



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής – Ανάπτυξη Λογισμικού και Τεχνικής
Νοημοσύνης»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Αναγνώριση Συναισθημάτων και Αντικειμένων Περιβάλλοντος σε βιντεοκλήση μέσω κινητού Face Recognition and Object Detection during videocall with mobile phone
Όνοματεπώνυμο Φοιτητή	Παναγιώτης Βαγγελάτος
Πατρώνυμο	Σπυρίδων
Αριθμός Μητρώου	ΜΠΣΠ19005
Επιβλέπων	Ευθύμιος Αλέπης, Αναπληρωτής Καθηγητής

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

Κωνσταντίνος Πατσάκης
Αναπληρωτής Καθηγητής

Ευθύμιος Αλέπης
Αναπληρωτής
Καθηγητής

Ευχαριστίες

Θα ήθελα να εκφράσω τις ευχαριστίες μου σε όλους όσους με βοήθησαν να ολοκληρώσω την παρούσα μεταπτυχιακή διατριβή. Ξεκινώντας θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή μου, κ. Αλέπη Ευθύμιο, για την πολύτιμη βοήθειά του και την υποστήριξή του.

Επίσης, θα ήθελα να εκφράσω τις ευχαριστίες μου στους γονείς μου και την γυναίκα μου για την αμέριστη συμπαράσταση, βοήθεια καθώς και για την κατανόηση και ανοχή τους σε όλη την διάρκεια των σπουδών μου.

Περίληψη

Στα πλαίσια της εργασίας αυτής, αναπτύχθηκε εφαρμογή για κινητές συσκευές Android όπου κατά την διάρκεια βιντεοκλήσης να ανιχνεύονται με την βοήθεια τεχνικής νοημοσύνης διάφορα χαρακτηριστικά προσώπου, όπως κατά πόσο ανοιχτά ή κλειστά είναι τα μάτια ή κατά πόσο χαμογελάει, καθώς και τα διάφορα αντικείμενα που υπάρχουν σε περιβάλλον.

Με βάση τις πληροφορίες αυτές, είναι σε θέση οι χρήστες της εφαρμογής αυτής να βλέπουν διάφορες αναφορές και επομένως να βγαίνουν διάφορα συμπεράσματα σχετικά όπως για το πότε είναι οι πιο βέλτιστες ώρες ώστε να γίνονται τα διάφορα meetings ή για το τι αντικείμενα προτιμούν να έχουν κοντά τους την ώρα των βιντεοκλήσεων.

Λέξεις Κλειδιά: Android, τεχνική νοημοσύνη, αναγνώριση χαρακτηριστικών προσώπου, αναγνώριση αντικειμένων περιβάλλοντος, Firebase Realtime Database, Firebase MLKit.

Abstract

In this thesis, we will describe the application that created for mobile phones with Android operation system. The application can recognize facial characteristics like if user has his/her eyes open or close or if the user smiles or not. In addition, the application can detect and the environmental objects. To accomplish this, the application uses the power of the artificial intelligence that provide the Google via the service of the Firebase MLKit of the Firebase Platform.

With the collected data, the user can display reports via the application and in addition for example a company find out when is the best part of the day to do the meetings or what objects has the users during the meetings.

Keywords: Android, artificial intelligence, face recognition, objects detection, Firebase Realtime Database, Firebase MLKit.

Περιεχόμενα

Ευχαριστίες	3
Περίληψη	4
Abstract	5
Περιεχόμενα	6
Εισαγωγή	7
Τεχνολογίες που χρησιμοποιήθηκαν	8
Android SDK	8
Android JetPack (AndroidX)	9
Firebase Authentication	10
Firebase Realtime Database	11
Firebase Machine Learning.....	12
Java Programming Language	13
Kotlin	14
Ανάλυση Android Εφαρμογής	15
Σύντομες πληροφορίες	15
Σύστημα πλοήγησης	16
Οθόνη: Splash	17
Οθόνη: Δημιουργία λογαριασμού	18
Οθόνη: Είσοδος.....	19
Οθόνη: Επαφές.....	20
Οθόνη: Προσθήκη / Επεξεργασία επαφών	22
Οθόνη: Καταγραφή συναισθημάτων και αντικειμένων	26
περιβάλλοντος	26
Οθόνη: Λίστα βιντεοκλήσεων (ανά επαφή).....	27
Οθόνη: Προβολή χαρακτηριστικών προσώπου και αντικειμένων περιβάλλοντος.....	28
Οθόνη: Αναφορές.....	32
Οθόνη: Βιντεοκλήσεις (Γενικές Αναφορές)	33
Οθόνη: Χαρακτηριστικά Προσώπου (Γενικές Αναφορές)	34
Οθόνη: Αντικείμενα ανά 3 έτη	35
Οθόνη: Ετικέτες αντικειμένων ανά 3 έτη.....	37
Συμπεράσματα	39
Βιβλιογραφία - Αναφορές.....	40
Πίνακας Εικόνων	42

Εισαγωγή

Σκοπός της μεταπτυχιακής διατριβής είναι η ανάπτυξη εφαρμογής για κινητές συσκευές Android, η οποία με την χρήση τεχνικής νοημοσύνης να ικανή να ανιχνεύει και να καταγράφει διάφορα χαρακτηριστικά προσώπου και αντικείμενα περιβάλλοντος κατά την διάρκεια βιντεοκλήσης. Με τα δεδομένα που θα έχουν συλλεχθεί, αν υποθέσουμε ότι θα τα χρησιμοποιεί κάποια εταιρία, θα είναι σε θέση να εξάγει αναφορές για το πότε είναι οι βέλτιστες ώρες να πραγματοποιούνται τα meetings ή τι αντικείμενα συνήθως υπάρχουν στον χώρο κατά την διάρκεια βιντεοκλήσεων.

Όπως είναι αναμενόμενο, για την ανάπτυξη μιας τέτοιου είδους εφαρμογής χρησιμοποιήθηκαν παραπάνω από μια τεχνολογίες. Συνοπτικά, εκτός από την χρήση του Android SDK, για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν οι τεχνολογίες Firebase MLKit για την ανάλυση εικόνων, την ανίχνευση των χαρακτηριστικών προσώπου και αντικειμένων περιβάλλοντος, η τεχνολογία Firebase Realtime Database για την αποθήκευση των παραπάνω δεδομένων, η τεχνολογία Firebase Authentication για την δημιουργία και είσοδο των χρηστών στην εφαρμογή καθώς και μερικές βιβλιοθήκες της Android JetPack σουίτας (βελτιωμένη έκδοση του Android SDK).

Τέλος, να αναφέρω ότι η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής είναι η Java. Επιλέχθηκε η Java αντί της Kotlin (επίσημη γλώσσα προγραμματισμού του Android) διότι είναι πιο «ώριμη» σε σχέση με την Kotlin άρα και πιο σταθερή καθώς και επειδή υπάρχει πλουσιότερη τεκμηρίωση (documentation) και παραδείγματα. Το περιβάλλον στο οποίο αναπτύχθηκε η εφαρμογή ήταν η τελευταία σταθερή (stable) έκδοση του Android Studio για λειτουργικό σύστημα Linux Debian 11 (x64) stable.

Τεχνολογίες που χρησιμοποιήθηκαν

Σε αυτή την ενότητα θα αναφερθούν όλες οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής και καθώς και θα αναλυθεί η κάθε μία τεχνολογία.

Android SDK

Το Android SDK είναι ένα σύνολο από διάφορα εργαλεία ανάπτυξης τα οποία είναι απαραίτητα για την ανάπτυξη της εφαρμογής όπως βιβλιοθήκες, εργαλεία εντοπισμού σφαλμάτων (debugger) και παραγωγής του τελικού εκτελέσιμου αρχείου που θα εκτελεστεί στην κινητή συσκευή. Εκτός από τα παραπάνω, το Android SDK περιέχει και έναν εξομοιωτή (emulator) ο οποίος βασίζεται στο QEMU, την τεκμηρίωση (documentation), τον πηγαίο κώδικα καθώς και μια πληθώρα από παραδείγματα.

Μέχρι προσφάτως το Android SDK υποστηρίζεται στα λειτουργικά συστήματα Linux, MAC OS X 10.5.8 και μεταγενέστερες εκδόσεις και Windows 7 και μεταγενέστερες εκδόσεις. Από το 2015 μέχρι και σήμερα, το επίσημο εργαλείο ανάπτυξης εφαρμογών Android είναι το Android Studio, ενώ μέχρι και το τέλος του 2014 το επίσημο IDE ήταν το Eclipse, χρησιμοποιώντας το Android Development Tool (ADT) plugin.

Σε αυτό το σημείο αξίζει να σημειωθεί ότι εκτός από το Android SDK, υπάρχει και το Android SDK Platform Tools και το οποίο δεν είναι ενσωματωμένο στο Android SDK και περιέχει ένα σύνολο από command-line εργαλεία όπως είναι το Android Debug Bridge (ADB) και το Fastboot το οποίο παρέχει εργαλεία για την τροποποίηση της Flash filesystem μέσω USB.

Τέλος, παρέχετε η δυνατότητα ανάπτυξης εφαρμογής Android με την χρήση της γλώσσας προγραμματισμού C++ χρησιμοποιώντας το Android NDK.

Android JetPack (AndroidX)

Το Android Jetpack (AndroidX) είναι μια σουίτα από βιβλιοθήκες και είναι μια βελτιωμένη έκδοση της Support Library του Android Platform και την οποία θα αντικαταστήσει μελλοντικά καθώς πλέον η Support Library δεν συντηρείται από την Google.

Αναπτύχθηκε από την Google υιοθετώντας τις καλύτερες πρακτικές (best practices) των αρχιτεκτονικών ανάπτυξης λογισμικού, μειώνοντας τον λεγόμενο boilerplate κώδικα καθώς και να προσφέρει υποστήριξη και στις παλιότερες εκδόσεις (backward compatibility). Η Android JetPack σουίτα έχει ξεχωριστό namespace (androidx.*) και η συντήρηση των εκάστοτε βιβλιοθηκών της γίνεται ανεξάρτητα. Όσο αφορά το versioning της κάθε βιβλιοθήκης, ακολουθείται το πρότυπο Semantic Version, ξεκινώντας από την έκδοση 1.0.0. Με αυτό τον τρόπο σε ένα έργο που χρησιμοποιούνται μερικές από τις βιβλιοθήκες του Android JetPack, σε περίπτωση που υπάρχει νέα έκδοση μιας βιβλιοθήκης δεν θα χρειαστεί να γίνει αναβάθμιση ολόκληρης της Android JetPack σουίτας αλλά μόνο της εκάστοτε βιβλιοθήκης.

