριση χειρονομιών. Για παράδειγμα, σε μια συσκευή με οθόνη αφής, οι χρήστες μπορούν να αλληλεπιδρούν με τη συσκευή χρησιμοποιώντας τα δάχτυλά τους για να αγγίξουν, και να σαρώσουν.

Τα τελευταία χρόνια, ο τομέας της τεχνητής νοημοσύνης (Artificial Intelligence, AI) έχει σημειώσει σημαντική πρόοδο στον τομέα της αλληλεπίδρασης ανθρώπου-υπολογιστή. Η τεχνητή νοημοσύνη έχει φέρει επανάσταση στον τρόπο με τον οποίο αλληλεπιδρούμε με την τεχνολογία, επιτρέποντας πιο διαισθητικές, φυσικές και εξατομικευμένες εμπειρίες. Η τεχνητή νοημοσύνη έχει φέρει επανάσταση στον τρόπο με τον οποίο αλληλεπιδρούμε με την τεχνολογία, επιτρέποντας πιο διαισθητικές, φυσικές και εξατομικευμένες εμπειρίες. Μια βασική εξέλιξη είναι η εφαρμογή αλγορίθμων μηχανικής μάθησης στη HCI, που επιτρέπει στα συστήματα να μαθαίνουν από τη συμπεριφορά των χρηστών και να προσαρμόζουν ανάλογα τις διεπαφές τους. Στην Εικόνα 1 αναφέρεται το σημείο τομής της τεχνητής νοημοσύνης (Artificial Intelligence, AI) και της αλληλεπίδρασης ανθρώπου-υπολογιστή (HCI) που είναι οι ευφυείς διεπαφές χρήστη (IUI).



Εικόνα 1 (Sonntag, 2017)

Οι ευφυείς διεπαφές χρήστη χρησιμοποιούν τεχνολογίες τεχνητής νοημοσύνης για την βελτίωση της εμπειρίας του χρήστη και έτσι επιχειρούν να λύσουν μερικά από τα προβλήματα των παραδοσιακών διεπαφών χρήστη (Shneiderman, 1997). Αυτές οι διεπαφές μπορούν να προσαρμόζονται στις ανάγκες και τις προτιμήσεις του χρήστη, να παρέχουν εξατομικευμένες συστάσεις και ακόμη και να ολοκληρώνουν εργασίες στην θέση του χρήστη (Alvarez-Cortes, Zarate, Uresti, & Zayas, 2009).

### 2.2.1 Μέθοδοι τεχνητής νοημοσύνης στα ΙUI

Η άνοδος των ΙUI οφείλεται στις αυξανόμενες δυνατότητες της τεχνητής νοημοσύνης και της μηχανικής μάθησης, οι οποίες επιτρέπουν πιο εξελιγμένες αλληλεπιδράσεις μεταξύ ανθρώπων και τεχνολογίας. Μέθοδοι ΑΙ έχουν εισαχθεί στο πεδίο των ΙUI για την δημιουργία μιας πιο φυσικής και αποτελεσματικής αλληλεπίδρασης ανθρώπου μηχανής. Μερικοί από τους τρόπους που χρησιμοποιούνται στα ΙUI είναι για την χρήση ευφυών βοηθών, έξυπνα συστήματα διδασκαλίας, έξυπνα συστήματα βοήθειας, συστήματα υποστήριξης αποφάσεων, ευφυείς πράκτορες κ.α. (Hefley & Murray, 1993).

#### 2.2.1.1 Μηχανική μάθηση

Μια από τις πιο δημοφιλείς μεθόδους στον τομέα της ΑΙ είναι η μηχανική μάθηση (Machine Learning - ML) εφαρμόζεται για τη δημιουργία λογισμικού με αναγνώριση ομιλίας, όραση υπολογιστή και άλλες εφαρμογές (Jordan & Mitchell, 2015). Αυτή η μέθοδος, παρέχει νέους τρόπους, ταχύτερης και ακριβέστερης προσαρμογής, σύστασης, βοήθειας, ανίχνευσης, απεικόνισης και χρήσης σε διεπαφές χρήστη, καθώς μαθαίνει από προηγούμενα δεδομένα και προσφέρει αυτόματη βελτίωση με την πάροδο του χρόνου, εξαλείφοντας την ανάγκη για μη αυτόματο προγραμματισμό εισόδων και αντίστοιχες εξόδους.



### 2.3 Προσαρμοστικές διεπαφές χρήστη

Η προσαρμοστική διεπαφή χρήστη (AUI) αναφέρεται σε μια διεπαφή χρήστη που είναι ικανή να αντιδρά σε αλλαγές του περιβάλλοντος χρήσης με συνεχή τρόπο (Akiki, Bandara, & Yu, 2015). Η βασική ιδέα των προσαρμοστικών διεπαφών χρήστη είναι ότι οι χρήστες είναι διαφορετικοί και, επομένως, έχουν διαφορετικές ανάγκες από ένα σύστημα. Το σύστημα πρέπει να προσαρμόζεται στο χρήστη, αντί να αναγκάζει το χρήστη να προσαρμόζεται στο σύστημα.

Τα συστήματα AUI αλλάζουν την δομή, τα περιεχόμενα και τα στοιχεία τους ανάλογα με τις ανάγκες και το πλαίσιο χρήσης του κάθε χρήστη (Matthias Schneider-Hufschmidt, 1993). Η χρήση της προσαρμοστικότητας είναι ωφέλιμη σε διάφορες εφαρμογές όπως για παράδειγμα, στην προσαρμογή της δομής των διατάξεων των ιστοσελίδων, δηλαδή, να οργανώσει εικονίδια σε οθόνες κινητών τηλεφώνων, ή να αναδιοργανώσει εφαρμογές στο μενού εφαρμογών. Επιπλέον, ανάλογα με την εφαρμογή, ο σκοπός ή ο στόχος μπορεί να διαφέρει, έτσι η προσαρμοστικότητα μπορεί να περιλαμβάνει την ελαχιστοποίηση του χρόνου επιλογής, την αύξηση της ευκρίνειας, τη μείωση του γνωστικού φορτίου, την αύξηση της εμπλοκής ή και τον συνδυασμό όλων αυτών. Υπάρχουν διαφορετικοί τύποι προσαρμογών που μπορούν να εκτελεστούν στις εφαρμογές συμπεριλαμβανομένου της παρουσίασης των γραφικών στοιχείων (θέση, μέγεθος, χρώματος, κ.λπ.) ή τη συμπεριφορά τους (αριθμός στοιχείων, κινούμενες εικόνες, κ.λπ.).

Επιπλέον, οι AUI τροποποιούν τα χαρακτηριστικά τους δυναμικά κατά το χρόνο εκτέλεσης με αλλαγές που προκαλούνται από τη συμπεριφορά του χρήστη (Schlungbaum, 1997). Τα χαρακτηριστικά και/ή η προηγούμενη συμπεριφορά κάθε χρήστη μοντελοποιούνται σε μια προσπάθεια προσαρμογής στις ανάγκες και τις επιθυμίες του. Οι τύποι των AUI χωρίζονται ανάλογα με τις μεταβλητές εισόδου που επηρεάζουν την προσαρμογή. Σύμφωνα με τους (Looije, Brake, & Neerincx, 2007) μερικές από τις μεταβλητές που επηρεάζουν την προσαρμοστικότητα είναι: Η μεταβλητή Χρήστης (User) που προσαρμόζεται το UI στις προτιμήσεις, τις γνώσεις και τις δεξιότητες του χρήστη. Η μεταβλητή Διεργασία (Task) που προσαρμόζεται σύμφωνα με την τρέχουσα εργασία του χρήστη ώστε η προσαρμογή να είναι σχετική και βοηθά τους χρήστες στην τρέχουσα δραστηριότητά τους. Η μεταβλητή Σύστημα (System) που προσαρμόζεται στις διαφορετικές δυνατότητες των συσκευών του χρήστη. Η μεταβλητή Πλαίσιο Χρήσης (Context) που προσαρμόζεται σύμφωνα με το τρέχον πλαίσιο του χρήστη μπορεί να διαδραματίσει σημαντικό ρόλο, ιδιαίτερα σε εφαρμογές για κινητές συσκευές.

Σε κάθε περίπτωση εφαρμογής, η προσαρμοστικότητα βασίζεται σε μοντέλα HCI (Oulasvirta, Kristensson, Bi, & Howes, 2018) για την επαρκή και ακριβή αποτύπωση των επιδράσεων αυτών των προσαρμογών στους επιλεγμένους στόχους. Ένα παράδειγμα είναι ο νόμος του Fitts (MacKenzie, 2009), όπου, μπορεί να χρησιμοποιηθεί για την προσαρμογή της θέσης και του μεγέθους ενός στοιχείου με σκοπό την ελαχιστοποίηση του χρόνου σκόπευσης.

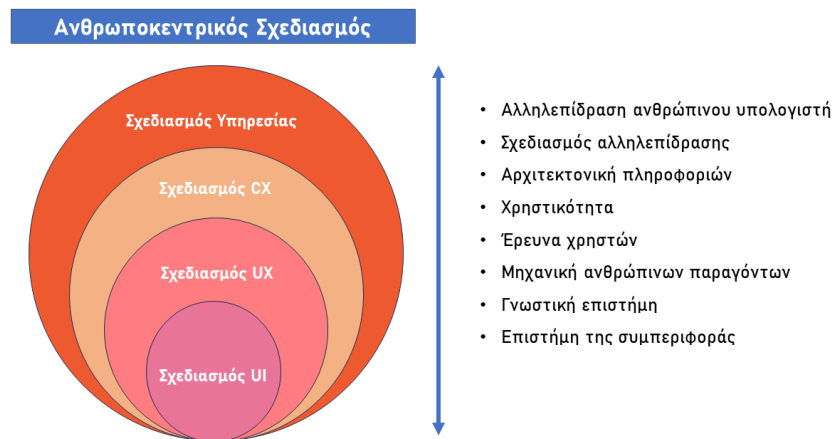
Η προσαρμογή του UI αντιστοιχίζεται σε διάφορες κατηγορίες. Κάποιες από αυτές τις κατηγορίες είναι μεταξύ των προσαρμόσιμων και των προσαρμοστικών συστημάτων (Stephanidis & Savidis, 2001). Παίρνοντας την περίπτωση των προσαρμόσιμων συστημάτων, οι τελικοί χρήστες έχουν τη δυνατότητα να προσαρμόσουν το UI στις ανάγκες τους, δηλαδή, οι χρήστες έχουν τον έλεγχο της προσαρμογής. Σε αντίθεση με τα προσαρμόσιμα συστήματα, τα προσαρμοστικά συστήματα έχουν εσωτερικούς μηχανισμούς για να πραγματοποιούν άμεσα την προσαρμογή, με ελάχιστο ή καθόλου έλεγχο από τους χρήστες. Ακόμη, υπάρχουν και άλλες ταξινομήσεις που κατηγοριοποιούν την εμπλοκή του χρήστη έναντι του συστήματος σε διάφορα στάδια της προσαρμογής, όπως η ταξινόμηση PDA-LPA η οποία με βάση τους (Bouzit, και συν., 2017), υποστήριξαν ότι παρέχει έναν λεπτομερή χαρακτηρισμό της εμπλοκής του τελικού χρήστη έναντι της αυτοδιαχείρισης του συστήματος σε σχέση με την αντίληψη, την απόφαση, τη δράση, τη μάθηση, την πρόβλεψη και την προσαρμογή.

Καταληκτικά, το γεγονός ότι υπάρχουν αρκετές προσεγγίσεις για τη δημιουργία προσαρμοστικών εφαρμογών για την προσβασιμότητα, δεν αλλάζει το γεγονός ότι έχουν εμφανιστεί και νέες προσεγγίσεις που επιτρέπουν στους χρήστες να προσαρμόζουν τις προτιμώμενες εφαρμογές ακόμη και πέρα από αυτό που υποστηρίζουν αυτές οι εφαρμογές. Ως εκ τούτου, είναι υποχρεωτική η προσαρμογή των διεπαφών χρήστη ώστε να ανταποκρίνονται σε όλες τις απαιτήσεις του διαφορετικού πλαισίου χρήσης και στις προτιμήσεις των χρηστών.

## 2.4 Ανθρωποκεντρικός σχεδιασμός

Η προσαρμοστική θεωρία διεπαφής χρήστη υπογραμμίζει επίσης τη σημασία της σχεδίασης με επίκεντρο τον χρήστη (UCD), η οποία είναι μια διαδικασία που βάζει τον χρήστη στο επίκεντρο της διαδικασίας σχεδιασμού.

Οι (Shneiderman, Plaisant, Cohen, & Jacobs, 2017) περιγράφουν τρεις βιώσιμες προσεγγίσεις για τη διεπαφή χρήστη και το σχεδιασμό HCI: συμμετοχικός σχεδιασμός, σχεδιασμός ευέλικτης αλληλεπίδρασης και ανθρωποκεντρικός σχεδιασμός. Παρόλο που οι δύο πρώτες προσεγγίσεις έχουν τα πλεονεκτήματά τους, η πρώτη μέθοδος υστερεί στην περιορισμένη συμμετοχή των χρηστών και στους μεγαλύτερους χρόνους παραγωγής ενώ η δεύτερη μέθοδος μπορεί να προκαλέσει σύγχυση στους χρήστες λόγω συχνών αλλαγών στο σχεδιασμό. Ωστόσο, ο ανθρωποκεντρικός σχεδιασμός δίνει προτεραιότητα στους χρήστες εμπλέκοντάς τους στη διαδικασία σχεδιασμού και λαμβάνοντας υπόψη τις προτιμήσεις τους, οδηγώντας σε φιλικές προς το χρήστη διεπαφές. Επίσης, αναφέρουν ότι ο ανθρωποκεντρικός σχεδιασμός έχει ως αποτέλεσμα σχέδια που αναπτύσσονται εύκολα, διατηρούνται και μπορούν να χρησιμοποιηθούν. Αυτή η προσέγγιση έχει εφαρμοστεί με επιτυχία σε πολυάριθμες μελέτες και έργα.



Εικόνα 2 Ανθρωποκεντρικός σχεδιασμός

Η προσέγγιση περιλαμβάνει προσθήκη των χρηστών στη διαδικασία σχεδιασμού, τη διεξαγωγή έρευνας χρηστών και την επαναληπτική δοκιμή και τελειοποίηση του σχεδίου με βάση τα σχόλια των χρηστών. Βοηθά, έτσι, να διασφαλιστεί ότι το τελικό προϊόν ανταποκρίνεται στις ανάγκες του χρήστη και είναι εύκολο στη χρήση. Ειδικότερα, η έρευνα χρηστών περιλαμβάνει τη συλλογή δεδομένων για τις ανάγκες, τις προτιμήσεις και τη συμπεριφορά των χρηστών μέσω μεθόδων όπως συνεντεύξεις, έρευνες και δοκιμές ευχρηστίας.

Μια άλλη βασική πτυχή του ανθρωποκεντρικού σχεδιασμού είναι η επαναληπτική δημιουργία πρωτοτύπων και η δοκιμή. Αντί να βασίζονται αποκλειστικά σε υποθέσεις ή γνώμες ειδικών, οι σχεδιαστές δημιουργούν πρωτότυπα χαμηλής πιστότητας και συλλέγουν σχόλια από πραγματικούς χρήστες. Αυτή η επαναληπτική διαδικασία επιτρέπει την έγκαιρη αναγνώριση και επίλυση προβλημάτων χρησιμότητας, διασφαλίζοντας ότι η τελική διεπαφή ανταποκρίνεται στις προσδοκίες των χρηστών. Η δοκιμή με πραγματικούς χρήστες παρέχει επίσης πολύτιμες πληροφορίες για τις προτιμήσεις των χρηστών, τις συμπεριφορές και τα πιθανά εμπόδια, βοηθώντας τους σχεδιαστές να βελτιώσουν και να βελτιστοποιήσουν ανάλογα τη διεπαφή.

### Ανθρωποκεντρικό μοντέλο σχεδιασμού



Εικόνα 3 Ανθρωποκεντρικό Μοντέλο σχεδιασμού

Πίνακας 1 Πλαίσιο μεθόδων HCD

Προσχέδιο	Πλαίσιο της χρήσης	Απαιτήσεις	Σχεδιασμός	Αξιολόγηση
Προγραμματισμός χρηστικότητας και οριοθέτηση	Ανάλυση του πλαισίου χρήσης	Προδιαγραφή απαιτήσεων χρήστη	Brainstorming	Συμμετοχική αξιολόγηση
Ανάλυση κόστους-οφέλους χρηστικότητας	Έρευνα υφιστάμενων χρηστών	Ομάδες εστίασης	Παράλληλος σχεδιασμός	Ευρετική ή αξιολόγηση εμπειρογνομόνων
	Μελέτη πεδίου / παρατήρηση χρήστη	Σενάρια χρήσης	Πρωτοτυποποίηση λογισμικού	Έρευνες χρηστών
	Ανάλυση εργασιών	Ανάλυση υφιστάμενου συστήματος / ανταγωνιστή	Κατευθυντήριες γραμμές και πρότυπα σχεδιασμού	Ερωτήσεις ικανοποίησης χρηστών
		Αντιστοίχιση εργασιών / συναρτήσεων	Αξιολόγηση γνωστικών φόρτο εργασίας	
		Απαιτήσεις χρήστη, χρηστικότητας και οργάνωσης		

Επιπλέον, η προσβασιμότητα αποτελεί ένα κρίσιμο κριτήριο του ανθρωποκεντρικού σχεδιασμού. Οι διεπαφές θα πρέπει να λαμβάνουν υπόψη τις διαφορετικές ανάγκες των χρηστών, συμπεριλαμβανομένων εκείνων με αναπηρίες ή αναπηρίες. Οι σχεδιαστές θα πρέπει να ακολουθούν καθιερωμένες οδηγίες και πρότυπα προσβασιμότητας, διασφαλίζοντας ότι οι διεπαφές είναι αντιληπτές, λειτουργικές και κατανοητές για όλους τους χρήστες. Συμμετέχοντας χρήστες με αναπηρίες και ενσωματώνοντας τα σχόλιά τους, οι σχεδιαστές μπορούν να αποκτήσουν γνώσεις για συγκεκριμένες προκλήσεις που αντιμετωπίζουν και να λάβουν τεκμηριωμένες αποφάσεις σχεδιασμού για να ξεπεράσουν αυτά τα εμπόδια.

## 2.5 Προσβασιμότητα

Η προσβασιμότητα αφορά την πρόσβαση και τη χρήση πληροφοριών, προϊόντων, συσκευών, υπηρεσιών ή περιβαλλόντων από άτομα στο μεγαλύτερο δυνατό βαθμό, συμπεριλαμβανομένων εκείνων που είναι άτομα με αναπηρία. Ο W3C (World Wide Web Consortium), ο διεθνής οργανισμός προτύπων για τον Παγκόσμιο Ιστό έχει σαφείς οδηγίες προσβασιμότητας περιεχομένου Ιστού (WCAG) (Consortium, Web Content Accessibility Guidelines, 2022) για την ανάπτυξη προσβάσιμων ιστοτόπων στους οποίους μπορούν να έχουν πρόσβαση χρήστες με ειδικές ανάγκες.

Με βάση την προσβασιμότητα στον ιστό, αναπτύσσουν περαιτέρω τα πρότυπα προσβασιμότητας για κινητές εφαρμογές (Consortium, Mobile Accessibility at W3C, 2022) λαμβάνοντας υπόψη χαρακτηριστικά κινητών όπως οθόνες αφής, μικρό μέγεθος οθόνης, χρήσεις σε διαφορετικές ρυθμίσεις όπως το έντονο ηλιακό φως κ.λπ. Εκτός από τις γενικές οδηγίες προσβασιμότητας, οι ερευνητές έχουν πρότεινε οδηγίες προσβασιμότητας για ειδικούς πληθυσμούς, όπως άτομα με προβλήματα όρασης (Park, Goh, & So, 2014), άτομα με προβλήματα ακοής (Swanepoel, Sousa, Smits, & Moore, 2019) ή άτομα μεγαλύτερης ηλικίας (Díaz-Bossini, Moreno, & Martínez, 2014).

Ταυτόχρονα, η Google και η Apple, ως οι βασικές οντότητες στην τεχνολογία κινητών και στην αγορά εφαρμογών, δημοσιεύουν τις οδηγίες προσβασιμότητας, τα εργαλεία ανάπτυξης λογισμικού (SDK) και τις σουίτες δοκιμών για εφαρμογές για κινητές συσκευές σε πλατφόρμες Android και iOS. Παρά τη σημασία αυτών των οδηγιών, οι σχεδιαστές και οι προγραμματιστές εφαρμογών αντιμετωπίζουν προκλήσεις όσον αφορά την κατανόηση και την ενσωμάτωσή τους στο σχεδιασμό εφαρμογών (Yan & Ramachandran, 2019). Η εφαρμογή της εργασίας αυτής, εστιάζει σε ζητήματα προσβασιμότητας σε εφαρμογές Android και εξετάζονται τις προτάσεις προσβασιμότητας που παρέχονται από την Google (Google, Google-Accessibility, 2023).

Οι φορητές συσκευές έχουν κάνει τεράστια άλματα στον σχεδιασμό της διεπαφής χρήστη, αλλά εξακολουθούν να υπάρχουν πολλές δυσκολίες, που πρέπει να αντιμετωπιστούν για να μπορέσουν να χρησιμοποιηθούν οι οθόνες αφής για κοινές εργασίες. Μια σύντομη περιγραφή διαφόρων ζητημάτων προσβασιμότητας σε εφαρμογές Android. Τα ζητήματα αυτά χωρίζονται σε τέσσερις κύριες κατηγορίες, επισήμανση περιεχομένου (content label), υλοποίηση UI στοιχείων, μέγεθος στόχου αφής και χαμηλή αντίθεση. Η επισήμανση περιεχομένου αντιμετωπίζει την ανάγκη για εναλλακτικά κείμενα για εικόνες/δράσεις που βοηθούν άτομα με προβλήματα όρασης. Τα ζητήματα που περιλαμβάνονται στην επισήμανση περιεχομένου είναι η έλλειψη ετικέτες κειμένου, οι διπλότυπες ετικέτες και οι περιττές

περιγραφές. Τα ζητήματα εφαρμογής UI σχετίζονται με προβλήματα που αντιμετωπίζουν χρήστες με κινητικά προβλήματα, όπως μη συμβατά εύρη με δυνατότητα κλικ, διπλότυπα όρια με δυνατότητα κλικ και ακατάλληλες περιγραφές περιεχομένου για επεξεργάσιμα πεδία. Επιπλέον εμπόδιο για τα άτομα με κινητικά προβλήματα αποτελεί το μέγεθος στόχου αφής που υπογραμμίζει τη δυσκολία ακριβούς αγγίγματος μικρών στόχων, γεγονός που εμποδίζει τη χρήση μίας εφαρμογής. Τέλος, τα προβλήματα χαμηλής αντίθεσης σχετίζονται με την ανεπαρκή αντίθεση στα στοιχεία κειμένου και εικόνας, επηρεάζοντας τους χρήστες με προβλήματα όρασης.

### 3 Μεθοδολογία

Σε αυτή την ενότητα, παρουσιάζονται οι λεπτομέρειες σχετικά με την προτεινόμενη μέθοδο. Αρχικά, για την ανάπτυξη της εφαρμογής για το σύστημα λεωφορείων της Αθήνας πραγματοποιήθηκε βιβλιογραφική ανασκόπηση σχετικά με την δημιουργία ανθρωποκεντρικής εφαρμογής με προσαρμοστικό περιβάλλον καθώς και σχετικά με την τηλεματική του ΟΑΣΑ για την διαχείριση του ανοιχτού API. Η ανάπτυξη μιας ανθρωποκεντρικής εφαρμογής για κινητά ακολουθεί μια προσέγγιση σχεδίασης με επίκεντρο τον χρήστη.

Αυτή η μεθοδολογία δίνει έμφαση στη συμμετοχή των τελικών χρηστών σε όλη τη διαδικασία ανάπτυξης, διασφαλίζοντας ότι οι ανάγκες και οι προτιμήσεις τους λαμβάνονται υπόψη σε κάθε στάδιο. Επομένως, εκτελέστηκε έρευνα για την κατανόηση των αναγκών και των προτιμήσεων των χρηστών. Διεξάχθηκαν συνεντεύξεις και έρευνες με χρήστες για τη συλλογή πληροφοριών σχετικά με τους περιορισμούς και τις προκλήσεις, που αντιμετωπίζουν οι μετακινούμενοι στην Αθήνα. Αυτές οι πληροφορίες βοήθησαν στον εντοπισμό των βασικών χαρακτηριστικών και λειτουργιών, που πρέπει να ενσωματωθούν στην εφαρμογή Android.

Έπειτα, πραγματοποιήθηκε έρευνα αγοράς για την ανάλυση υπαρχουσών εφαρμογών για κινητές συσκευές για συστήματα μεταφορών στην Αθήνα και σε άλλες πόλεις. Αυτή η έρευνα παρέχει πολύτιμες πληροφορίες για τις βέλτιστες πρακτικές, τα επιτυχημένα χαρακτηριστικά και τις πιθανές παγίδες που πρέπει να αποφευχθούν κατά τη διαδικασία ανάπτυξης. Το επόμενο βήμα αποτελεί τον σχεδιασμό και ανάπτυξη της διεπαφής χρήστη και των βασικών χαρακτηριστικών της εφαρμογής. Το βήμα περιλαμβάνει τον καθορισμό της δομής και οργάνωσης των πληροφοριών εντός της εφαρμογής δηλαδή τη δημιουργία ενός σαφούς συστήματος πλοήγησης, την κατηγοριοποίηση χαρακτηριστικών και την εξασφάλιση ευκολίας πρόσβασης σε σχετικές πληροφορίες. Καταληκτικά, διεξάχθηκαν δοκιμές χρηστικότητας με χρήστες για την αξιολόγηση των αποφάσεων σχεδιασμού και την μελλοντική αντιμετώπιση τυχόν ζητημάτων χρηστικότητας ή τομείς προς βελτίωση.



### 3.1 Στάδια πρώιμης έρευνας

Στα πρώτα στάδια της έρευνας για την ανάπτυξη ανθρωποκεντρικής εφαρμογή μέσω μεταφοράς για κινητές συσκευές, εμπλέκονται αρκετά βήματα και εκτιμήσεις για να διασφαλιστεί μια προσέγγιση με επίκεντρο τον χρήστη. Το πρώτο βήμα αφορά την αξιολόγηση των αναγκών των χρηστών. Η έρευνα ξεκίνησε με την κατανόηση των χρηστών-στόχων και των αναγκών τους. Αυτό περιλαμβάνει τη διεξαγωγή ερευνών και παρατηρήσεων χρηστών για τη συλλογή πληροφοριών σχετικά με τις προτιμήσεις και τις προσδοκίες τους όταν πρόκειται να χρησιμοποιήσουν μια εφαρμογή μεταφοράς με λεωφορείο. Η κατανόηση των δημογραφικών στοιχείων των χρηστών, των ταξιδιωτικών συνθηκών και της τεχνολογικής επάρκειας είναι απαραίτητη για την προσαρμογή της εφαρμογής στις συγκεκριμένες απαιτήσεις τους.

Έπειτα, από την διεξαγωγή συνεντεύξεων με τους χρήστες συγκεντρώθηκαν οι στόχοι και οι απαιτήσεις από την εφαρμογή. Παρακάτω, παρουσιάζεται ο Πίνακας 2 με την ανάλυση των απαιτήσεων των χρηστών.

Πίνακας 2 Ανάλυσης Απαιτήσεων Χρηστών

<b>Ανάλυση Απαιτήσεων Χρηστών</b>	<b>Περιγραφή</b>
<b>Προφίλ χρήστη</b>	Αποτελείται από επιβάτες, κατοίκους και τουρίστες στην Αθήνα
<b>Ανάγκες χρήστη</b>	Παρακολούθηση λεωφορείων σε πραγματικό χρόνο, εμφάνιση δρομολογίων, πληροφορίες των χρονοδιαγραμμάτων, επιλογές προσβασιμότητας
<b>Περιορισμοί</b>	Ανακριβή δρομολογία λεωφορείων, δυσκολία εύρεσης στάσεων λεωφορείων, έλλειψη ενημερώσεων σε πραγματικό χρόνο
<b>Προτιμήσεις χρήστη</b>	Απλή και διαισθητική διεπαφή, προσαρμοστική διεπαφή χρήστη που ανταποκρίνεται στις διαφορετικές ανάγκες και προτιμήσεις των χρηστών
<b>Προσωπική ανάπτυξη (Personal Development)</b>	Δημιουργία προσωπικοτήτων χρηστών που αντιπροσωπεύουν διαφορετικές ομάδες χρηστών και τους στόχους, τα χαρακτηριστικά τους
<b>Τεχνικές απαιτήσεις</b>	Συμβατότητα πλατφόρμας Android, ενσωμάτωση με σύστημα παρακολούθησης GPS, υποστήριξη πολλαπλών γλωσσών

Αυτός ο πίνακας παρέχει μια επισκόπηση της ανάλυσης που διεξήχθη και των απαιτήσεων των χρηστών που εντοπίστηκαν κατά τα πρώτα στάδια ανάπτυξης. Χρησιμεύει ως αναφορά για την καθοδήγηση της διαδικασίας σχεδιασμού και ανάπτυξης, διασφαλίζοντας ότι η εφαρμογή αντιμετωπίζει τις συγκεκριμένες ανάγκες, προτιμήσεις, προσδοκίες και περιορισμούς των χρηστών-στόχων.

### 3.2 Παρόμοιες εφαρμογές συγκοινωνιών

Σε αυτή την υπό-ενότητα, αναλύονται μερικές υπάρχουσες εφαρμογές μεταφοράς με MMM στην αγορά παρέχει πολύτιμες πληροφορίες για τα δυνατά και τα αδύνατα σημεία των ανταγωνιστών. Αυτή η έρευνα βοηθά στον εντοπισμό περιοχών βελτίωσης και πιθανών διαφοροποιήσεων για την εφαρμογή της εργασίας. Μελετώντας τις δυνατότητες, τη χρηστικότητα και τα σχόλια των χρηστών των υπάρχουσών εφαρμογών, είναι δυνατή η απόκτηση σημαντικών πληροφοριών για τις βέλτιστες πρακτικές και να αποφευχθούν τυχών παγίδες.

#### 3.2.1 Εφαρμογές συγκοινωνίας

Οι κινητές εφαρμογές για τη μεταφορά με μέσα μαζική μεταφοράς έχουν γίνει ολοένα και πιο δημοφιλείς στις αστικές περιοχές, παρέχοντας στους χρήστες πληροφορίες σε πραγματικό χρόνο, σχεδιασμό διαδρομής και άλλες χρήσιμες λειτουργίες. Αυτή η ανασκόπηση στοχεύει στη σύγκριση και αξιολόγηση επιλεγμένων εφαρμογών για κινητά για τη μεταφορά με μέσα μαζική μεταφοράς, επισημαίνοντας τα θετικά και τα αρνητικά τους και προτείνοντας πιθανούς τομείς βελτίωσης.

Κάθε εφαρμογή για κινητά για τις δημόσιες συγκοινωνίες είναι εξοπλισμένη με μια σειρά λειτουργιών που έχουν σχεδιαστεί για να διευκολύνουν την κίνηση μέσα σε μια πόλη χρησιμοποιώντας διάφορες διαθέσιμες επιλογές μεταφοράς. Μια τέτοια εφαρμογή συγκοινωνίας λειτουργεί ως ένας απαραίτητος ταξιδιωτικός σύμβουλος, έχοντας πλήρη γνώση κάθε γωνιάς μιας πόλης. Είναι έτοιμο να προσφέρει συμβουλές προγραμματισμού ταξιδιού και να παρέχει ενημερώσεις σε πραγματικό χρόνο για μία διαδρομή, παρακολουθώντας τις τοποθεσίες των οχημάτων. Για να εκπληρώσει αποτελεσματικά το σκοπό της, η εφαρμογή πρέπει να επιτύχει μια ισορροπία μεταξύ μιας απλής διεπαφής χρήστη και μιας ισχυρής λειτουργικότητας. Ενώ απαιτείται σημαντική προσπάθεια σχεδιασμού για την ανάπτυξη του front-end, η κατασκευή του backend της εφαρμογής αποτελεί ακόμη μεγαλύτερη πρόκληση. Οι προγραμματιστές πρέπει να ενσωματώσουν την εφαρμογή με πλατφόρμες χαρτών, πρακτορεία δημόσιων μεταφορών, εταιρείες κοινής χρήσης ποδηλάτων, παρόχους υπηρεσιών πληρωμών και επεξεργαστές δικτύου συναλλαγών. Αυτός ο περίπλοκος ιστός συνδέσεων εξασφαλίζει απρόσκοπτη εμπειρία χρήστη και αποτελεσματικό συντονισμό μεταξύ διαφόρων παρόχων υπηρεσιών.

Η αναζήτηση εμπορικών εφαρμογών εντόπισε πάνω από 100 μοναδικές εφαρμογές που σχετίζονται με ενεργή μεταφορά από την Apple App Store και Google Play Store και είναι διαθέσιμες σε διάφορες. Καθώς τα περισσότερα smartphone παγκοσμίως τρέχουν σε Android λογισμικό υπάρχουν περισσότερες εφαρμογές για το σύστημα Android, φυσικά περισσότερες εφαρμογές βρίσκονται στο Google Play Store από το Apple App Store, το οποίο αναπτύσσει μόνο εφαρμογές iOS για iPhone.

### 3.2.2 Εφαρμογή OASA Telematics

#### 3.2.2.1 Περιγραφή εφαρμογής

Η πρώτη εφαρμογή OASA Telematics App του ΟΑΣΑ αποτελεί ένα ολοκληρωμένο εργαλείο σχεδιασμένο για να παρέχει στους χρήστες εύκολη πρόσβαση σε πληροφορίες και υπηρεσίες που σχετίζονται με το σύστημα μεταφορών στην Αθήνα. Αναπτύχθηκε το 2016 για να βελτιώσει την εμπειρία μετακίνησης για κατοίκους και επισκέπτες, η εφαρμογή προσφέρει μια σειρά λειτουργιών, όπως παρακολούθηση λεωφορείων σε πραγματικό χρόνο, προγραμματισμός ταξιδιού, χρονοδιαγράμματα, ειδοποιήσεις και άλλα. Αυτή η περιγραφή στοχεύει να παρέχει μια επισκόπηση της εφαρμογής για κινητά του ΟΑΣΑ και των βασικών λειτουργιών της.