Μερικές από τις πιο συχνά χρησιμοποιούμενες βιβλιοθήκες της Android JetPack σουίτας είναι:

- ❖ Η βιβλιοθήκη WorkManager. Η βιβλιοθήκη αυτή υλοποιεί τις απαραίτητες λειτουργίες για την εκτέλεση επαναλαμβανόμενων διεργασιών στο παρασκήνιο (background).
- ❖ Την βιβλιοθήκη Room, η οποία χρησιμοποιείται για την εύκολη διαχείριση SQLite βάσης δεδομένων καθώς και τον εύκολο τρόπο εκτέλεσης CRUD.
- ❖ Την βιβλιοθήκη Navigation, η οποία χρησιμοποιείται για την πλοήγηση του χρήστη μεταξύ των activities και fragments. Η βιβλιοθήκη έχει υλοποιήσει το δύσκολο κομμάτι για τους προγραμματιστές, που αφορά με σωστή μετάβαση από fragment σε fragment καθώς και την σωστή μετάβαση σε προηγούμενη οθόνη.
- ❖ Την βιβλιοθήκη CameraX, η οποία έχει επιλύσει τα προβλήματα συμβατότητας με παλιότερες εκδόσεις καθώς και έχει υλοποιημένα 3 βασικά User Cases που χρησιμοποιούνται σχεδόν από κάθε προγραμματιστή. Αυτά είναι η προβολή εικόνας, η ανάλυση εικόνας και η λήψη και η αποθήκευση εικόνας.
- ❖ Μια πληθώρα από βιβλιοθήκες που οδηγούν τον προγραμματιστή να κάνει χρήση της αρχιτεκτονικής MVVM (Model, View, ViewModel), την εκτέλεση εργασιών ασύγχρονα το background κάνοντας χρήση των βιβλιοθηκών LiveData (Java) ή Flow (Kotlin), την δυναμική ενημέρωση των δεδομένων ενός activity ή fragment μέσω του feature databinding κ.α.

Firestore Authentication

Δεδομένου ότι η εφαρμογή θα πρέπει να είναι σε θέση να αναγνωρίζει και να πιστοποιεί τους χρήστες, έπρεπε να αναπτυχθεί ένα σύστημα εγγραφής, εισόδου και αυθεντικοποίησης χρηστών. Για την επίλυση της ανάγκης αυτής επιλέχθηκε η Cloud υπηρεσία της Google, η Firebase Authentication.

Η υπηρεσία αυτή παρέχει ένα ολοκληρωμένο και εύχρηστο SDK καθώς και έτοιμες βιβλιοθήκες για την ανάπτυξη οθονών που έχουν υλοποιημένες τις λειτουργίες αυτές. Ένας ακόμα σημαντικός παράγοντας που επιλέχθηκε η υπηρεσία αυτή είναι ότι υποστηρίζει μια πληθώρα από τρόπους αυθεντικοποίησης, η οποίες είναι:

- ❖ Χρήση email και κωδικού πρόσβασης,
- ❖ Χρήση τηλεφώνου,
- ❖ Χρήση λογαριασμού Facebook, Google, Play Games, Game Center, Twitter, GitHub, Yahoo, Microsoft και Apple
- ❖ Είσοδος στο σύστημα ως ανώνυμος χρήστης

Μελλοντικά, αν η εφαρμογή Android υλοποιηθεί και σε άλλες πλατφόρμες, επειδή η Firebase Authentication είναι Cloud υπηρεσία, θα μπορεί ο εκάστοτε χρήστης να συνδέεται από όποια συσκευή επιθυμεί με τα ίδια στοιχεία.

Firestore Realtime Database

Η υπηρεσία Firestore Realtime Database της Google παρέχει την δυνατότητα της αποθήκευσης δεδομένων σε Cloud βάση δεδομένων σε πραγματικό χρόνο. Ο τύπος της βάσης δεδομένων στην υπηρεσία αυτής είναι NoSQL. Εκτός από την αποθήκευση των δεδομένων, δίνεται και η δυνατότητα του συγχρονισμού των δεδομένων σε πραγματικό χρόνο σε όλες τις συνδεδεμένες συσκευές, συμπεριλαμβανομένου και κινητών συσκευών, καθώς η Google έχει βελτιστοποιήσει την υπηρεσία αυτή για κινητές συσκευές.

Επιπρόσθετα, και ειδικά για τις κινητές συσκευές, παρέχετε η δυνατότητα και της offline αποθήκευσης δεδομένων. Για να επιτευχθεί αυτό, αν η κινητή συσκευή δεν έχει πρόσβαση στο διαδίκτυο τότε το SDK αποθηκεύει τα δεδομένα που θα σταλούν από την κινητή συσκευή στην cloud βάση δεδομένων στον σκληρό δίσκο της κινητής συσκευής και όταν υπάρξει σύνδεση στο διαδίκτυο, τότε τα στέλνει στην cloud βάση δεδομένων.

Ένα ακόμα από τα πλεονεκτήματα της Firestore Realtime Database είναι ότι τα δεδομένα είναι προσβάσιμα από κάθε συσκευή, είτε είναι κινητό, ταμπλέτα είτε ακόμα και από τον περιηγητή ιστού (browser) χωρίς να υπάρχει κάποιος ενδιάμεσος (application) server. Η ασφάλεια και ο έλεγχος εγκυρότητας των δεδομένων γίνεται μέσω κανόνων οι οποίοι εκτελούνται κάθε φορά που γίνεται αποθήκευση και ανάκτηση των δεδομένων.

Τέλος, η Google παρέχει το Blaze pricing plan το οποίο η χρέωση γίνεται δυναμικά ανάλογα με την χρήση των πόρων. Σε αυτό το πλάνο, παρέχεται η δυνατότητα της χρήσης πολλών στιγμιότυπων Firestore Realtime Databases στο ίδιο Firestore έργο. Με αυτόν τον τρόπο επιτυγχάνεται η εύκολη και γρήγορα επεκτασιμότητα (scalability) και επομένως επιτυγχάνεται και η αύξηση της απόδοσης (Performance) της εκάστοτε εφαρμογής.

Firestore Machine Learning

Η υπηρεσία Firestore Machine Learning παρέχει είτε την δυνατότητα χρήσης ήδη εκπαιδευμένων μοντέλων της Google, για συγκεκριμένες περιπτώσεις είτε την χρήση custom μοντέλων τα οποία τα ανεβάζουν οι χρήστες στην Google. Επίσης, με το Firestore Machine Learning SDK η εφαρμογή μπορεί να χρησιμοποιήσει είτε ένα μοντέλο στο cloud, δηλαδή όλη η διαδικασία εκτέλεσης για γίνει στο cloud της Google και να σταλούν τα δεδομένα πίσω στην εφαρμογή, είτε να «αποθηκευτεί» το μοντέλο στο κινητό και όλη η διαδικασία εκτέλεσης για την εύρεση αποτελεσμάτων να γίνει στην κινητή συσκευή. Και στις δυο περιπτώσεις υπάρχουν πλεονεκτήματα και μειονεκτήματα.

Για παράδειγμα, όταν επιλέγετε η εκτέλεση ανάλυσης των δεδομένων στο cloud της Google τα αποτελέσματα είναι πιο ακριβής καθώς δεν υπάρχει ο περιορισμός των πόρων (resources) που θα χρησιμοποιηθούν, όμως έχουμε κατανάλωση του network bandwidth καθώς πρέπει να σταλούν τα δεδομένα που θα αναλυθούν στο cloud και να ληφθούν τα αποτελέσματα από την εφαρμογή.

Πρόσφατα η Google δημιούργησε μια νέα βιβλιοθήκη, την ML Kit, η οποία παρέχει εκπαιδευμένα και έτοιμα προς χρήση μοντέλα τα οποία είναι πλήρως βελτιστοποιημένα για κινητές συσκευές και η ανάλυση των δεδομένων εκτελείται στην κινητή συσκευή σε πραγματικό χρόνο. Οι περιπτώσεις χρήστη που υποστηρίζει το ML Kit είναι:

- ❖ Ανάγνωση barcode
- ❖ Εντοπισμός προσώπου (όχι όμως αναγνώριση προσώπου)
- ❖ Εύρεση ετικετών
- ❖ Εντοπισμός και παρακολούθηση αντικειμένων
- ❖ Αναγνώριση κειμένου
- ❖ Αναγνώριση ψηφιακού μελανιού
- ❖ Εντοπισμός ανθρώπινης πόζας
- ❖ Διαχωρισμός ανθρώπου από το παρασκήνιο

και άλλα.

Java Programming Language

Η γλώσσα προγραμματισμού Java είναι μια υψηλού επιπέδου αμιγώς αντικειμενοστραφή γλώσσα προγραμματισμού που αναπτύχθηκε από τον James Gosling, Mike Sheridan and Patrick Naughton το Ιούνιο του 1991 και έχει παρόμοια σύνταξη με την C/C++. Για την εκτέλεση ενός προγράμματος που έχει αναπτυχθεί σε Java θα πρέπει να έχει εγκατασταθεί το Java Virtual Machine (JVM) το οποίο εκτελεί το bytecode που έχει παράγει ο compiler κατά την μεταγλώττιση του προγράμματος. Το γεγονός ότι το πρόγραμμα εκτελείτε στο JVM, έχει το πλεονέκτημα ότι το πρόγραμμα θα εκτελεστεί σε όλους τους υπολογιστές χωρίς να πρέπει να γίνει compile το πρόγραμμα (το λεγόμενο WORA, write once, run anywhere). Η Java αναπτύχθηκε με γνώμονα τις παρακάτω 5 αρχές:

- ❖ Να είναι απλή, αντικειμενοστραφής και γνώριμη
- ❖ Να είναι ασφαλής και να «αντέχει» στα σφάλματα
- ❖ Να μην έχει κάποια συγκεκριμένη αρχιτεκτονική και να είναι μεταφέρσιμη
- ❖ Να είναι αποδοτική
- ❖ Να είναι διερμηνευμένη, να υποστηρίζει νήματα (threads) και να είναι δυναμική

Πέρα από τα παραπάνω η Java έχει πάρα πολύ καλή και δομημένη τεκμηρίωση καθώς και πάρα πολλά παραδείγματα όπως και άπειρες 3rd Party βιβλιοθήκες και frameworks. Δεδομένου όλων των παραπάνω η Google είχε επιλέξει την Java για την ανάπτυξη εφαρμογών σε κινητές συσκευές.

Kotlin

Η γλώσσα προγραμματισμού Kotlin αναπτύχθηκε από την JetBrains ως έργο που ξεκίνησε τον Ιούλιο του 2011. Η πρώτη επίσημη έκδοση έγινε διαθέσιμη προς το κοινό στις 16 Φεβρουαρίου του 2016. Ο βασικός σκοπός ανάπτυξης της Kotlin ήταν να καλύψει ένα εύρος από χαρακτηριστικά που δεν ήταν ικανές να καλύψουν άλλες γλώσσες προγραμματισμού που ήθελε ο δημιουργός της, Dmitry Jemerov, τα οποία μεν κάλυπτε η γλώσσα προγραμματισμού Scala αλλά είχε αργό χρόνο μεταγλώττισης και την καθιστούσε μη αποδοτική.

Η Kotlin σχεδιάστηκε με γνώμονα να είναι πλήρως συμβατή με την Java καθώς τρέχει σε επίπεδο Java Virtual Machine (JVM) και το οποίο βασίζεται στην Java Class Library. Πλέον είναι η επίσημη γλώσσα προγραμματισμού για την ανάπτυξη Android εφαρμογών. Εκτός όμως την ανάπτυξη Android εφαρμογών, μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών σε επίπεδο JVM, iOS, macOS, watchOS, Windows, Linux, Javascript, WebAssembly και native κώδικα (μέσω του LLVM).

Ένα εξίσου σημαντικό χαρακτηριστικό της Kotlin είναι ότι μπορεί αναπτυχθεί εφαρμογή για κινητές συσκευές που να τρέχει και σε Android και σε iOS, όπως για παράδειγμα επιτυγχάνεται μέσω Xamarin της Microsoft, Flutter της Google κ.α, έχοντας την λογική της εφαρμογής, το λεγόμενο Business Logic, ως κοινή βάση και ότι αφορά τα UI components μεταγλωττίζονται ανάλογα με το λογισμικό (Android ή iOS).

Ανάλυση Android Εφαρμογής

Σε αυτό το κεφάλαιο θα αναλυθεί η Android εφαρμογή που αναπτύχθηκε στα πλαίσια της διπλωματικής εργασίας.

Σύντομες πληροφορίες

Ξεκινώντας, συνολικά δημιουργήθηκαν 1 activity και 18 fragments, ακολουθώντας την single-activity αρχιτεκτονική που αξιοποιείται πλήρως από την Navigation βιβλιοθήκη της Android JetPack σουίτας. Για την επίτευξη καθαρού και οργανωμένου κώδικα καθώς και της αύξησης της απόδοσης όσο αφορά την διαχείριση των UI components ενεργοποιήθηκε το χαρακτηριστικό του Android SDK, View Binding, το οποίο παράγει αυτόματα μια κλάση ανά fragment (και activity) που περιέχει όλα τα UI components ως ιδιότητες (αντικείμενα) και μέσω των κλάσεων αυτών παρέχεται η πρόσβαση σε αυτά, χρησιμοποιώντας τα UIDs που έχουν οριστεί στο αντίστοιχο XML layout αρχείο. Με αυτό τον τρόπο επιτυγχάνεται η αντικατάσταση της μεθόδου “findViewById()” κατά την οποία για να βρεθεί το UI Component θα πρέπει να «διαβαστεί» το «δέντρο ιεραρχίας» και σε περιπτώσεις όπου έχουμε περίπλοκο layout, δεν είναι αποδοτικό.

Για πολλά χρόνια υπήρχε το πρόβλημα κατά το οποίο τα δεδομένα των οθονών χανόντουσαν όταν η οθόνη περιστρέφονταν. Η λύση ήταν ο προγραμματιστής να αποθηκεύει τα δεδομένα που δεν πρέπει να χαθούν στο “Bundle” αντικείμενο και όταν το fragment ή το activity δημιουργούνταν ξανά, να φορτώνονται τα δεδομένα από αυτό. Σε όλη την διαδικασία αυτή υπήρχε ο περιορισμός ότι στο “Bundle” αντικείμενο μπορούσαν να αποθηκευτούν μικρός όγκος δεδομένων. Το πρόβλημα το έλυσε η Google μέσω των βιβλιοθηκών Lifecycle και ViewModel της Android JetPack σουίτας. Πιο συγκεκριμένα, ο προγραμματιστής αποθηκεύει τα δεδομένα στο ViewModel αντικείμενο το οποίο είναι lifecycle-aware, δηλαδή δεν επηρεάζεται από τον κύκλο ζωής του activity ή fragment και επομένως τα δεδομένα είναι κάθε στιγμή διαθέσιμα μέχρι να «καταστραφεί» το ViewModel αντικείμενο.

Άλλη μια πολύ χρήσιμη βιβλιοθήκη της Android JetPack σουίτας είναι η LiveData η οποία είναι σε θέση να ενημερώνει αυτόματα τις τιμές των UI components, είτε με τις τιμές που εισάγει ο χρήστης από την οθόνη, είτε με δεδομένα τα οποία προήλθαν από την βάση δεδομένων είτε οτιδήποτε άλλο (Two-Way data binding), χωρίς να πρέπει να επέμβει ο προγραμματιστής. Για να επιτευχθεί το παραπάνω αποτέλεσμα, η βιβλιοθήκη LiveData έχει υλοποιηθεί με βάση το Observable design pattern.

Σύστημα πλοήγησης

Για την πλοήγηση μεταξύ των fragments ανάλογα με τις επιλογές του χρήστη χρησιμοποιήθηκε η βιβλιοθήκη Navigation της Android JetPack σουίτας. Προτιμήθηκε η βιβλιοθήκη αυτή γιατί αφενός είναι επίσημη βιβλιοθήκη της Google καθώς και επειδή αναλαμβάνει η ίδια η βιβλιοθήκη το δύσκολο κομμάτι των fragment transactions με αποτέλεσμα η εφαρμογή να είναι πιο σταθερή. Επιπρόσθετα αναλαμβάνει την σωστή διαχείριση της λειτουργικότητας του back button.

Για να μπορέσει να γίνει χρήση του Navigation, στο layout (xml) του “βασικού” activity προσθέσαμε το component NavHost και στο xml αρχείο (nav_graph.xml) ορίσαμε ποιες οθόνες συνδέονται μεταξύ τους, το πως θα γίνεται η μετάβαση μεταξύ των fragments και τι τιμές θα μεταφέρονται από το ένα fragment στο άλλο αν υφίσταται ανάγκη.

Για την μεταφοράς δεδομένων μεταξύ των fragments χρησιμοποιήσαμε το SafeArgs το οποίο ανέπτυξε η Google για το Gradle. Το SafeArgs plugin έχει το πλεονέκτημα ότι είναι type safety και επομένως η εφαρμογή γίνεται ακόμα πιο σταθερή.

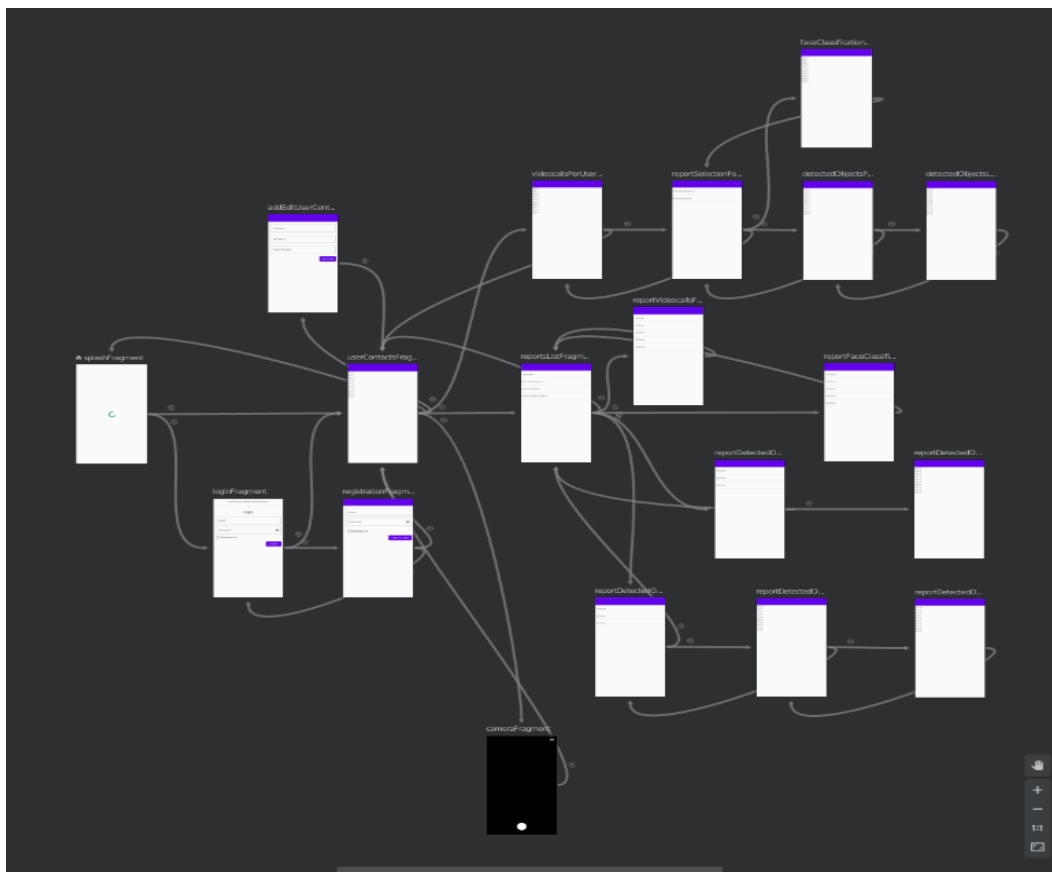


Figure 1: nav_graph

Οθόνη: Splash

Η οθόνη Splash δημιουργήθηκε με σκοπό όταν ο χρήστης εκτελεί την εφαρμογή, να βλέπει ένα μπάρα (κυκλική) που φορτώνει όσο εκτελούνται οι απαραίτητες διεργασίες στο παρασκήνιο (background) καθώς η εφαρμογή αναπτύχθηκε με γνώμονα την αντιμετώπιση του προβλήματος “Application Not Responding (ARN)”. Το πρόβλημα αυτό προκαλείτε όταν το UI Thread (ή Main Thread) δεν ανταποκρίνεται στα user inputs για μερικά δευτερόλεπτα (περίπου για 5 δευτερόλεπτα).

Οι διεργασίες που εκτελούνται στο παρασκήνιο αφορούν τον έλεγχο για το αν ο χρήστης έχει επιλέξει να «θυμάται» η εφαρμογή τα στοιχεία σύνδεσης ή όχι και αντίστοιχα να γίνεται αυτόματα η είσοδος στην εφαρμογή και η μεταφορά του χρήστη στην οθόνη «Επαφές».