Η εφαρμογή προσφέρει τις ακόλουθες δυνατότητες στο κύριο μενού της:

- *Πλησιέστερες στάσεις:* Αυτή η λειτουργία προσδιορίζει αυτόματα την πλησιέστερη στάση λεωφορείου ή τρόλεϊ με βάση την τοποθεσία του χρήστη.
- *Τοποθεσίες λεωφορείων:* Οι χρήστες μπορούν να δουν τις θέσεις των λεωφορείων σε πραγματικό χρόνο στον χάρτη. Η ενοποίηση χαρτών και υπηρεσιών γεωγραφικού εντοπισμού χρησιμοποιείται για την εμφάνιση θέσεων οχημάτων σε πραγματικό χρόνο, κοντινών στάσεων και βέλτιστων διαδρομών.
- *Αφίξεις:* Αυτή η ενότητα παρέχει ενημερώσεις σχετικά με τις ώρες άφιξης των λεωφορείων για όλες τις γραμμές σε διάφορες στάσεις.
- *Βέλτιστη διαδρομή:* Οι χρήστες μπορούν να αναζητήσουν την πιο αποτελεσματική διαδρομή χρησιμοποιώντας τη λειτουργία MMM, λαμβάνοντας υπόψη την τρέχουσα τοποθεσία τους.
- *Γραμμές:* Πληροφορίες σχετικά με όλες τις διαδρομές λεωφορείων και τρόλεϊ είναι διαθέσιμες σε αυτήν την ενότητα.
- *Ειδήσεις:* Οι χρήστες μπορούν να έχουν πρόσβαση σε ανακοινώσεις που σχετίζονται με τις αστικές συγκοινωνίες και άλλες σχετικές ειδήσεις.

Επιπλέον, οι χρήστες έχουν την επιλογή να αποθηκεύσουν τις στάσεις που χρησιμοποιούν συχνά στην ενότητα «Αγαπημένα». Με αυτόν τον τρόπο, μπορούν εύκολα να έχουν πρόσβαση σε πληροφορίες σε πραγματικό χρόνο για όλες τις διαδρομές που διέρχονται από αυτές τις στάσεις, διευκολύνοντας τη διαδικασία σχεδιασμού του ταξιδιού τους.

### 3.2.2.2 Αξιολόγηση εφαρμογής

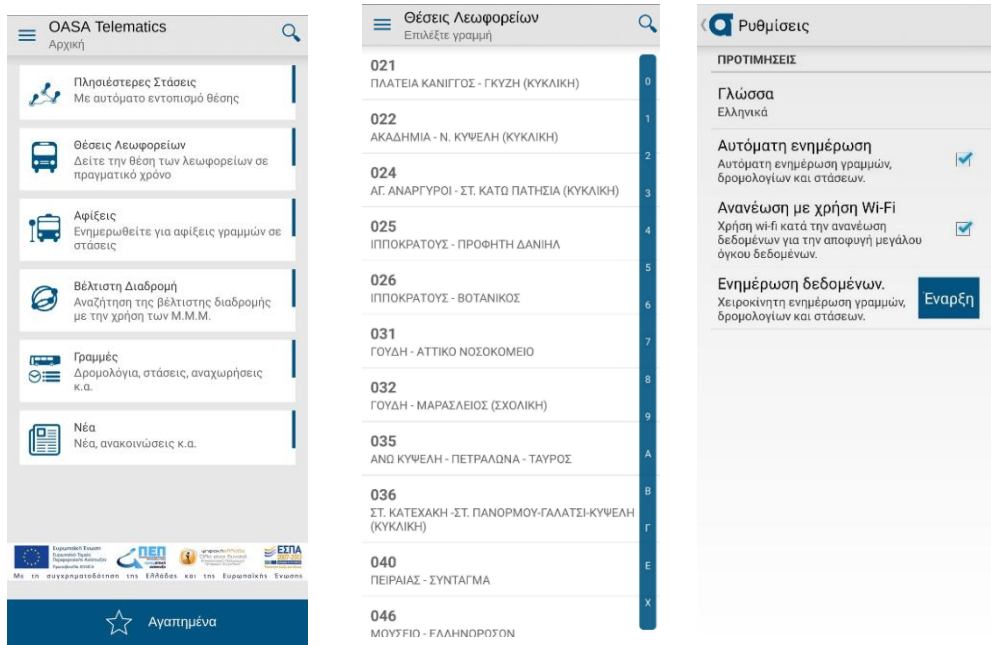
Βασισμένη σε προσωπικές εμπειρίες και κριτικές χρηστών από το Google Play Store, η εφαρμογή για κινητά OASA Telematics έχει αρκετές θετικές και αρνητικές πτυχές. Οι χρήστες εκτιμούν τη δυνατότητα παρακολούθησης λεωφορείων σε πραγματικό χρόνο της εφαρμογής, η οποία τους επιτρέπει να παρακολουθούν με ακρίβεια την τοποθεσία και τις ώρες άφιξης των λεωφορείων. Αυτή η λειτουργία επιτρέπει καλύτερο προγραμματισμό ταξιδιού και μειώνει τους χρόνους αναμονής στις στάσεις λεωφορείων.

Ωστόσο, αναφέρονται περιστασιακές τεχνικές δυσλειτουργίες, όπως αργούς χρόνους φόρτωσης, ανακρίβειες στα δεδομένα παρακολούθησης λεωφορείων. Αυτά τα ζητήματα μπορεί να είναι απογοητευτικά και να εμποδίζουν τη συνολική χρηστικότητα της εφαρμογής. Επιπλέον, αναφέρεται ότι η εφαρμογή μπορεί περιστασιακά να παρέχει ελλιπείς ή ανακριβείς πληροφορίες σχετικά με δρομολόγια λεωφορείων, στάσεις ή ώρες άφιξης. Ενώ η εφαρμογή διαθέτει φιλική διεπαφή χρήστη, ορισμένοι χρήστες βρίσκουν την πλοήγησή της ελαφρώς μπερδεμένη λόγω της παρουσίας πολλαπλών καρτελών και δραστηριοτήτων.

Επιπρόσθετα, το στυλ της διεπαφής χρήστη της εφαρμογής θεωρείται ξεπερασμένο σε σύγκριση με τις προτεινόμενες οδηγίες σχεδίασης του Android Material Design (Google, Material Design, 2021). Η ενημέρωση της διεπαφής χρήστη ώστε να ευθυγραμμιστεί με τις σύγχρονες αρχές σχεδίασης θα μπορούσε να βελτιώσει τη συνολική εμπειρία χρήστη και να κάνει την εφαρμογή πιο ελκυστική οπτικά και διαισθητική στη χρήση. Επί του παρόντος, η εφαρμογή δεν ενσωματώνεται με συστήματα αγοράς εισιτηρίων και πληρωμών. Πολλοί χρήστες εκφράζουν την επιθυμία τους για τη δυνατότητα αγοράς, αποθήκευσης και χρήσης εισιτηρίων εντός της εφαρμογής, απλοποιώντας, έτσι, τη διαδικασία έκδοσης εισιτηρίων και βελτιώνοντας την ευκολία.

Καταληκτικά, η εφαρμογή δεν λαμβάνει υπόψη το ζήτημα προσβασιμότητα. Υπάρχει μια αξιοσημείωτη απουσία προσβάσιμων επιλογών, όπως πλοήγηση με φωνητική καθοδήγηση ή ρυθμιζόμενα μεγέθη γραμματοσειράς, που περιορίζει τη χρηστικότητα για άτομα με προβλήματα όρασης ή κινητικότητας. Η

ενσωμάτωση χαρακτηριστικών προσβασιμότητας θα ενίσχυε σημαντικά τη χρηστικότητα της εφαρμογής για όλους τους χρήστες.



Εικόνα 4 Διεπαφή χρήστη εφαρμογής OASA Telematics

### 3.2.3 Εφαρμογή Moovit

#### 3.2.3.1 Περιγραφή εφαρμογής

Το Moovit είναι μια αρκετά δημοφιλής εφαρμογή για κινητά σχεδιασμένη για χρήστες μέσω μαζικής μεταφοράς, προσφέροντας ένα ολοκληρωμένο σύνολο λειτουργιών για τη διευκόλυνση ταξιδιωτικών εμπειριών. Το Moovit, που κυκλοφόρησε το 2012 ως δωρεάν εφαρμογή και τοποθετείται ως πάροχος Mobility as a Service (MaaS). Ουσιαστικά, το Moovit λειτουργεί ως σχεδιαστής ταξιδιών, ενσωματώνοντας πληροφορίες. Η εφαρμογή Moovit μπορεί εύκολα να εγκατασταθεί σε οποιαδήποτε ηλεκτρονική συσκευή με δυνατότητα web, παρέχοντας πληροφορίες σχετικά με τις ώρες ταξιδιού, τα δρομολόγια και το κόστος για διάφορες επιλογές μεταφοράς, όπως δημόσια μέσα μεταφοράς, ταξί, κοινή χρήση αυτοκινήτου, κοινή χρήση ποδηλάτων και κοινή χρήση σκούτερ. Η εφαρμογή συνδυάζει δεδομένα τόσο από ιδιωτικούς όσο και από δημόσιους μεταφορείς, χρησιμοποιώντας ως αποθετήριο πληροφοριών για τα δημόσια μέσα μεταφοράς σε πραγματικό χρόνο. Οι χρήστες συνεισφέρουν στο σύστημα χωρίς να λαμβάνουν πληρωμές ή εκπτώσεις. Το Moovit ενθαρρύνει τη δέσμευση των χρηστών μέσω λειτουργιών παιχνιδιών, όπως ένα σύστημα σε επίπεδο σημείου, όπου οι χρήστες κερδίζουν πόντους για τις συνεισφορές και την πρόοδό τους σε επίπεδα, ενισχύοντας τη φήμη τους στην κοινότητα και προωθώντας τη χρήση της εφαρμογής.

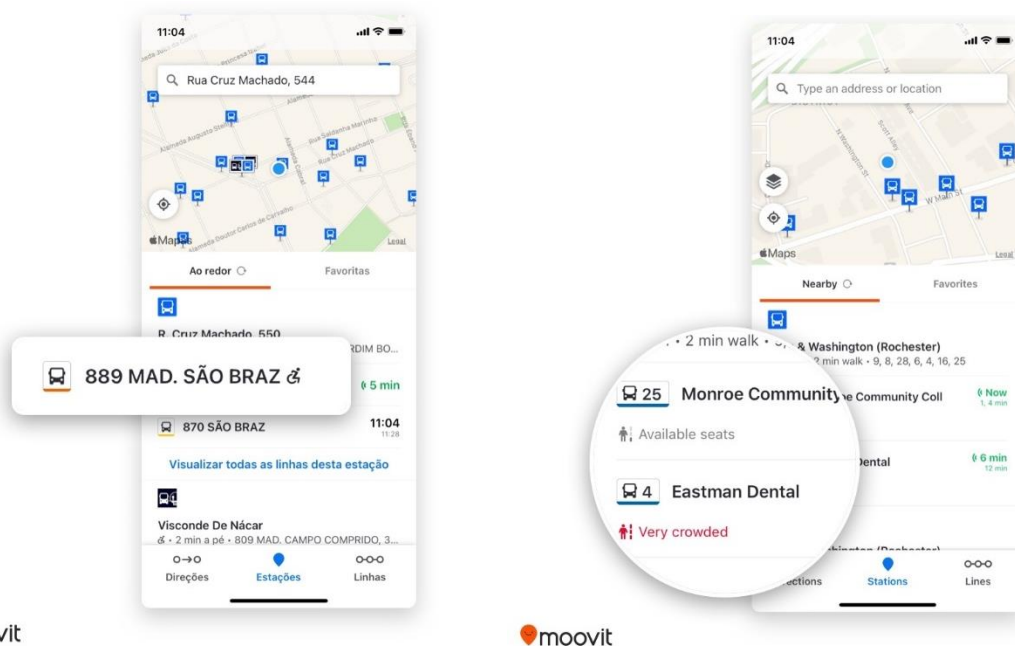
Έως το 2023, το Moovit είναι διαθέσιμο σε περισσότερες από 3.400 πόλεις σε 112 χώρες και υποστηρίζει 45 γλώσσες με 1,7 δισεκατομμύρια χρήστες σε όλες τις πλατφόρμες - Android, iOS και Web (Inc., About Moovit, 2023).

Η εταιρεία τονίζει επίσης τη σημασία της χρήσης ανοιχτών δεδομένων σε πραγματικό χρόνο, καθώς παρέχει στους χρήστες μια σαφέστερη κατανόηση των συνθηκών μεταφοράς, διευκολύνοντας έτσι τον ευκολότερο προγραμματισμό του ταξιδιού (Bus-news, 2021.b). Οι (Arabghalizi & Labrinidis, 2019) υπογραμμίζουν τη σημασία της ενίσχυσης της ευφυΐας και της προβλεψιμότητας των δημόσιων συγκοινωνιών για την ενίσχυση της αξιοπιστίας, της ασφάλειας και της αποτελεσματικότητάς τους. Σύμφωνα με αυτόν τον στόχο, το Moovit έχει ενσωματώσει ένα API που υποδεικνύει τη διαθεσιμότητα ελεύθερου χώρου και προσβασιμότητας με αναπηρικά αμαξίδια σε οχήματα μεταφοράς, όπως λεωφορεία προσβάσιμα από αναπηρικά αμαξίδια (Inc., Moovit now shows how crowded your bus is in real-time, and if it's wheelchair accessible, 2021). Αυτή η δυνατότητα απλοποιεί τον προγραμματισμό ταξιδιού για άτομα με αναπηρίες και τους δίνει τη δυνατότητα να χρησιμοποιούν τα δημόσια μέσα μεταφοράς. Αυτή η λύση επιτρέπει σε άτομα με αναπηρία με αναπηρικά αμαξίδια να προγραμματίσουν ολόκληρο το ταξίδι τους, συμπεριλαμβανομένων των επιλογών οχημάτων, των σταθμών, των διαδρομών και των κατάλληλων γραμμών.

### 3.2.3.2 Αξιολόγηση εφαρμογής

Οι χρήστες επαίνεσαν την εφαρμογή Moovit για την εντυπωσιακή γκάμα χαρακτηριστικών και την ικανότητά της να παρέχει ολοκληρωμένες πληροφορίες μεταφοράς. Η φιλική προς το χρήστη διεπαφή της εφαρμογής έχει λάβει επίσης θετικά σχόλια καθώς οι χρήστες βρίσκουν την εφαρμογή εύκολη στην πλοήγηση και την κατανόηση, καθιστώντας την προσβάσιμη σε άτομα με διαφορετικά επίπεδα τεχνολογικής. Ο διαισθητικός σχεδιασμός και η σαφής διάταξη της διεπαφής χρήστη συμβάλλουν στην εμπειρία χρήστη, επιτρέποντας στους χρήστες να έχουν γρήγορη πρόσβαση στις πληροφορίες που χρειάζονται χωρίς καμία σύγχυση ή απογοήτευση. Επιπλέον, η δέσμευση του Moovit να ενσωματώνει ανοιχτά δεδομένα σε πραγματικό χρόνο έχει εκτιμηθεί ιδιαίτερα. Οι χρήστες εκτιμούν την ακρίβεια και την αξιοπιστία των παρεχόμενων πληροφοριών, δίνοντάς τους τη δυνατότητα να έχουν μια σαφέστερη εικόνα των συνθηκών μεταφοράς. Η ενσωμάτωση δεδομένων σε πραγματικό χρόνο βοηθά επίσης τους χρήστες να προγραμματίσουν τα ταξίδια τους πιο αποτελεσματικά, λαμβάνοντας υπόψη παράγοντες όπως οι τρέχουσες τοποθεσίες των οχημάτων, οι διακοπές λειτουργίας, ακόμη και τα επίπεδα συνωστισμού.

Από την άλλη, ορισμένοι χρήστες έχουν εκφράσει ανησυχίες για περιστασιακά προβλήματα απόδοσης με την εφαρμογή Moovit. Έχουν αναφερθεί περιπτώσεις αργών χρόνων φόρτωσης, καθυστερημένης απόκρισης και περιστασιακών απρόοπτων τερματισμών. Αν και αυτά τα ζητήματα δεν φαίνεται να είναι ευρέως διαδεδομένα, μπορεί να είναι απογοητευτικά για τους χρήστες που βασίζονται στην εφαρμογή για ενημερώσεις και προγραμματισμό μεταφοράς σε πραγματικό χρόνο. Η βελτίωση της απόδοσης και της σταθερότητας της εφαρμογής θα βελτιώσει τη συνολική εμπειρία χρήστη. Επιπλέον, ορισμένοι χρήστες έχουν προτείνει ότι το Moovit θα μπορούσε να επωφεληθεί από περαιτέρω επιλογές προσαρμογής. Ενώ η εφαρμογή παρέχει μια σειρά από επιλογές μεταφοράς, οι χρήστες θα ήθελαν περισσότερο έλεγχο στις εμφανιζόμενες πληροφορίες και εξατομίκευση της διεπαφής. Για παράδειγμα, οι χρήστες έχουν εκφράσει την επιθυμία για περισσότερη προσαρμογή χρωμάτων και δυνατότητα προσαρμογής ρυθμίσεων με βάση τις προσωπικές προτιμήσεις. Η ενσωμάτωση τέτοιων χαρακτηριστικών προσαρμογής θα επέτρεπε στους χρήστες να προσαρμόσουν την εφαρμογή στις συγκεκριμένες ανάγκες και προτιμήσεις τους.



Εικόνα 5 Στιγμιότυπα UI Moovit (Inc., About Moovit, 2023)

### 3.2.4 Εφαρμογή Citymapper

#### 3.2.4.1 Περιγραφή εφαρμογής

Το Citymapper είναι μια δημοφιλής εφαρμογή για κινητά που προσφέρει ολοκληρωμένες πληροφορίες συγκοινωνίας και υπηρεσίες πλοήγησης σε χρήστες σε διάφορες πόλεις παγκοσμίως. Ιδρύθηκε το 2011 και κυκλοφόρησε για να παρέχει δρομολόγια λεωφορείων για τους Ολυμπιακούς του Λονδίνου το 2012. Σύντομα, ξεπέρασε τα σύνορα του Λονδίνου και συνέχισε να περιλαμβάνει μία προς μία διαφορετικές πόλεις σε όλο τον κόσμο. Έως το 2023, είναι διαθέσιμο σε πάνω από 107 πόλεις και περιοχές (Citymapper, 2023). Το Citymapper παρέχει στους χρήστες πληροφορίες συγκοινωνίας σε πραγματικό χρόνο, σχεδιασμό διαδρομής και βήμα προς βήμα πλοήγηση για συστήματα δημόσιων μεταφορών, συμπεριλαμβανομένων λεωφορείων, τρένων, μετρό, τραμ και άλλων τρόπων μεταφοράς. Η εφαρμογή ενσωματώνει ένα ευρύ φάσμα λειτουργιών, συμπεριλαμβανομένης της δημιουργίας ταξιδιών που χρησιμοποιούν δεδομένα σε πραγματικό χρόνο. Μόλις ο χρήστης επιλέξει τις τοποθεσίες αναχώρησης και προορισμού, η εφαρμογή παρέχει πολλαπλές επιλογές για να φτάσει στον επιθυμητό προορισμό. Παράλληλα με τις προτάσεις διαδρομών, ο χρήστης έχει εναλλακτικές παραλλαγές, όπως διαδρομές μόνο για λεωφορεία ή επιλογές ασφαλείς για βροχή. Η εφαρμογή παρέχει επίσης πληροφορίες σχετικά με το πλαίσιο, όπως ο καιρός στον προορισμό. Η προβολή λεπτομερειών διαδρομής περιλαμβάνει μια εμφανή απεικόνιση χάρτη που δείχνει το επιλεγμένο δρομολόγιο ταξιδιού, συνοδευόμενο από έναν πίνακα που παραθέτει τα διάφορα τμήματα του ταξιδιού.

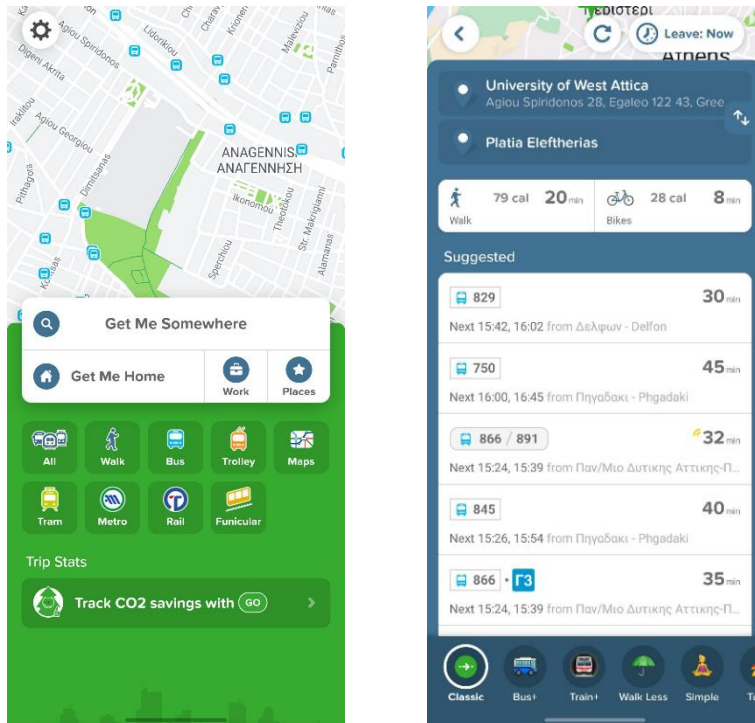
Επιπλέον, η εφαρμογή ενθαρρύνει την ενεργή συμμετοχή των χρηστών στην αξιολόγηση και διόρθωση των δεδομένων μίας διαδρομής. Η ενεργή συμμετοχή πραγματοποιείται όταν οι χρήστες παρέχουν σχόλια μέσω των εργαλείων που είναι ενσωματωμένα στη διεπαφή της εφαρμογής. Αρχικά, οι χρήστες μπορούν να αναφέρουν ζητήματα στην επιλεγμένη διαδρομή τους, επομένως, να βοηθήσουν στη διόρθωση και τον καθαρισμό των δεδομένων καθώς και στον αλγόριθμο δρομολόγησης σε περίπτωση που αντιμετωπίσουν κάποιο πρόβλημα. στη διαδρομή που εμφανίζεται από την εφαρμογή. Έπειτα, υπάρχει η λειτουργία βελτίωση δεδομένων, μέσω της οποίας οι χρήστες μπορούν να υποβάλλουν τους τύπους δεδομένων τα οποία το Citymapper δεν έχει πρόσβαση.

Η διεπαφή του Citymapper διαθέτει μια μεγάλη προβολή χάρτη που εμφανίζει το επιλεγμένο δρομολόγιο ταξιδιού. Αυτή η οπτική αναπαράσταση συμπληρώνεται από έναν πίνακα που παραθέτει τα επιμέρους μέρη του ταξιδιού, προσφέροντας λεπτομερή ανάλυση κάθε τμήματος. Ο συνδυασμός οπτικοποίησης χάρτη και λεπτομερειών δρομολογίου βελτιώνει την εμπειρία χρήστη και διευκολύνει τους χρήστες να πλοηγηθούν και να κατανοήσουν τις διαδρομές τους.



### 3.2.4.2 Αξιολόγηση εφαρμογής

Η εφαρμογή Citymapper έχει συγκεντρώσει σημαντικό αριθμό κριτικών χρηστών, παρέχοντας πολύτιμες πληροφορίες για τα δυνατά και τα αδύνατα σημεία της. Μεταξύ των θετικών που επισημάνθηκαν από τους χρήστες είναι η ακρίβεια της εφαρμογής στην παροχή πληροφοριών συγκοινωνίας σε πραγματικό χρόνο και η διαισθητική διεπαφή χρήστη της. Οι χρήστες εκτιμούν την ικανότητα της εφαρμογής να παρέχει έγκαιρες ενημερώσεις σχετικά με τις ώρες άφιξης, τις καθυστερήσεις και τις διακοπές των υπηρεσιών, δίνοντάς τους τη δυνατότητα να προγραμματίζουν αποτελεσματικά τα ταξίδια τους. Ο σχεδιασμός και η πλοήγηση του Citymapper διευκολύνουν τους χρήστες να εισάγουν τις τοποθεσίες τους, να έχουν πρόσβαση σε επιλογές συγκοινωνίας και να προβάλλουν λεπτομερείς πληροφορίες διαδρομής, συμβάλλοντας σε μια θετική εμπειρία χρήστη. Ωστόσο, η καμπύλη εκμάθησης που σχετίζεται με τις πολλές δυνατότητες της εφαρμογής έχει επίσης σημειωθεί από ορισμένους χρήστες, καθώς μπορεί να χρειαστεί χρόνος για να περιηγηθούν σε όλες τις επιλογές και να κατανοήσουν τις δυνατότητες της εφαρμογής.



Εικόνα 6 Στιγμιότυπα εφαρμογής Citymapper

### 3.3 Περιβάλλον Υλοποίησης

Σε αυτή την υπό-ενότητα γίνεται αναφορά σχετικά με το περιβάλλον υλοποίησης της εργασίας. Κατά τη διάρκεια του σταδίου ιδεών και πρωτοτύπων για την ανάπτυξη της εφαρμογής μεταφοράς MMM, αξιοποιήθηκε το εργαλείο Android Studio με χρήση της γλώσσας προγραμματισμού Java. Η τελική εφαρμογή έχει αναπτυχθεί με το εργαλείο Android Studio (Google, Android Studio IDE, 2022). Το Android Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE), ανεπτυγμένο από την Google, ειδικά σχεδιασμένο για την ανάπτυξη Android εφαρμογών. Παρέχει μια σταθερή βάση για τη δημιουργία ποιοτικών και αποτελεσματικών Android εφαρμογών καθώς παρέχει μια φιλική προς τον χρήστη διεπαφή, ισχυρό επεξεργαστή κώδικα, ένα ολοκληρωμένο σύνολο εργαλείων, την ενοποίηση με το Android SDK και τον εξομοιωτή και διαχείριση εικονικών και φυσικών συσκευών.



Εικόνα 7 Λογότυπο Android Studio (Google, Android Studio IDE, 2022)

Χρησιμοποιώντας το Android Studio, έγινε δυνατή η δημιουργία της αρχικής σχεδίασης της διεπαφής χρήστη (UI) της εφαρμογής μέσω ενός οπτικού επεξεργαστή (visual editor). Έτσι, είναι δυνατή η εύκολη τοποθέτηση στοιχείων διεπαφής χρήστη, ορισμός διατάξεων και να τακτοποίηση στοιχείων για την δημιουργία επιθυμητής εμφάνισης και αίσθησης της εφαρμογής. Το Android Studio προσφέρει επίσης ένα πλούσιο σύνολο προτύπων σχεδίασης, θεμάτων και στυλ που προσαρμόστηκαν ώστε να ταιριάζουν με τις απαιτήσεις της χρήστη της εφαρμογής.

Πίνακας 3 Πίνακας Στοιχείων Περιβάλλοντος ανάπτυξης

Στοιχεία περιβάλλοντος ανάπτυξης εργασίας
Installed Version: Android Studio Flamingo
Version: 2022.2.1 Patch 1
Language: Java
Runtime version: 17.0.6+0-b2043.56-9586694 amd64
Operating System: Windows 10 Education
Στοιχεία Device Emulator
Version: Pixel 6 with Google Play
API: 33
Android 13.0

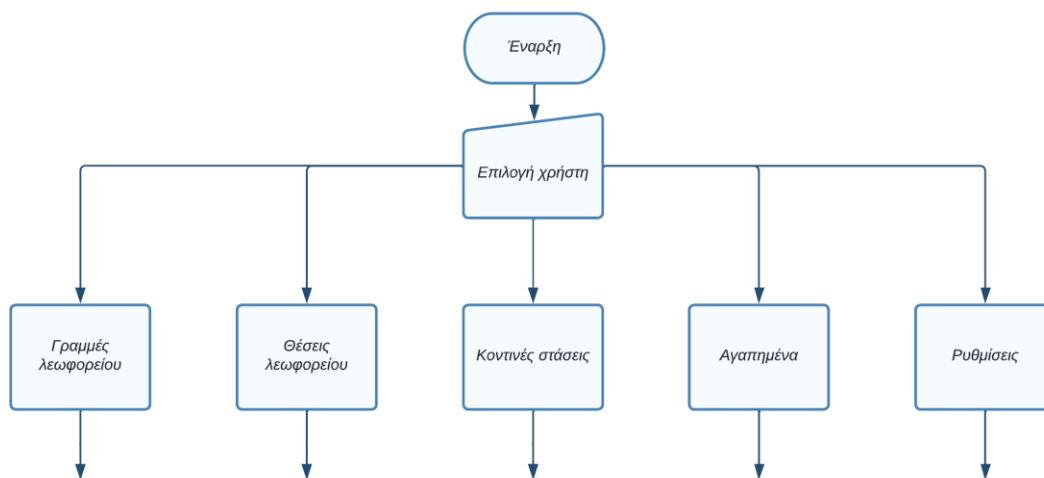
Επιπρόσθετα, για τη δημιουργία πρωτοτύπων και την προσομοίωση αλληλεπιδράσεων εφαρμογών, το Android Studio παρέχει έναν εξομοιωτή. Αυτός ο εξομοιωτής επιτρέπει την δοκιμή της εφαρμογής σε διάφορες εικονικές συσκευές Android, διασφαλίζοντας ότι η διεπαφή χρήστη και η λειτουργικότητα λειτουργούν όπως προβλέπεται σε διαφορετικά μεγέθη και αναλύσεις οθόνης. Με την εκτέλεση της εφαρμογής στον εξομοιωτή, συγκεντρώθηκαν πολύτιμες πληροφορίες, δοκιμάστηκαν βελτιώσεις και επαναληπτικές αλλαγές της σχεδίασης και της εμπειρίας χρήστη πριν τις πραγματικές δοκιμές συσκευής.

Όσον αφορά τη γλώσσα κωδικοποίησης, η Java (Arnold, Gosling, & Holmes, 2005) επιλέχθηκε ως η κύρια γλώσσα για την ανάπτυξη της εφαρμογής Android. Η Java αποτελεί μια ευέλικτη και ευρέως διαδεδομένη γλώσσα προγραμματισμού, φημισμένη για τη σταθερότητα, την απόδοση και την εκτεταμένη υποστήριξη βιβλιοθήκης. Χρησιμοποιήθηκε, συνεπώς, η Java για την ανάπτυξη της λογικής και της λειτουργικότητας της εφαρμογής μεταφοράς λεωφορείων, συμπεριλαμβανομένου του χειρισμού των αλληλεπιδράσεων των χρηστών, της ανάκτησης δεδομένων από API, της εφαρμογής λειτουργιών παρακολούθησης σε πραγματικό χρόνο και της διαχείρισης της διατήρησης δεδομένων σε βάσεις δεδομένων.

## 4 Αρχιτεκτονική και Ανάπτυξη Εφαρμογής

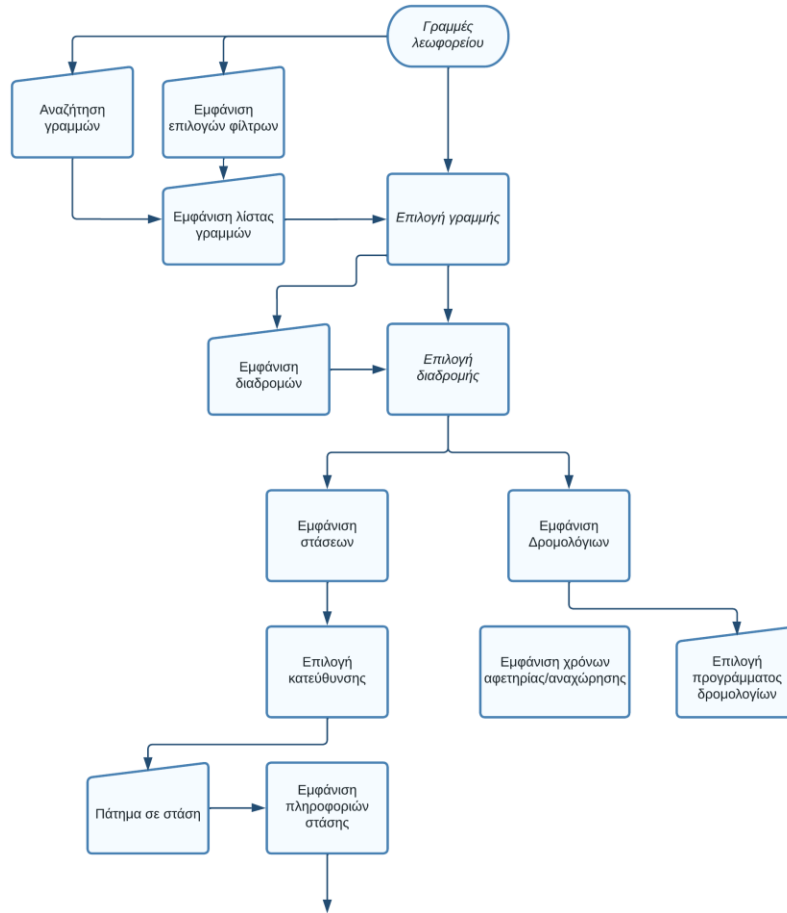
### 4.1 Αρχιτεκτονική και σχεδιασμός

Ο σχεδιασμός του διαγράμματος ροής εφαρμογών για την εφαρμογή μας περιλαμβάνει τη δομή της πλοήγησης και των αλληλεπιδράσεων του χρήστη εντός της εφαρμογής. Το διάγραμμα ροής ξεκινά με την κύρια οθόνη ως σημείο εισόδου της εφαρμογής. Από εκεί, οι χρήστες παρουσιάζονται με διάφορα χαρακτηριστικά οργανωμένα σε διακριτές ενότητες. Κάθε χαρακτηριστικό αντιπροσωπεύει μια συγκεκριμένη λειτουργικότητα ή πληροφορίες με τις οποίες μπορούν να αλληλεπιδράσουν οι χρήστες.