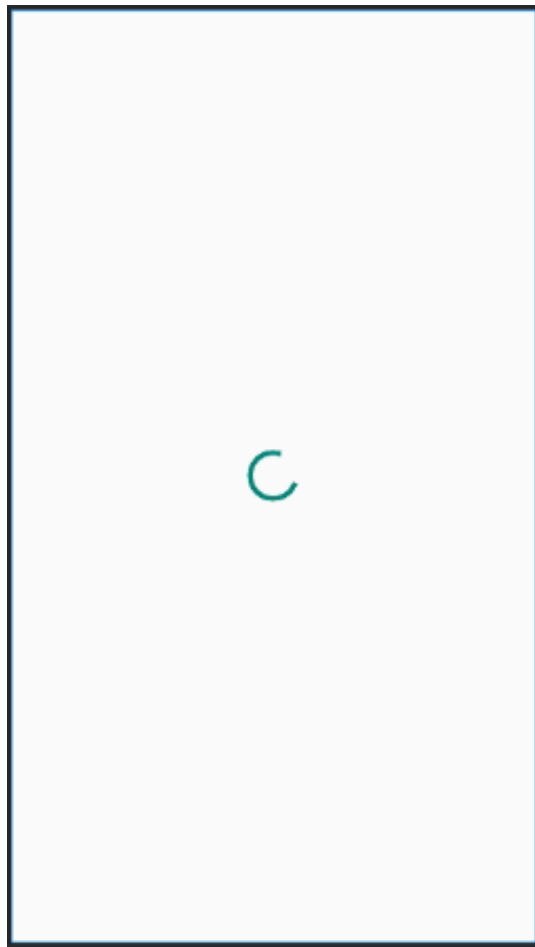


Figure 2: Οθόνη Splash

Οθόνη: Δημιουργία λογαριασμού

Στην οθόνη αυτήν παρέχεται στον χρήστη η δυνατότητα να δημιουργήσει έναν νέο λογαριασμό για να μπορέσει να συνδεθεί στο σύστημα. Η δημιουργία του λογαριασμού γίνεται με την χρήση Firebase Authentication βιβλιοθήκης, χρησιμοποιώντας για στοιχεία σύνδεσης ένα Email και έναν κωδικό πρόσβασης.

Αφού ολοκληρωθεί επιτυχώς η δημιουργία του λογαριασμού, τότε ο χρήστης μεταφέρεται αυτόματα στην οθόνη «Είσοδος». Σε περίπτωση που παρουσιαστεί οποιαδήποτε δυσλειτουργία, τότε εμφανίζεται στην οθόνη του χρήστη ένα παράθυρο με το μήνυμα σφάλματος.

Τέλος όπως φαίνονται και στις παρακάτω εικόνες υπάρχει ένα “Checkbox” με την περιγραφή “Remember me”. Όταν είναι επιλεγμένο τότε η εφαρμογή θα “θυμάται” το email και τον κωδικό πρόσβαση (αποθηκεύοντάς τα σε ένα secured shared preference) και ενημερώνεται το αντίστοιχο πεδίο στην τοπική βάση και την από την επόμενη φορά που ο χρήστης θα ανοίγει την εφαρμογή, τότε θα γίνεται αυτόματα είσοδος στην εφαρμογή χωρίς να χρειάζεται ο χρήστης κάθε φορά να εισάγει το email και τον κωδικό πρόσβασής του.

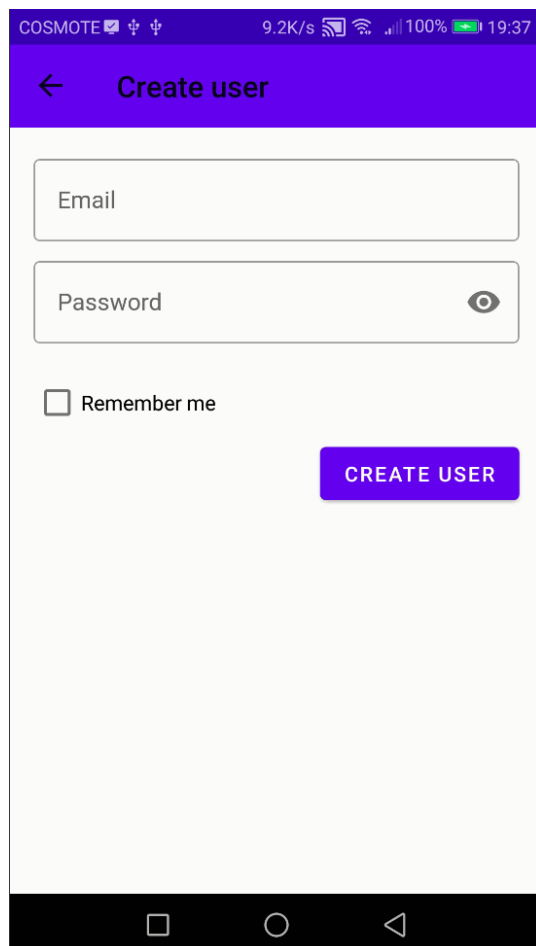


Figure 3: Οθόνη δημιουργίας λογαριασμού

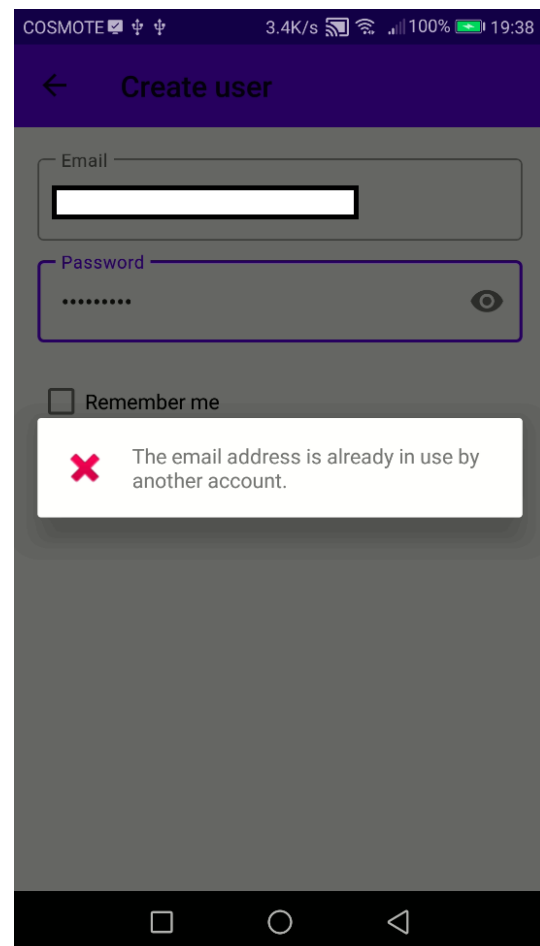


Figure 4: Οθόνη δημιουργίας λογαριασμού - Σφάλμα

Οθόνη: Είσοδος

Στην οθόνη αυτήν, παρέχεται η δυνατότητα στον χρήστη να συνδεθεί στο σύστημα (εφαρμογή) χρησιμοποιώντας τα στοιχεία σύνδεσης που συμπληρώθηκαν στην οθόνη της δημιουργίας λογαριασμού. Όταν πατηθεί το κουμπί «Είσοδος», τότε τα στοιχεία που συμπλήρωσε ο χρήστης στέλνονται στην υπηρεσία Firebase Authentication όπου και γίνεται η ταυτοποίησή τους. Αν τα στοιχεία είναι έγκυρα, τότε ο χρήστης μεταφέρετε στην οθόνη «Επαφές», ενώ στην περίπτωση που δεν είναι έγκυρα, τότε ενημερώνεται ο χρήστης για το σφάλμα που παρουσιάστηκε.

Ακόμα, παρέχεται η δυνατότητα στον χρήστη, αν επιθυμεί, να θυμάται η εφαρμογή τα στοιχεία σύνδεσης και να συνδέεται αυτόματα κάθε φορά ο χρήστης εκτελεί την εφαρμογή. Σε αυτή την περίπτωση, τα στοιχεία αποθηκεύονται κρυπτογραφημένα. Για την ασφαλή αποθήκευση των στοιχείων σύνδεσης, χρησιμοποιείται μια ειδική κλάση της Android JetPack σουίτας, η «EncryptedSharedPreferences», που κρυπτογραφεί τα δεδομένα κάνοντας χρήση του συμμετρικού αλγορίθμου κρυπτογράφησης AES-256 πριν την αποθήκευσή τους.

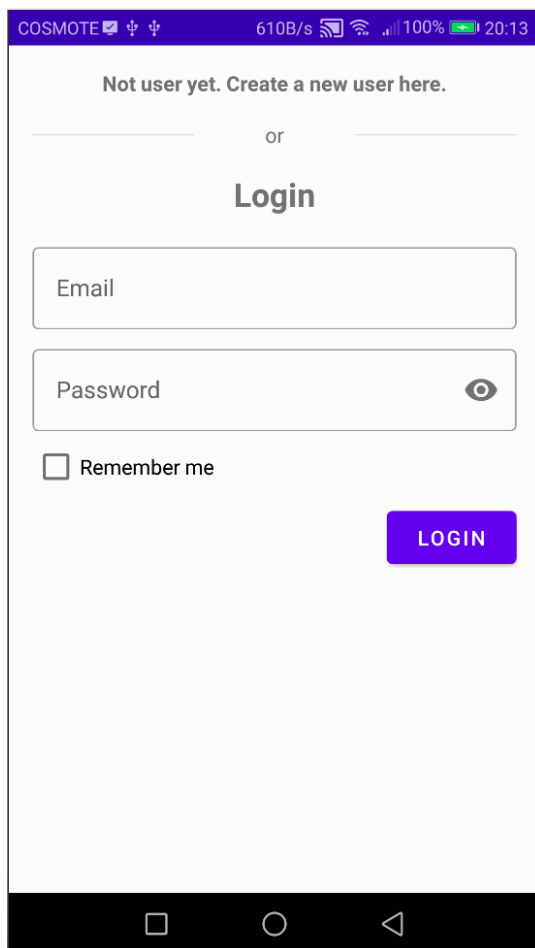


Figure 5: Οθόνη Είσοδος

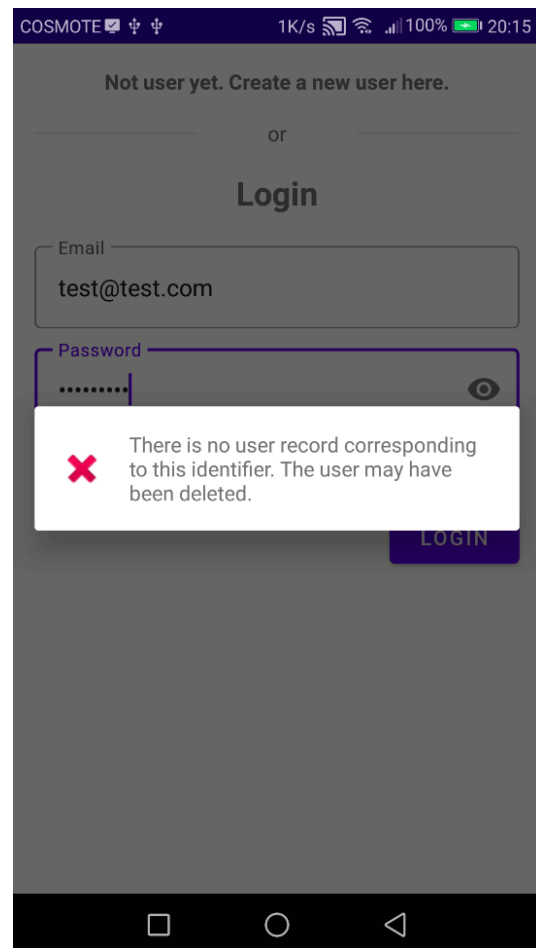


Figure 6: Οθόνη Είσοδος - Σφάλμα

Οθόνη: Επαφές

Η οθόνη «Επαφές» είναι η βασική οθόνη της εφαρμογής καθώς περιέχει τις επαφές του χρήστη που θα μπορεί να πραγματοποιηθεί βιντεοκλήση. Πέρα όμως από αυτήν την ενέργεια, παρέχονται και άλλες βασικές ενέργειες στον χρήστη όπως είναι η εισαγωγή, επεξεργασία και διαγραφή επαφών, η προβολή των στατιστικών για κάθε μια βιντεοκλήση που έχει πραγματοποιηθεί καθώς και η προβολή διαφόρων στατιστικών που αφορούν συνολικά όλες τις βιντεοκλήσεις από όλους του ενεργούς χρήστες της εφαρμογής.

Οι επαφές που εμφανίζονται στην οθόνη αυτή είναι αποθηκευμένες στην Firebase Realtime Database. Κάθε φορά που η οθόνη αυτή φορτώνεται τότε οι επαφές του συνδεδεμένου χρήστη φορτώνονται σε πραγματικό χρόνο από την cloud βάση δεδομένων και με αυτόν τον τρόπο εξασφαλίζεται ότι ο χρήστης θα έχει πάντα την πιο πρόσφατη λίστα επαφών.

Επίσης, στην οθόνη αυτή γίνεται χρήση της αρχιτεκτονικής MVVM (ModelView – ViewModel) ώστε αφενός ο κώδικας να είναι πιο σωστά δομημένος και αφετέρου να εξασφαλίζεται ότι τα δεδομένα (επαφές) που περιέχει η οθόνη, δεν θα χάνονται κατά την αλλαγή του lifecycle της οθόνης (πχ, κατά την περιστροφή της οθόνης). Για την προβολή των επαφών στην οθόνη χρησιμοποιήθηκε το component RecyclerView έναντι του ListView διότι ως πιο καινούργια τεχνολογία συνιστάτε από την Android Development Team. Μερικά από τα οφέλη του RecyclerView είναι ότι είναι πιο αποδοτικά όταν έχουμε μεγάλο όγκο δεδομένων να εμφανιστούν (λόγο του τρόπου που έχει υλοποιηθεί) καθώς και ότι το RecyclerView έχει βασιστεί το component lifecycle της Android JetPack σουίτας.

Τέλος, κάθε φορά που φορτώνεται η οθόνη αυτή, εκτός από τις επαφές, λαμβάνονται από την cloud βάση δεδομένων και όλα τα δεδομένα για όλες τις βιντεοκλήσεις για όλους τους ενεργούς χρήστες της εφαρμογής, που αφορούν:

- ❖ Βιντεοκλήσεις
- ❖ Χαρακτηριστικά προσώπου
- ❖ Εντοπισμένα αντικείμενα περιβάλλοντος
- ❖ Ετικέτες εντοπισμένων αντικειμένων περιβάλλοντος

και τα οποία αποθηκεύονται στην τοπική SQLite βάση δεδομένων της εκάστοτε συσκευής. Ο λόγος που επιλέχθηκε η συγκεκριμένη οθόνη για την λειτουργία αυτή είναι ότι πριν ο χρήστης προβεί στην εμφάνιση στατιστικών, είτε των δικών του είτε γενικών στατιστικών, θα πρέπει να έχει πάντα τα τελευταία δεδομένα που περιέχει η cloud βάση δεδομένων.

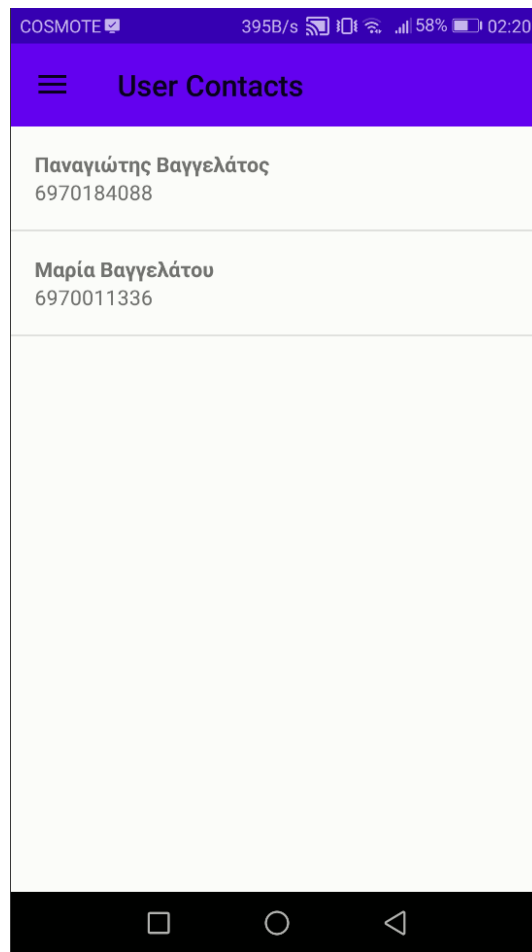


Figure 7: Οθόνη Επαφές

Οθόνη: Προσθήκη / Επεξεργασία επαφών

Πριν ο χρήστης μπορέσει να πραγματοποιήσει οποιαδήποτε βιντεοκλήση, θα πρέπει να υπάρχουν διαθέσιμες επαφές και για αυτό τον σκοπό δημιουργήθηκε η οθόνη αυτή. Για την προσθήκη επαφών ο χρήστης θα πρέπει στην οθόνη «Επαφές» να επιλέξει πάνω αριστερά τις 3 οριζόντιες παύλες ώστε να εμφανιστεί το navigation drawer και να κάνει κλικ στην επιλογή «Προσθήκη Επαφής».

Στο νέο παράθυρο που θα εμφανιστεί, ο χρήστης συμπληρώνει το ονοματεπώνυμο και το τηλέφωνο και στην συνέχεια κάνοντας κλικ στο κουμπί «Προσθήκη», τα δεδομένα αυτά αποθηκεύονται στην cloud βάση δεδομένων (Firebase Realtime Database) και αυτόματα εμφανίζεται η επαφή αυτή στην οθόνη «Επαφές».

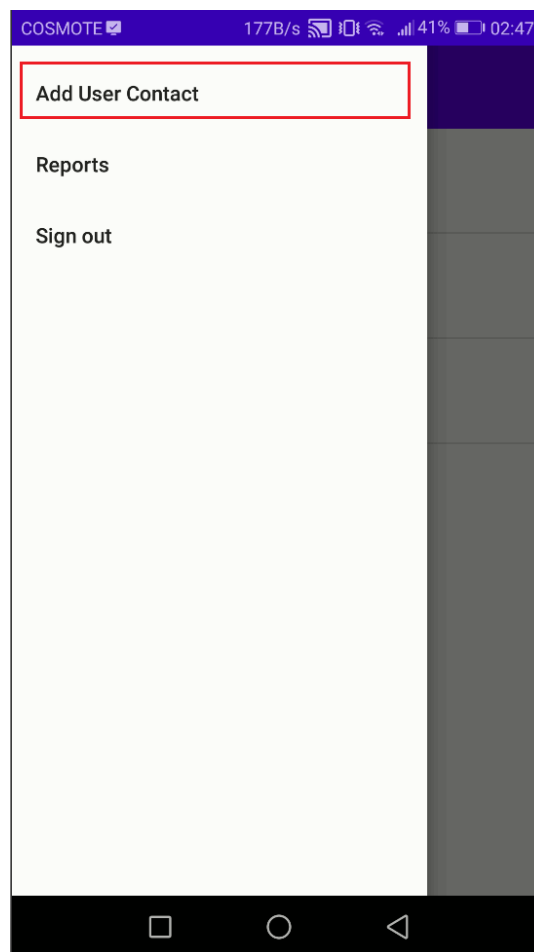
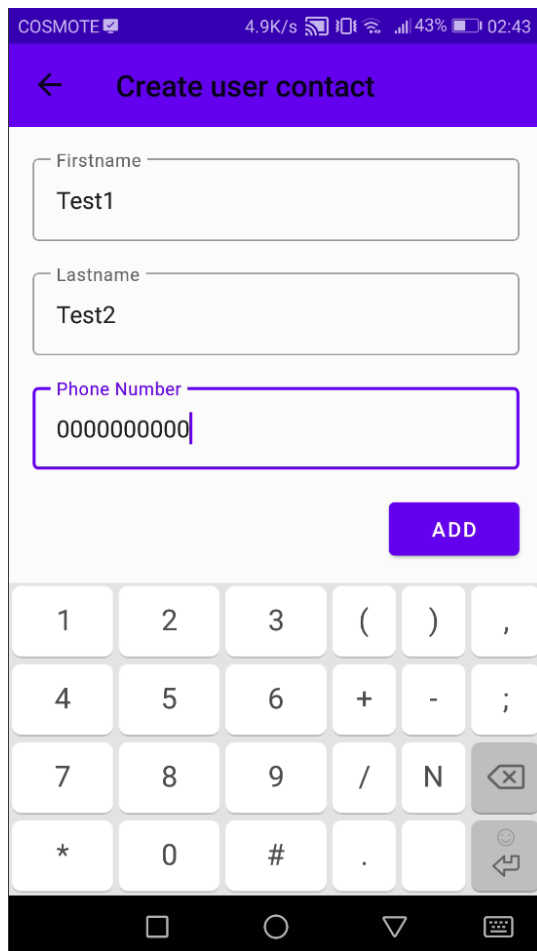
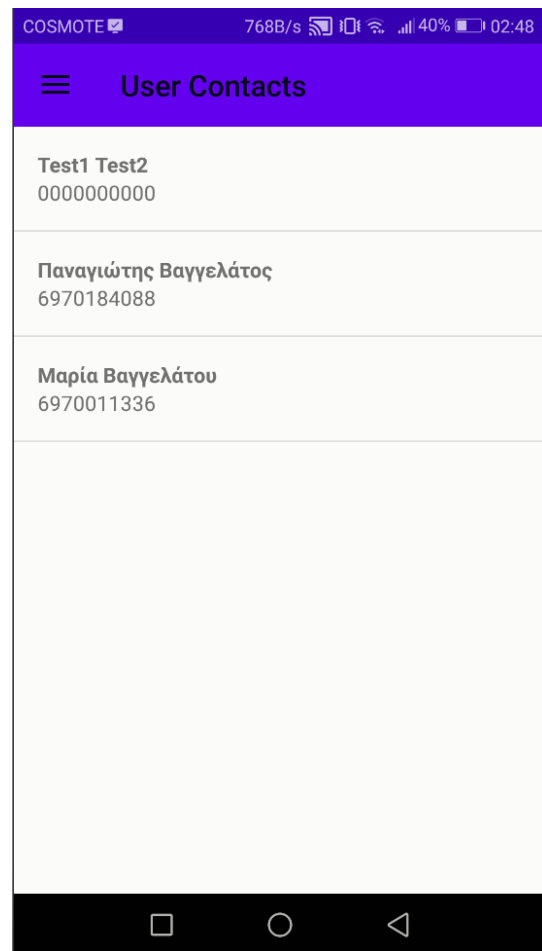