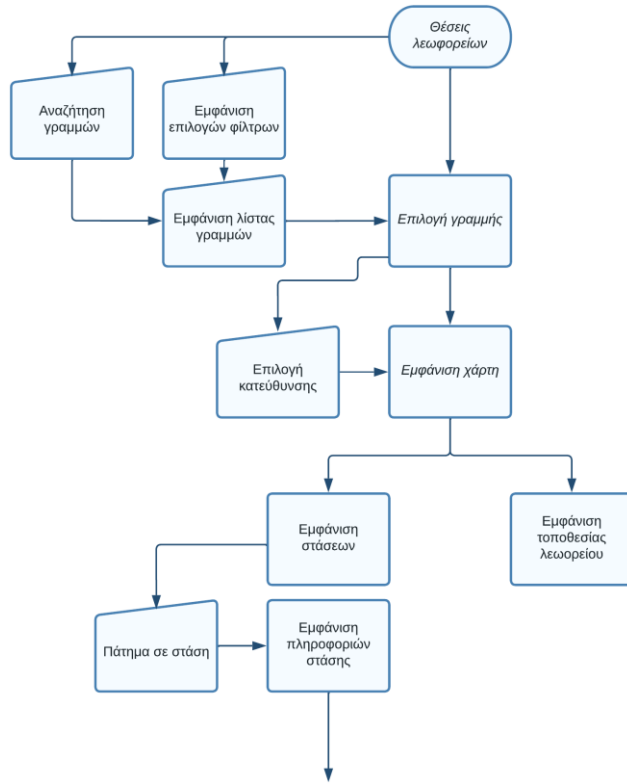
Διάγραμμα 1 Διάγραμμα ροής εφαρμογής αρχικής οθόνης

Για την δημιουργία των διαγραμμάτων ροής, δόθηκε προσοχή στη λογική ακολουθία των ενεργειών και των μεταβάσεων μεταξύ των οθονών. Το Διάγραμμα 2 ξεκινά με την επιλογή εμφάνισης και αναζήτησης γραμμών λεωφορείου, επιτρέποντας στους χρήστες να έχουν πρόσβαση σε πληροφορίες σχετικά με τις στάσεις και τα δρομολόγια λεωφορείων και τρόλεϊ. Αυτή η ροή ακολουθείται από την επιλογή γραμμής και διαδρομής. Στην συνέχεια, προβάλλονται οι στάσεις της γραμμής και δρομολόγια από την αφετηρία και προς το τέρμα.

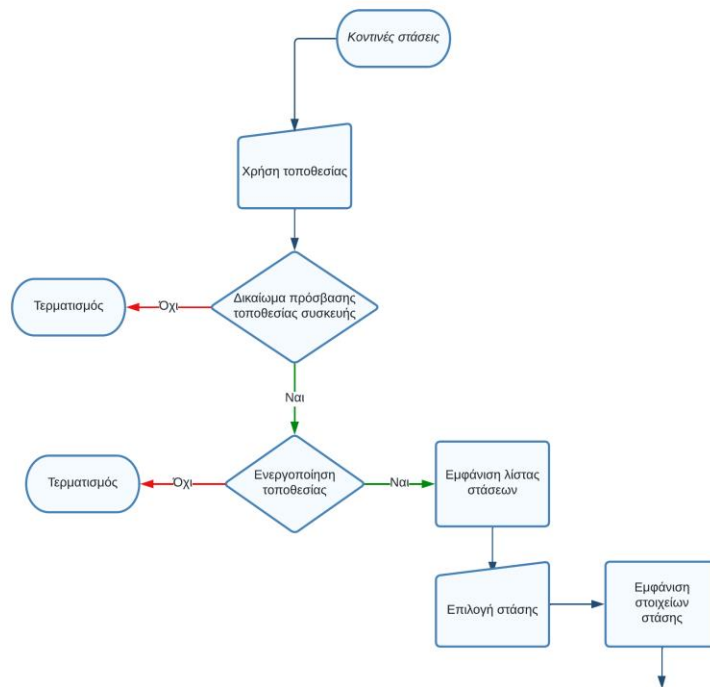


Διάγραμμα 2 Διάγραμμα ροής γραμμών λεωφορείου

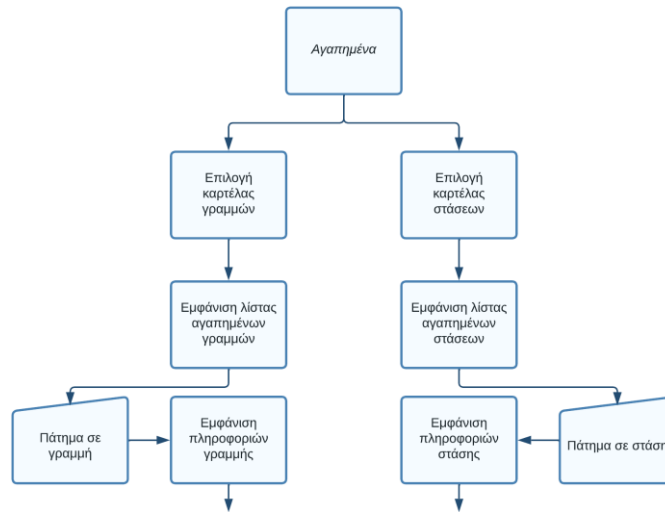
Το Διάγραμμα 3 εμφανίζει ένα άλλο σημαντικό χαρακτηριστικό που είναι η δυνατότητα εμφάνισης θέσεων και στάσεων λεωφορείων στο χάρτη. Αυτή η δυνατότητα παρέχει μια οπτική αναπαράσταση των θέσεων του λεωφορείου στο χάρτη και βοηθά τους χρήστες να εντοπίσουν το λεωφορείο και τις στάσεις της γραμμής.



Διάγραμμα 3 Διάγραμμα ροής θέσης λεωφορείων

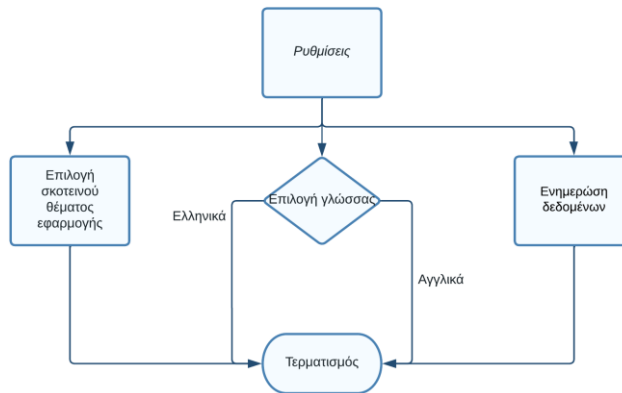


Διάγραμμα 4 Διάγραμμα ροής κοιμητικών στάσεων

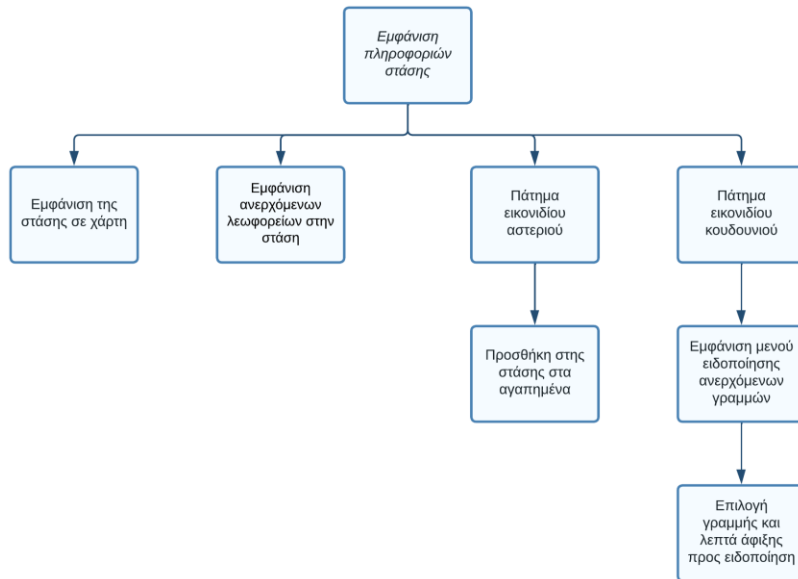


Διάγραμμα 5 Διάγραμμα ροής επιλογής αγαπημένων

Το Διάγραμμα 5 περιλαμβάνει τη δυνατότητα προβολής αγαπημένων γραμμών και στάσεων, επιτρέποντας στους χρήστες να έχουν πιο γρήγορη πρόσβαση στις πληροφορίες που χρησιμοποιούνται συχνά. Επιπλέον το Διάγραμμα 6, εμφανίζει την ενότητα ρυθμίσεων παρέχει επιλογές προσαρμογής, όπως αλλαγή του θέματος από σκοτεινή σε φωτεινή λειτουργία και επιλογή της προτιμώμενης γλώσσας. Αυτές οι επιλογές καλύπτουν τις μεμονωμένες προτιμήσεις των χρηστών.



Διάγραμμα 6 Διάγραμμα ροής ρυθμίσεων



Διάγραμμα 7 Διάγραμμα ροής εμφάνισης στάσης

Το Διάγραμμα 7 εμφανίζει την διαδικασία εμφάνισης πληροφοριών μίας στάσης. Εμφανίζεται αρχικά η στάση στον χάρτη και οι χρόνοι άφιξης των γραμμών στη συγκεκριμένη στάση. Ο χρήστης, έτσι, μπορεί σε πραγματικό χρόνο να παρακολουθεί τις ώρες άφιξης σε μια στάση που έχει επιλέξει. Επιπλέον, μπορεί να επιλέξει την αποθήκευση της στάσης στα αγαπημένα και την ενεργοποίηση ειδοποιήσεων κατά την άφιξη μίας επιλεγμένης γραμμής.

Στην συνέχεια εμφανίζουμε την αρχιτεκτονική του ευφυή πράκτορα που χρησιμοποιήθηκε για την προσαρμογή του UI. Το διάγραμμα ροής ξεκινά με τον κόμβο έναρξης. Ο πράκτορας συλλέγει σχετικές μετρήσεις από την εφαρμογή για κινητά. Αυτές οι μετρήσεις μπορεί να περιλαμβάνουν αλληλεπιδράσεις χρηστών, μοτίβα χρήσης, προτιμήσεις ή οποιαδήποτε άλλα σημεία δεδομένων που παρέχουν πληροφορίες για τη συμπεριφορά και το περιβάλλον του χρήστη. Ο πράκτορας αναλύει τις μετρήσεις που συλλέγονται για την κατανόηση της συμπεριφοράς του χρήστη. Αυτή η ανάλυση βοηθά στον καθορισμό των κατάλληλων ενεργειών που πρέπει να γίνουν με βάση προκαθορισμένους κανόνες. Ο πράκτορας αντιστοιχίζει τις μετρήσεις με τους καθορισμένους κανόνες για να προσδιορίσει ποιους κανόνες ικανοποιούνται με βάση το τρέχον πλαίσιο. Εάν ένας κανόνας ταιριάζει, ο πράκτορας ενεργοποιεί τη σχετική ενέργεια. Η ενέργεια μπορεί να είναι οποιαδήποτε προκαθορισμένη εργασία ή συμπεριφορά που εκτελεί ο πράκτορας εντός της εφαρμογής. Το διάγραμμα ροής επιστρέφει στο βήμα «Συλλογή μετρικών» για τη συνεχή συλλογή και ανάλυση μετρήσεων καθώς ο χρήστης αλληλεπιδρά με την εφαρμογή.





Διάγραμμα 8 Διάγραμμα ροής ευφυή πράκτορα

## 4.2 Υλοποίηση και Ανάπτυξη

Η εφαρμογή τηλεματικής ΟΑΣΑ αναπτύχθηκε ως παράδειγμα της ανάπτυξης ανθρωποκεντρικής εφαρμογής με χρήση τεχνικών τεχνητής νοημοσύνης για την δημιουργία ενός προσαρμοστικού UI. Η εφαρμογή υπηρεσιών της τηλεματική του ΟΑΣΑ δημιουργήθηκε ως ένα προσαρμοστικό, ανθρωποκεντρικό λογισμικό υποστηρίζοντας δύο βασικούς τύπους προσαρμογής πληροφορίας, οπτικοποίησης (παρουσίαση) και προσαρμογής της διεπαφής χρήστη.

### 4.2.1 Χαρακτηριστικά εφαρμογής

Τα βασικά χαρακτηριστικά της εφαρμογής χωρίζονται στις παρακάτω λειτουργίες:

- Εμφάνιση και αναζήτηση γραμμών και πρόγραμμα δρομολογίων για όλα τα λεωφορεία και τρόλεϊ του ΟΑΣΑ. Ο χρήστης μπορεί να αναζητήσει την γραμμή και κατεύθυνση που επιθυμεί και να παρακολουθήσει τις στάσεις και τα δρομολόγια της.
- Εμφάνιση χρόνων άφιξης γραμμών από συγκεκριμένη στάση και τοποθεσία στάσης σε χάρτη. Ο χρήστης μπορεί σε πραγματικό χρόνο να παρακολουθήσει τους χρόνους άφιξης σε μία στάση που έχει επιλέξει.
- Εμφάνιση θέσεων λεωφορείων και στάσεων στον χάρτη μίας συγκεκριμένης γραμμής. Εμφανίζονται και ενημερώνονται σε πραγματικό χρόνο θέσεις των λεωφορείων μίας κατεύθυνσης μίας γραμμής.
- Αποθήκευση αγαπημένων γραμμών και στάσεων. Ο χρήστης μπορεί είτε από τη λίστα των γραμμών είτε από τη δραστηριότητα επιλεγμένης στάσης να αποθηκεύσει τις γραμμές και στάσεις αντίστοιχα. Οι αποθηκευμένες βρίσκονται στην λειτουργία “αγαπημένα”.
- Ρυθμίσεις για αλλαγές της εφαρμογής κατά την προτίμηση του χρήστη. Οι ρυθμίσεις που παρέχονται είναι οι παρακάτω:
- Αλλαγή θέματος Σκούρο/Ανοιχτό: Αλλαγή εμφάνισης όλων των στοιχείων της εφαρμογής σε σκούρο ή ανοιχτό θέμα που προτιμά ο χρήστης.
- Επιλογή γλώσσας μεταξύ αγγλικών και ελληνικών: Αλλαγή της γλώσσας των κειμένων της εφαρμογής καθώς και των γραμμών και στάσεων.
- Ενημέρωση δεδομένων: Χειροκίνητη ενημέρωση των γραμμών για τυχόν αλλαγές.

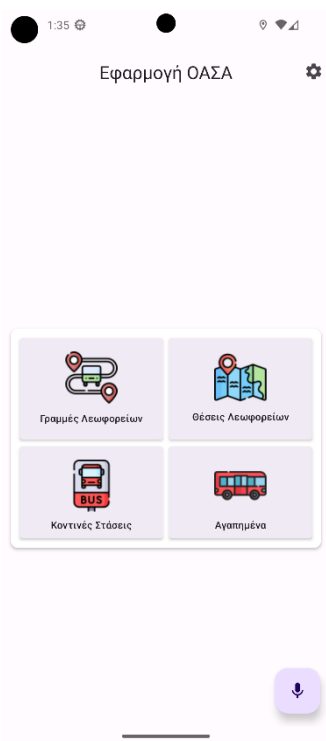
Παρακάτω περιγράφονται περιληπτικά οι βασικές τεχνικές προδιαγραφές που υλοποιήθηκαν στην εφαρμογή είναι οι εξής:

#### 4.2.2 Διεπαφή χρήστη

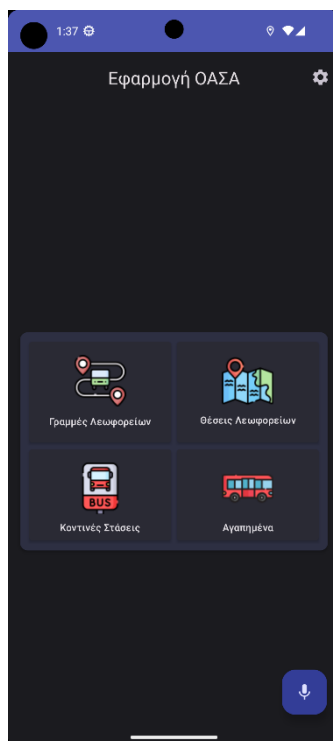
Η διεπαφή του χρήστη αποτελείται από 8 δραστηριότητες (Activities) και 4 fragments για την πλοήγηση του χρήστη κατά την εκτέλεση των λειτουργιών της εφαρμογής καθώς και για την αλλαγή ορισμένων ρυθμίσεων.

##### 4.2.2.1 Δραστηριότητες

Το *MainActivity* αποτελεί την αρχική διεπαφή του χρήστη με την εφαρμογή που περιέχει έναν πίνακα με κουμπιά για κάθε λειτουργία καθώς και ένα κουμπί ρυθμίσεων στο Toolbar. Το *MainActivity* αφορά την κεντρική οθόνη της εφαρμογής που περιέχει όλες τις λειτουργίες που παρέχονται. Η διεπαφή αποτελείται από ένα *CoordinatorLayout* που περιέχει ένα *AppBarLayout* και *MaterialToolbar* για την εμφάνιση της πάνω μπάρας με τον τίτλο της εφαρμογής και ένα κουμπί μενού για της ρυθμίσεις. Έπειτα, υπάρχει ένα *LinearLayout* που περιέχει ένα *TableLayout* με κάθε *TableRow* να περιέχει ένα *MaterialCardView* με μία εικόνα και έναν τίτλο μέσα σε ένα *RelativeLayout*.

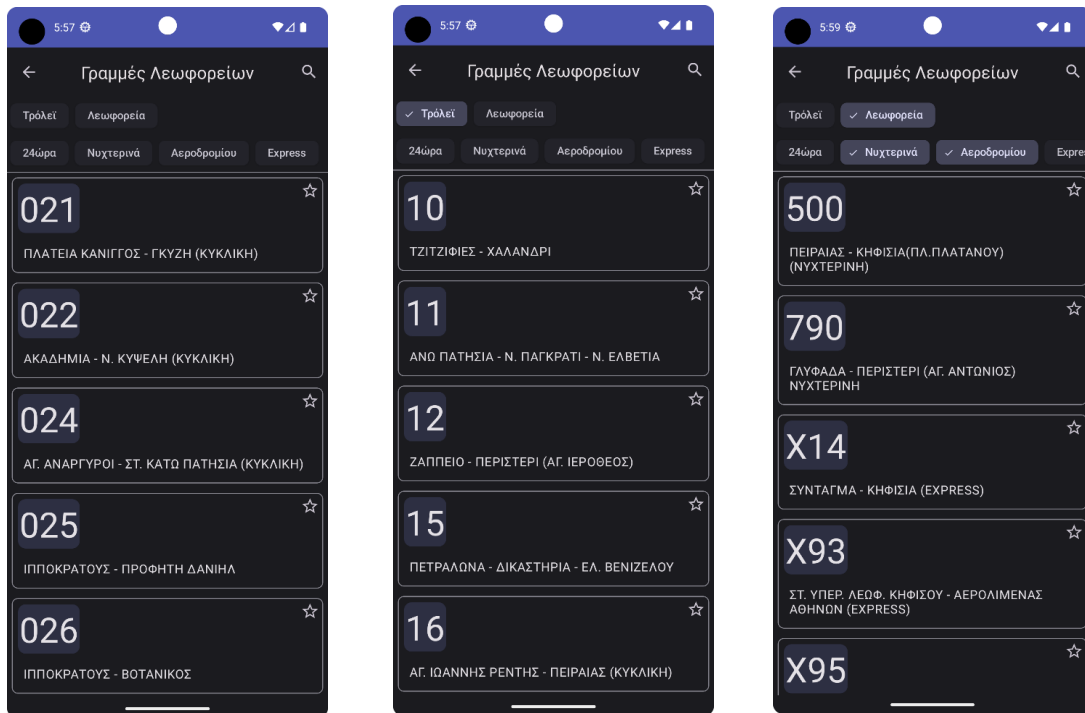


Εικόνα 9 MainActivity Layout (Ανοιχτό θέμα)



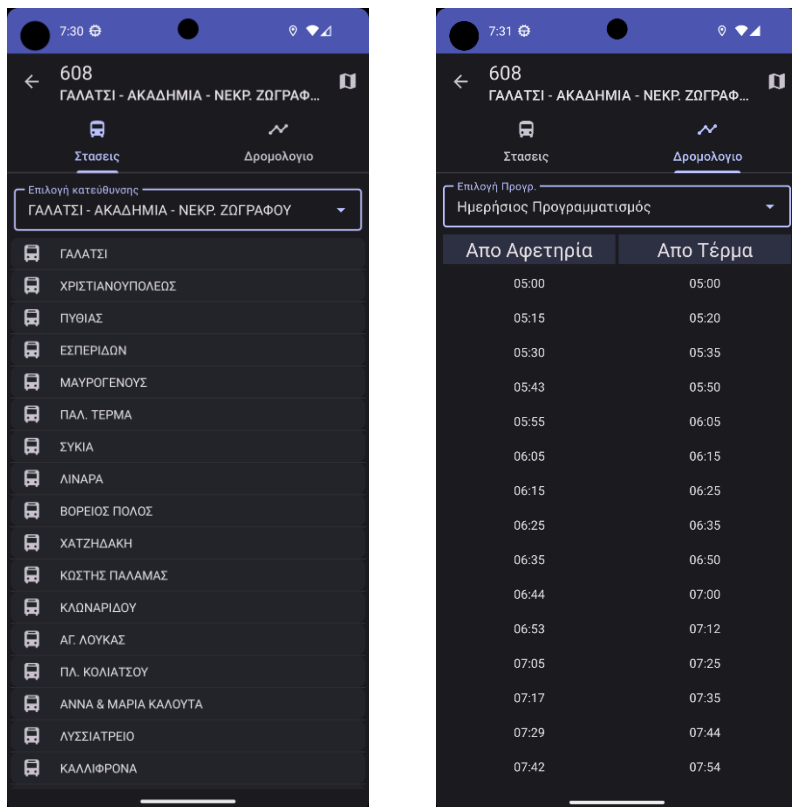
Εικόνα 8 MainActivity Layout (Σκούρο θέμα)

Το *BusLineActivity* περιέχει μία scrollable λίστα (*RecyclerView*) με τις γραμμές λεωφορείων (*custom RecyclerViewAdapter*). Επίσης, υπάρχουν δύο σειρές με chip κουμπιά για την εφαρμογή φίλτρου των γραμμών λεωφορείου (λεωφορεία, τρόλεϊ, νυχτερινές, 24ωρες, εξπρές και αεροδρομίου). Επιπλέον, στο *Toolbar* υπάρχει η δυνατότητα της αναζήτησης των γραμμών (*filter*). Αν επιλέξει ένα αντικείμενο της λίστας εμφανίζεται ένα *Dialog* κουτί με τις διαθέσιμες διαδρομές της γραμμής. Όταν ο χρήστης επιλέξει μία διαδρομή μεταβαίνει στο *RouteActivity*. Το *BusLineActivity* περιέχει μία λίστα από αντικείμενα *Line* που εμφανίζονται στο *RecyclerView* μέσω του *LinesAdapter*. Η μεταβλητή *ProgressBar* χρησιμοποιείται για να εμφανιστεί ένα στρογγυλό *process bar* κατά την διάρκεια του *API request*. Η κλήση στο *API* για τις πληροφορίες γραμμών περιλαμβάνεται στο Παράρτημα - Κώδικας 11 Ανάκτηση στοιχείων γραμμών από *API*.



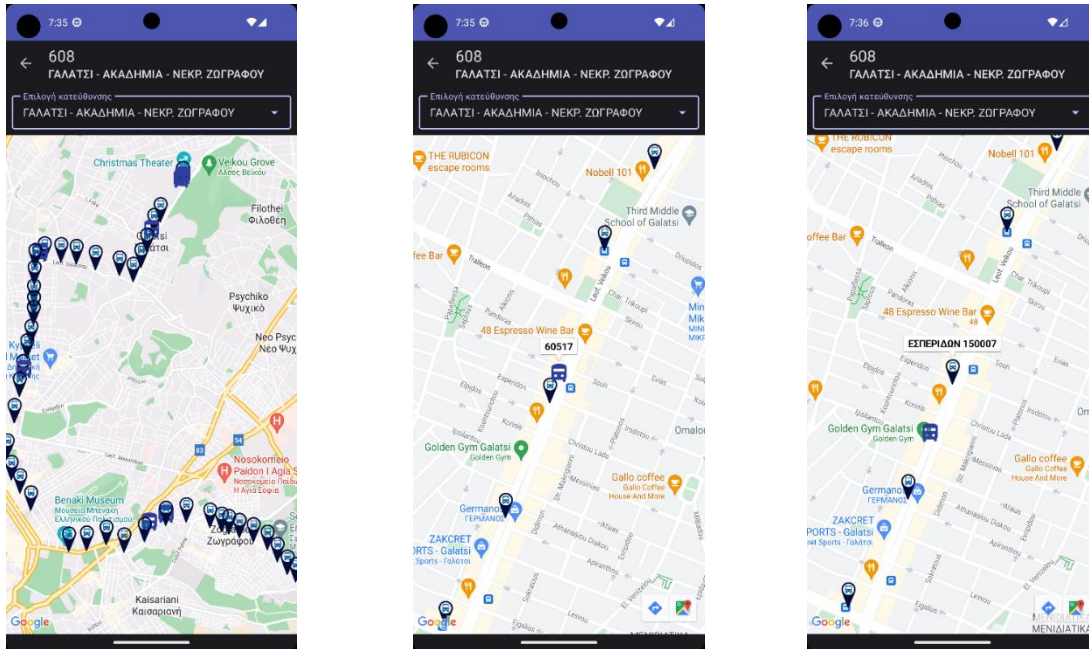
Εικόνα 10 Στιγμιότυπο λίστας γραμμών λεωφορείων

Η *RouteActivity* δραστηριότητα που περιέχει όλες τις πληροφορίες σχετικά με την γραμμή που έχει επιλέξει ο χρήστης. Στην δραστηριότητα εμφανίζεται ένα *ViewPager* με τις στάσεις (*StopsFragment*) και ένα με στοιχεία των δρομολογίων (*ScheduleFragment*). Το *StopsFragment* περιέχει ένα *AutoCompleteTextView* για την επιλογή της κατεύθυνσης και μία λίστα από τις στάσεις (*custom ListView*). Ο χρήστης μπορεί να επιλέξει μία στάση και να μεταβεί στο *StopActivity*. Στο *ScheduleFragment* εμφανίζονται δύο πίνακες με τα δρομολόγια της γραμμής από την αφετηρία και από το τέρμα.



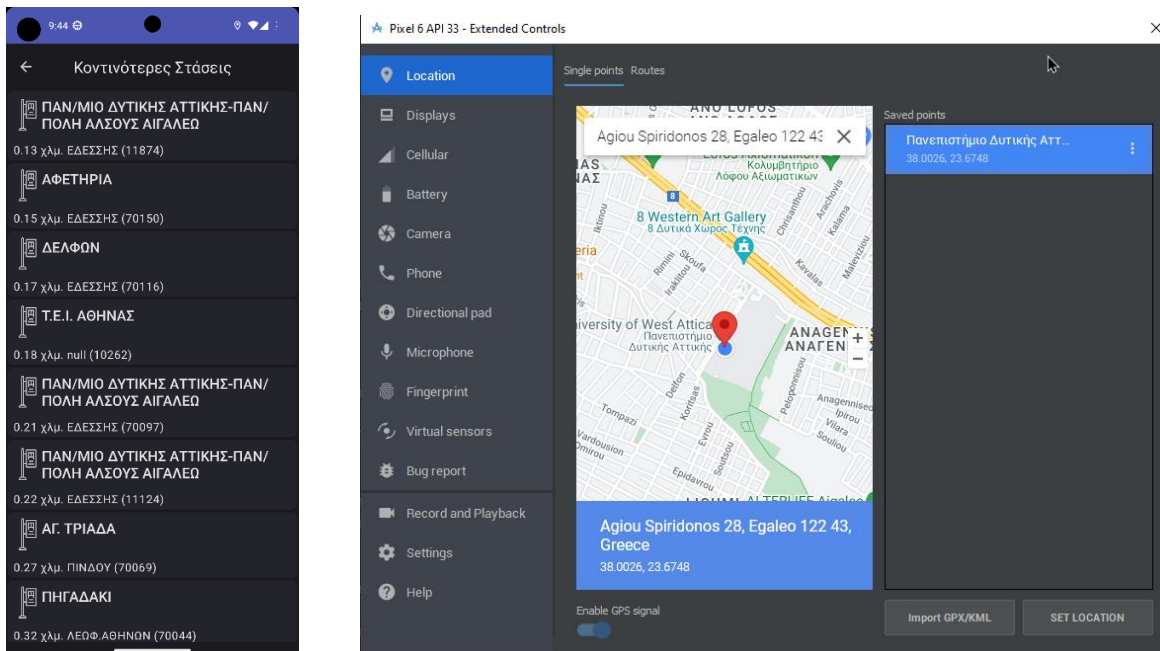
Εικόνα 11 Στιγμιότυπο εμφάνισης στάσεων και δρομολογίων γραμμής

Η δραστηριότητα *BusLocations* εμφανίζει τις στάσεις και τα λεωφορεία μίας γραμμής στο χάρτη. Ο χρήστης επιλέγει την κατεύθυνση της γραμμής μέσω ενός *AutoCompleteTextView* με τις διαθέσιμες κατευθύνσεις. Επιπλέον, αν ο χρήστης πατήσει στον τίτλο μίας στάσης θα μεταβεί στην *StopActivity* δραστηριότητα που εμφανίζει τα λεωφορεία που φτάνουν στην στάση. Η κλάση *BusLocations* περιέχει μεταβλητές για το *GoogleMap*, την επιλεγμένη διαδρομή το κωδικό γραμμής και ορισμένες λίστες με αντικείμενα «*Stop*». Επιπλέον, περιέχει ένα *HashMap* με «*Marker, Stop*» για τα σημεία των στάσεων στον χάρτη και ένα *HashMap* με «*Marker*» και «*LanLng*» για τις συντεταγμένες των λεωφορείων στον χάρτη. Τέλος, περιέχει μία μεταβλητή *Timer* για την συνεχής ενημέρωση της τοποθεσίας των λεωφορείων.



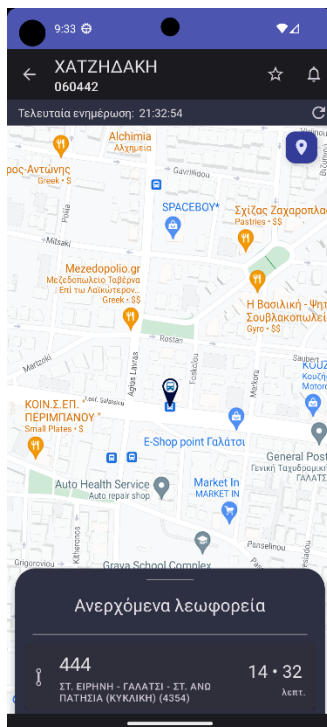
Εικόνα 12 Στιγμιότυπα εμφάνισης θέσης λεωφορείου στον χάρτη

Το *ClosestStops* εμφανίζει μία λίστα από τις κοντινότερες στάσεις με βάση την τοποθεσία του χρήστη (location). Ο χρήστης μπορεί να πατήσει όποιο αντικείμενο της λίστας και να μεταβεί στην *StopActivity*. Η δραστηριότητα εκτελεί το API request *getClosestStops* με παραμέτρους τις συντεταγμένες του χρήστη και εμφανίζει μία λίστα με τα *Stops*. Η διαδικασία ανάκτησης της τοποθεσίας του χρήστη είναι παρόμοια με την δραστηριότητα *StopActivity*.

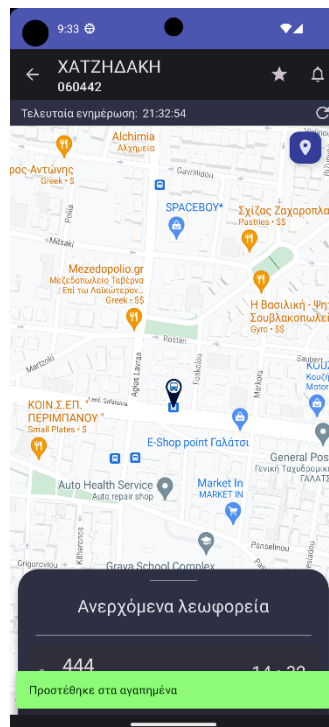


Η δραστηριότητα *StopActivity* εμφανίζει την στάση στον χάρτη και με ένα bottom sheet που εμφανίζονται τα λεωφορεία και οι χρόνοι άφιξης τους στην στάση. Η δραστηριότητα μπορεί να εκτελεστεί από τον πίνακα στάσεων στην *RouteActivity*, από το label μίας στάσης στον χάρτη του *BusLocations*, από την *ClosestStops* και από την λίστα αγαπημένων *FavStopsFragment*.

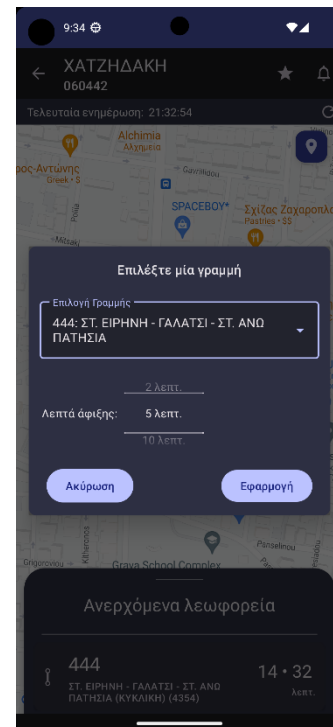
Το layout της δραστηριότητας περιέχει δύο βασικά στοιχεία, το *Map Fragment* και το *BottomSheet Fragment*. Επιπλέον, υπάρχει ένα *MaterialToolBar* που εμφανίζει το όνομα της στάσης και τον κωδικό και περιέχει δύο επιλογές μενού. Η πρώτη επιλογή χρησιμοποιείται για την προσθήκη της στάσης στα αγαπημένα και η δεύτερη ανοίγει ένα *Dialog Box* για την ενημέρωση άφιξης λεωφορείου με *Notification*. Κάτω από το *TopBar* υπάρχει ένα layout που εμφανίζει την ώρα της τελευταίας ενημέρωσης και ένα κουμπί για την ενημέρωση των δεδομένων άφιξης. Τέλος, υπάρχει ένα *FloatingActionButton* που χρησιμοποιείται για την εμφάνιση της τοποθεσίας του χρήστη στον χάρτη.



Εικόνα 13 Εμφάνιση στάσης και λεωφ.