Figure 8: Navigation Drawer - Προσθήκη Επαφής

**Figure 9: Οθόνη Δημιουργίας Επαφής****Figure 10: Οθόνη Επαφές - Νέα Επαφή**

Η ίδια οθόνη χρησιμοποιείται και για την επεξεργασία μιας υπάρχουσας επαφής. Για να επεξεργαστεί ο χρήστης μια επαφή, στην οθόνη «Επαφές» πιέζει παρατεταμένα την επαφή που επιθυμεί ώστε να εμφανιστούν οι 3 διαθέσιμες επιλογές (Επεξεργασία, Βιντεοκλήσεις και Διαγραφή) και κάνοντας κλικ στην επιλογή «Επεξεργασία» τότε εμφανίζεται ένα νέο παράθυρο με τα υπάρχοντα στοιχεία της επαφής αυτής, όπου μπορεί ο χρήστης να τα επεξεργαστεί. Κάνοντας κλικ στο κουμπί «Ενημέρωση», τα νέα δεδομένα αποθηκεύονται στην cloud βάση δεδομένων και την ίδια στιγμή ενημερώνονται αυτόματα τα στοιχεία στην οθόνη «Επαφές».

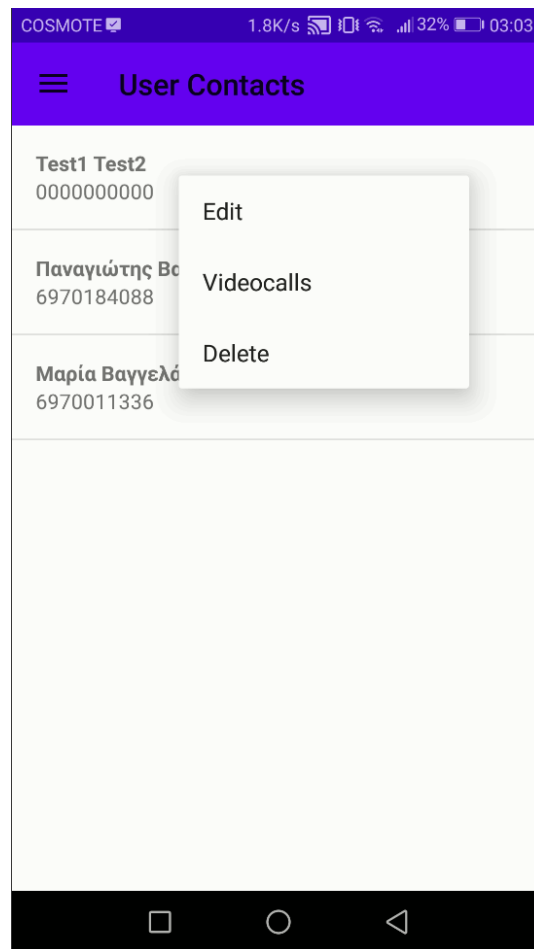


Figure 11: Οθόνη Επαφές - Επεξεργασία Επαφής

COSMOTE 2.7K/s 31% 03:05

← Edit user contact

Firstname
Σπύρος

Lastname
Βαγγελάτος

Phone Number
0000000001

UPDATE

**Figure 12: Οθόνη Επεξεργασίας
Επαφής**

COSMOTE 2.2K/s 31% 03:05

☰ User Contacts

Σπύρος Βαγγελάτος
0000000001

Παναγιώτης Βαγγελάτος
6970184088

Μαρία Βαγγελάτου
6970011336

**Figure 13: Οθόνη Επαφής -
Ενημερωμένη Επαφή**

Οθόνη: Καταγραφή συναισθημάτων και αντικειμένων

περιβάλλοντος

Στην οθόνη αυτή πραγματοποιείτε η βιντεοκλήση και η καταγραφή συναισθημάτων και αντικειμένων περιβάλλοντος. Αρχικά πριν ο χρήστης ξεκινήσει βιντεοκλήση με την επαφή που επιθυμεί, θα πρέπει να επιτρέψει στην εφαρμογή να αποκτήσει πρόσβαση στην κάμερα. Στην περίπτωση που δεν επιτραπεί η πρόσβαση, τότε ο χρήστης θα μεταφερθεί αυτόματα στην οθόνη «Επαφές», ενώ αν επιτραπεί η πρόσβαση, τότε θα ξεκινήσει η βιντεοκλήση.

Όταν ξεκινάει η βιντεοκλήση χρησιμοποιείται αυτόματα η μπροστινή κάμερα ενώ έχει δοθεί η δυνατότητα στον χρήστη να μπορεί να κάνει εναλλαγή μεταξύ την μπροστινής και της πίσω κάμερας. Επειδή το προγραμματιστικό κομμάτι όσο αφορά την κάμερα είναι αρκετά δύσκολο να υλοποιηθεί σωστά και να δουλεύει στις περισσότερες συσκευές της αγοράς και σε διάφορες εκδόσεις Android, γίνεται χρήση της βιβλιοθήκης AndroidX Camera της Android JetPack σουίτας που έχει υλοποιηθεί με γνώμονα την επίλυση αυτών των προβλημάτων. Βασικό χαρακτηριστικό της βιβλιοθήκης είναι τα User Cases που πρακτικά είναι έτοιμες διεργασίες που μπορούν να χρησιμοποιηθούν χωρίς ο προγραμματιστής να πρέπει να υλοποιήσει οτιδήποτε αφορά την κάμερα. Μέχρι αυτή την στιγμή η βιβλιοθήκη παρέχει τα παρακάτω User Cases:

- ❖ Προβολή, που έχει υλοποιημένη την λειτουργικότητα της προβολής φωτογραφίας στην οθόνη,
- ❖ Ανάλυση εικόνας, που παρέχει πρόσβαση στο buffer της κάμερας και επομένως ο προγραμματιστής έχει πρόσβαση στην εικόνα του κάθε στιγμιότυπου (frame) της κάμερας και που μπορεί να χρησιμοποιηθεί η εικόνα αυτή από άλλα APIs, όπως Firebase MLKit
- ❖ Λήψη εικόνα, που έχει υλοποιημένη την λειτουργικότητα της αποθήκευση της εικόνας σε υψηλή ποιότητα.

Όσο αφορά την καταγραφή συναισθημάτων και αντικειμένων περιβάλλοντος, χρησιμοποιήθηκε το User Case «Ανάλυση Εικόνας» της βιβλιοθήκης AndroidX Camera καθώς συνεργάζεται με το API της υπηρεσίας Firebase MLKit. Η εφαρμογή κάθε 5 δευτερόλεπτα διαβάζει το buffer της κάμερας του συγκεκριμένου στιγμιότυπου (frame) και στέλνει την συγκεκριμένη εικόνα στην υπηρεσία Firebase MLKit όπου γίνεται η ανάλυσή της για τον εντοπισμό των χαρακτηριστικών προσώπου και για τον εντοπισμό αντικειμένων περιβάλλοντος. Όταν ολοκληρωθεί η παραπάνω διαδικασία και σταλούν τα αποτελέσματα πίσω, τότε τα αποτελέσματα αποθηκεύονται στην cloud βάση δεδομένων. Αν και γνωρίζοντας ότι με αυτό τον τρόπο έχουμε μεγαλύτερη κατανάλωση network bandwidth, επιλέχθηκε αυτή η λογική διότι εξασφαλίζεται το γεγονός ότι η cloud βάση δεδομένων θα περιέχει πάντα τα τελευταία δεδομένα βιντεοκλήσης, το οποίο είναι σημαντικό για τις αναφορές που παρέχονται στον χρήστη.

Τέλος, όταν ο χρήστης επιλέξει να τερματιστεί η βιντεοκλήση, τότε ενημερώνεται στην cloud βάση δεδομένων και η ώρα τερματισμού της εκάστοτε βιντεοκλήσης.

Οθόνη: Λίστα βιντεοκλήσεων (ανά επαφή)

Στην οθόνη αυτή εμφανίζονται όλες οι βιντεοκλήσεις που έχουν πραγματοποιηθεί για την επαφή που έχει επιλέξει ο χρήστης στην οθόνη «Επαφές». Η σειρά που εμφανίζονται είναι κατά αύξουσα σειρά, δηλαδή από την νεότερη προς την παλιότερη. Για την κάθε μια βιντεοκλήση εμφανίζονται πληροφορίες όπως την ημερομηνία που πραγματοποιήθηκε η βιντεοκλήση καθώς και την ώρα έναρξης και τερματισμού της κάθε μία βιντεοκλήσης.