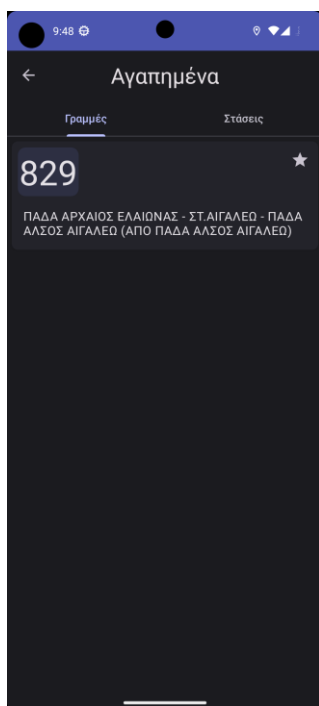


Εικόνα 14 Προσθήκη στάσης στα αγαπημένα

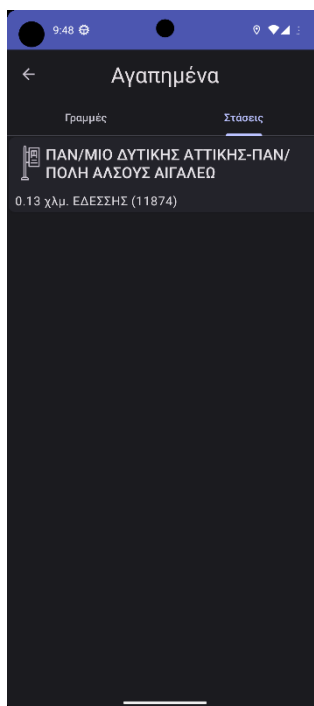


Εικόνα 15 Λειτουργία ειδοποίησης άφιξης λεωφ.

Το FavouriteActivity περιέχει ένα ViewPager από μία σελίδα για τις αγαπημένες γραμμές και την άλλη για τις αγαπημένες στάσεις του χρήστη. Από αυτή τη δραστηριότητα μπορεί να μεταβεί στις βασικές δραστηριότητες των στάσεων και γραμμών (StopActivity, RouteActivity). Το Fragment FavLinesFragment περιέχει ένα RecyclerView με τα αντικείμενα Line που βρίσκονται στην λίστα αγαπημένων του χρήστη. Κατά την δημιουργία του Fragment καλεί την μέθοδο isLanguage από την δραστηριότητα FavouriteActivity για την εμφάνιση των στοιχείων στην κατάλληλη γλώσσα. Η δραστηριότητα δημιουργεί ένα DatabaseHelper αντικείμενο και εκχωρεί τις γραμμές λεωφορείων που είναι αποθηκευμένες στην SQLite βάση. Αν η βάση δεν περιέχει δεδομένα εκτελεί το API request (webGetLinesWithMLInfo) για τις διαθέσιμες διαδρομές των αγαπημένων γραμμών κατά το πάτημα μίας γραμμής της λίστας.



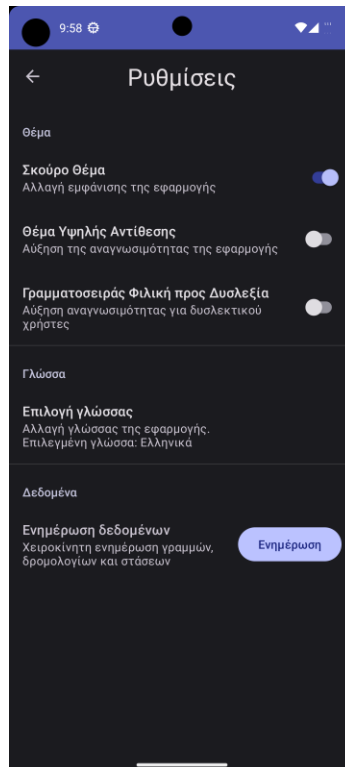
Εικόνα 17 Εμφάνιση αγαπημένων γραμμών



Εικόνα 16 Εμφάνιση αγαπημένων στάσεων



Η SettingsActivity περιέχει ένα settings\_layout.xml με τα Root Preferences του χρήστη. Η δραστηριότητα δίνει την δυνατότητα στον χρήστη να κάνει κάποιες αλλαγές, όπως, αλλαγή θέματος και αλλαγή γλώσσας. Ακόμη, υπάρχει η επιλογή ενημέρωσης των αλλαγών ώστε ο χρήστης να μπορεί να δει το αποτέλεσμα της αλλαγής που έκανε, έτσι ώστε να είναι σίγουρος για την επιλογή του. Τα τρία χαρακτηριστικά της δραστηριότητας αυτής λειτουργούν χρησιμοποιώντας την κλάση SettingsFragment, όπου ανάλογα την αλλαγή ή την ενημέρωση που θέλει να εκτελέσει ο χρήστης.



Εικόνα 18 Εμφάνιση ρυθμίσεων

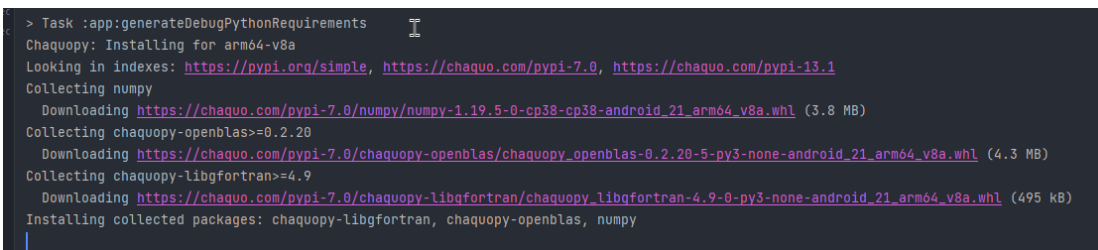
### 4.2.3 Ενσωμάτωση Python SDK

Για την δημιουργία και εκτέλεση python scripts χρησιμοποιήθηκε το Chaquopy 14.0. Το Chaquopy είναι ένα ισχυρό εργαλείο που επιτρέπει την ενσωμάτωση της γλώσσας Python σε μια εφαρμογή Android, επιτρέποντας στους προγραμματιστές να αξιοποιήσουν το εκτεταμένο οικοσύστημα Python για τα έργα τους Android (Ltd, n.d.). Το Chaquopy ενσωματώνεται με το Android Studio, παρέχοντας υποστήριξη για επεξεργασία κώδικα, εντοπισμό σφαλμάτων και άλλα εργαλεία ανάπτυξης. Αυτό βελτιστοποιεί τη διαδικασία ανάπτυξης και επιτρέποντας γνωστές ροές εργασίας. Το Chaquopy βελτιστοποιεί την εκτέλεση του κώδικα Python σε συσκευές Android χρησιμοποιώντας τη συλλογή Just-In-Time (JIT) και άλλες βελτιώσεις απόδοσης. Αυτό εξασφαλίζει αποτελεσματική εκτέλεση και ελαχιστοποιεί τα έξοδα, παρέχοντας μια απρόσκοπτη εμπειρία χρήστη.

Ένα από τα αρχικά βήματα στη χρήση του Chaquopy είναι η επεξεργασία των ιδιοτήτων Gradle για τον καθορισμό της επιθυμητής έκδοσης Python (version 3.8) και η εγκατάσταση των βιβλιοθηκών pandas και scikit-learn μέσω pip που θα χρησιμοποιηθούν.

```
plugins {  
    id 'com.android.application' version '7.1.3' apply false  
    id 'com.android.library' version '7.1.3' apply false  
    id 'com.google.android.libraries.mapsplatform.secrets-gradle-plugin' version '2.0.1' apply false  
    id 'com.chaquo.python' version '14.0.2' apply false  
}
```

Μόλις αποθηκευτούν οι αλλαγές, το Chaquopy κατεβάζει αυτόματα και εγκαθιστά τις καθορισμένες βιβλιοθήκες Python χρησιμοποιώντας pip κατά τη διαδικασία κατασκευής. Αυτό διασφαλίζει ότι οι απαιτούμενες εξαρτήσεις Python είναι άμεσα διαθέσιμες για χρήση της εφαρμογής σας Android.



```
> Task :app:generateDebugPythonRequirements  
Chaquopy: Installing for arm64-v8a  
Looking in indexes: https://pypi.org/simple, https://chaquo.com/pypi-7.0, https://chaquo.com/pypi-13.1  
Collecting numpy  
  Downloading https://chaquo.com/pypi-7.0/numpy/numpy-1.19.5-0-cp38-cp38-android\_21\_arm64\_v8a.whl (3.8 MB)  
Collecting chaquopy-openblas=0.2.20  
  Downloading https://chaquo.com/pypi-7.0/chaquopy-openblas/chaquopy\_openblas-0.2.20-5-py3-none-android\_21\_arm64\_v8a.whl (4.3 MB)  
Collecting chaquopy-libgfortran>=4.9  
  Downloading https://chaquo.com/pypi-7.0/chaquopy-libgfortran/chaquopy\_libgfortran-4.9-0-py3-none-android\_21\_arm64\_v8a.whl (495 kB)  
Installing collected packages: chaquopy-libgfortran, chaquopy-openblas, numpy
```

Εικόνα 19 Στιγμιότυπο εγκατάστασης βιβλιοθηκών

Ενώ το Chaquopy προσφέρει πολλά πλεονεκτήματα, υπάρχουν ορισμένα μειονεκτήματα ή περιορισμούς κατά τη χρήση του εργαλείου. Αρχικά, ενώ το Chaquopy παρέχει βελτιστοποιήσεις όπως η μεταγλώττιση JIT για τη βελτίωση της απόδοσης του κώδικα Python σε συσκευές Android, δεν πετυχαίνει το ίδιο επίπεδο απόδοσης με τον κώδικα Java. Η Python, ως γλώσσα ερμηνείας, μπορεί να εισάγει περιορισμούς

επιβάρυνσης και απόδοσης σε σύγκριση με τον εξαιρετικά βελτιστοποιημένο μεταγλωττισμένο κώδικα. Επίσης η ενσωμάτωση της Python με το Chaquopy απαιτεί τη συμπερίληψη του χρόνου εκτέλεσης της Python και των εξαρτήσεων εντός του πακέτου APK. Αυτό οδηγεί στην αύξηση του μεγέθους της εφαρμογής, η οποία μπορεί να επηρεάσει τους χρόνους λήψης και εγκατάστασης, καθώς και τις απαιτήσεις αποθήκευσης συσκευής. Το πακέτο APK της αρχική εφαρμογής είχε μέγεθος 14.8 Mb ενώ με την ενσωμάτωση της Python το μέγεθος έφτασε τα 168 Mb.

#### 4.2.4 Ανάπτυξη Front-End

Η ανάπτυξη του front-end είναι ήταν σπουδαίας σημασίας για την δημιουργία της ανθρωποκεντρικής εφαρμογής μας με προσαρμοστικό περιβάλλον εργασίας χρήστη. Το κεφάλαιο αυτό εστιάζει στην υλοποίηση στοιχείων διεπαφής χρησιμοποιώντας το Android Studio και Material Design. Ακολουθώντας αυτές τις οδηγίες, η διεπαφή της εφαρμογής στοχεύει να προσφέρει μια ευχάριστη εμπειρία χρήστη σε διαφορετικές συσκευές και μεγέθη οθόνης.

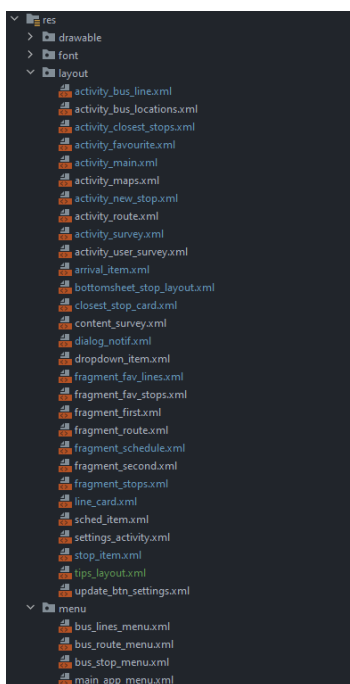
##### 4.2.4.1 Σχεδιασμό Διάταξης και Πόροι XML

Αυτό το κεφάλαιο εμβαθύνει στη διαδικασία σχεδιασμού ευέλικτων διατάξεων και στην αποτελεσματική χρήση των πόρων XML στο Android Studio. Με την κατανόηση των αρχών και των τεχνικών που συζητούνται σε αυτό το κεφάλαιο, οι προγραμματιστές μπορούν να δημιουργήσουν οπτικά ελκυστικές και φιλικές προς το χρήστη διεπαφές που προσαρμόζονται απρόσκοπτα σε διαφορετικά μεγέθη και προσανατολισμούς οθόνης.

Μία από τις βασικές πτυχές του σχεδιασμού διάταξης είναι ο ορισμός στοιχείων UI με τη χρήση πόρων XML. Το Android Studio παρέχει έναν οπτικό επεξεργαστή διάταξης που επιτρέπει την εύκολη προσθήκη στοιχείων UI στην οθόνη και την προσαρμογή των ιδιοτήτων τους. Μέσω της χρήσης ιδιοτήτων XML, καθορίζεται η συμπεριφορά, εμφάνιση και τοποθεσία κάθε στοιχείου UI. Αυτή η προσέγγιση εξασφαλίζει σαφή διαχωρισμό μεταξύ του επιπέδου παρουσίασης και της λογικής της εφαρμογής, καθιστώντας τον κώδικα ευκολότερο στη διαχείριση και τη συντήρηση.

Για τη δημιουργία ευέλικτων διατάξεων (layouts), λαμβάνεται υπόψη η ποικιλία μεγεθών και προσανατολισμών οθόνης στις οποίες θα εμφανίζεται η εφαρμογή τους. Εφαρμόζονται περιορισμοί και οδηγίες διάταξης για να βοηθήσει στην επίτευξη του προσαρμοστικού σχεδιασμού. Οι περιορισμοί επιτρέπουν στα στοιχεία του UI να προσαρμόζονται και να διατηρούν τις σχετικές θέσεις τους καθώς αλλάζει το μέγεθος της οθόνης, εξασφαλίζοντας μια συνεπή διάταξη σε όλες τις συσκευές. Οι

κατευθυντήριες γραμμές παρέχουν οπτικές αναφορές για την ευθυγράμμιση και την τοποθέτηση των στοιχείων UI, βοηθώντας την διατήρηση ενός ισορροπημένου και αρμονικού σχεδιασμό.



Εικόνα 20 XML αρχεία που χρησιμοποιούνται στην εφαρμογή

Οι αρχές του Material Design παίζουν σημαντικό ρόλο στο σχεδιασμό της διάταξης. Το Material Design δίνει έμφαση στην καθαρή και μοντέρνα αισθητική, με έμφαση στην τυπογραφία, το λευκό χώρο και τις διαισθητικές αλληλεπιδράσεις. Με την ενσωμάτωση των κατευθυντήριων γραμμών του Material Design, προσπαθήσαμε να δημιουργήσουμε οπτικά ευχάριστες διεπαφές που συνάδουν με άλλες εφαρμογές Android και παρέχουν μια οικεία εμπειρία χρήστη. Αυτό περιλαμβάνει τη χρήση προκαθορισμένων στυλ και θεμάτων Material Design για να εξασφαλιστεί μια συνεκτική οπτική ταυτότητα σε όλη την εφαρμογή. Μερικά από τα αντικείμενα Material Design που συμπεριλήφθηκαν στην ανάπτυξη της εφαρμογής είναι, Material3 Theme Light, Material3 Theme Dark, MaterialToolbar, MaterialCardView, MaterialButton, Material3.TabLayout κ.α.

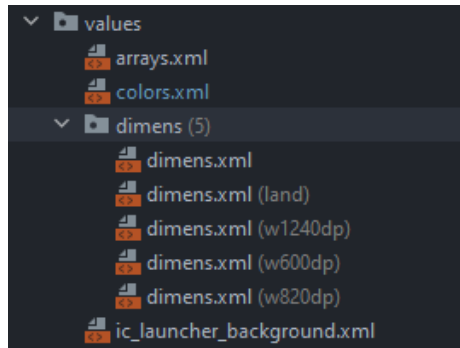
#### 4.2.4.2 Σχεδιασμός προσαρμοστικής διεπαφής για προσβασιμότητα

Ο προσαρμοστικός σχεδιασμός του περιβάλλοντος εργασίας περιλαμβάνει την εξέταση των προτιμήσεων του χρήστη και την ανάλογη προσαρμογή της διεπαφής. Το Android Studio παρέχει λειτουργίες όπως η νυχτερινή λειτουργία και η προσαρμογή θεμάτων, επιτρέποντας στους χρήστες να επιλέγουν τα προτιμώμενα χρωματικά σχήματα και οπτικά στυλ. Με την εφαρμογή αυτών των

χαρακτηριστικών, η εφαρμογή μπορεί να παρέχει μια εξατομικευμένη εμπειρία, ενισχύοντας την ικανοποίηση και τη δέσμευση των χρηστών.

Εκτός από την προσαρμογή σε διαφορετικές συσκευές και προτιμήσεις χρηστών, ένα προσαρμοστικό περιβάλλον εργασίας θα πρέπει επίσης να υποστηρίζει διαφορετικές μεθόδους εισαγωγής. Οι συσκευές Android προσφέρουν μια σειρά από μηχανισμούς εισόδου, όπως οθόνες αφής, φωνητικές εντολές συρόμενες κινήσεις χειρός κ.α.. Ο σχεδιασμός του UI ώστε να είναι συμβατό με διάφορες μεθόδους εισόδου διασφαλίζει ότι οι χρήστες μπορούν να αλληλεπιδρούν με την εφαρμογή χρησιμοποιώντας την προτιμώμενη μέθοδο εισόδου τους, ενισχύοντας τη χρηστικότητα και την προσβασιμότητα.

Άλλο ένα στοιχείο του προσαρμοστικού σχεδιασμού UI είναι ο χειρισμός διαφορετικών μεγεθών και πυκνοτήτων οθόνης. Οι συσκευές Android διαθέτουν διάφορες πυκνότητες εικονοστοιχείων, από οθόνες χαμηλής πυκνότητας έως οθόνες υψηλής πυκνότητας. Για να καλύψουν αυτή τη μεταβλητότητα, παρέχουμε πολλαπλές εκδόσεις στοιχείων εικόνας βελτιστοποιημένων για διαφορετικές πυκνότητες οθόνης. Με την ενσωμάτωση πόρων που αφορούν την πυκνότητα, όπως εικόνες και εικονίδια, η εφαρμογή μπορεί να διατηρήσει την οπτική συνοχή και να αποφύγει την εικονοποίηση ή το θόλωμα σε συσκευές με διαφορετικές πυκνότητες οθόνης.



Εικόνα 21 Διαφοροποίηση πυκνοτήτων

Κατά την ανάπτυξη του προσαρμοστικού UI της εφαρμογής μας, ήταν απαραίτητο να ληφθούν υπόψη οι συγκεκριμένες ανάγκες και προτιμήσεις των διαφόρων κατηγοριών χρηστών. Ο στόχος μας ήταν να δημιουργηθεί μια εφαρμογή που θα μπορεί να προσαρμόζει τη διεπαφή και τη λειτουργικότητά της ώστε να ανταποκρίνεται σε συγκεκριμένες κατηγορίες χρηστών. Οι αναγνωρισμένες κατηγορίες χρηστών περιλαμβάνουν αρχάριους χρήστες, χρήστες με αναπηρίες ή ειδικές ανάγκες και μεγαλύτερες ηλικιακές ομάδες.

Συλλέγοντας δεδομένα και μετρήσεις από τις αλληλεπιδράσεις των χρηστών, είναι δυνατόν να εντοπιστούν διακριτές κατηγορίες χρηστών και να προσαρμόσουμε τις επιλογές της εφαρμογής αναλόγως. Σε αυτό το κεφάλαιο, θα διερευνήσουμε τον τρόπο με τον οποίο η εφαρμογή μπορεί να προσαρμοστεί για αρχάριους χρήστες, χρήστες με αναπηρίες ή ειδικές ανάγκες και ηλικιακές ομάδες μεγαλύτερης ηλικίας.

### Κατηγορία αρχάριου χρήστη

Ο αρχάριος χρήστης αναφέρεται σε κάποιον που είναι νέος στη χρήση μιας συγκεκριμένης εφαρμογής Android ή μπορεί να έχει περιορισμένη εμπειρία με την τεχνολογία κινητής τηλεφωνίας γενικά. Οι αρχάριοι χρήστες συχνά δεν είναι εξοικειωμένοι με τα χαρακτηριστικά, τις λειτουργίες και τη διεπαφή χρήστη της εφαρμογής. Μπορεί να χρειάζονται περισσότερη καθοδήγηση και βοήθεια για να πλοηγηθούν στην εφαρμογή και να κατανοήσουν πώς λειτουργεί. Οι αρχάριοι χρήστες παρουσιάζουν συνήθως ορισμένες συμπεριφορές και χαρακτηριστικά που τους διακρίνουν από τους πιο έμπειρους χρήστες, καθώς συνήθως δυσκολεύονται με βασικές εργασίες, όπως η εύρεση συγκεκριμένων λειτουργιών, η κατανόηση των επιλογών του μενού ή η εκτέλεση ενεργειών εντός της εφαρμογής. Μπορεί επίσης να αντιμετωπίζουν δυσκολίες στην αποτελεσματική ολοκλήρωση των εργασιών και να χρειάζονται περισσότερο χρόνο για να επιτύχουν τους στόχους τους εντός της εφαρμογής.

Για έναν αρχάριο χρήστη, η καμπύλη εκμάθησης που σχετίζεται με τη χρήση της εφαρμογής μπορεί να είναι απότομη. Συνεπώς είναι αναγκαία μία πρόσθετη υποστήριξη και σαφείς οδηγίες για να ξεπεράσουν τις αρχικές προκλήσεις και να γίνουν πιο ικανοί στη χρήση της εφαρμογής.

Στην εφαρμογή, αναλύονται διάφορες μετρήσεις, όπως ο αριθμός των χαμένων κλικ, ο χρόνος εκτέλεσης της εφαρμογής και το ποσοστό των ολοκληρωμένων λειτουργιών της εφαρμογής. Αυτές οι μετρήσεις βοηθούν την εντόπιση των τομών στους οποίους οι αρχάριοι μπορεί να δυσκολεύονται ή να θεωρούν δύσκολη την πλοήγηση στην εφαρμογή. Για την αντιμετώπιση αυτών των ζητημάτων, υλοποιούνται προσαρμοστικά χαρακτηριστικά, όπως συμβουλές καθοδήγησης και επαρκή ανατροφοδότηση σχετικά με τις αλληλεπιδράσεις αφής, όπως δόνηση και ήχος όταν πατιούνται κουμπιά. Αυτές οι υποδείξεις και οι μηχανισμοί ανατροφοδότησης παρέχουν σαφείς οδηγίες και βοηθούν τους αρχάριους χρήστες να κατανοήσουν καλύτερα τη λειτουργικότητα της εφαρμογής και να πλοηγηθούν στα χαρακτηριστικά της. Μειώνοντας τα χαμένα κλικ και προσφέροντας καθοδήγηση, στοχεύουμε στη βελτίωση της συνολικής χρηστικότητας και της εμπειρίας χρήσης για τους αρχάριους χρήστες.

### Κατηγορία χρηστών με ειδικές ανάγκες

Οι χρήστες με ειδικές ανάγκες αναφέρονται σε άτομα που έχουν ειδικές καταστάσεις που επηρεάζουν την ικανότητά τους να χρησιμοποιούν μία εφαρμογή Android. Αυτές ειδικές ανάγκες μπορεί να περιλαμβάνουν, μεταξύ άλλων, οπτικές βλάβες, βλάβες ακοής, κινητικές αναπηρίες, γνωστικές αναπηρίες, και γλωσσικές ή μαθησιακές αναπηρίες. Κατά το σχεδιασμό μιας εφαρμογής Android, είναι σημαντικό να λαμβάνονται υπόψη οι ποικίλες ανάγκες των χρηστών με ειδικές ανάγκες, ώστε να διασφαλίζεται η προσβασιμότητα. Αυτοί οι χρήστες ενδέχεται να χρειάζονται ειδικές προσαρμογές, διευκολύνσεις ή υποστηρικτικές τεχνολογίες για να αλληλεπιδράσουν και να πλοηγηθούν αποτελεσματικά στην εφαρμογή.

Για τους χρήστες με προβλήματα όρασης, η εφαρμογή θα πρέπει να παρέχει λειτουργίες όπως συμβατότητα με αναγνώστες οθόνης ή λειτουργίες μετατροπής κειμένου σε ομιλία, οι οποίες μπορούν να μετατρέπουν το κείμενο και τις πληροφορίες στην οθόνη σε ηχητική έξοδο. Επιπλέον, τα θέματα υψηλής αντίθεσης ή παραμετροποιήσιμα χρωματικά σχήματα μπορούν να βοηθήσουν τους χρήστες με χαμηλή όραση ή αχρωματοψία να αντιληφθούν και να διακρίνουν καλύτερα τα στοιχεία της διεπαφής. Επιπρόσθετα, οι χρήστες με προβλήματα ακοής μπορεί να επωφεληθούν από υπότιτλους ή οπτικές ενδείξεις για σημαντικές ηχητικές πληροφορίες. Η παροχή εναλλακτικών τρόπων λήψης πληροφοριών, όπως ειδοποιήσεις με βάση το κείμενο ή ανατροφοδότηση με δόνηση, μπορεί να βελτιώσει την εμπειρία τους. Οι κινητικές αναπηρίες μπορεί να δυσκολέψουν τις ακριβείς αλληλεπιδράσεις αφής πρόκληση για ορισμένους χρήστες. Η ενσωμάτωση χαρακτηριστικών όπως μεγαλύτεροι στόχοι αφής, χειρισμοί που βασίζονται σε χειρονομίες ή επιλογές φωνητικής εισαγωγής μπορούν να βελτιώσουν την προσβασιμότητα για χρήστες με κινητικές αναπηρίες.

Οι γνωστικές διαταραχές ή οι μαθησιακές δυσκολίες μπορεί να απαιτούν απλοποιημένες διεπαφές, σαφείς οδηγίες και οπτικά βοηθήματα για να διευκολύνουν την κατανόηση και την ολοκλήρωση της εργασίας. Η χρήση συνεπούς και ξεκάθαρης γλώσσας, καθώς και η ελαχιστοποίηση των περισπασμών και της πολύπλοκης πλοήγησης, μπορούν να υποστηρίξουν τους χρήστες με αυτές τις ανάγκες.

Για την εξυπηρέτηση χρηστών με αναπηρίες ή ειδικές ανάγκες, όπως οπτική αναπηρία, αχρωματοψία και δυσλεξία, εστιάζονται συγκεκριμένα σημεία μετρικών δεδομένων, όπως οι συντεταγμένες των αγγιγμάτων που πλησιάζουν ένα κουμπί χωρίς να το πατάει, οι προσπάθειες μεγέθυνσης της οθόνης και η καθυστέρηση μεταξύ των εισόδων. Με βάση αυτές τις μετρήσεις, χρησιμοποιούνται προσαρμοστικές επιλογές για την ενίσχυση της προσβασιμότητας. Για τους χρήστες με προβλήματα όρασης, αυξάνεται τη

φωτεινότητα του τηλεφώνου στο 100% για την βελτίωση της ορατότητας. Επιπλέον, ενεργοποιούνται οι ρυθμίσεις μετατροπής κειμένου σε ομιλία για την παροχή ηχητικής ανατροφοδότησης και ενεργοποιείται η λειτουργία ενός μεγεθυντικού φακού αφής που βοηθά στην ανάγνωση μικρών κειμένων. Επιπλέον, η ενσωμάτωση μιας γραμματοσειρά φιλική προς τη δυσλεξία, όπως η γραμματοσειρά OpenDyslexic (González, 2023), οδηγεί στην βελτίωση της αναγνωσιμότητας για τους χρήστες με δυσλεξία.

### Κατηγορία ομάδων μεγαλύτερης ηλικίας

Η ηλικιακή ομάδα των ηλικιωμένων αναφέρεται σε άτομα που συνήθως κατηγοριοποιούνται ως ηλικιωμένοι πολίτες ή ηλικιωμένοι ενήλικες. Στο πλαίσιο μιας εφαρμογής Android, αυτή η δημογραφική ομάδα περιλαμβάνει άτομα που βρίσκονται σε μεταγενέστερα στάδια της ζωής τους και ενδέχεται να έχουν συγκεκριμένα χαρακτηριστικά και ανάγκες που διαφέρουν από τους νεότερους χρήστες. Όταν σχεδιάσαμε την εφαρμογή για την προσβασιμότητα των ηλικιωμένων ομάδων, λάβαμε υπόψη τις μοναδικές απαιτήσεις και προτιμήσεις τους. Μπορεί να έχουν λιγότερη εμπειρία με τα smartphones ή να χρειάζονται πρόσθετη υποστήριξη για να πλοηγηθούν και να χρησιμοποιήσουν αποτελεσματικά τις λειτουργίες μιας εφαρμογής Android. Ένα βασικό στοιχείο για τους ηλικιωμένους χρήστες θα μπορούσε να είναι η μείωση της οπτικής οξύτητας. Αυτό μπορεί να περιλαμβάνει δυσκολίες στην ανάγνωση μικρών κειμένων ή στη διάκριση μικρών στοιχείων διεπαφής. Για να εξυπηρετήσει αυτούς τους χρήστες, η εφαρμογή θα πρέπει να προσφέρει επιλογές για την αύξηση του μεγέθους της γραμματοσειράς, τη χρήση σαφούς και ευανάγνωστης τυπογραφίας και την παροχή επαρκούς αντίθεσης μεταξύ του κειμένου και των στοιχείων του φόντου. Επίσης, οι κινητικές δεξιότητες μπορεί να επηρεαστούν από τις αλλαγές που σχετίζονται με την ηλικία, με αποτέλεσμα τη μείωση της επιδεξιότητας και της ακρίβειας στις αλληλεπιδράσεις με την αφή. Ο σχεδιασμός μεγαλύτερων στόχων αφής, η ελαχιστοποίηση της ανάγκης για πολύπλοκες χειρονομίες και η παροχή οπτικής ανατροφοδότησης κατά την αφή μπορούν να βελτιώσουν τη χρηστικότητα της εφαρμογής για τους ηλικιωμένους χρήστες. Επιπρόσθετα, η παροχή επιλογών για φωνητική εισαγωγή ή φωνητικών βοηθών μπορεί να είναι επωφελής για τους ηλικιωμένους χρήστες που μπορεί να προτιμούν ή να απαιτούν αλληλεπίδραση με την εφαρμογή χωρίς χέρια. Επιπλέον, η ενσωμάτωση ηχητικών ενδείξεων, όπως προφορικές ειδοποιήσεις ή ήχοι επιβεβαίωσης, μπορεί να βοηθήσει τους ηλικιωμένους χρήστες να ενημερώνονται για σημαντικά γεγονότα της εφαρμογής.

Στην εφαρμογή μας, για τις μεγαλύτερες ηλικιακές ομάδες, αναλύονται μετρικές όπως οι προσπάθειες κύλισης σε συγκεκριμένα σημεία χωρίς τη δυνατότητα κύλισης, η μεγάλη διάρκεια του κλικ καθώς και



την προσπάθεια μεγέθυνσης στοιχείων της οθόνης. Αυτές οι μετρικές υποδεικνύουν πιθανές δυσκολίες που μπορεί να αντιμετωπίζουν οι χρήστες μεγαλύτερης ηλικίας κατά την αλληλεπίδραση με την εφαρμογή. Για να αντιμετωπιστούν αυτές τις προκλήσεις, εισάγονται προσαρμοστικές επιλογές προσαρμοσμένες στις ανάγκες τους. Συνδυάζεται ένα θέμα υψηλής αντίθεσης την βελτίωση της ορατότητας και την διασφάλιση της σαφούς διαφοροποίησης μεταξύ των στοιχείων του UI. Υλοποιείται η λειτουργία φωνητικής εισαγωγής για να παρέχει μια εναλλακτική μέθοδο πλοήγησης, μειώνοντας την εξάρτηση από μικρά κουμπιά ή πολύπλοκες χειρονομίες. Επιπλέον, ενσωματώνουμε μια επιλογή για την αύξηση του μεγέθους της γραμματοσειράς, ώστε να βελτιωθεί η αναγνωσιμότητα για τους χρήστες μεγαλύτερης ηλικίας. Με την προσαρμογή της διεπαφής και των μεθόδων αλληλεπίδρασης της εφαρμογής, στοχεύουμε στην παροχή μιας πιο διαισθητικής και φιλικής προς τον χρήστη εμπειρίας για τις ομάδες μεγαλύτερης ηλικίας.

Λαμβάνοντας υπόψη τις μοναδικές απαιτήσεις κάθε κατηγορίας χρηστών πραγματοποιείται η προσαρμογή των επιλογών της εφαρμογής με βάση τα δεδομένα και τις μετρήσεις, για την βελτίωση της χρηστικότητας και της προσβασιμότητας.

#### 4.2.5 Προσαρμογή μέσω μηχανικής μάθησης

Σε αυτό το κεφάλαιο, παρουσιάζεται η υλοποίηση αλγορίθμων μηχανικής μάθησης στην android εφαρμογή για την προσαρμογή της διεπαφής χρήστη. Ο στόχος της ενσωμάτωσης των μεθόδων τεχνητής νοημοσύνης (AI) είναι η μείωση των προσπαθειών των χρηστών κατά την επιλογή δρομολογίων λεωφορείων, την κύλιση στις οθόνες και την πλοήγηση στην εφαρμογή. Προσαρμόζοντας τη διεπαφή χρήστη (UI) με βάση τις προτιμήσεις των χρηστών από παλιότερα δεδομένα, η εφαρμογή στοχεύει να βελτιώσει τη χρηστικότητα και να μειώσει το ποσοστό εγκατάλειψης. Στην συνέχεια θα διερευνηθεί η συλλογή δεδομένων του χρήστη, η πρόβλεψη προτιμώμενων γραμμών λεωφορείου και η ενσωμάτωση αλγορίθμων μηχανικής μάθησης στην εφαρμογή Android.

##### 4.2.5.1 Συλλογή δεδομένων

Για την εξατομίκευση της εμπειρίας χρήσης της εφαρμογής, συλλέγονται σχετικά δεδομένα χρήστη. Αυτά τα δεδομένα περιλαμβάνουν συντεταγμένες γεωγραφικού μήκους και πλάτους για τον προσδιορισμό της τοποθεσίας του χρήστη, καθώς και την ώρα και την ημερομηνία χρήσης της εφαρμογής. Η συλλογή τέτοιων πληροφοριών επιτρέπει στην εφαρμογή να κατανοεί τις προτιμήσεις των χρηστών βάσει ιστορικών μοτίβων και να προσαρμόζει ανάλογα τη διεπαφή χρήστη.