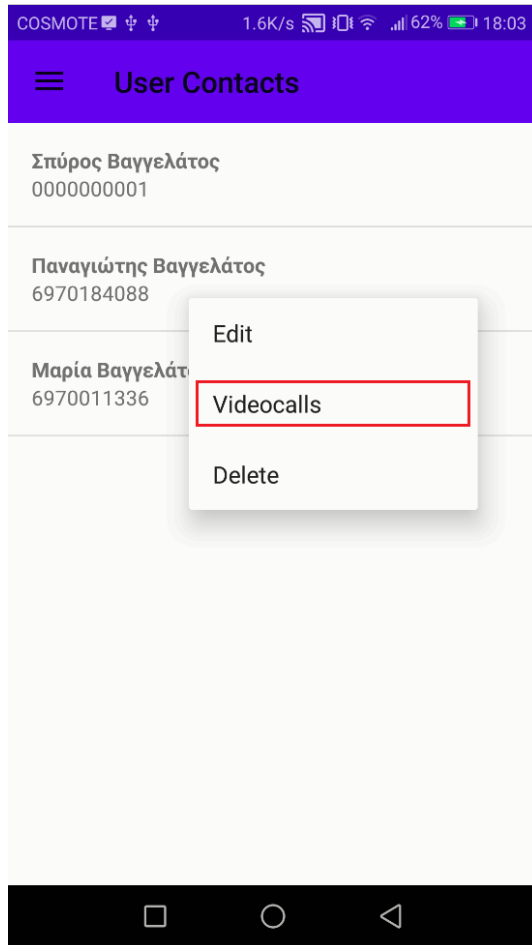


Figure 14: Οθόνη Επαφές – Μενού Στατιστικά βιντεοκλήσεων

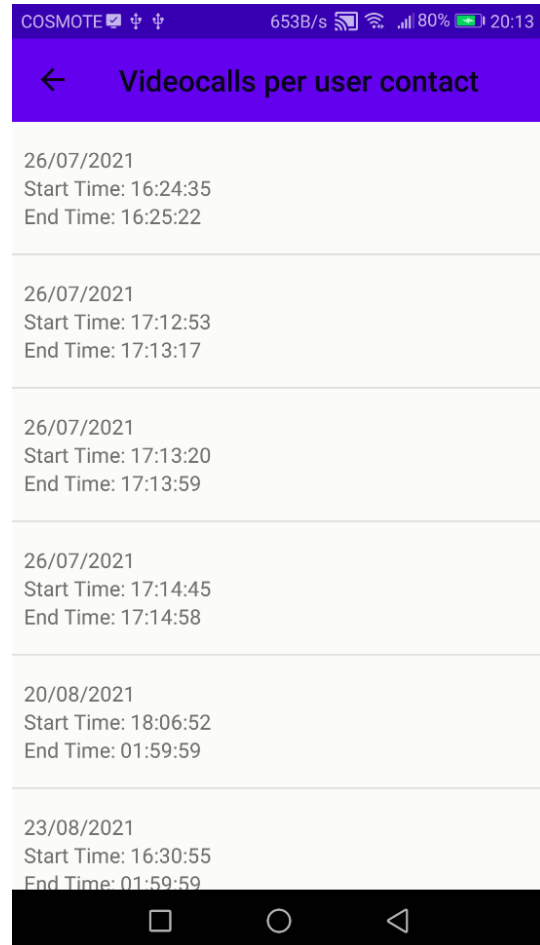


Figure 15: Οθόνη Στατιστικά Βιντεοκλήσεων

Οθόνη: Προβολή χαρακτηριστικών προσώπου και αντικειμένων περιβάλλοντος

Κάνοντας ο χρήστης κλικ στην βιντεοκλήση που επιθυμεί (της εικόνας 15), θα μεταβεί στην οθόνη «Αναφορές». Στην οθόνη αυτή ο χρήστης έχει την δυνατότητα να επιλέξει τι είδους στατιστικά επιθυμεί να προβάλει. Οι 2 κατηγορίες στατιστικών είναι «Χαρακτηριστικά Προσώπου» και «Εντοπισμένα Αντικείμενα».

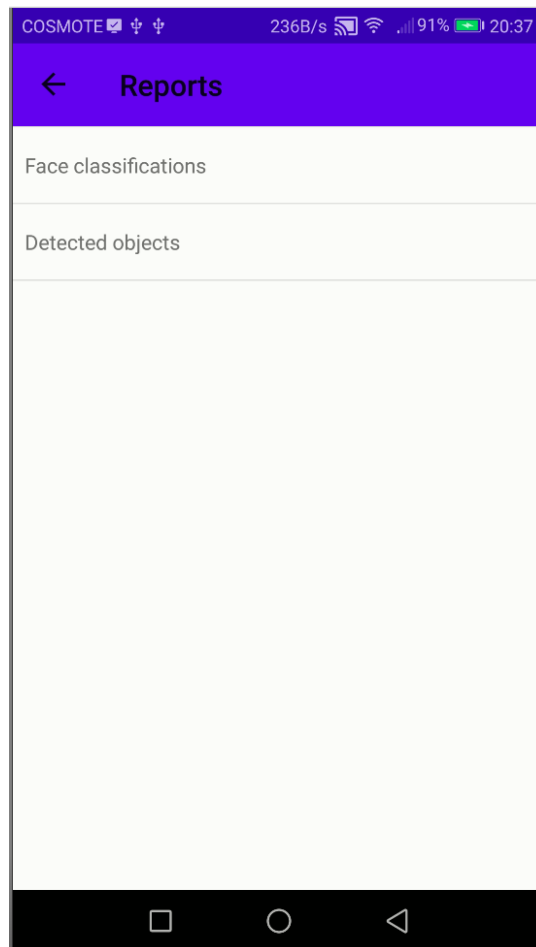
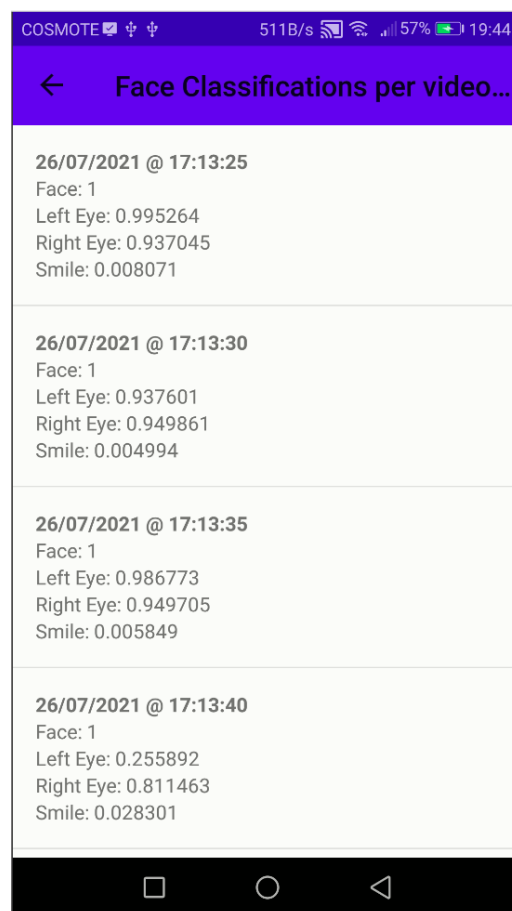


Figure 16: Οθόνη Αναφορές

Επιλέγοντας την επιλογή «Χαρακτηριστικά Προσώπου», ο χρήστης να μεταβεί στην οθόνη «Χαρακτηριστικά Προσώπου ανά Βιντεοκλήση». Στην οθόνη αυτή υπάρχει η λίστα με όλα τα στιγμιότυπα εικόνων που αναλύθηκαν από την υπηρεσία Firebase MLKit. Ξεκινώντας αναφέρεται η ημερομηνία και η ώρα που τραβήχτηκε το κάθε ένα στιγμιότυπο εικόνας. Επιπρόσθετα εμφανίζεται ποιο πρόσωπο αφορούν τα χαρακτηριστικά προσώπου που αναλύθηκαν. Τέλος εμφανίζονται οι πληροφορίες σχετικά με το πόσο ανοιχτά είναι τα μάτια και κατά πόσο χαμογελάει ή όχι το πρόσωπο που αναλύθηκε. Όταν οι τιμές των προηγούμενων χαρακτηριστικών προσώπου κυμαίνονται μεταξύ του 0 και 1. Όταν η τιμή που αφορά κατά πόσο ανοιχτά ή όχι είναι τα μάτια πλησιάζει το 1 τότε τόσο πιο ανοιχτά είναι τα μάτια ενώ αντίστοιχα όσο πιο κοντά είναι στο 0 τόσο πιο κλειστά είναι. Την ίδια λογική έχει και το χαμόγελο, δηλαδή όσο η τιμή του χαμόγελου είναι κοντά στο 1 τόσο πιο πολύ το πρόσωπο χαμογελάει και αντίστροφα όσο πιο κοντά είναι η τιμή στο 0, το πρόσωπο δεν χαμογελάει (δεν σημαίνει όμως ότι το πρόσωπο είναι λυπημένο).



Time	Face	Left Eye	Right Eye	Smile
26/07/2021 @ 17:13:25	1	0.995264	0.937045	0.008071
26/07/2021 @ 17:13:30	1	0.937601	0.949861	0.004994
26/07/2021 @ 17:13:35	1	0.986773	0.949705	0.005849
26/07/2021 @ 17:13:40	1	0.255892	0.811463	0.028301

Figure 17: Χαρακτηριστικά Προσώπου ανά Βιντεοκλήση

Επιλέγοντας την επιλογή «Εντοπισμένα Αντικείμενα», ο χρήστης να μεταβεί στην οθόνη «Αντικείμενα ανά Βιντεοκλήση». Στην οθόνη αυτή υπάρχει η λίστα με όλα τα εντοπισμένα αντικείμενα που αναγνωρίστηκαν από την υπηρεσία Firebase MLKit. Οι πληροφορίες που προβάλλονται στην οθόνη αυτή είναι η ημερομηνία και η ώρα που έγινε ο εντοπισμών αντικειμένων, το πλήθος των εντοπισμένων αντικειμένων καθώς και πόσες ετικέτες αντικειμένων αναγνωρίστηκαν.

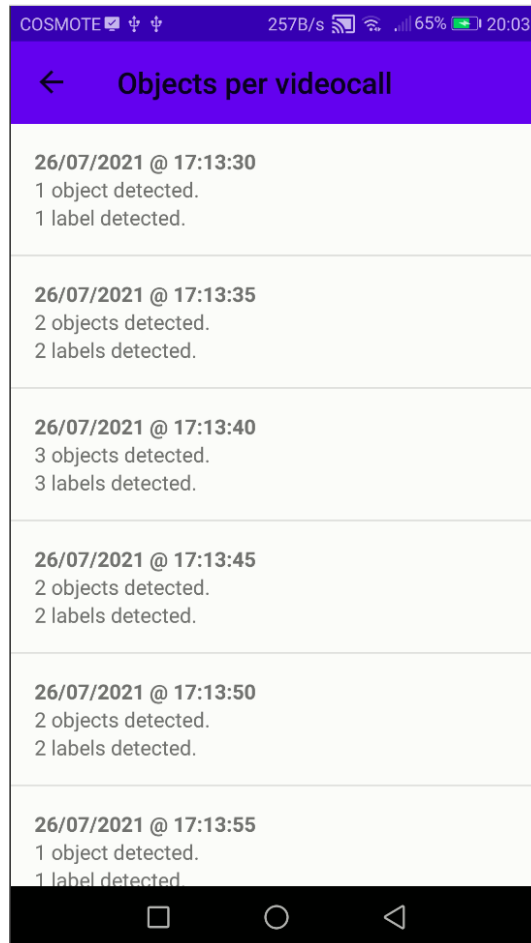


Figure 18: Οθόνη Αντικείμενα ανά Βιντεοκλήση

Επιπλέον, όταν ο χρήστη κάνει κλικ πάνω σε μια γραμμή, τότε θα μεταβεί στην οθόνη «Ετικέτες Αντικειμένων ανά Βιντεοκλήση και Ωρα». Η οθόνη αυτή περιέχει τι ετικέτες των αντικειμένων που αναγνωρίστηκαν.