Κατά την εκκίνηση της εφαρμογής ζητείται από τον χρήστη η ενεργοποίηση της τοποθεσίας της συσκευής του χρήστη καθώς και η αποδοχή της χρήσης του αποθηκευτικού χώρου του κινητού (Κώδικας 16, Κώδικας 17). Μόλις φτάσει ο χρήστης στην επιλογή μίας γραμμής αποθηκεύονται τα δεδομένα που θα χρησιμοποιηθούν για την προσαρμογή του UI (Κώδικας 18). Χρησιμοποιείται ο πίνακας `user_interactions` στην SQLite βάση δεδομένων που καταγράφονται τα στοιχεία (Κώδικας 8). Στην συνέχεια τα δεδομένα όπως φαίνονται στον Πίνακας 4 αποθηκεύονται σε ένα αρχείο μορφής CSV για να τα χειριστεί ο κώδικας `python`.

Πίνακας 4 Παράδειγμα δεδομένων CSV αρχείου

Id	latitude	longitude	day_of_week	time_of_day (ms)	bus_line
1	23.674484	38.002723	1	32405045	811
2	23.812254	38.017002	1	52545480	421
3	23.674488	38.001628	2	32155097	811
4	23.745696	37.976219	2	69203508	608
5	23.812256	38.017001	2	50402900	421
6	23.674482	38.002721	2	35598015	811
7	23.812257	38.017005	3	50400100	421
8	23.673639	38.001628	3	32569036	811
9	23.673630	38.001638	4	31456482	829
10	23.812255	38.017003	4	51522231	421
11	23.745696	37.976219	4	75284641	608
12	23.673639	38.001628	4	36045684	811
13	23.812249	38.017006	5	54853977	421
14	23.674482	38.002722	5	38461098	811
15	23.674483	38.002725	6	9415605	10
16	23.674480	38.002727	6	9857814	10

#### 4.2.5.2 Αλγόριθμοι μηχανικής μάθησης

Αρκετοί αλγόριθμοι μηχανικής μάθησης δοκιμάστηκαν και συγκρίθηκαν ως προς την αποτελεσματικότητά τους στην πρόβλεψη της προτιμώμενης γραμμής διαύλου του χρήστη εντός της εφαρμογής Android. Οι αλγόριθμοι που εξετάστηκαν για αξιολόγηση περιλάμβαναν το Naive Bayes, το Stochastic Gradient Descent, το K-Nearest Neighbors (KNN), τον Support Vector Classifier (SVC) και το Random Forest. Παρακάτω παρουσιάζονται λίγα λόγια για κάθε αλγόριθμο. Μετά από αξιολόγηση και

σύγκριση, επιλέχθηκε ο αλγόριθμος με την καλύτερη απόδοση με βάση μετρήσεις όπως η ακρίβεια, η ακρίβεια, η ανάκληση και η βαθμολογία F1 (Κώδικας 20). Αυτός ο αλγόριθμος έδειξε την ικανότητα αποτελεσματικής πρόβλεψης της προτιμώμενης γραμμής λεωφορείου του χρήστη με βάση τα δεδομένα που συλλέγονται, παρέχοντας μια σταθερή βάση για την εφαρμογή προσαρμοστικής διεπαφής χρήστη.

Ο αλγόριθμος Naïve Bayes αφορά ένα πιθανολογικό αλγόριθμο που υποθέτει την ανεξαρτησία μεταξύ των χαρακτηριστικών (Russell & Norvig, 2003). Χρησιμοποιείται ευρέως σε εργασίες ταξινόμησης κειμένου και φιλτραρίσματος ανεπιθύμητων μηνυμάτων (Metsis, Androutsopoulos, & Paliouras, 2006). Ο Naïve Bayes υπολογίζει την πιθανότητα κάθε κλάσης δεδομένων των χαρακτηριστικών εισόδου και επιλέγει την κλάση με την υψηλότερη πιθανότητα. Είναι γνωστός για την απλότητά του, τη γρήγορη προπόνηση και τους χρόνους πρόβλεψης. Παρά την υπόθεση της ανεξαρτησίας χαρακτηριστικών, το Naïve Bayes συχνά αποδίδει καλά στην πράξη και μπορεί να χειριστεί αποτελεσματικά δεδομένα υψηλών διαστάσεων (Bielza, Li, & Larrañaga, 2011). Ωστόσο, μπορεί να αντιμετωπίζει πολύπλοκες σχέσεις μεταξύ των χαρακτηριστικών και μπορεί να μην καταγράφει πιο διαφοροποιημένα μοτίβα στα δεδομένα. Ο αλγόριθμος αξιολογήθηκε για την αποτελεσματικότητά του στην πρόβλεψη της προτιμώμενης γραμμής λεωφορείου του χρήστη με βάση τα δεδομένα του παρελθόντος χρήστη. Λαμβάνοντας υπόψη τα ιστορικά δεδομένα του χρήστη που έχει επιλέξει συγκεκριμένες γραμμές λεωφορείων, δεδομένης της τοποθεσίας, της ώρας της ημέρας και της ημέρας της εβδομάδας, ο Naïve Bayes υπολογίζει τις πιθανότητες επιλογής κάθε γραμμής λεωφορείων. Στη συνέχεια, ο αλγόριθμος προβλέπει την πιο πιθανή γραμμή λεωφορείου με βάση αυτά τα δεδομένα. Ο αλγόριθμος Naïve Bayes παρέχει γρήγορους χρόνους εκπαίδευσης και πρόβλεψης, καθιστώντας τον κατάλληλο για προβλέψεις σε πραγματικό χρόνο στην εφαρμογή Android.

Ο Stochastic Gradient Descent είναι ένας αλγόριθμος βελτιστοποίησης που χρησιμοποιείται συνήθως σε γραμμικά μοντέλα (Toulis, Airoidi, & Rennie, 2014). Είναι ιδιαίτερα χρήσιμο με σύνολα δεδομένων μεγάλης κλίμακας, καθώς ενημερώνει τις παραμέτρους του μοντέλου με βάση μικρές παρτίδες δεδομένων και όχι με ολόκληρο το σύνολο δεδομένων. Το SGD εκτελεί επαναληπτικά ενημερώσεις κλίσης, με στόχο την ελαχιστοποίηση της απώλειας. Η αποτελεσματικότητά του το καθιστά κατάλληλο για εργασίες με μεγάλο αριθμό δειγμάτων και χαρακτηριστικών. Ωστόσο, η σύγκλιση του SGD μπορεί να επηρεαστεί από τον ρυθμό εκμάθησης και την επιλογή των υπερπαραμέτρων, που απαιτούν προσεκτική ρύθμιση. Επιπλέον, ενδέχεται να αντιμετωπίζει θορυβώδη δεδομένα ή δεδομένα με σύνθετες σχέσεις που δεν μπορούν να αναπαρασταθούν αποτελεσματικά από ένα γραμμικό μοντέλο. Σε αυτή την εφαρμογή, ο αλγόριθμος βελτιστοποιεί το γραμμικό μοντέλο με βάση τα ιστορικά δεδομένα χρήστη,

βρίσκοντας τους συντελεστές που αντιπροσωπεύουν καλύτερα τη σχέση μεταξύ των χαρακτηριστικών και της προτιμώμενης γραμμής λεωφορείου. Με την ικανότητά του να χειρίζεται αποτελεσματικά μεγάλα σύνολα δεδομένων, το SGD αποδεικνύεται μια βιώσιμη επιλογή για την πρόβλεψη της προτιμώμενης γραμμής λεωφορείου του χρήστη σε πραγματικό χρόνο εντός της εφαρμογής.

Το K-Nearest Neighbors (KNN) είναι ένας μη παραμετρικός αλγόριθμος που ταξινομεί νέες παρουσίες με βάση την πλειοψηφία των πλησιέστερων γειτόνων τους (Guo, Wang, Bell, & Bi, 2004). Είναι απλό να κατανοηθεί και να εφαρμοστεί, αλλά μπορεί να είναι υπολογιστικά ακριβό για μεγάλα σύνολα δεδομένων. Μπορεί να χειριστεί την ταξινόμηση πολλαπλών τάξεων και είναι ανθεκτικό σε θορυβώδη δεδομένα εκπαίδευσης. Ωστόσο, το κύριο μειονέκτημά του είναι το υπολογιστικό κόστος κατά την πρόβλεψη, καθώς απαιτεί τον υπολογισμό των αποστάσεων από όλα τα δείγματα εκπαίδευσης. Αυτό μπορεί να κάνει το KNN λιγότερο αποτελεσματικό για μεγάλα σύνολα δεδομένων και χώρους χαρακτηριστικών υψηλών διαστάσεων. Επιπλέον, δεν καταγράφει παγκόσμια μοτίβα και μπορεί να αντιμετωπίσει προβλήματα με μη ισορροπημένες τάξεις. Σε αυτή την εργασία, ο αλγόριθμος συγκρίνει τα δεδομένα χρήστη με τα ιστορικά δεδομένα άλλων χρηστών που επέλεξαν συγκεκριμένες γραμμές διαύλου. Μετρώντας την απόσταση μεταξύ των παρουσιών στο χώρο χαρακτηριστικών, το KNN προσδιορίζει τους k πλησιέστερους γείτονες και εκχωρεί την πιο συχνή γραμμή διαύλου μεταξύ αυτών ως πρόβλεψη για τον χρήστη.

Ο Support Vector Machine (SVM) είναι ένας ισχυρός αλγόριθμος που κατασκευάζει ένα υπερεπίπεδο για να διαχωρίσει διαφορετικές κλάσεις (Srivastava & Bhambhu, 2010). Το SVM μπορεί να χειριστεί εργασίες γραμμικής και μη γραμμικής ταξινόμησης χρησιμοποιώντας διαφορετικές συναρτήσεις πυρήνα, όπως συνάρτηση γραμμικής, πολυωνυμικής ή ακτινικής βάσης (RBF). Είναι γνωστό για την αποτελεσματικότητά του στην αντιμετώπιση δεδομένων υψηλών διαστάσεων ενώ μπορεί να χειριστεί αποτελεσματικά μεγάλα σύνολα δεδομένων. Στην εφαρμογή, ο αλγόριθμος βρίσκει το βέλτιστο υπερεπίπεδο που διαχωρίζει καλύτερα τα ιστορικά δεδομένα χρήστη με βάση την τοποθεσία, την ώρα της ημέρας και την ημέρα της εβδομάδας για να προβλέψει με ακρίβεια την προτιμώμενη γραμμή διαύλου. Χρησιμοποιώντας διαφορετικές συναρτήσεις πυρήνα, το SVC μπορεί να χειριστεί εργασίες γραμμικής και μη γραμμικής ταξινόμησης. Η ικανότητά του να χειρίζεται δεδομένα υψηλών διαστάσεων και να παρέχει καλές δυνατότητες γενίκευσης καθιστά τον SVC έναν χρήσιμο αλγόριθμο για την πρόβλεψη της προτιμώμενης γραμμής ωστόσο, ο χρόνος εκπαίδευσης του SVC μπορεί να είναι σχετικά αργός σε σύγκριση με άλλους αλγόριθμους.

Το Random Forest είναι ένας αλγόριθμος εκμάθησης συνόλου που συνδυάζει πολλαπλά δέντρα αποφάσεων για να κάνει προβλέψεις (Ali, Khan, Ahmad, & Maqsood, 2012). Μπορεί να καταγράψει πολύπλοκες σχέσεις και αλληλεπιδράσεις μεταξύ χαρακτηριστικών και είναι κατάλληλο για χρήση σε παράλληλους και καταναμημένους υπολογιστές. Ωστόσο, το κύριο μειονέκτημα είναι η αυξημένη πολυπλοκότητα και ο χρόνος εκπαίδευσης σε σύγκριση με τα μεμονωμένα δέντρα αποφάσεων. Ο προσεκτικός συντονισμός των υπερπαραμέτρων, όπως ο αριθμός των δέντρων και το μέγιστο βάθος, είναι απαραίτητος για την αποφυγή υπερβολικής προσαρμογής (overfitting) και την επίτευξη βέλτιστης απόδοσης. Σε αυτή την εργασία, κάθε δέντρο απόφασης εκπαιδεύεται στα ιστορικά δεδομένα χρήστη, λαμβάνοντας υπόψη την τοποθεσία, την ώρα της ημέρας και την ημέρα της εβδομάδας ως χαρακτηριστικά και την επιλεγμένη γραμμή διαύλου ως μεταβλητή στόχο. Συγκεντρώνοντας τις προβλέψεις των μεμονωμένων δέντρων, το Random Forest προσδιορίζει την πιο πιθανή προτιμώμενη γραμμή διαύλου για τα δεδομένα χρήστη. Το Random Forest είναι γνωστό για την ανθεκτικότητά του έναντι της υπερπροσαρμογής, τα μέτρα σημασίας χαρακτηριστικών και τον αποτελεσματικό χειρισμό μεγάλων συνόλων δεδομένων, καθιστώντας το έναν κατάλληλο αλγόριθμο για την πρόβλεψη της προτιμώμενης γραμμής του χρήστη στην εφαρμογή.

Στο Κώδικας 21 πραγματοποιείται η εκπαίδευση και αξιολόγηση των μοντέλων με τα δεδομένα από δοκιμές τις εφαρμογής. Εκπαιδεύτηκε κάθε μοντέλο με προσαρμοσμένες τις παράμετρούς, όπως το ρυθμό εκμάθησης, την ισχύ τακτοποίησης ή τον αριθμό των κρυφών επιπέδων, χρησιμοποιώντας τεχνικές όπως η διασταυρούμενη επικύρωση για να βρεθούν οι βέλτιστες τιμές. Αξιολογήθηκαν τα εκπαιδευμένα μοντέλα χρησιμοποιώντας κατάλληλες μετρήσεις αξιολόγησης, όπως accuracy, precision, recall, f1-score (Κώδικας 20) . Παρακάτω, παρουσιάζεται πίνακας με τα αποτελέσματα των δοκιμών.

Πίνακας 5 Αποτελέσματα αξιολόγησης μοντέλων κατηγοριοποίησης

	Naïve Bayes	SGD	KNN	SVM	Random Forest	Decision Tree
<b>Accuracy</b>	50%	50%	100%	66.67%	90%	90%
<b>Precision</b>	30%	25%	100%	80%	85%	82%
<b>Recall</b>	50%	50%	100%	66.67%	90%	90%
<b>F1-score</b>	37.5%	33.3%	100%	62.5%	85.5%	90%

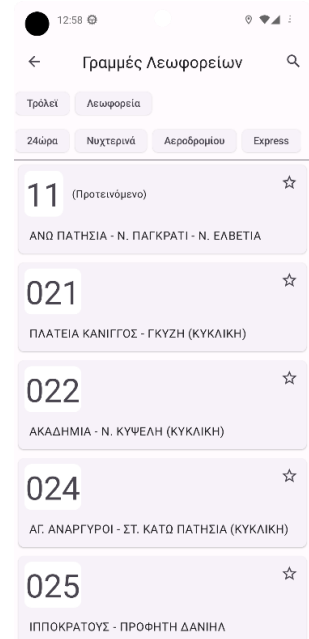
Με βάση τα παραπάνω αποτελέσματα της αξιολόγησης και τις συγκεκριμένες απαιτήσεις της εφαρμογής, επιλέχθηκε το KNN μοντέλο.

### 4.2.5.3 Αποτελέσματα προσαρμογής

Η διαδικασία προσαρμογής της διεπαφής χρήστη της εφαρμογής περιλαμβάνει την ενσωμάτωση της πρόβλεψης μηχανικής μάθησης με τη διεπαφή χρήστη για να προσφέρει την βέλτιστη εμπειρία. Κατά την εισαγωγή του χρήστη στην λειτουργία εμφάνισης λίστας των λεωφορείων συλλέγεται και αποθηκεύεται το ιστορικό αλληλεπίδρασης του χρήστη, συμπεριλαμβανομένης της τοποθεσίας, της ώρας της ημέρας και της ημέρας της εβδομάδας (Κώδικας 22). Αυτά τα δεδομένα θα χρησιμοποιηθούν ως είσοδος για το μοντέλο μηχανικής εκμάθησης για την πρόβλεψη της γραμμής διαύλου. Έπειτα, χρησιμοποιείται το εκπαιδευμένο μοντέλο μηχανικής μάθησης για την ανάλυση του ιστορικού αλληλεπίδρασης του χρήστη και προβλέπει την πιο πιθανή γραμμή λεωφορείου που σκοπεύει να ακολουθήσει ο χρήστης. Αυτή η πρόβλεψη βασίζεται σε μοτίβα και συσχετισμούς που ανακαλύφθηκαν στα δεδομένα εκπαίδευσης. Μόλις επιτευχθεί η πρόβλεψη γραμμής διαύλου, τροποποιείται η διεπαφή χρήστη της εφαρμογής για να ενσωματώσει την προβλεπόμενη γραμμή. Προσδιορίζεται η λίστα γραμμών λεωφορείων και προστίθεται η προβλεπόμενη γραμμή λεωφορείου στην κορυφή της λίστας για να εξασφαλισθεί η γρήγορη και εύκολη πρόσβαση από τον χρήστη (Κώδικας 19). Επιπλέον, εμφανίζεται ένα κείμενο δίπλα στον κωδικό της γραμμής για την ενημέρωση του χρήστη. Το κείμενο παρέχει πληροφορίες, όπως "Προτεινόμενο", υποδεικνύοντας ότι η πρόβλεψη είναι προσαρμοσμένη στις προτιμήσεις του χρήστη. Ακολουθούν στιγμιότυπα δοκιμών της παραπάνω μεθόδου:

```
I /data/data/com.example.oasa_app/files/chaquopy/AssetFinder/app/dataset/OASADatabase_db_user_interactions.csv
I id bus_line route_line ... day_of_week time_of_day Unnamed: 7
I 0 2 11 1995 ... 7 46401813 NaN
I 1 3 40 3922 ... 7 46417967 NaN
I 2 4 10 1993 ... 7 46433414 NaN
I 3 5 11 1995 ... 7 46445723 NaN
I 4 6 500 1890 ... 7 46467829 NaN
I
I [5 rows x 8 columns]
W /data/data/com.example.oasa_app/files/chaquopy/AssetFinder/requirements/sklearn/neighbors/_classification.py:
I [10 10]
I latitude longitude day_of_week time_of_day
I 0 38.0196117 23.7560217 7 46507330
I Predicted bus_line and route_line: [11]
I 11
```

Databases							
user_interactions							
Live updates							
	id	bus_line	route_line	latitude	longitude	day_of_week	time_of_day
>	1	1	11	38.0196117	23.7560217	7	46365357
>	2	2	11	38.0196117	23.7560217	7	46401813
>	3	3	040	38.0196117	23.7560217	7	46417967
>	4	4	10	38.0196117	23.7560217	7	46433414
>	5	5	11	38.0196117	23.7560217	7	46445723
>	6	6	500	38.0196117	23.7560217	7	46467829
>	7	7	790	38.0196117	23.7560217	7	46503683



#### 4.2.6 Ανάπτυξη Back-End

Το κεφάλαιο αυτό επικεντρώνεται στην υλοποίηση των back-end συστατικών της ανθρωποκεντρικής εφαρμογής Android για το σύστημα μεταφοράς με λεωφορείο. Διαδραματίζονται οι τρόποι της σύνδεσης της εφαρμογής με το API τηλεματικής του ΟΑΣΑ, την διαχείριση της εμμονής των δεδομένων (data persistence), την μετατροπή των δεδομένων του API σε αντικείμενα του μοντέλου και στην ενσωμάτωση βασικών λειτουργιών για μια εμπειρία χρήστη.

Η ενσωμάτωση με το API τηλεματικής του ΟΑΣΑ έχει σημαντικό ρόλο στη διαδικασία ανάπτυξης του back-end. Περιλαμβάνει τη σύνδεση της εφαρμογής με το API χρησιμοποιώντας τη βιβλιοθήκη Volley της Google, η οποία απλοποιεί τα αιτήματα δικτύου και παρέχει αποτελεσματικό χειρισμό των απαντήσεων του API<sup>1</sup>. Με την ενσωμάτωση με το API, η εφαρμογή αποκτά πρόσβαση σε πληθώρα πληροφοριών που σχετίζονται με αφίξεις λεωφορείων, δρομολόγια, διαδρομές και στάσεις. Η ενσωμάτωση αυτή επιτρέπει στους χρήστες να ανακτούν δεδομένα σε πραγματικό χρόνο και να ενημερώνονται για την κατάσταση των δρομολογίων των λεωφορείων τους.

##### 4.2.6.1 OASA Telematics API

Όλες οι πληροφορίες των δρομολογίων και γραμμών του ΟΑΣΑ ανακτώνται από το ανοιχτό API του OASA telematics (ΟΑΣΑ, 2016). Τα στοιχεία που λαμβάνονται από την εφαρμογή είναι σε μορφή JSON (array/object) η οποία υποστηρίζεται ευρέως και αναλύεται εύκολα για περαιτέρω επεξεργασία.

Για την εκτέλεση των API requests στην εφαρμογή συμβουλευτήκαμε την σελίδα<sup>2</sup> του (Παπαιωάννου, 2016). Το API έχει την παρακάτω φόρμα:

```
http://telematics.oasa.gr/api/?act=action&p1=paramr1&p2=param2
```

Για την υποβολή αιτήσεων στο API, η εφαρμογή χρησιμοποιεί αιτήσεις POST, προσθέτοντας τις απαραίτητες μεταβλητές στις παραμέτρους URL. Κάθε λειτουργία εντός της εφαρμογής αντιστοιχεί σε ένα συγκεκριμένο τελικό σημείο του API και εξυπηρετεί έναν ξεχωριστό σκοπό για την ανάκτηση σχετικών δεδομένων. Παρακάτω αναφέρονται οι λειτουργίες που χρησιμοποιούνται στην εφαρμογή.

- «GetRoutesForLine» παίρνει τον κωδικό γραμμής ως παράμετρο και επιστρέφει τις διαδρομές που σχετίζονται με αυτήν τη γραμμή. Αυτή η λειτουργία επιτρέπει στους χρήστες να

<sup>1</sup> <https://google.github.io/volley/>

<sup>2</sup> <https://oasa-telematics-api.readthedocs.io/en/latest/>

εξερευνήσουν τις διάφορες διαδρομές που είναι διαθέσιμες για μια συγκεκριμένη γραμμή λεωφορείων.

- «WebGetLinesWithMLInfo» ανακτά λεπτομέρειες για όλες τις γραμμές μαζί με τα δεδομένα για την κύρια γραμμή. Παρέχει ολοκληρωμένες πληροφορίες σχετικά με τις γραμμές, συμπεριλαμβανομένων των χρονοδιαγραμμάτων, των στάσεων και των σχετικών βασικών γραμμών.
- «GetClosestStops» παρέχει βοήθεια στους χρήστες στην εύρεση κοντινών στάσεων λεωφορείων. Παρέχοντας γεωγραφικό πλάτος και μήκος ως παραμέτρους, επιστρέφει μια λίστα με στάσεις σε κοντινή απόσταση από την καθορισμένη τοποθεσία.
- «getStopArrivals» επιτρέπει στους χρήστες να παρακολουθούν τις ώρες άφιξης των λεωφορείων σε μια συγκεκριμένη στάση. Παρέχοντας τον κωδικό στάσης ως παράμετρο, η συνάρτηση ανακτά πληροφορίες σχετικά με τα λεωφορεία που φτάνουν σύντομα και τις αντίστοιχες ώρες άφιξής τους.
- «getBusLocation» παίρνει τον κωδικό και τη διαδρομή του λεωφορείου ως παραμέτρους και επιστρέφει τις συντεταγμένες των λεωφορείων. Αυτή η λειτουργία επιτρέπει στους χρήστες να παρακολουθούν τις τρέχουσες θέσεις των λεωφορείων κατά μήκος των επιλεγμένων διαδρομών τους.
- «getDailySchedule» παρέχει τις ώρες των ημερήσιων δρομολογίων για τα δρομολόγια των λεωφορείων, καθορίζοντας τις ώρες άφιξης και αναχώρησης τόσο για τις κατευθύνσεις «come» και «go».
- «getScheduleDaysMasterline» λαμβάνει τον κωδικό διαδρομής ως παράμετρο και επιστρέφει τις ημέρες κατά τις οποίες λειτουργεί η κύρια γραμμή, διευκολύνοντας τον ακριβή προγραμματισμό του χρονοδιαγράμματος.
- «getSchedLines» ανακτά τους χρόνους διαδρομής λαμβάνοντας ως παραμέτρους τον κωδικό γραμμής και την κύρια γραμμή. Επιστρέφει τα χρονοδιαγράμματα για την καθορισμένη γραμμή, επιτρέποντας στους χρήστες να έχουν πρόσβαση στις ακριβείς ώρες των λεωφορείων κατά μήκος της διαδρομής.
- Τέλος, το «webGetStops» ανακτά τις στάσεις που πραγματοποιούνται από μια συγκεκριμένη διαδρομή. Παρέχοντας τον κωδικό διαδρομής ως παράμετρο, επιστρέφει μια λίστα στάσεων που σχετίζονται με αυτήν τη διαδρομή.



```
[
  {
    "StopCode": "10340",
    "StopID": "000134",
    "StopDescr": "ΓΑΛΑΤΣΙ",
    "StopDescrEng": "GALATSI",
    "StopStreet": null,
    "StopStreetEng": null,
    "StopHeading": "214",
    "StopLat": "38.0265555",
    "StopLng": "23.7614695",
    "RouteStopOrder": "1",
    "StopType": "0",
    "StopAmea": "0"
  }, {
    "StopCode": "150038",
    "StopID": "150038",
    "StopDescr": "ΧΡΙΣΤΙΑΝΟΥΠΟΛΕΟΣ",
    "StopDescrEng": "ΧΡΙΣΤΙΑΝΟΥΠΟΛΕΟΣ",
    "StopStreet": "ΛΕΩΔ.ΟΜΟΡΘΟΚΚΑΗΣΙΑΣ",
    "StopStreetEng": null,
    "StopHeading": "217",
    "StopLat": "38.0214937",
    "StopLng": "23.7575178",
    "RouteStopOrder": "2",
    "StopType": "0",
    "StopAmea": "0"
  }
],
```

Εικόνα 22 Παράδειγμα API response

#### 4.2.6.2 Βιβλιοθήκη Volley

Το Volley είναι μια βιβλιοθήκη Android που αναπτύχθηκε από την Google και απλοποιεί τη διαδικασία υποβολής αιτημάτων δικτύου σε εφαρμογές Android (Google, Volley, 2022). Παρέχει έναν απλό και αποτελεσματικό τρόπο χειρισμού αιτημάτων HTTP, καθιστώντας το ιδανική επιλογή για την ενοποίηση API και τη λήψη δεδομένων από απομακρυσμένους διακομιστές. Το Volley προσφέρει πολλά βασικά χαρακτηριστικά που το κάνουν δημοφιλή στην ανάπτυξη εφαρμογών Android. Ένα από τα κύρια πλεονεκτήματά του είναι η ικανότητά του να χειρίζεται ασύγχρονες λειτουργίες δικτύου, διασφαλίζοντας ότι τα αιτήματα δικτύου δεν μπλοκάρουν το κύριο νήμα διεπαφής χρήστη και δεν προκαλούν προβλήματα απόδοσης. Αυτό καθιστά την εφαρμογή πιο αποκριτική και την αποτρέπει από το να παγώσει ή να μην ανταποκρίνεται κατά τη λειτουργία του δικτύου. Ένα ακόμη χαρακτηριστικό είναι ο μηχανισμός αυτόματης ουράς αιτημάτων και προσωρινής αποθήκευσης. Το Volley διαχειρίζεται μια ουρά αιτημάτων δικτύου, επιτρέποντας την ταυτόχρονη αποστολή πολλαπλών αιτημάτων, ενώ παράλληλα διαχειρίζεται αποτελεσματικά τις συνδέσεις δικτύου. Παρέχει επίσης ενσωματωμένους μηχανισμούς προσωρινής αποθήκευσης που μπορούν να χρησιμοποιηθούν για την προσωρινή αποθήκευση αποκρίσεων δικτύου, μειώνοντας την ανάγκη για επαναλαμβανόμενες αιτήσεις δικτύου και βελτιώνοντας τη συνολική απόδοση της εφαρμογής.

Συνεπώς, το Volley απλοποιεί τη διαδικασία υποβολής διαφορετικών τύπων αιτημάτων δικτύου, συμπεριλαμβανομένων των GET, POST, PUT, DELETE και άλλων καθώς περιέχει ένα σύνολο κλάσεων και διεπαφών που χειρίζονται επικοινωνίες δικτύου, όπως κλάσεις Request και Response για τον καθορισμό της αίτησης και την επεξεργασία της απόκρισης, αντίστοιχα. Επίσης, η βιβλιοθήκη περιλαμβάνει

υποστήριξη για την ανάλυση διαφόρων τύπων μορφών δεδομένων, όπως αρχεία JSON, XML καθώς και αρχεία εικόνας. Προσφέρει ενσωματωμένη υποστήριξη αιτημάτων JSON και XML, καθιστώντας εύκολη την ανάλυση των απαντήσεων JSON ή XML από τον διακομιστή και την εξαγωγή των απαιτούμενων δεδομένων για περαιτέρω επεξεργασία. Στην εφαρμογή υλοποιήθηκε μια κλάση Singleton με όνομα OASAService για την ενσωμάτωση του RequestQueue και των λειτουργιών του Volley. Το μοτίβο Singleton διασφαλίζει ότι υπάρχει μόνο μία παρουσία της κλάσης σε όλο τον κύκλο ζωής της εφαρμογής, προωθώντας την αποτελεσματική χρήση της μνήμης και αποφεύγοντας τις διενέξεις στα αιτήματα δικτύου. Η κλάση Singleton που ενσωματώνει το αντικείμενο RequestQueue και παρέχει μεθόδους για την υποβολή αιτημάτων, χειρισμό ακύρωσης αιτημάτων και χειρισμού σφαλμάτων.

Το επόμενο βήμα υλοποίησης είναι η μετατροπή των δεδομένων που λαμβάνονται από το API σε μοντέλα που μπορούν εύκολα να χρησιμοποιηθούν στην Android. Για αυτό το σκοπό δημιουργήθηκαν τα μοντέλα όπως Arrival, Bus, Line, Route, Schedule, ScheduleDaysML, και Stop. Αυτά τα μοντέλα ενσωματώνουν τα σχετικά δεδομένα και παρέχουν μεθόδους για την πρόσβαση και τον χειρισμό των δεδομένων αποτελεσματικά. Το πρώτο βήμα στη διαδικασία μετατροπής είναι ο σχεδιασμός και η δημιουργία των απαραίτητων κατηγοριών μοντέλων που θα αντιπροσωπεύουν τις διαφορετικές οντότητες στο σύστημα μεταφορών. Κάθε κλάση μοντέλου θα πρέπει να έχει χαρακτηριστικά που αντιστοιχούν στα σχετικά δεδομένα που ανακτώνται από το API. Οι κλάσεις μοντέλα που χρησιμοποιήθηκαν είναι οι εξής:

Arrival: περιέχει στοιχεία για τις αφίξεις των λεωφορείων.

- route\_code: κωδικός διαδρομής
- veh\_code: κωδικός λεωφορείου
- btime2: λίστα από ώρες άφιξης
- lineID: κωδικός γραμμής
- lineDesc: περιγραφή γραμμής (ελληνικά)
- lineDescEn: περιγραφή γραμμής (αγγλικά)
- Bus: περιέχει στοιχεία των λεωφορείων.
- vehicle\_num: αριθμός λεωφορείου
- location: συντεταγμένες λεωφορείου (Pair<String, String>)

Line: περιέχει στοιχεία για τις αφίξεις των λεωφορείων.

- ml\_code: master line κωδικός
- sdc\_code: sdc κωδικός
- line\_code: κωδικός γραμμής
- line\_id: αλφαριθμητικό γραμμής
- line\_desc: περιγραφή γραμμής (ελληνικά)

- line\_desc\_en: περιγραφή γραμμής (αγγλικά)

Route: περιέχει στοιχεία για τις αφίξεις των λεωφορείων.

- LineCode: αλφαριθμητικό γραμμής
- RouteCode: αλφαριθμητικό δρομολογίου
- RouteId: κωδικός δρομολογίου
- RouteActive: αλφαριθμητικό που δείχνει αν το δρομολόγιο είναι ενεργό
- RouteDescr: περιγραφή δρομολογίου (ελληνικά)
- RouteDescrEng: περιγραφή δρομολογίου (αγγλικά)
- RouteDistance: συνολική απόσταση διαδρομής δρομολογίου
- RouteType: τύπος δρομολογίου
- LineID: αλφαριθμητικό γραμμής
- LineDescr: περιγραφή γραμμής (ελληνικά)
- LineDescrEng: περιγραφή γραμμής (αγγλικά)
- MasterLineCode: κωδικός master line

Schedule: περιέχει στοιχεία για τις αφίξεις των λεωφορείων.

- timetable: λίστα από αλφαριθμητικά που δείχνουν τον χρονοδιάγραμμα μίας γραμμής.

ScheduleDaysML: περιέχει στοιχεία για τις αφίξεις των λεωφορείων.

- sdc\_code: κωδικός sdc
- sdc\_descr: περιγραφή schedule (ελληνικά)
- sdc\_descr\_eng: περιγραφή schedule (αγγλικά)

Stop: περιέχει στοιχεία για τις στάσεις των λεωφορείων.

- StopCode: κωδικός στάσης
- StopID: αλφαριθμητικό στάσης
- StopDescr: περιγραφή στάσης (ελληνικά)
- StopDescrEng: περιγραφή στάσης (αγγλικά)
- StopStreet: διεύθυνση στάσης (ελληνικά)
- StopStreetEng: διεύθυνση στάσης (ελληνικά)
- StopHeading:
- StopLat: γεωγραφικό μήκος
- StopLng: γεωγραφικό πλάτος
- RouteStopOrder: ταξινόμηση στάσεων στη γραμμή
- StopType: τύπος στάσης
- StopAmea: πρόσβαση από άτομα με ειδικές ανάγκες
- StopDistance: απόσταση στάσης

Για την βελτίωση της απόδοσης και να την μείωση των χρόνων φόρτωσης, είναι αποθηκεύσαμε μερικά από τα δεδομένα του API σε μια τοπική βάση δεδομένων SQLite. Η βάση δεδομένων SQLite χρησιμοποιήθηκε για την αποθήκευση σχετικών πληροφοριών, όπως λεπτομέρειες γραμμής, πληροφορίες διαδρομής και δεδομένα στάσεων. Εφαρμόζοντας ένα επίπεδο βάσης δεδομένων, η

εφαρμογή μπορεί να ανακτήσει δεδομένα από τη βάση δεδομένων αντί να κάνει επαναλαμβανόμενα αιτήματα API, με αποτέλεσμα ταχύτερη πρόσβαση στα δεδομένα και πιο ομαλή εμπειρία χρήστη.

lines		stops	
ml_code	text	stop_code	text
sd_code	text	stop_id	text
line_id	text	stop_desc	text
line_code	text	stop_desc_en	text
line_desc	text	stop_street	text
line_desc_eng	text	stop_street_en	text
		stop_heading	text
		stop_lat	text
		stop_lng	text
		route_stop_order	text
		stop_type	text
		stop_amea	text
		stop_dist	text

Εικόνα 23 Διάγραμμα βάσης δεδομένων

Τέλος, έγινε χρήση των κοινόχρηστων προτιμήσεων (Shared Preferences) για την αποθήκευση των προτιμήσεων των χρηστών. Οι Κοινόχρηστες Προτιμήσεις επιτρέπουν στους χρήστες να αποθηκεύουν τις αγαπημένες τους στάσεις ή γραμμές, καθώς και τις προσαρμογές της διεπαφής χρήστη σύμφωνα με τις προτιμήσεις τους. Χρησιμοποιώντας τις Κοινόχρηστες Προτιμήσεις, η εφαρμογή προσφέρει μια εξατομικευμένη εμπειρία για κάθε χρήστη.

#### 4.2.7 Ευφυείς πράκτορες για προσαρμοστικό UI

Κατά την ανάπτυξη της εφαρμογής μας, ήταν σημαντικό να παρέχεται μια διεπαφή χρήστη που να ανταποκρίνεται στις διαφορετικές ανάγκες και προτιμήσεις διαφορετικών κατηγοριών χρηστών. Αυτό το κεφάλαιο διερευνά την ενσωμάτωση έξυπνων πρακτόρων στη διαδικασία ανάπτυξης της προσαρμογής της διεπαφής χρήστη με βάση τις αλληλεπιδράσεις των χρηστών και συγκεκριμένους κανόνες. Δημιουργήθηκε μια κλάση με το όνομα ReflexAgent, η οποία προκαλεί τις προσαρμογές στη διεπαφή χρήστη και εφαρμόζει λογική βασισμένη σε κανόνες για την τροποποίηση των στοιχείων διεπαφής χρήστη.

Αρχικά, οι ευφυείς πράκτορες μπορούν να αναγνωρίσουν μοτίβα και να λάβουν τεκμηριωμένες αποφάσεις σχετικά με τον τρόπο προσαρμογής της διεπαφής χρήστη ώστε να ανταποκρίνεται στις ανάγκες του χρήστη. Αυτοί οι πράκτορες ενεργούν ως υπεύθυνοι λήψης αποφάσεων, αναλύοντας δεδομένα χρήστη, πληροφορίες σχετικά με τα συμφραζόμενα και προκαθορισμένους κανόνες για την

προσαρμογή της διεπαφής χρήστη σε πραγματικό χρόνο. Ενσωματώνοντας ευφυείς πράκτορες, η διεπαφή χρήστη μπορεί να προσφέρει μια εξατομικευμένη και προσαρμοσμένη εμπειρία για κάθε χρήστη, ενισχύοντας τη χρηστικότητα και τη συνολική ικανοποίηση των χρηστών.

Συγκεκριμένα σε αυτή την εφαρμογή συγκεντρώνουν δεδομένα σχετικά με τη συμπεριφορά, τις προτιμήσεις και το πλαίσιο των χρηστών, επιτρέποντάς τους να κατανοήσουν τις ανάγκες και τις προθέσεις του χρήστη. Αυτά τα δεδομένα μπορεί να περιλαμβάνουν παράγοντες όπως η τοποθεσία του χρήστη, οι προηγούμενες ενέργειες και τα ιστορικά μοτίβα χρήσης. Με τη συνεχή παρακολούθηση και ανάλυση αυτών των δεδομένων, οι ευφυείς πράκτορες μπορούν να λαμβάνουν τεκμηριωμένες αποφάσεις σχετικά με τον τρόπο προσαρμογής της διεπαφής χρήστη για την καλύτερη εξυπηρέτηση του χρήστη. Ένα σημαντικό χαρακτηριστικό των ευφύων πρακτόρων στην εφαρμογή είναι η ικανότητά τους να αιτιολογούν και να λαμβάνουν αποφάσεις με βάση προκαθορισμένους κανόνες. Αυτοί οι κανόνες ορίζουν τις προϋποθέσεις και τις αντίστοιχες ενέργειες για την προσαρμογή της διεπαφής χρήστη. Για παράδειγμα, εάν ένας χρήστης αναζητά συχνά συγκεκριμένες διαδρομές λεωφορείων, ο ευφυής πράκτορας μπορεί να δώσει προτεραιότητα στην εμφάνιση αυτών των διαδρομών ή να παρέχει γρήγορη πρόσβαση σε αυτές. Οι κανόνες αυτοί βασίζονται σε κατηγορίες χρηστών, προτιμήσεις ή συγκεκριμένους, διασφαλίζοντας ότι οι τροποποιήσεις της διεπαφής χρήστη είναι σχετικές και χρήσιμες για κάθε χρήστη. Στον Πίνακα 6 εμφανίζονται αναλυτικά οι κανόνες και οι ενέργειες που συμβαίνουν στο σύστημα για την τροποποίηση της διεπαφής χρήστη.

Πίνακας 6 Κανόνες και ενέργειες προσαρμογής

ID	Κανόνας	Περιγραφή κανόνα	Συνθήκη	Περιγραφή συνθήκης	Ενέργειες
1	user_not_found_information	Ο χρήστης δεν βρήκε πληροφορίες			Συμβουλές χρήσης
2	user_missed_clicks	Μεγάλος αριθμός πατημάτων σε σημείων χωρίς κουμπιά (missed clicks)	missedClicks > 3	Καταγραφή πλήθος λάθος πατημάτων στο onTouch	Εμφάνιση επαρκής ανατροφοδότησης (haptics, vibration, audio). Αύξηση φωτεινότητας οθόνης. Λειτουργία μεγεθυντικού φακού κατά το άγγιγμα. Λειτουργία φωνητικής πλοήγησης.
3	user_is_idle	Μεγάλη διάρκεια χρήσης της εφαρμογής	runTime > 120s	Χρόνος έναρξης και τερματισμού εφαρμογής onCreate και onDestroy	Αύξηση φωτεινότητας οθόνης. Συμβουλές χρήσης. Επαρκής ανατροφοδότησης (haptics, vibration, audio).
4	user_input_delay	Παύση μεταξύ δύο πατημάτων	DelayBetweenTouch > 10s	Χρόνος καθυστέρησης μεταξύ δύο πατήματα ενεργειών onTouch	Αύξηση φωτεινότητας οθόνης. Φιλική γραμματοσειρά σε δυσλεξικού χρήστες (OpenDyslexic Font).
5	user_long_press	Ο χρήστης εκτελεί παρατεταμένο πάτημα	clickDuration > 2s	Χρόνος έναρξης και λήξης πατήματος onTouch	Λειτουργία μεγεθυντικού φακού κατά το άγγιγμα.
6	user_near_presses	Συντεταγμένες του αγγιγμάτων που πλησιάζουν σε κουμπί χωρίς πάτημα	nearPresses > 2	Σύγκριση συντεταγμένων με το περίγραμμα των κουμπιών	Αύξηση φωτεινότητας οθόνης. Συνδυασμός οπτικοακουστικών συστημάτων για την πλοήγηση της εφαρμογής (haptics, vibration, audio). Ενεργοποίηση λειτουργίας high contrast.
7	user_attempts_zooming	Προσπάθεια μεγέθυνσης της οθόνης (zoom)	zoomGesture = true	Ανίχνευση κίνησης δύο αγγιγμάτων και zoom κίνησης	Λειτουργία μεγεθυντικού φακού κατά το άγγιγμα. Συμβουλές χρήσης.
8	user_attempts_scrolling	Προσπάθεια κύλισης σε σημεία χωρίς την δυνατότητα	timesTriesToMove > 3	Ανίχνευση κύλισης του αγγιγματος σε στατικά activities	Συνδυασμός οπτικοακουστικών συστημάτων για την πλοήγηση της εφαρμογής (haptics, vibration, audio). Συμβουλές χρήσης.

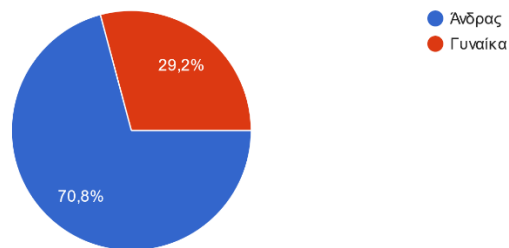
## 5 Αξιολόγηση εφαρμογής

Ο έλεγχος μιας εφαρμογής είναι ζωτικής σημασίας για τη δημιουργία μιας ανθρωποκεντρικής εφαρμογής. Διαδραματίζει σημαντικό ρόλο στην κατανόηση του πόσο καλά ευθυγραμμίζεται η εφαρμογή με τις ανάγκες, τις προτιμήσεις και τις προσδοκίες των χρηστών. Με τη συλλογή σχολίων από χρήστες και τη διεξαγωγή διεξοδικών ελέγχων, μπορούν να εντοπιστούν οι τομείς προς βελτίωση, να επικυρωθούν τις αποφάσεις σχεδιασμού και να διασφαλισθεί ότι η εφαρμογή πληροί τους επιδιωκόμενους στόχους παροχής της θετικής εμπειρίας του χρήστη.

Στην λήξη της ανάπτυξης της εφαρμογής, διαμοιράστηκε και πραγματοποιήθηκε η συλλογή σχολίων και κριτικών από χρήστες, συμπεριλαμβανομένων γνωστών, φίλων και φοιτητών του Πανεπιστημίου Δυτική Αττικής, για την αξιολόγηση της απόδοσής της και την ικανοποίηση των χρηστών.

Ακολουθεί μια κριτική της εφαρμογής με βάση τις εμπειρίες τους.

Επιλέξτε το φύλο σας  
24 απαντήσεις

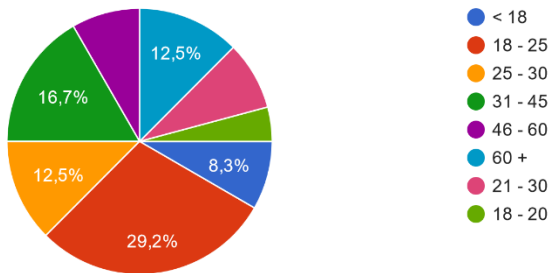


Γράφημα 1 Απεικόνιση φύλου

Το Γράφημα 1 μας εμφανίζει ότι από τα 25 άτομα που δοκίμασαν την εφαρμογή, το σύνολο των ατόμων που απάντησαν στο ερωτηματολόγιο ήταν 24, από τους οποίους οι 17 ήταν άντρες και οι 7 ήταν γυναίκες.

Οι συμμετέχοντες στην έρευνα που αξιολογούν την εφαρμογή καλύπτουν ένα ευρύ φάσμα ηλικιακών ομάδων, ενισχύοντας την σημασία των προσαρμοστικών λειτουργιών. Όπως φαίνεται στο Γράφημα 2, οι δύο ηλικιακές ομάδες αποτελούν την πλειοψηφία, είναι η ηλικιακή ομάδα 18 - 25 ετών με 7 και η ηλικιακή ομάδα 31-45 ετών με 4 ανήκουν. Επιπλέον, 3 άτομα ανήκουν στην ηλικιακή ομάδα άνω των 60, ενώ οι υπόλοιποι συμμετέχοντες είναι είτε 46-60 ετών είτε κάτω των 18 ετών.

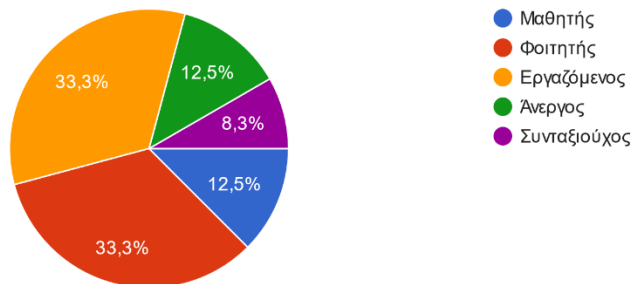
Ποια είναι η ηλικία σας;  
24 απαντήσεις



Γράφημα 2 Απεικόνιση Εύρους Ηλικίας

Σύμφωνα με το Γράφημα 3 η πλειοψηφία των χρηστών είναι είτε φοιτητές είτε εργαζόμενοι με επόμενη κατηγορία τον μαθητή και τελευταία τον συνταξιούχο.

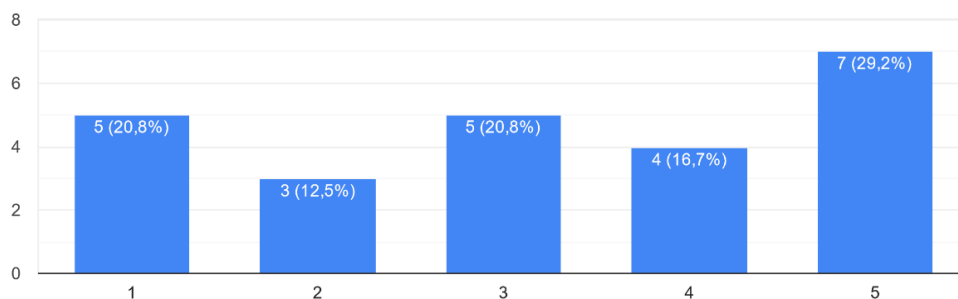
Τι επαγγέλεστε;  
24 απαντήσεις



Γράφημα 3 Απεικόνιση επαγγέλματος

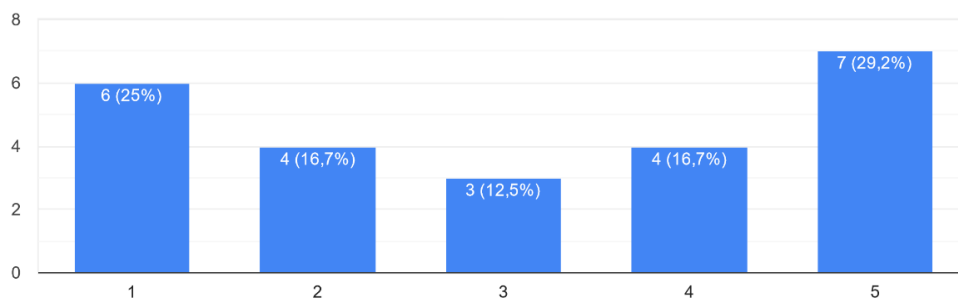


Πώς θα αξιολογούσατε την τεχνολογική σας εμπειρία;  
24 απαντήσεις



Γράφημα 4 Απεικόνιση τεχνολογικής εμπειρίας

Πόσο συχνά χρησιμοποιείτε την τεχνολογία στο επάγγελμά σας;  
24 απαντήσεις



Γράφημα 5 Απεικόνιση τεχνολογίας στο επάγγελμα

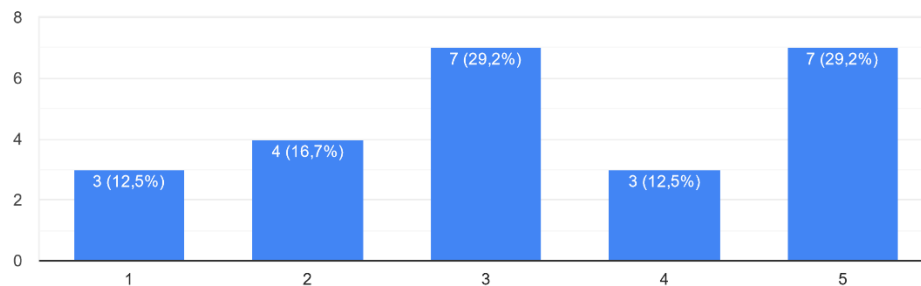
Τα Γράφημα 4 και Γράφημα 5 απεικονίζουν το εύρος των χρηστών με καθόλου και με αρκετά καλή τεχνολογική γνώση. Οι χρήστες που έχουν συνηθίσει να χρησιμοποιούν την τεχνολογία στην εργασία και την καθημερινή τους ζωή μπορεί να έχουν υψηλότερες προσδοκίες για την αποτελεσματικότητα, λειτουργικότητα και ευκολία χρήσης στις εφαρμογές για κινητά που χρησιμοποιούν. Επιπλέον, οι χρήστες που είναι συνηθισμένοι να χρησιμοποιούν εκτενώς την τεχνολογία έχουν συχνά συγκεκριμένες προτιμήσεις και συνήθειες που αναμένουν ότι οι εφαρμογές για κινητές συσκευές θα ικανοποιούν που περιλαμβάνει προσαρμόσιμες ρυθμίσεις, επιλογές εξατομίκευσης ή προσαρμοστικό UI.

Αντίθετα, οι χρήστες που έχουν περιορισμένη έκθεση στην τεχνολογία ενδέχεται να δυσκολεύονται να πλοηγηθούν στη διεπαφή της εφαρμογής, να κατανοήσουν τις δυνατότητές της ή να χρησιμοποιήσουν

αποτελεσματικά τις λειτουργίες της. Αυτή η έλλειψη εξοικείωσης με την τεχνολογία μπορεί να οδηγήσει σε απογοήτευση και να εμποδίσει τη συνολική εμπειρία τους με την εφαρμογή για κινητά.

Επιπλέον, οι χρήστες που δεν είναι συνηθισμένοι στη χρήση τεχνολογίας μπορεί να έχουν χαμηλότερες προσδοκίες ή διαφορετικές προτιμήσεις όσον αφορά τη χρηστικότητα και το σχεδιασμό της εφαρμογής. Μπορεί να προτιμούν απλούστερες διεπαφές, σαφέστερες οδηγίες ή πιο εύχρηστες αλληλεπιδράσεις.

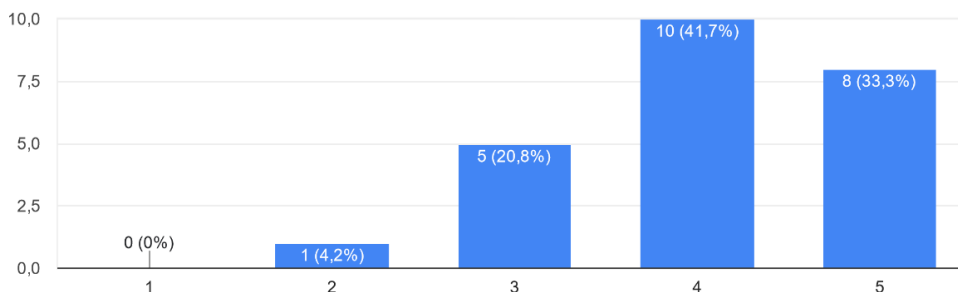
Πόσο εξοικειωμένοι είστε με το θέμα της εφαρμογής;  
24 απαντήσεις



Γράφημα 6 Εξοικείωση με το θέμα

Όπως φαίνεται από το Γράφημα 6 οι περισσότεροι χρήστες είναι εξοικειωμένοι με το θέμα του OASA έχοντας εμπειρία με τις εφαρμογές MMM. Η εξοικείωση αυτή πιθανά να τους επιτρέπει να πλοηγούνται στην εφαρμογή με ευκολία. Είναι πιθανό να κατανοήσουν την ορολογία, τις έννοιες και τα χαρακτηριστικά πιο γρήγορα, οδηγώντας σε μια μειωμένη καμπύλη μάθησης. Αυτή η εξοικείωση τους επιτρέπει να αλληλεπιδρούν με την εφαρμογή με μεγαλύτερη αυτοπεποίθηση, διευκολύνοντάς τους να επιτύχουν τους στόχους και τις εργασίες τους. Επιπλέον, η εξοικείωση με το θέμα επιτρέπει στους χρήστες να έχουν μια πιο ξεκάθαρη εικόνα του τι θέλουν από την εφαρμογή. Μπορούν να θέσουν συγκεκριμένους στόχους και προτιμήσεις με βάση τις προηγούμενες γνώσεις τους, επιτρέποντάς τους να χρησιμοποιούν πιο αποτελεσματικά τις δυνατότητες της εφαρμογής. Αυτή η ευθυγράμμιση μεταξύ των προσδοκιών των χρηστών και των δυνατοτήτων της εφαρμογής συμβάλλει σε μια πιο ικανοποιητική εμπειρία. Ωστόσο, είναι σημαντικό να σημειωθεί ότι η εξοικείωση δεν πρέπει να αποτελεί προϋπόθεση για τη χρήση της εφαρμογής όπως φαίνονται στα γραφήματα αξιολόγησης χρήσης. Η διαισθητική και φιλική προς το χρήστη διεπαφή γεφυρώνει το χάσμα για τους χρήστες που είναι λιγότερο εξοικειωμένοι με το θέμα.

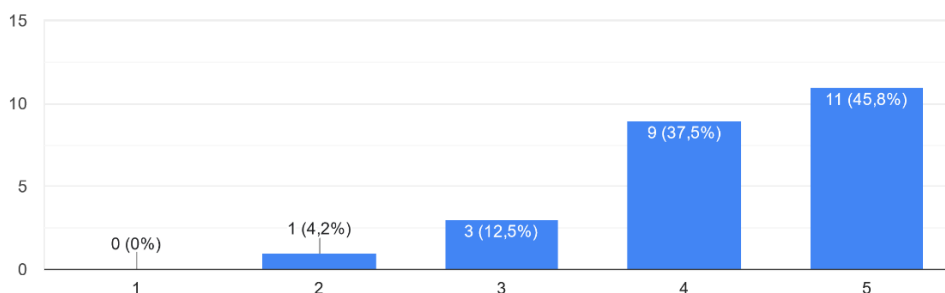
Πόσο εύκολα μπορούσατε να πλοηγηθείτε στην εφαρμογή και να κατανοήσετε τα διάφορα χαρακτηριστικά της;  
24 απαντήσεις



Γράφημα 7 Πλοήγηση και κατανόηση λειτουργιών

Σύμφωνα με το Γράφημα 7 η εφαρμογή έχει συγκεντρώσει ως επί το πλείστον θετικές κριτικές, με τους χρήστες να εκφράζουν την ευκολία κατά την πλοήγηση στην εφαρμογή και την κατανόηση των διάφορων λειτουργιών της. Ο ανθρωποκεντρικός σχεδιασμός και η διαισθητική διεπαφή της εφαρμογής έχουν παίξει σημαντικό ρόλο στη βελτίωση της συνολικής εμπειρίας χρήστη. Εστιάζοντας στη διαισθητική πλοήγηση, τις οπτικές ενδείξεις και τις κατανοητές επεξηγήσεις χαρακτηριστικών, η εφαρμογή έχει δημιουργήσει ένα περιβάλλον όπου οι χρήστες αισθάνονται σίγουροι για τις αλληλεπιδράσεις τους. Αυτή η προσοχή στη χρηστικότητα και την εμπειρία χρήστη έχει αναμφίβολα συνεισφέρει στη θετική υποδοχή της εφαρμογής και στην υψηλή ικανοποίηση των χρηστών.

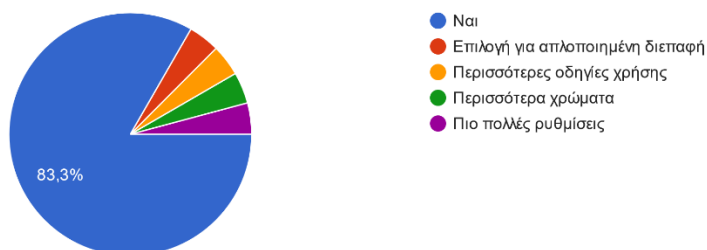
Βρήκατε τη διεπαφή χρήστη της εφαρμογής προσαρμοστική και ανταποκρίνεται στις ανάγκες σας;  
24 απαντήσεις



Γράφημα 8 Αξιολόγηση UI

Σύμφωνα με το Γράφημα 8 οι χρήστες επαίνεσαν την εφαρμογή για την ικανότητά της να προσαρμόζεται και να ανταποκρίνεται άμεσα ενισχύοντας τη συνολική εμπειρία και την ικανοποίησή τους. Επιπλέον, οι χρήστες επαίνεσαν την εφαρμογή για την προληπτική της φύση στην πρόβλεψη των αναγκών τους με την προσπάθεια να «μαθαίνει» από τη συμπεριφορά και τις προτιμήσεις των χρηστών, παρέχοντας εξατομικευμένες προτάσεις και συντομεύσεις.

Καταφέρατε να προσαρμόσετε τη διεπαφή ή τις ρυθμίσεις της εφαρμογής ώστε να ταιριάζουν καλύτερα στις προτιμήσεις σας; Εάν όχι, ποιες επιπλέον επιλογές προσαρμογής θα θέλατε να δείτε;  
24 απαντήσεις



Γράφημα 9 Αξιολόγηση προσαρμοστικότητας UI

Με βάση το Γράφημα 9 και Γράφημα 10 οι χρήστες έχουν εκφράσει την ικανοποίησή τους για το διαθέσιμο επίπεδο προσαρμογής, δίνοντάς τους τη δυνατότητα να δημιουργήσουν μια εξατομικευμένη εμπειρία χρήστη. Ωστόσο, ορισμένοι χρήστες εξέφρασαν την επιθυμία για πρόσθετες επιλογές προσαρμογής για να βελτιώσουν περαιτέρω την εμπειρία τους.

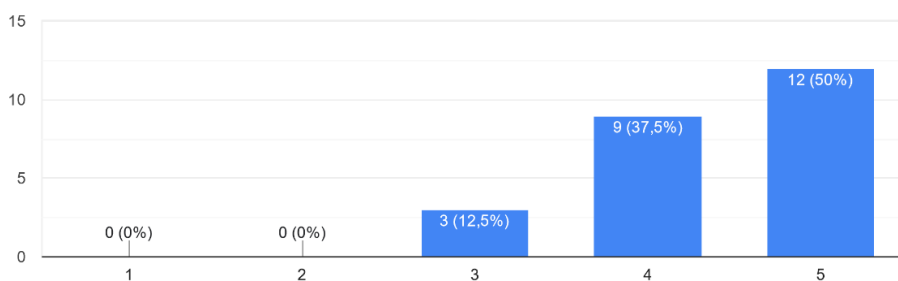
Μία από τις προτεινόμενες επιλογές είναι μια ρύθμιση για μια απλούστερη διεπαφή χρήστη (UI) που θα αφαιρεί την περιττή ακαταστασία και βελτιστοποιεί τα γραφικά της εφαρμογής. Αυτό θα ωφελούσε τους χρήστες που προτιμούν έναν μινιμαλιστικό και απλό σχεδιασμό, ενισχύοντας την ευκολία χρήσης τους.

Οι χρήστες εξέφρασαν επίσης την επιθυμία για περισσότερους οδηγούς χρήσης και σεμινάρια εντός της εφαρμογής. Πρόσθετες οδηγίες και συμβουλές εργαλείων μπορούν να βοηθήσουν τους χρήστες να ανακαλύψουν κρυφές λειτουργίες, να κατανοήσουν τις προηγμένες λειτουργίες και να αξιοποιήσουν στο έπακρο την εφαρμογή. Αυτοί οι οδηγοί χρήσης θα ήταν ιδιαίτερα πολύτιμοι για νέους χρήστες ή χρήστες που θέλουν να εξερευνήσουν τις δυνατότητες της εφαρμογής εκτενέστερα.

Μια άλλη επιλογή προσαρμογής που ανέφεραν οι χρήστες είναι η διαθεσιμότητα περισσότερων χρωμάτων διεπαφής χρήστη. Η παροχή μιας ευρύτερης γκάμας επιλογών χρωμάτων θα επέτρεπε στους χρήστες να επιλέξουν ένα συνδυασμό χρωμάτων που αντηχεί με τις προσωπικές τους προτιμήσεις ή προσφέρει καλύτερη αναγνωσιμότητα για άτομα με προβλήματα όρασης. Αυτή η αυξημένη ευελιξία στις

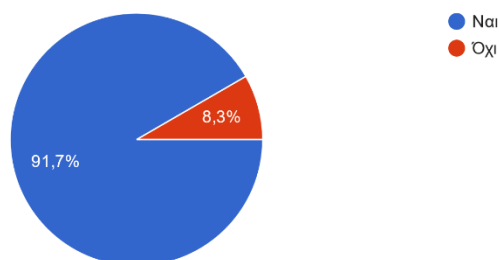
επιλογές χρωμάτων μπορεί να συμβάλει σε μια πιο ευχάριστη και προσιτή εμπειρία χρήστη. Τέλος, οι χρήστες εξέφρασαν ενδιαφέρον για πρόσθετες επιλογές προσαρμογής στις ρυθμίσεις της εφαρμογής. Αυτό θα μπορούσε να περιλαμβάνει πιο λεπτομερή έλεγχο σε διάφορες πτυχές, όπως το μέγεθος γραμματοσειράς, τα στυλ εικονιδίων ή τη προσαρμογή χειρονομιών. Επεκτείνοντας στο μέλλον το εύρος των διαθέσιμων ρυθμίσεων προσαρμογής, οι χρήστες θα έχουν περισσότερο έλεγχο στην προσαρμογή της εφαρμογής στις συγκεκριμένες προτιμήσεις και τα μοτίβα χρήσης τους.

Πόσο ικανοποιημένοι είστε με τη συνολική εμπειρία χρήστη της εφαρμογής για την κάλυψη των μεταφορικών σας αναγκών;  
24 απαντήσεις



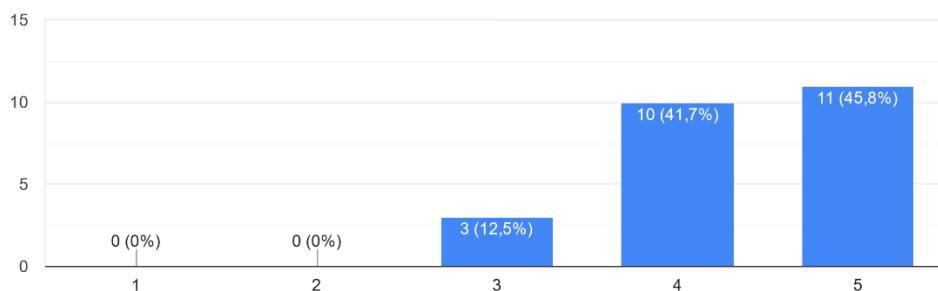
Γράφημα 10 Αξιολόγηση συνολικής εμπειρίας χρήστη

Προσέφερε η εφαρμογή πρόσθετες λειτουργίες ή υπηρεσίες που βελτίωσαν τη συνολική εμπειρία σας με τα μέσα μαζικής μεταφοράς;  
24 απαντήσεις



Γράφημα 11 Αξιολόγηση λειτουργιών

Πόσο καλά ανταποκρίθηκε η εφαρμογή στους συγκεκριμένους στόχους και απαιτήσεις σας όσον αφορά την πρόσβαση σε πληροφορίες λεωφορείων και δημόσιων συγκοινωνιών;  
24 απαντήσεις

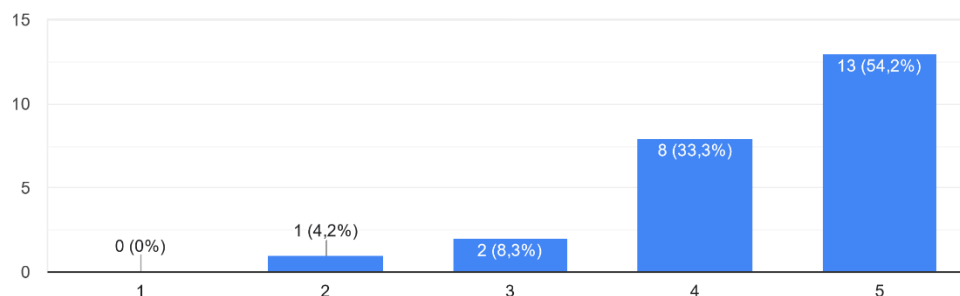


Γράφημα 12 Αξιολόγηση στόχων και απαιτήσεων

Στην συνέχεια, τα Γράφημα 11 και Γράφημα 12 απεικονίζουν ότι η πλειοψηφία των χρηστών έχουν εκφράσει υψηλά επίπεδα ικανοποίησης με την ικανότητα της εφαρμογής να παρέχει μια απρόσκοπτη και αποτελεσματική εμπειρία μεταφοράς. Δίνοντας προτεραιότητα στις ανάγκες των χρηστών και προσπαθώντας συνεχώς να βελτιώνει τη λειτουργικότητα της εφαρμογής, η εφαρμογή κατάφερε να προσφέρει μια εστιασμένη στο χρήστη και ικανοποιητική εμπειρία χρήσης.

Το Γράφημα 13 φανερώνει ότι οι χρήστες εκφράζουν μεγάλη πιθανότητα να συστήσουν την εφαρμογή σε άλλους που χρησιμοποιούν τα μέσα μαζικής μεταφοράς. Συνεπώς, ο ανθρωποκεντρικός σχεδιασμός της εφαρμογής και οι νέες λειτουργίες της έχουν οδηγήσει σε ισχυρή υποστήριξη από τους χρήστες της.

Πόσο πιθανό είναι να προτείνετε την εφαρμογή σε άλλους που χρησιμοποιούν τα μέσα μαζικής μεταφοράς;  
24 απαντήσεις



Γράφημα 13 Απεικόνιση προτεινόμενων νέων χρηστών

Έχετε προτάσεις για τη βελτίωση των λειτουργιών προσβασιμότητας της εφαρμογής ή της συνολικής εμπειρίας χρήστη;

4 απαντήσεις

Λειτουργία βελτιστής διαδρομής από ένα σημείο στο άλλο

Βελτίωση της απόδοσης της εφαρμογής

εμπλουτισμός εφαρμογής με καταγραφή σχολίων για την αναφορά προβλημάτων και ζήτηση βοήθειας

Κάπως πιο απλή δομή της εφαρμογής

#### Γράφημα 14 Προτάσεις βελτίωσης

Στο Γράφημα 14 η εφαρμογή έχει λάβει πολύτιμες προτάσεις για τη βελτίωση των δυνατοτήτων προσβασιμότητας και της συνολικής εμπειρίας χρήστη, αποδεικνύοντας τη δέσμευση της εφαρμογής να ανταποκρίνεται στις ανάγκες των χρηστών και να βελτιώνει τη λειτουργικότητά της.

Μία από τις προτάσεις που παρέχονται είναι να η δημιουργίας δυνατότητας σχεδιασμού διαδρομής της εφαρμογής από το ένα σημείο στο άλλο. Οι χρήστες έχουν τονίσει την έλλειψη της λειτουργίας αυτής καθώς είναι σημαντική η δημιουργία ενός αποτελεσματικού και ακριβούς υπολογισμού της διαδρομής που λαμβάνει υπόψη παράγοντες όπως οι συνθήκες κυκλοφορίας, οι εναλλακτικοί τρόποι μεταφοράς και οι ενημερώσεις σε πραγματικό χρόνο.

Μια ακόμη πρόταση των χρηστών είναι η βελτίωση της συνολικής απόδοσης της εφαρμογής. Οι χρήστες έχουν εκφράσει την επιθυμία για ταχύτερους χρόνους φόρτωσης, ομαλότερη μετάβαση μεταξύ οθονών και βελτιωμένη απόκριση στις αλληλεπιδράσεις των χρηστών. Η βελτίωση της απόδοσης της εφαρμογής θα εξασφάλιζε μια απρόσκοπτη και ρευστή εμπειρία χρήστη, μειώνοντας την απογοήτευση και αυξάνοντας την ικανοποίηση των χρηστών.

Οι χρήστες συνέστησαν επίσης τον εμπλουτισμό της εφαρμογής με λειτουργίες που τους επιτρέπουν να καταγράψουν σχόλια, να αναφέρουν προβλήματα και να ζητούν βοήθεια μέσα από την εφαρμογή. Με την ενσωμάτωση ενός μηχανισμού ανάδρασης (feedback) στην εφαρμογή, οι χρήστες θα μπορούν εύκολα να επικοινωνήσουν τις εμπειρίες τους, να αναφέρουν προβλήματα που αντιμετωπίζουν και να αναζητήσουν άμεση βοήθεια όταν χρειάζεται. Αυτή η διαδραστική λειτουργία θα ενθάρρυνε την αίσθηση δέσμευσης και συμμετοχής, επιτρέποντας στην εφαρμογή να αντιμετωπίζει τις ανησυχίες των χρηστών και να παρέχει την απαραίτητη υποστήριξη.

Επιπλέον, οι χρήστες έχουν προτείνει την απλοποίηση της δομής της εφαρμογής για τη βελτίωση της χρηστικότητας. Με την απλοποίηση της πλοήγησης μέσα στην εφαρμογή, οι χρήστες πιστεύουν ότι η συνολική εμπειρία χρήστη θα ήταν πιο διαισθητική και φιλική προς το χρήστη. Η απλοποίηση της δομής της εφαρμογής θα βοηθούσε τους χρήστες να βρίσκουν εύκολα τις επιθυμητές λειτουργίες και λειτουργίες, μειώνοντας το γνωστικό φορτίο και βελτιώνοντας την αποτελεσματικότητα στη χρήση της εφαρμογής.

Συμπερασματικά, οι προτάσεις για τη βελτίωση των δυνατοτήτων της εφαρμογής και της συνολικής εμπειρίας χρήστη καταδεικνύουν τη δεκτικότητα της εφαρμογής στα σχόλια των χρηστών και τη δέσμευσή της για συνεχή βελτίωση. Με την ενσωμάτωση βελτιστοποιήσεων στη λειτουργία σχεδιασμού διαδρομής, τη βελτίωση της απόδοσης της εφαρμογής, την ενσωμάτωση μηχανισμών ανάδρασης και την απλοποίηση της δομής της εφαρμογής, η εφαρμογή μπορεί να προσφέρει μια πιο προσιτή και ανθρωποκεντρική εμπειρία. Λαμβάνοντας υπόψη αυτές τις προτάσεις μπορεί να καταλήξουμε σε μια εφαρμογή που ανταποκρίνεται στις εξελισσόμενες ανάγκες και προσδοκίες των χρηστών της, οδηγώντας τελικά σε μεγαλύτερη ικανοποίηση των χρηστών.



## 6 Συμπεράσματα

Σε αυτό το έργο, έχει αναπτυχθεί μια ανθρωποκεντρική εφαρμογή Android που στοχεύει στη βελτίωση της προσβασιμότητας και στη βελτίωση της εμπειρίας του χρήστη για τα μέσα μαζικής μεταφοράς. Η εφαρμογή χρησιμοποιεί προσαρμοστικές τεχνικές διεπαφής χρήστη για να προσαρμόσει δυναμικά τη διεπαφή της με βάση τις προτιμήσεις των χρηστών, τις δυνατότητες της συσκευής και τους παράγοντες περιβάλλοντος. Ο πρωταρχικός στόχος ήταν να γίνουν οι δημόσιες συγκοινωνίες πιο προσβάσιμες και βολικές για ένα ευρύ φάσμα χρηστών. Καθ' όλη τη διάρκεια του έργου, η εστίαση βρισκόταν γύρω από την ενσωμάτωση αρχών σχεδιασμού με επίκεντρο τον χρήστη, αξιοποιώντας προσαρμοστικές στρατηγικές διεπαφής χρήστη και χρησιμοποιώντας τις πιο σύγχρονες τεχνολογίες για να δημιουργήσουμε μια απρόσκοπτη και χωρίς αποκλεισμούς εμπειρία. Η εφαρμογή αντιμετώπισε, με σχετική επιτυχία, αρκετές προκλήσεις προσβασιμότητας και παρείχε, μια σειρά χαρακτηριστικών για την ενίσχυση της χρηστικότητας και της προσβασιμότητας, σε όλους τους χρήστες, συμπεριλαμβανομένων και εκείνων με αναπηρίες.

Με την εκτέλεση του προσαρμοστικού περιβάλλοντος εργασίας χρήστη, η εφαρμογή ήταν σε θέση να προσαρμόσει δυναμικά τη διάταξη, τις γραμματοσειρές, τους συνδυασμούς χρωμάτων και τα μοτίβα αλληλεπίδρασης με βάση τις προτιμήσεις των χρηστών και τους παράγοντες περιβάλλοντος, όπως οι δυνατότητες της συσκευής και η τοποθεσία. Αυτό επέτρεψε στους χρήστες να προσαρμόσουν την εμπειρία τους, διευκολύνοντας την περιήγησή τους στην εφαρμογή, να αποκτήσουν πρόσβαση σε σχετικές πληροφορίες και να αλληλεπιδράσουν με διάφορες λειτουργίες. Επιπλέον, η ενσωμάτωση δεδομένων των δημόσιων συγκοινωνιών σε πραγματικό χρόνο μέσω API, όπως το OASA Telematics API, επέτρεψε στους χρήστες να έχουν πρόσβαση σε ενημερωμένες πληροφορίες σχετικά με διαδρομές, δρομολόγια και τοποθεσίες λεωφορείων. Η παραπάνω διαδικασία βελτίωσε την ακρίβεια και την αξιοπιστία των πληροφοριών της εφαρμογής, βοηθώντας τους χρήστες να προγραμματίσουν τα ταξίδια τους πιο αποτελεσματικά και να παραμείνουν ενημερωμένοι για τυχόν διακοπές ή καθυστερήσεις. Ωστόσο, παρά την πρόοδο που έχει σημειωθεί όσον αφορά την ανάπτυξη μιας προσβάσιμης και φιλικής προς τον χρήστη εφαρμογής, εξακολουθούν να υπάρχουν τομείς που επιδέχονται περαιτέρω βελτίωση. Ορισμένοι περιορισμοί, όπως η ελλιπής λειτουργία υποστήριξης ξένης γλώσσας πέραν της Αγγλικής ή η απόδοση σε συσκευές χαμηλού επιπέδου και πιθανές ανησυχίες σχετικά με το απόρρητο των δεδομένων, πρέπει να αντιμετωπιστούν σε μελλοντικές εργασίες για την εξέλιξη της εφαρμογής.

Η μεθοδολογία ενσωμάτωσης προβλέψεων μηχανικής μάθησης με τη διεπαφή χρήστη μιας εφαρμογής Android για την παροχή εξατομικευμένων προτάσεων προσφέρει σημαντικά οφέλη για τους χρήστες.

Αξιοποιώντας το ιστορικό αλληλεπίδρασης του χρήστη και χρησιμοποιώντας ένα εκπαιδευμένο μοντέλο μηχανικής εκμάθησης, η εφαρμογή μπορεί να προβλέψει την πιο πιθανή γραμμή λεωφορείου με βάση παράγοντες όπως η τοποθεσία, η ώρα της ημέρας και η ημέρα της εβδομάδας. Αυτή η πρόβλεψη στη συνέχεια ενσωματώνεται στη διεπαφή χρήστη, τοποθετώντας την στην κορυφή της λίστας γραμμών λεωφορείων και συνοδεύεται από προτεινόμενο κείμενο. Αυτή η προσέγγιση απλοποιεί την εμπειρία του χρήστη, επιτρέποντας γρήγορη και εύκολη πρόσβαση στην προβλεπόμενη γραμμή χωρίς την ανάγκη χειροκίνητης αναζήτησης ή κύλισης.

## 6.1 Περιορισμοί

Σε αυτή την υπό-ενότητα αναφέρονται οι περιορισμοί που συναντήσαμε κατά την ανάπτυξη της εφαρμογής. Αρχικά, οι περιορισμοί δεδομένων και πληροφοριών αφορούν την ακρίβεια και την κάλυψη των δεδομένων. Η εφαρμογή βασίζεται στην ακρίβεια και τη διαθεσιμότητα των δεδομένων που παρέχονται από το ανοιχτό API τηλεματικής του ΟΑΣΑ. Εάν υπάρχουν οποιεσδήποτε ανακρίβειες ή καθυστερήσεις στα δεδομένα, μπορεί να επηρεάσει τις πληροφορίες σε πραγματικό χρόνο που εμφανίζονται στους χρήστες, όπως ώρες άφιξης λεωφορείων ή ενημερώσεις διαδρομής. Επιπλέον, υπάρχουν περιορισμοί ενσωμάτωσης που αφορούν την ενοποίηση της εφαρμογής με εξωτερικές υπηρεσίες, όπως API τηλεματικής του ΟΑΣΑ. Ωστόσο, οποιεσδήποτε αλλαγές στη δομή του API θα επηρεάσουν τη λειτουργικότητα και τη διαθεσιμότητα δεδομένων της εφαρμογής. Επιπλέον, υπάρχουν τα συγκεκριμένα μειονεκτήματα, ο περιορισμός της γλώσσας και της τοπικής προσαρμογής. Η εφαρμογή είναι διαθέσιμη μόνο σε δύο γλώσσες (ελληνικά και αγγλικά) και δεν διαθέτει τοπική προσαρμογή για συγκεκριμένες περιοχές, έτσι ενδέχεται να δημιουργηθεί εμπόδιο στη χρηστικότητα και στην προσβασιμότητά της εφαρμογής για χρήστες που προτιμούν ή απαιτούν διαφορετικές επιλογές γλώσσας και τοποθεσίας. Άλλο ένα, λειτουργικό, πρόβλημα που υπάρχει στην εφαρμογή βρίσκεται στη σύνδεση του διαδικτύου, δηλαδή, οι χρήστες, που δεν έχουν πρόσβαση στο διαδίκτυο, δεν έχουν πρόσβαση και σε ορισμένες πληροφορίες όπως στάσεις γραμμών, χρόνοι δρομολογίων, κ.α.

Ένας περιορισμός της μεθοδολογίας της προσαρμογής του UI με μηχανική μάθηση είναι η εξάρτηση από ιστορικά δεδομένα αλληλεπίδρασης, τα οποία μπορεί να μην αντικατοπτρίζουν πάντα με ακρίβεια τις τρέχουσες προτιμήσεις ή τα μοτίβα ταξιδιού του χρήστη. Αλλαγές στη ρουτίνα ή απρόβλεπτες περιστάσεις μπορεί να οδηγήσουν σε ανακρίβειες στις προβλέψεις. Η ενσωμάτωση δεδομένων σε πραγματικό χρόνο και μηχανισμών ανάδρασης των χρηστών θα μπορούσε να βοηθήσει στην αντιμετώπιση αυτού του περιορισμού, επιτρέποντας στο σύστημα να προσαρμόζεται και να βελτιώνει τις προβλέψεις του με την πάροδο του χρόνου.

## 6.2 Μελλοντικές επεκτάσεις

Η εφαρμογή έχει ανάγκη από αρκετές επεκτάσεις σε διαφορετικά επίπεδα, όπως την προσαρμοστικότητα της εφαρμογής, τα τμήματα που αφορούν την διεπαφή χρήστη και τις δυνατότητες της σχετικά με την καλύτερη δυνατή απόδοση που μπορεί να έχει.

Όσον αφορά την προσαρμοστικότητα της εφαρμογής, θα μπορούσε να βελτιωθεί η μηχανή κανόνων προσθέτοντας πιο εξελιγμένους κανόνες και συνθήκες με βάση τη συμπεριφορά, τις προτιμήσεις και το περιβάλλον των χρηστών. Ακόμη, εξετάζεται το ενδεχόμενο να ενσωματωθούν τεχνικές μηχανικής εκμάθησης για να προσαρμόζεται δυναμικά η διεπαφή χρήστη με βάση τις αλληλεπιδράσεις και τα μοτίβα που δημιουργούνται με τον χρόνο από τους χρήστες. Επιπλέον, μια μελλοντική επέκταση θα ήταν ο εμπλουτισμός των επιλογών του χρήστη για να εξατομικευτεί και να προσαρμοστεί η εμπειρία διεπαφής χρήστη. Μια επιπρόσθετη εξέλιξη θα ήταν, να δοθεί η δυνατότητα στους χρήστες να επιλέγουν διαφορετικά θέματα, διατάξεις, μεγέθη γραμματοσειρών και συνδυασμούς χρωμάτων σύμφωνα με τις προτιμήσεις τους. Μια άλλη, θα ήταν να παρέχεται η επιλογή για την ενεργοποίηση/απενεργοποίηση συγκεκριμένων στοιχείων ή λειτουργιών διεπαφής χρήστη με βάση τις ατομικές ανάγκες όπως, τα φίλτρα αναζήτησης, κ.α.

Σχετικά με την προσαρμογή του UI μέσω μηχανικής μάθησης, οι μελλοντικές βελτιώσεις θα μπορούσαν να περιλαμβάνουν τη βελτίωση της ακρίβειας πρόβλεψης με την ενσωμάτωση περισσότερων χαρακτηριστικών με βάση τα συμφραζόμενα, όπως καιρικές συνθήκες, κυκλοφοριακή συμφόρηση ή γεγονότα που συμβαίνουν στη γύρω περιοχή. Αυτό θα μπορούσε να παρέχει μια πιο ολοκληρωμένη κατανόηση των μοτίβων μετακινήσεων του χρήστη και να επιτρέψει πιο ακριβείς προβλέψεις. Επιπλέον, οι δυνατότητες εξατομίκευσης, όπως η δυνατότητα στους χρήστες να παρέχουν ρητή ανατροφοδότηση σχετικά με τις προβλεπόμενες γραμμές λεωφορείων ή η προσαρμογή των προτιμήσεών τους, θα μπορούσαν να βελτιώσουν περαιτέρω την ακρίβεια και τη συνάφεια των προβλέψεων. Αυτή η προσέγγιση με γνώμονα τον χρήστη θα δώσει τη δυνατότητα στα άτομα να συμμετέχουν ενεργά στη βελτίωση του μοντέλου πρόβλεψης ώστε να ανταποκρίνεται καλύτερα στις συγκεκριμένες ανάγκες τους.

Σχετικά με την διεπαφή χρήστη και τις λειτουργίες της εφαρμογής, βασικός στόχος είναι η συνεχής βελτίωση του UI και συνεπώς και της εμπειρίας του χρήστη, δηλαδή, η βελτίωση της διάταξης των στοιχείων και πληροφοριών της εφαρμογής, η χρήση έξυπνων εικονιδίων και γραφικών κ.α. Συγκεκριμένα για την βελτίωση του UI, προτείνεται, στην αρχική σελίδα της εφαρμογής, να επανασχεδιαστεί για να παραλείψει την εμφάνιση των επιλογών για τον χρήστη και να επιλέξει

αυτόματα στο μενού τις γραμμές των λεωφορείων. Στην ιδανική περίπτωση, κατά την εκκίνηση της εφαρμογής, θα εμφανιζόταν μια λίστα με τις επιθυμητές ώρες άφιξης. Προς το παρόν, οι χρήστες πρέπει να περάσουν από δύο σελίδες και ένα αναδυόμενο παράθυρο, έπειτα να κάνουν κύλιση σε μια λίστα και να κάνουν δύο επιλογής πατήματα. Με την αλλαγή της διεπαφής χρήστη, θα επιτευχθεί η μείωση του χρόνου της ροής και συνεπώς η ελαχιστοποίηση της προσπάθειας πλοήγησης και επιλογής από τον χρήστη.

Επιπλέον, συγκεκριμένα για την εξέλιξη της λειτουργικότητας της εφαρμογής θα μπορούσαν να ενσωματωθούν πρόσθετοι τρόποι μεταφοράς, όπως υπηρεσίες κοινής χρήσης ποδηλάτων, πλατφόρμες κοινής χρήσης ταξί ή ακόμα και εναλλακτικές λειτουργίες όπως ηλεκτρικά σκούτερ. Η παροχή μιας ολοκληρωμένης προβολής πολλαπλών επιλογών μεταφοράς στην εφαρμογή θα μπορούσε να δώσει στους χρήστες τη δυνατότητα να προγραμματίσουν τις μετακινήσεις τους πιο αποτελεσματικά.

Στη μελλοντική ανάπτυξη με χρήση το Chaquory για προσαρμογή διεπαφής χρήστη με βάση τη μηχανική μάθηση, υπάρχουν αρκετοί δρόμοι προς εξερεύνηση προκειμένου να βελτιωθεί η απόδοση της εφαρμογής και να μειωθούν οι απαιτήσεις πόρων σε τηλέφωνα Android, ελαχιστοποιώντας ενδεχομένως την ανάγκη για ενσωμάτωση Chaquory. Θα μπορούσαν να αξιοποιηθούν οι εκτεταμένες βιβλιοθήκες και τα πλαίσια, όπως το TensorFlow Lite ή το PyTorch Mobile, που είναι διαθέσιμα στο οικοσύστημα Android για την προσαρμογή διεπαφής χρήστη χωρίς εξωτερικές εξαρτήσεις Python. Αυτή η προσέγγιση εξαλείφει την ανάγκη της Python, μειώνοντας την καθυστέρηση και την κατανάλωση πόρων.

Αναφορικά με την απόδοση και τους ταχύτερους χρόνους φόρτωσης των δεδομένων από API, μία μελλοντική επέκταση της εφαρμογής θα ήταν η χρήση πιο σύγχρονων βιβλιοθηκών δικτύωσης όπως η OkHttp<sup>3</sup> και η Retrofit<sup>4</sup> (Square, Retrofit: A type-safe HTTP client for Android and Java, 2023). Η βιβλιοθήκη OkHttp είναι ένας ισχυρός HTTP client για Android που προσφέρει ένα πιο προηγμένο και ευέλικτο σύνολο λειτουργιών σε σύγκριση με την βιβλιοθήκη Volley. Η OkHttp παρέχει επίσης ένα εύχρηστο API για την πραγματοποίηση ασύγχρονων κλήσεων δικτύου και το χειρισμό των απαντήσεων. Αναλυτικά, διαθέτει ενσωματωμένη υποστήριξη για λειτουργίες όπως η συγκέντρωση συνδέσεων, η προσωρινή αποθήκευση απόκρισης και το HTTP/2, τα οποία μπορούν να βελτιώσουν σημαντικά την απόδοση των αιτημάτων API. Αυτό είναι ιδιαίτερα ωφέλιμο για τις κλήσεις API που επιστρέφουν την ίδια απόκριση συχνά ή και για στατικά δεδομένα που δεν αλλάζουν συχνά. Με τη χρήση της προσωρινής

---

<sup>3</sup> <https://square.github.io/okhttp/>

<sup>4</sup> <https://square.github.io/retrofit/>

αποθήκευσης, η OkHttp μειώνει την ανάγκη για αιτήματα δικτύου, με αποτέλεσμα βελτιωμένη απόδοση και μειωμένο φόρτο διακομιστή. Μια ακόμη σημαντική και ευρέως χρησιμοποιούμενη βιβλιοθήκη είναι η Retrofit, η οποία απλοποιεί τη διαδικασία υποβολής αιτημάτων API επιτρέποντάς τον ορισμό μιας διεπαφής με σχολιασμένες μεθόδους που αντιπροσωπεύουν τα API end points. Αξίζει να σημειωθεί, ότι η Retrofit παρέχει υποστήριξη σειριοποίησης και αποσειροποίησης για JSON και XML, απλοποιώντας τον χειρισμό των αποκρίσεων API, με αποτέλεσμα, να βελτιώσει την αναγνωσιμότητα και τη συντήρηση του κώδικα. Συνοψίζοντας, οι παραπάνω τεχνολογίες είναι ικανές να οδηγήσουν στην ταχύτερη ανάκτηση των δεδομένων και στο μειωμένο φόρτο διακομιστή βελτιώνοντας έτσι την ανταπόκριση και την ομαλότητα της εφαρμογής.

## 7 Βιβλιογραφία

- Akiki, P., Bandara, A., & Yu, Y. (2015). Adaptive model-driven user interface development systems. doi:10.1145/2597999
- Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012, 09 01). Random Forests and Decision Trees. *International Journal of Computer Science Issues(IJCSI)*.
- Alvarez-Cortes, V., Zarate, V. H., Uresti, J. A., & Zayas, B. E. (2009, 12). Current Challenges and Applications for Adaptive User Interfaces. doi:10.5772/7745
- Arabghalizi, T., & Labrinidis, A. (2019). *How full will my next bus be? A Framework to Predict Bus Crowding Levels*. School of Computing and Information - Department of.
- Arnold, K., Gosling, J., & Holmes, D. (2005). *The Java programming language*. Addison Wesley Professional.
- Ballantyne, M. a.-G. (2018). Study of Accessibility Guidelines of Mobile Applications. Στο *Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia* (σσ. 305-315). ACM.
- Benyon, D., & Murray, D. (1993). Applying user modeling to human-computer interaction design. σσ. 199-225. doi:https://doi.org/10.1007/BF00849555
- Bielza, C., Li, G., & Larrañaga, P. (2011). Multi-dimensional classification with Bayesian networks. *International Journal of Approximate Reasoning*, σσ. 705-727.
- Bouzit, S., Calvary, G., Coutaz, J., Chene, D., Petit, E., & Vanderdonckt, J. (2017, Μάιος 11). The PDA-LPA Design Space for User Interface. *Eleventh IEEE International Conference on Research*.
- Citymapper. (2023). *71 European cities in 31 countries*. Ανάκτηση από <https://citymapper.com/news/2382/71-european-cities-in-31-countries>
- Consortium, W. W. (2022). *Mobile Accessibility at W3C*. Ανάκτηση από <https://www.w3.org/WAI/standards-guidelines/mobile/>
- Consortium, W. W. (2022). *Web Content Accessibility Guidelines*. Ανάκτηση από <https://www.w3.org/WAI/standards-guidelines/wcag/>
- Department, S. S. (2022, Φεβρουάριος). *Number of smartphone subscriptions worldwide from 2016 to 2021, with forecasts from 2022 to 2027*. Ανάκτηση από Statista: <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>
- Díaz-Bossini, J., Moreno, L., & Martínez, P. (2014). Towards Mobile Accessibility for Older People: A User Centered Evaluation. *International Conference on Universal Access in Human-Computer Interaction*, 58-68.
- Ehlert, P. (2003, Φεβρουάριος). Intelligent User Interfaces: Introduction and Survey.
- González, A. (2023). *OpenDyslexic: A typeface for Dyslexia*. Ανάκτηση από <https://opendyslexic.org/>
- Google. (2021). *Material Design*. Ανάκτηση από <https://m3.material.io/>.
- Google. (2022). *Android Studio IDE*. Ανάκτηση από <https://developer.android.com/studio>
- Google. (2022). *Volley*. Ανάκτηση από <https://github.com/google/volley>

- Google. (2023). Google-Accessibility. <https://developer.android.com/guide/topics/ui/accessibility>.
- Guo, G., Wang, H., Bell, D., & Bi, Y. (2004, 08 08). KNN Model-Based Approach in Classification.
- Hefley, W. E., & Murray, D. (1993). *Intelligent User Interfaces*. New York: Association for Computing Machinery. doi:<https://doi.org/10.1145/169891.169892>
- Inc., M. (2021, Φεβρουάριος). *Moovit now shows how crowded your bus is in real-time, and if it's wheelchair accessible*. Ανάκτηση από <https://moovit.com/press-releases/realtime-crowding-info-wheelchair-accessibility-feature/>
- Inc., M. (2023). *About Moovit*. Ανάκτηση από <https://moovit.com/about-us/>
- Jordan, M. I., & Mitchell, T. M. (2015). *Machine learning: Trends, perspectives, and prospects*. American Association for the Advancement of Science. doi:10.1126/science.aaa8415
- Looije, R., Brake, G., & Neerincx, M. (2007). Usability engineering for mobile maps. 532-539.
- Ltd, C. (χ.χ.). *Chaquopy: Python SDK for Android*. Ανάκτηση από <https://chaquo.com/chaquopy/>
- MacKenzie, I. S. (2009). Fitts' Law as a Research and Design Tool in Human-Computer Interaction.
- Manaris, B. (1998). Natural Language Processing: A Human-Computer Interaction Perspective. *Advances in Computers*.
- Matthias Schneider-Hufschmidt, T. K. (1993). Adaptive user interfaces: principles and practice.
- Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006). Spam filtering with Naive Bayes—which Naive Bayes? *Third conference on email and anti-spam* .
- Miraz, M. H., Ali, M., & Excell, P. S. (2020). Adaptive user interfaces and universal usability through plasticity of user interface design.
- Oulasvirta, A., Kristensson, P. O., Bi, X., & Howes, A. (2018). *Computational Interaction*.
- Park, K., Goh, T., & So, H.-J. (2014). Toward accessible mobile application design: developing mobile application accessibility guidelines for people with visual impairment.
- Roser, M., Ritchie, H., & Mathieu, E. (2023). *Technological Change*. Ανάκτηση από <https://ourworldindata.org/technological-change>
- Russell, S., & Norvig, P. (2003). *Artificial Intelligence: A Modern Approach* (2nd ed.).
- Schlungbaum, E. (1997). *Individual user interfaces and model-based user interface software tools*. New York: Association for Computing Machinery. doi:<https://doi.org/10.1145/238218.238330>
- Shneiderman, B. (1997). Direct manipulation for comprehensible, predictable and controllable user interfaces. 33-39.
- Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. (2017). *Designing the User Interface*. Pearson.
- Siricharoen, W. (2018). Understanding Social Interaction with Human Computer Interaction (HCI) Adaptation. *EAI Endorsed Transactions on Context-aware Systems and Applications*. doi:10.4108/eai.13-7-2018.160762
- Sonntag, D. (2017). *Intelligent User Interfaces - A Tutorial*.

- Square. (2023). *OkHttp*. Ανάκτηση από <https://square.github.io/okhttp/>
- Square. (2023). *Retrofit: A type-safe HTTP client for Android and Java*. Ανάκτηση από <https://square.github.io/retrofit/>
- Srivastava, D., & Bhambhu, L. (2010, 02 01). Data classification using support vector machine. *Journal of Theoretical and Applied Information Technology*.
- Stephanidis, C., & Savidis, A. (2001). Universal Access in the Information Society: Methods, Tools, and Interaction Technologies. σσ. 40-55.
- Swanepoel, D. W., Sousa, K. C., Smits, C., & Moore, D. R. (2019). Mobile applications to detect hearing impairment: opportunities and. *Bull World Health Organ*, 717-718.
- Theres, V. S., Christina, S., Eiband, M., & Buschek, D. (2020). What is "intelligent" in intelligent user interfaces?: a meta-analysis of 25 years of IUI. doi:10.1145/3377325.3377500
- Toulis, P., Airoidi, E., & Rennie, J. (2014). Statistical analysis of stochastic gradient methods for generalized linear models. *Proceedings of the 31st International Conference on Machine Learning*, σσ. 667-675.
- Yan, S., & Ramachandran, P. G. (2019). The Current Status of Accessibility in Mobile Apps.
- Zheng, M., Cheng, S., & Xu, Q. (2016). Context-based mobile user interface.
- ΟΑΣΑ. (2016). Ανάκτηση από Συγκοινωνίες Αθηνών: <https://www.oasa.gr/telematics/>
- ΟΑΣΑ. (2016).
- Παπαιωάννου, Γ. (2016). OASA Telematics API's documentation.



## ΠΑΡΑΡΤΗΜΑ – ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ

### Ερωτηματολόγιο αξιολόγησης εφαρμογής μέσω μεταφοράς Αθήνας

Το παρόν ερωτηματολόγιο αποτελεί μέρος της διπλωματικής εργασίας του ΠΑ.Δ.Α. Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών. Στοχεύει στην αξιολόγηση του σχεδιασμού και ανάπτυξης της εφαρμογής Android σχετικά με τα λεωφορεία και τρόλεϊ του ΟΑΣΑ.

Εκτιμώμενος χρόνος ερωτηματολογίου είναι περίπου 10 λεπτά.

[Συνδεθείτε στο Google](#), για να αποθηκεύσετε την πρόοδό σου. [Μάθετε περισσότερα](#)

\* Υποδεικνύει απαιτούμενη ερώτηση

Επιλέξτε το φύλο σας

- Άνδρας
- Γυναίκα
- Άλλο: \_\_\_\_\_

Ποια είναι η ηλικία σας; \*

- < 18
- 18 - 25
- 25 - 30
- 31 - 45
- 46 - 60
- 60 +

Τι επαγγέλεστε;

- Μαθητής
- Φοιτητής
- Εργαζόμενος
- Άνεργος
- Συνταξιούχος

Πώς θα αξιολογούσατε την τεχνολογική σας εμπειρία;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πολύ καλή

Πόσο συχνά χρησιμοποιείτε την τεχνολογία στο επάγγελμά σας;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πόσο εξοικειωμένοι είστε με το θέμα της εφαρμογής;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πόσο εύκολα μπορούσατε να πλοηγηθείτε στην εφαρμογή και να κατανοήσετε τα διάφορα χαρακτηριστικά της;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Βρήκατε τη διεπαφή χρήστη της εφαρμογής προσαρμοστική και ανταποκρίνεται στις ανάγκες σας;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Καταφέρατε να προσαρμόσετε τη διεπαφή ή τις ρυθμίσεις της εφαρμογής ώστε να ταιριάζουν καλύτερα στις προτιμήσεις σας; Εάν όχι, ποιες επιπλέον επιλογές προσαρμογής θα θέλατε να δείτε;

- Ναι
- Άλλο: \_\_\_\_\_

Πόσο ικανοποιημένοι είστε με τη συνολική εμπειρία χρήστη της εφαρμογής για την κάλυψη των μεταφορικών σας αναγκών;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Προσέφερε η εφαρμογή πρόσθετες λειτουργίες ή υπηρεσίες που βελτίωσαν τη συνολική εμπειρία σας με τα μέσα μαζικής μεταφοράς;

- Ναι
- Όχι

Πόσο καλά ανταποκρίθηκε η εφαρμογή στους συγκεκριμένους στόχους και απαιτήσεις σας όσον αφορά την πρόσβαση σε πληροφορίες λεωφορείων και δημόσιων συγκοινωνιών;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Πόσο πιθανό είναι να προτείνετε την εφαρμογή σε άλλους που χρησιμοποιούν τα μέσα μαζικής μεταφοράς;

	1	2	3	4	5	
Καθόλου	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Πάρα πολύ

Έχετε προτάσεις για τη βελτίωση των λειτουργιών προσβασιμότητας της εφαρμογής ή της συνολικής εμπειρίας χρήστη;

Η απάντησή σας \_\_\_\_\_

## ΠΑΡΑΡΤΗΜΑ - ΚΩΔΙΚΑΣ

```
@SuppressWarnings("ClickableViewAccessibility")
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    // init user data class
    userData = new UserData(this);
    // get time when app is open
    startTime = System.currentTimeMillis();

    Log.d("stats", "Start time app: " + startTime);

    // get settings from sharedPrefs
    SharedPreferences settings = getSharedPreferences(SHARED_PREFS, 0);
    boolean hasEnteredSurvey = settings.getBoolean(HAS_ENTERED_SURVEY, false);
    dyslexicFontEnabled = settings.getBoolean(DYSLEXIC_FONT, false);
    isHighContrastTheme = settings.getBoolean(HIGHCONTRAST_THEME, false);

    // set theme
    setTheme(isHighContrastTheme ? R.style.Theme_OASA_App_Theme_HighContrast : R.style.Theme_OASA_App_Theme);
    setTheme(dyslexicFontEnabled ? R.style.Theme_OASA_App_Theme_Dyslexic_Font : R.style.Theme_OASA_App_Theme);

    setContentView(R.layout.activity_main);

    // get instance of ReflexAgent class
    reflexAgent = ReflexAgent.getInstance(getApplicationContext(), this);
}
```

### Κώδικας 1 MainActivity

```
// Αρχικοποίηση των haptic/sound feedback και μεγενθυτικού φακού
// give user feedback on touch button (vibrate)
vibrate = (Vibrator) getSystemService(Context.VIBRATOR_SERVICE);
// setup click sound for sound feedback
buttonClickPlayer = MediaPlayer.create(this, R.raw.click_sound);

// setup magnifier
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) {
    magnifier = new Magnifier.Builder(tableLayout).setSize(400, 400).setInitialZoom(2.0f).build();
    magnifier.show(tableLayout.getWidth() / 2.0f, tableLayout.getHeight() / 2.0f);
}

// Χρήση της δόνησης
if (vibrate != null) {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        vibrate.vibrate(VibrationEffect.createOneShot(200, VibrationEffect.DEFAULT_AMPLITUDE));
    } else {
        vibrate.vibrate(200);
    }
}

// Button pressed - ενεργοποίηση ήχου
if (buttonClickPlayer != null) {
    buttonClickPlayer.start();
}
```

### Κώδικας 2 Haptic και sound feedback

```

// Συνάρτηση για την ενεργοποίηση voice-to-text
// Create an intent that can start the Speech Recognizer activity
private void displaySpeechRecognizer() {
    Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
RecognizerIntent.LANGUAGE_MODEL_WEB_SEARCH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, language_flag ? "en_US" : "el");
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_PREFERENCE, "el_GR");
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, getString(R.string.voice_prompt)); // Set
the prompt message in the desired language
    // This starts the activity and populates the intent with the speech text.
    startActivityForResult(intent, SPEECH_REQUEST_CODE);
}

```

*Κώδικας 3 Ενεργοποίηση voice-to-text δραστηριότητας*

```

public class UsageTip {
    private String title;
    private String[] description;

    public UsageTip(String title, String[] description){
        this.title = title;
        this.description = description;
    }

    public String getTitle(){
        return title;
    }

    public int getDescSize() {
        return description.length;
    }

    public String getDescription(int index){
        if(index < description.length - 1){
            return description[index];
        }else {
            return "";
        }
    }
}

```

*Κώδικας 4 UsageTip κλάση*

```

@Override
public boolean onTouchEvent(MotionEvent event) {
    // get Coordinates of touch
    float x = event.getX();
    float y = event.getY();

    // get number of fingers on screen
    pointerCount = event.getPointerCount();
    long touch_delay = -1L;
    // get User presses
    if (event.getAction() == MotionEvent.ACTION_DOWN){
        pressTime = System.currentTimeMillis();
        if (lastEvent != -1L) touch_delay = System.currentTimeMillis() - lastEvent;
        lastEvent = System.currentTimeMillis();
        if (releaseTime != -1L) duration = pressTime - releaseTime;

        float raw_x = event.getRawX();
        float raw_y = event.getRawY();

        int[] location = new int[2];
        tableLayout.getLocationOnScreen(location);
        int viewX = location[0];
        int viewY = location[1];

        int viewWidth = tableLayout.getWidth();
        int viewHeight = tableLayout.getHeight();
        int viewRight = viewX + viewWidth;
        int viewBottom = viewY + viewHeight;
        int tolerance = 100; // pixels of tolerance

        // check if raw x, y are within the tolerance from the view's bounds
        if (raw_x > viewX - tolerance && raw_x < viewRight + tolerance && raw_y > viewY -
tolerance && raw_y < viewBottom + tolerance) {
            near_presses++; // user pressed near the main table of buttons
        }
    } else if (event.getAction() == MotionEvent.ACTION_UP) {

        releaseTime = System.currentTimeMillis();
        duration = releaseTime - pressTime;
        missed_clicks++;
        if(userScrolling){
            timesTriesToMove++;
            userScrolling = false;
        }
    } else if (event.getAction() == MotionEvent.ACTION_MOVE){
        userScrolling = true;
    }
    int x_ = (int) event.getX();
    int y_ = (int) event.getY();

    int[] location = new int[2];
    bus_linesCardView.getLocationOnScreen(location);
    int buttonX = location[0];
    int buttonY = location[1];
    int buttonWidth = bus_linesCardView.getWidth();
    int buttonHeight = bus_linesCardView.getHeight();
}

```

Κώδικας 5 Συλλογή μετρικών input 1/2

```

if (x_ >= buttonX && x_ < buttonX + buttonWidth && y_ >= buttonY && y_ < buttonY + buttonHeight) {
    // Touch event occurred on the button
    // Button pressed
    if (buttonClickPlayer != null) {
        buttonClickPlayer.start();
    }
    Toast.makeText(this, "Pressed mapscardview", Toast.LENGTH_SHORT).show();
}

if (!isHighContrastTheme){
    isHighContrastTheme = reflexAgent.enableHighContrast;
    if(isHighContrastTheme){
        new MaterialAlertDialogBuilder(this)
            .setTitle(R.string.highcontrast_title)
            .setMessage(R.string.highcontrast_suggest)
            .setPositiveButton("OK", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    SharedPreferences sharedPreferences = getSharedPreferences(SHARED_PREFS, 0);
                    SharedPreferences.Editor editor = sharedPreferences.edit();
                    editor.putBoolean(HIGHCONTRAST_THEME, isHighContrastTheme);
                    editor.apply();
                    restartActivity();
                }
            }).setNegativeButton(getString(R.string.negative_choice), null)
            .show();
    }
}

long app_runtime = System.currentTimeMillis() - startTime;

Log.d("stats", "Duration pressed: " + pressTime);
Log.d("stats", "Button pressed: " + bus_line_clicks);
Log.d("stats", "Coordinates touch: x:" + x + ", y: " + y);
Log.d("stats", "Missed clicks " + missed_clicks);
Log.d("stats", "Click duration " + duration);
Log.d("stats", "Times tried to Move " + timesTriesToMove);
Log.d("stats", "User scrolling " + userScrolling);
Log.d("stats", "Touch pointers " + pointerCount);
Log.d("stats", "Near presses " + near_presses);
Log.d("stats", "App runtime " + app_runtime);
Log.d("stats", "Touch delay " + touch_delay);

userData.setPressCoords(new Pair<>(x, y));
userData.setMissedClicks(missed_clicks);
userData.setClickDuration(duration);
userData.setTimesTriesToMove(timesTriesToMove);
userData.setPointerCount(pointerCount);
userData.setNearPresses(near_presses);
userData.setRunTime(app_runtime);
scaleGestureDetector.onTouchEvent(event); // check zoom gesture
userData.LogAction(duration, x, y, pointerCount, new Date(System.currentTimeMillis()));

List<String> userdata = userData.getLogger();

// agent processing the state with user's data:
reflexAgent.processState(userData);

return true;
}

```

Κώδικας 6 Συλλογή μετρικών input 2/2



```

// Λειτουργία του voice-to-speech
@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == SPEECH_REQUEST_CODE && resultCode == RESULT_OK) {
        List<String> results = data.getStringArrayListExtra(
            RecognizerIntent.EXTRA_RESULTS);

        String spokenText = results.get(0);
        if (spokenText.equalsIgnoreCase(getString(R.string.open_bus_lines))) {

            OpenBusLineActivity();
        }
        else if (spokenText.equalsIgnoreCase(getString(R.string.open_bus_lines))) {
            OpenClosestStopsActivity();
        }

        . . .

        else if (spokenText.equalsIgnoreCase(getString(R.string.open_favourites)) ||
            spokenText.equalsIgnoreCase(getString(R.string.open_fav_lines))) {
            OpenFavouritesActivity(0);
        }
        else if (spokenText.equalsIgnoreCase(getString(R.string.open_favourites)) ||
            spokenText.equalsIgnoreCase(getString(R.string.open_fav_stops))) {
            OpenFavouritesActivity(1);
        }
        else if (spokenText.contains(getString(R.string.open_settings))) {
            OpenSettingsActivity();
        }
    }
}
}

```

Κώδικας 7 Λειτουργία voice-to-speech

```

public void addInteraction(String busSelected, String routeSelected,
    double latitude, double longitude, int dayOfWeek, long timeOfDay) {
    SQLiteDatabase db = getWritableDatabase();

    ContentValues values = new ContentValues();
    values.put(COLUMN_BUS_SELECTED, busSelected);
    values.put(COLUMN_ROUTE_SELECTED, routeSelected);
    values.put(COLUMN_LATITUDE, latitude);
    values.put(COLUMN_LONGITUDE, longitude);
    values.put(COLUMN_DAY_OF_WEEK, dayOfWeek);
    values.put(COLUMN_TIME_OF_DAY, timeOfDay);

    long newRowId = db.insert(TABLE_INTERACTIONS, null, values);
    if (newRowId == -1) {
        // Error occurred while inserting the record
        Log.e("Data Insertion", "Error occurred while inserting the record into the dataset
table");
    } else {
        // Record inserted successfully
        Log.d("Data Insertion", "Record inserted successfully. New row ID: " + newRowId);
    }
}
}

```

Κώδικας 8 Εισαγωγή στοιχείων στον πίνακα user\_interactions

```

public class GuideTipsFragment extends DialogFragment {
    private List<UsageTip> tips = new ArrayList<UsageTip>({});
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.tips_layout, container, false);
        View title = v.findViewById(R.id.tip_title);
        tips.add(new UsageTip("Πλοήγηση στο Κύριο Μενού", new String[]{"Κατά την εκκίνηση της εφαρμογής, θα δείτε το κύριο μενού με τέσσερα κουμπιά: Γραμμές λεωφορείων, Τοποθεσίες λεωφορείων, Πιο κοντινές στάσεις και Αγαπημένο λεωφορείο και στάσεις", "Χρησιμοποιήστε τα 4 κουμπιά πατώντας τα και αποκτήσετε πρόσβαση σε διάφορες λειτουργίες της εφαρμογής."}));
        tips.add(new UsageTip("Αναζήτηση γραμμές λεωφορείου", new String[]{"Πατήστε το κουμπί \"Γραμμές λεωφορείων\" για να δείτε όλες τις διαθέσιμες γραμμές λεωφορείων.", "Μπορείτε να αναζητήσετε μια συγκεκριμένη γραμμή λεωφορείου εισάγοντας το όνομα ή τον αριθμό της στη γραμμή αναζήτησης.", "Επιλέξτε μια γραμμή λεωφορείου για να δείτε λεπτομερείς πληροφορίες, συμπεριλαμβανομένων των στάσεων κατά μήκος της διαδρομής και των ωρών άφιξης για κάθε στάση."}));

        . . .

        tips.add(new UsageTip("Αναζήτηση γραμμές λεωφορείου", new String[]{"Πατήστε το κουμπί \"Γραμμές λεωφορείων\" για να δείτε όλες τις διαθέσιμες γραμμές λεωφορείων.", "Μπορείτε να αναζητήσετε μια συγκεκριμένη γραμμή λεωφορείου εισάγοντας το όνομα ή τον αριθμό της στη γραμμή αναζήτησης.", "Επιλέξτε μια γραμμή λεωφορείου για να δείτε λεπτομερείς πληροφορίες, συμπεριλαμβανομένων των στάσεων κατά μήκος της διαδρομής και των ωρών άφιξης για κάθε στάση."}));
        tips.add(new UsageTip("Ρύθμιση ειδοποιήσεων", new String[]{"Εάν θέλετε να σας υπενθυμίζεται η ώρα άφιξης ενός συγκεκριμένου λεωφορείου, κάντε κλικ στο εικονίδιο ειδοποίησης (αντιπροσωπεύεται από ένα κουδούνι ή ένα ξυπνητήρι) ενώ προβάλλετε τα στοιχεία του λεωφορείου.", "Μπορείτε να αναζητήσετε μια συγκεκριμένη γραμμή λεωφορείου εισάγοντας το όνομα ή τον αριθμό της στη γραμμή αναζήτησης.", "Επιλέξτε μια γραμμή λεωφορείου για να δείτε λεπτομερείς πληροφορίες, συμπεριλαμβανομένων των στάσεων κατά μήκος της διαδρομής και των ωρών άφιξης για κάθε στάση."}));

        UsageTip currentTip = getRandomTip();
        final int[] i = {0};
        ((TextView)title).setText(currentTip.getTitle());
        View desc = v.findViewById(R.id.tip_desc);
        ((TextView)desc).setText(currentTip.getDescription(i[0]));
        View next_btn = v.findViewById(R.id.next_tip);

        ((MaterialButton)next_btn).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(currentTip.getDescSize() - 1 < i[0]){
                    i[0]++;
                    View descr = v.findViewById(R.id.tip_desc);
                    ((TextView) descr).setText(currentTip.getDescription(i[0]));
                }
                else {
                    ((MaterialButton) next_btn).setText("Τελος");
                    dismiss();
                }
            }
        });
    }

    return v;
}
// Get a random tip from the array of tips
private UsageTip getRandomTip() {
    Random rand = new Random();
    int index = rand.nextInt(tips.size());
    return tips.get(index);
}
}

```

Κώδικας 9 Διάλογος με τα tips

```

// Singleton class
public class OASAService {
    private Context context;
    private static OASAService instance = null;
    //for Volley API
    public RequestQueue requestQueue;

    public OASAService(Context context) {
        this.context = context;
        requestQueue = getRequestQueue();
    }

    public static synchronized OASAService getInstance(Context context) {
        if (null == instance)
            instance = new OASAService(context);
        return instance;
    }

    public RequestQueue getRequestQueue() {
        if(requestQueue == null){
            requestQueue = Volley.newRequestQueue(context.getApplicationContext());
        }
        return requestQueue;
    }

    // this is so we don't need to pass context each time
    public static synchronized OASAService getInstance() {
        if (null == instance) {
            throw new IllegalStateException(OASAService.class.getSimpleName() +
                " is not initialized, call getInstance(...) first");
        }
        return instance;
    }

    public<T> void addToRequestQueue(Request<T> request){
        requestQueue.add(request);
    }

    public void cancelPendingRequests(Object tag) {
        if (requestQueue != null) {
            requestQueue.cancelAll(tag);
        }
    }
}

```

Κώδικας 10 OASAService κλάση

```

private static final String BASE_URL = "http://telematics.oasa.gr/";
private static final String API_URL = BASE_URL + "api/";
public void getLines() {
    String query = "?act=webGetLinesWithMLInfo";
    String req_url = API_URL + query;

    JsonRequest request = new JsonRequest(Request.Method.GET, req_url, null,
        new Response.Listener<JSONArray>() {
            @Override
            public void onResponse(JSONArray response) {
                try {
                    ArrayList<Line> linesList = new ArrayList<>();
                    // JSON array containing information from the API.
                    for (int i = 0; i < response.length(); i++) {
                        JSONObject temp_obj = response.getJSONObject(i);
                        String mlCode = temp_obj.getString("ml_code");
                        String sdcCode = temp_obj.getString("sdc_code");
                        String lineCode = temp_obj.getString("line_code");
                        String lineID = temp_obj.getString("line_id");
                        String lineDesc = temp_obj.getString("line_descr");
                        String lineDescEn = temp_obj.getString("line_descr_eng");
                        Line temp = new Line(mlCode, sdcCode, lineCode, lineID, lineDesc, lineDescEn);
                        linesList.add(temp);
                        db.insertLine(temp);
                    }
                    lines = new ArrayList<>(linesList);
                    updateList();
                    progressBar.setVisibility(View.GONE);
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        },
        new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                if (error instanceof TimeoutError || error instanceof NoConnectionError) {
                    Toast.makeText(getApplicationContext(),
                        getString(R.string.error_network_timeout),
                        Toast.LENGTH_LONG).show();
                } else if (error instanceof AuthFailureError) {
                    //TODO
                } else if (error instanceof ServerError) {
                    //TODO
                } else if (error instanceof NetworkError) {
                    //TODO
                } else if (error instanceof ParseError) {
                    //TODO
                }
            }
        }
    );
    request.setTag(this);
    OASAService.getInstance().addToRequestQueue(request);
}

```

*Κώδικας 11 Ανάκτηση στοιχείων γραμμών από API*

```

// εύρεση κίνησης ζουμ και χρήση reflexAgent
scaleGestureDetector = new ScaleGestureDetector(this, new
ScaleGestureDetector.SimpleOnScaleGestureListener() {
    @Override
    public boolean onScale(ScaleGestureDetector detector) {
        // Check if the scale gesture indicates zooming in
        if (detector.getScaleFactor() > 1.0f) {
            // Gather relevant data
            userData.setZoomGesture(true);
            // Call the ReflexAgent to enable the "enableZoom" rule
            reflexAgent.processState(userData);
            Log.d("stats", "ZoomGesture found");
        } else {
            Log.d("stats", "ZoomGesture not found ");
        }
        return true;
    }
});

```

Κώδικας 12 Ανίχνευσης ζούμ και κάλεσμα reflexAgent

```

// process the current state and take appropriate actions
public void processState(UserData userData) {
    for (String ruleCondition : RULE_CONDITIONS) {
        if (ruleCondition.equals("user_missed_clicks") && userData.MissedClicksRule()) {
            performActions(RULE_ACTION.get("user_missed_clicks"));
        } else if (ruleCondition.equals("user_attempts_zooming") && userData.ZoomInRule()) {
            performActions(RULE_ACTION.get("user_attempts_zooming"));
        } else if (ruleCondition.equals("user_near_presses") && userData.NearPressesRule()) {
            performActions(RULE_ACTION.get("user_near_presses"));
        } else if (ruleCondition.equals("user_is_idle") && userData.IdleUserRule()) {
            performActions(RULE_ACTION.get("user_is_idle"));
        } else if (ruleCondition.equals("user_input_delay") && userData.DelayBetweenTouchRule()) {
            performActions(RULE_ACTION.get("user_input_delay"));
        } else if (ruleCondition.equals("user_long_press") && userData.TouchDurationRule()) {
            performActions(RULE_ACTION.get("user_long_press"));
        } else if (ruleCondition.equals("user_attempts_scrolling") && userData.TriedScrollRule()) {
            performActions(RULE_ACTION.get("user_attempts_scrolling"));
        }
    }
}

```

Κώδικας 13 Επεξεργασία κατάστασης από δεδομένα χρήστη

```

// perform the actions based on the rule
private void performActions(List<String> actions) {
    for (String action : actions){
        switch (action) {
            case "adjustBrightness":
                setBrightness();
                break;
            case "enableZoom":
                enableZoom();
                break;
            case "giveUsageTips":
                giveUsageTips();
                break;
            case "enableSoundHapticFeedback":
                enableSoundHapticFeedback();
                break;
            case "applyHighContrastMode":
                applyHighContrastMode();
                break;
            case "applyDyslexicFont":
                applyDyslexicFont();
                break;
            case "enableVoiceInput":
                enableVoiceInput();
                break;
            default:
                System.out.println("Unknown action.");
        }
    }
}

```

*Κώδικας 14 Εφαρμογή ενεργειών*

```

private static final Map<String, List<String>> RULE_ACTION = new HashMap<String, List<String>>()
{
    put("user_not_found_information", new ArrayList<String>(){ add("giveUsageTips"); });
    put("user_missed_clicks", new ArrayList<String>(){ add("adjustBrightness");
add("enableZoom"); add("enableVoiceInput"); });
    put("user_attempts_zooming", new ArrayList<String>(){ add("enableZoom");
add("giveUsageTips"); });
    put("user_attempts_scrolling", new ArrayList<String>(){ add("enableSoundHapticFeedback");
add("giveUsageTips"); });
    put("user_near_presses", new ArrayList<String>(){ add("applyHighContrastMode");
add("adjustBrightness"); add("enableSoundHapticFeedback");});
    put("user_is_idle", new ArrayList<String>(){ add("giveUsageTips"); add("adjustBrightness");
add("enableVoiceInput"); });
    put("user_input_delay", new ArrayList<String>(){ add("adjustBrightness");
add("applyDyslexicFont");});
    put("user_long_press", new ArrayList<String>(){ add("enableSoundHapticFeedback");
add("adjustBrightness"); });
};

```

*Κώδικας 15 Χάρτης από ενέργειες ανά κανόνα*

```

// Check Permission for external storage
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
    if (checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE) !=
PackageManager.PERMISSION_GRANTED) {
        requestPermissions(new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE},
REQUEST_WRITE_EXTERNAL_STORAGE);
    }
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.R) {
        if(!Environment.isExternalStorageManager()) {
            Snackbar.make(findViewById(android.R.id.content), "Permission needed!",
Snackbar.LENGTH_INDEFINITE)
                .setAction("Settings", new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {

                        try {
                            Uri uri = Uri.parse("package:" + BuildConfig.APPLICATION_ID);
                            Intent intent = new
Intent(Settings.ACTION_MANAGE_APP_ALL_FILES_ACCESS_PERMISSION, uri);
                            startActivity(intent);
                        } catch (Exception ex){
                            Intent intent = new Intent();
                            intent.setAction(Settings.ACTION_MANAGE_ALL_FILES_ACCESS_PERMISSION);
                            startActivity(intent);
                        }
                    }
                })
                .show();
        }
    }
}
}
}
}

```

*Κώδικας 16 Έλεγχος άδειας για επεξεργασία εξωτερικού χώρου αποθήκευσης*

```

// get user location:
fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);
checkLocationSettings(); // check location permission

if (checkGoogleServices()) // check google services
    checkGPS(); // check users location
else {
    // show error message in snack bar
    Snackbar snackbar = Snackbar
        .make(coordinatorLayout,
com.google.android.gms.base.R.string.common_google_play_services_enable_text, Snackbar.LENGTH_LONG);
    snackbar.setBackgroundTint(getResources().getColor(R.color.md_theme_dark_errorContainer));
    snackbar.setActionTextColor(getResources().getColor(R.color.md_theme_dark_error));
    snackbar.setTextColor(getResources().getColor(R.color.md_theme_dark_error));
    snackbar.show();
}
}

```

*Κώδικας 17 Έλεγχος άδειας τοποθεσίας κινητού*

```

private void getLocation() {
    fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);
    if (ActivityCompat.checkSelfPermission(BusLineActivity.this, Manifest.permission.ACCESS_FINE_LOCATION)
    != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        return;
    }
    fusedLocationClient.getLastLocation()
        .addOnSuccessListener(this, new OnSuccessListener<Location>() {
            @Override
            public void onSuccess(Location location) {
                // Got last known location. In some rare situations this can be null.
                if (location != null) {
                    // Logic to handle location object
                    progressBar.setVisibility(View.VISIBLE);
                    LocalDate date = null;
                    if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.O) {
                        date = LocalDate.now();

                        DayOfWeek dow = date.getDayOfWeek();
                        int dayOfWeek = dow.getValue();
                        long timeOfDay = LocalTime.now().get(ChronoField.MILLI_OF_DAY);
                        latitude = location.getLatitude();
                        longitude = location.getLongitude();

                        String new_data = "";
                        if (latitude != 0 && longitude != 0){
                            new_data = String.join(";", String.valueOf(latitude),
String.valueOf(longitude),
                                String.valueOf(dayOfWeek), String.valueOf(timeOfDay));
                        }

                        Python py = Python.getInstance();
                        PyObject module = py.getModule("knn_predict");
                        if (!new_data.equals("")) {
                            PyObject obj = module.callAttr("main", new_data);
                            String result = obj.toString();
                            System.out.println(result);
                            if(!result.equals("Not enough data")){
                                Toast.makeText(getApplicationContext(), "Predicted " + result,
Toast.LENGTH_SHORT).show();

                                    addPredictedLine(result);
                                }else{
                                    Toast.makeText(getApplicationContext(), result, Toast.LENGTH_SHORT).show();
                                }
                            }
                        }
                    }
                });
            }
        });
}

```

*Κώδικας 18 Καταγραφή στοιχείων χρήσης και χρήση ML μοντέλου*

```

private void addPredictedLine(String prediction){
    filteredLines = new ArrayList<>();
    filteredLines.addAll(lines);

    for (Line line : lines)
        if(Objects.equals(line.getLine_id(), prediction))
            filteredLines.add(0, line);

    // Update the RecyclerView with filtered data
    linesAdapter.suggested(prediction);
    linesAdapter.filterList(filteredLines, false);
}

```

*Κώδικας 19 Εισαγωγής προτεινόμενης γραμμής στην αρχή της λίστας*



```

# Read data from csv file
data = pd.read_csv('./data/OASADatabase_db-user_interactions.csv', delimiter=';')

print(data.describe())

# get X, y from the data (features and label)
X = data[['latitude', 'longitude', 'day_of_week', 'time_of_day']]
y = data[['bus_line']]

# split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)

# Naive Bayes
nb = GaussianNB()
naive = model_evaluation(nb, "Naive Bayes")

# Stochastic Gradient Descent
sgd = SGDClassifier(max_iter=5000, random_state=0)
model_evaluation(sgd, "Stochastic Gradient Descent")

# K-Nearest Neighbors
knn_clf = KNeighborsClassifier(n_neighbors=3)
model_evaluation(knn_clf, "KNN")

# Support Vector Machine
svm = SVC(decision_function_shape="ovo")
model = model_evaluation(svm, "Support Vector Machine")

rf = RandomForestClassifier(n_estimators=1000, max_depth=10, random_state=0, warm_start=True)
model = model_evaluation(rf, "RandomForestClassifier")

# Create Decision Tree classifier object
clf = DecisionTreeClassifier(criterion="entropy", max_depth=3)
dt_model = model_evaluation(clf, "Decision Tree classifier")

```

*Κώδικας 20 Αξιολόγηση μοντέλων*

```

# Function to evaluate and print the model performance metrics
def model_evaluation(model_, title="Default"):
    trained_model = model_.fit(X_train, y_train)
    y_pred = trained_model.predict(X_test)

    # compute accuracy of the model
    print(model_.score(X_test, y_test))
    print(classification_report(y_test, y_pred))
    print('f1_score ', title, ': ', round(f1_score(y_test, y_pred, average='weighted'), 5), '\n')
    print('precision ', title, ': ', round(precision_score(y_test, y_pred, average='weighted'), 5), '\n')
    print('recall ', title, ': ', round(recall_score(y_test, y_pred, average='weighted'), 5), '\n')
    print('Accuracy ', title, ': ', round(accuracy_score(y_test, y_pred), 5), '\n')

    # Compute the confusion matrix
    cm = confusion_matrix(y_test, y_pred)

    # Plot the confusion matrix.
    sns.heatmap(cm, annot=True)
    plt.ylabel('Prediction', fontsize=13)
    plt.xlabel('Actual', fontsize=13)
    plt.title('Confusion Matrix', fontsize=17)
    plt.show()

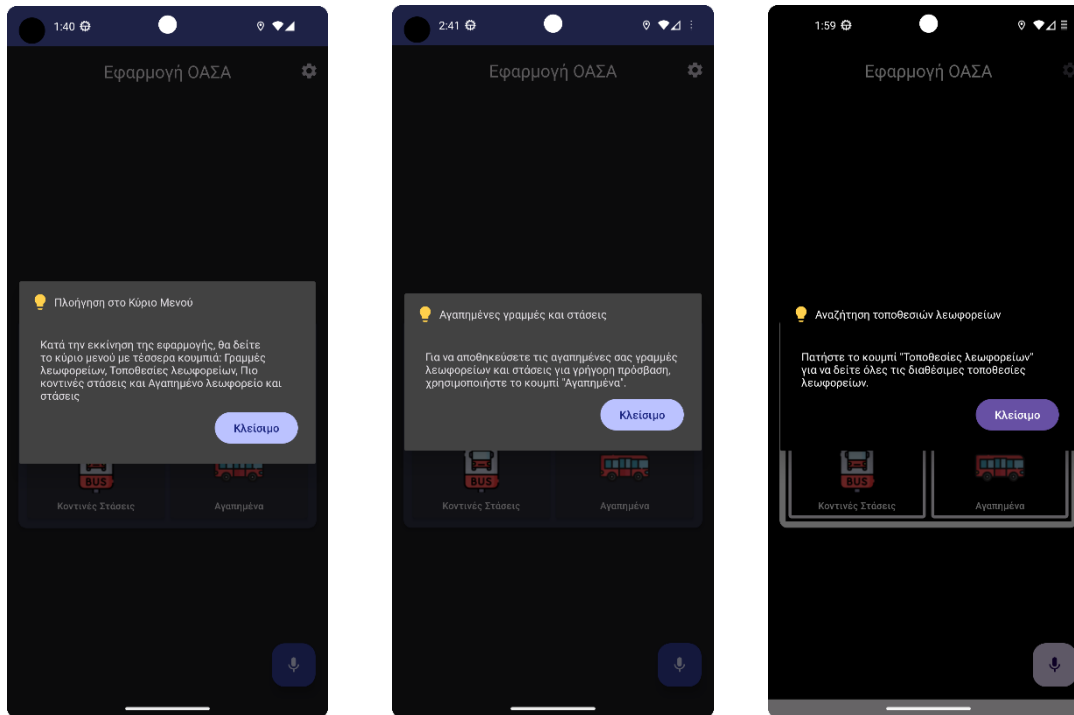
    return model_

```

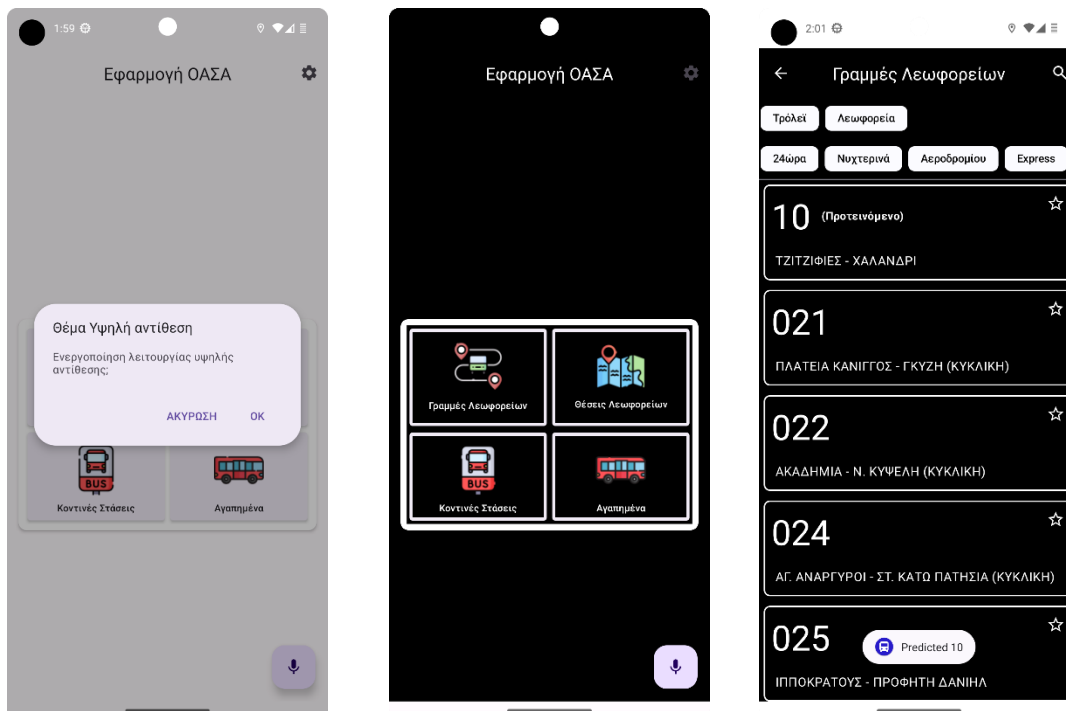
*Κώδικας 21 Συνάρτηση εκπαίδευσης και εκτίμησης μοντέλων*



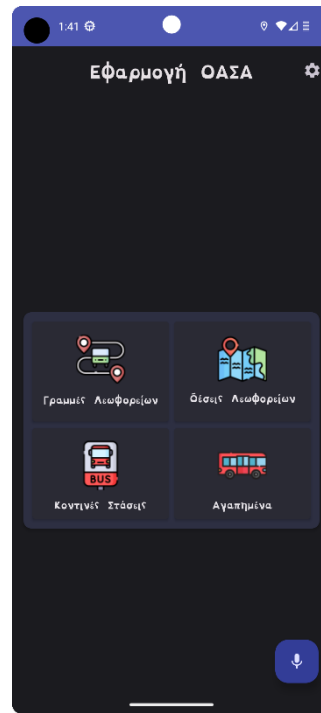
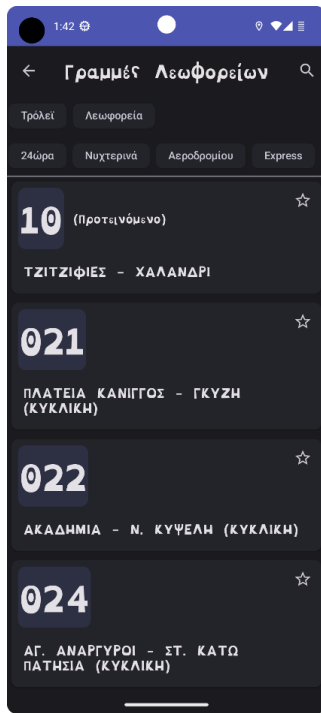
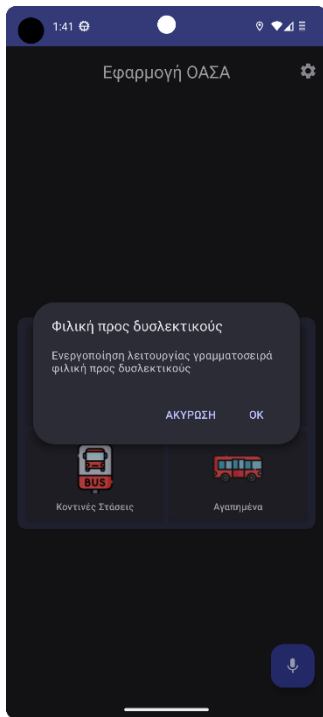
## ΠΑΡΑΡΤΗΜΑ – ΣΤΙΓΜΙΟΤΥΠΑ ΕΚΤΕΛΕΣΗΣ



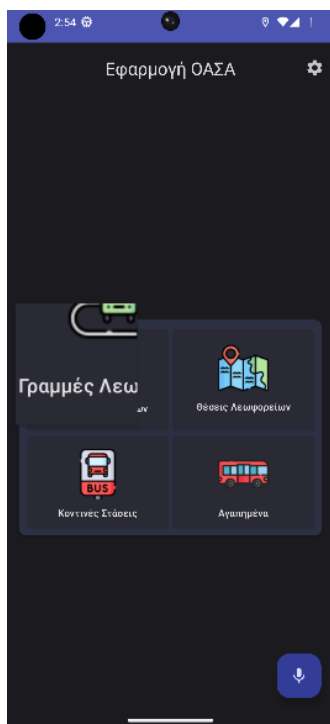
Στιγμιότυπο 1 Εμφάνιση οδηγιών χρήσης



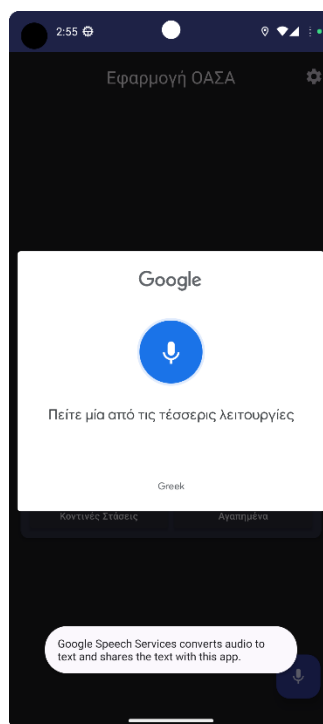
Στιγμιότυπο 2 Ενεργοποίηση θέματος υψηλής αντίθεσης



Στιγμιότυπο 3 Ενεργοποίηση γραμματοσειράς φιλικής προς δυσλεκτικούς



Στιγμιότυπο 5 Λειτουργία μεγθυντικού φακού



Στιγμιότυπο 4 Λειτουργία φωνητικών εντολών