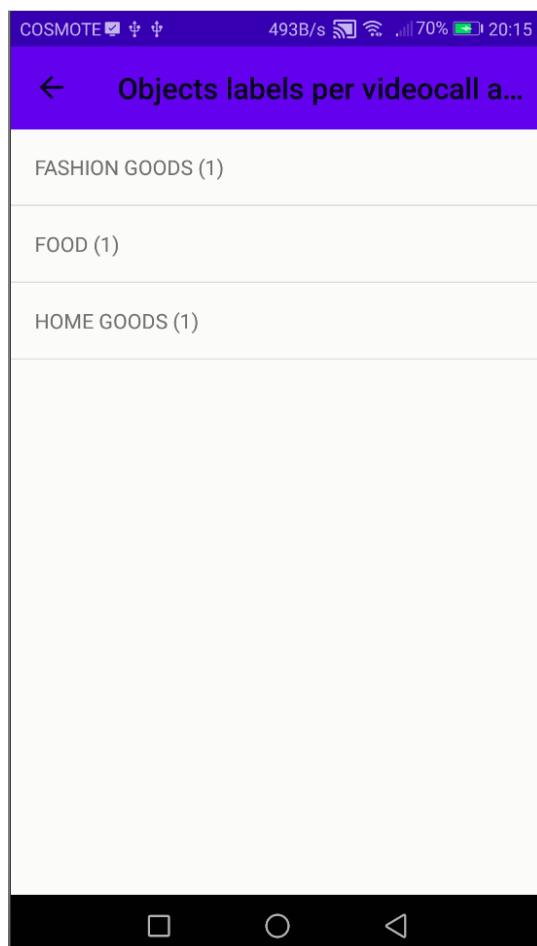


Figure 19: Οθόνη Ετικέτες Αντικειμένων ανά Βιντεοκλήση και Ωρα

Οθόνη: Αναφορές

Εκτός από τις αναφορές - στατιστικά για κάθε μια βιντεοκλήση του κάθε ένα χρήστη ξεχωριστά, παρέχετε στον χρήστη η δυνατότητα να δει αναφορές - στατιστικά που αφορούν όλες τις βιντεοκλήσεις από όλους του χρήστες. Για να μπορέσει να δει αυτές τις αναφορές - στατιστικά, θα πρέπει στην οθόνη «Επαφές» να επιλέξει πάνω αριστερά τις 3 οριζόντιες παύλες ώστε να εμφανιστεί το navigation drawer και να κάνει κλικ στην επιλογή «Αναφορές».

Κάνοντας κλικ στην επιλογή «Αναφορές», ο χρήστης θα μεταβεί στην οθόνη «Αναφορές» όπου και υπάρχουν 4 επιλογές να επιλέξει. Οι επιλογές αυτές είναι:

Βιντεοκλήσεις (Videocalls)

- ❖ Χαρακτηριστικά Προσώπου (Face Classifications)
- ❖ Εντοπισμένα Αντικείμενα (Detected Objects)
- ❖ Ετικέτες Εντοπισμένων Αντικειμένων (Detected Objects Labels)

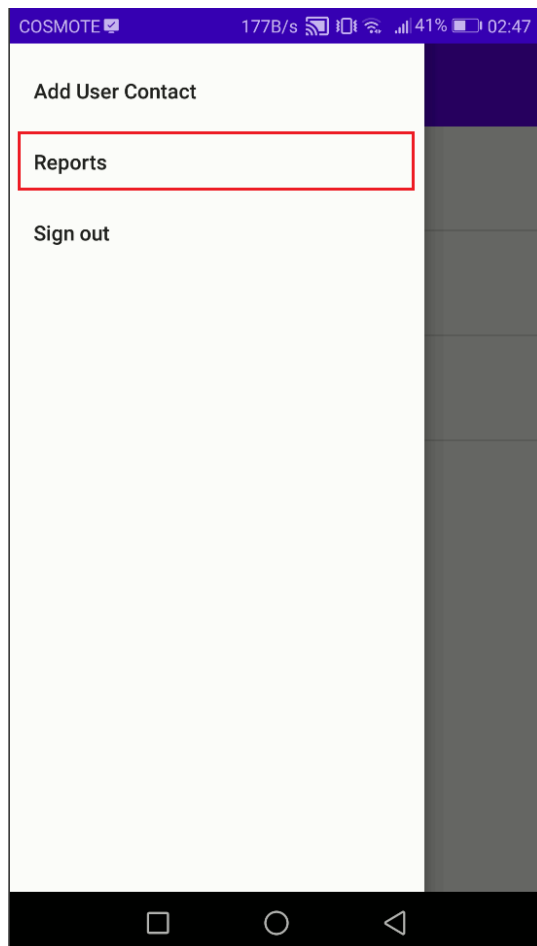


Figure 20: Navigation Drawer - Αναφορές

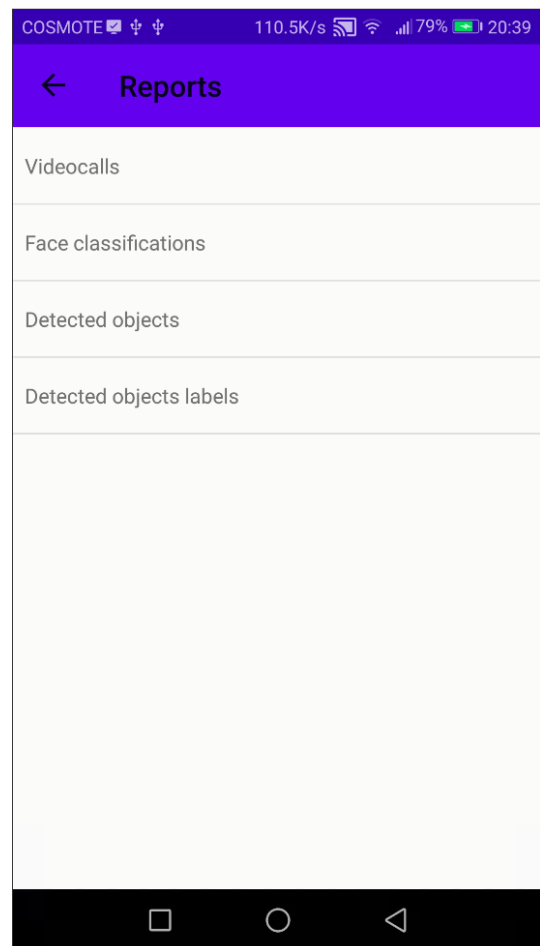


Figure 21: Οθόνη Αναφορές

Οθόνη: Βιντεοκλήσεις (Γενικές Αναφορές)

Κάνοντας ο χρήστης κλικ στην πρώτη επιλογή, Βιντεοκλήσεις, θα μεταβεί στην οθόνη «Βιντεοκλήσεις». Στην οθόνη αυτή υπάρχουν 5 διαφορετικά στατιστικά που μπορεί να δει ο χρήστης. Το πρώτο στατιστικό είναι το πλήθος των βιντεοκλήσεων που έχουν γίνει συνολικά από όλους τους χρήστες της εφαρμογής. Τα υπόλοιπα 4 αφορούν το πόσες βιντεοκλήσεις πραγματοποιήθηκαν πρωί, μεσημέρι, απόγευμα και βράδυ. Στα 4 αυτά στατιστικά εκτός από το πλήθος των βιντεοκλήσεων, αναγράφεται και το ποσοστό επί της εκατό με βάση το συνολικό πλήθος των βιντεοκλήσεων.

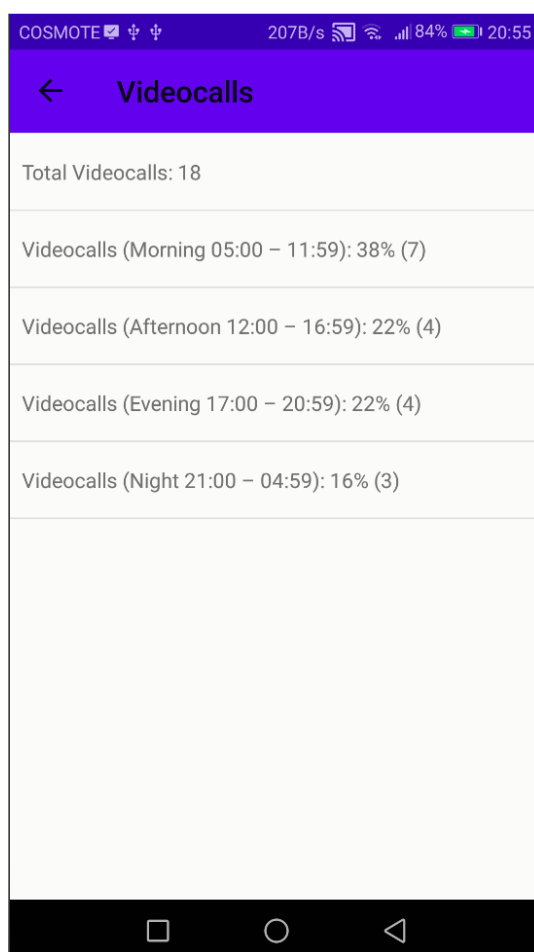


Figure 22: Οθόνη Βιντεοκλήσεις (Γενικές Αναφορές)

Οθόνη: Χαρακτηριστικά Προσώπου (Γενικές Αναφορές)

Κάνοντας ο χρήστης κλικ στην επιλογή «Χαρακτηριστικά Προσώπου» τότε φορτώνεται η οθόνη «Χαρακτηριστικά Προσώπου». Στην οθόνη αυτή περιέχονται πληροφορίες σχετικά για το πλήθος των προσώπων που αναλύθηκαν από την υπηρεσία Firebase MLKit από όλες τις βιντεοκλήσεις από όλους τους χρήστες. Επιπλέον, περιέχονται πληροφορίες σχετικά με το πόσα πρόσωπα είχαν ανοιχτά και τα δύο μάτια, πόσα κλειστά και τα δύο μάτια, πόσα πρόσωπα είχαν ένα από τα δύο μάτια ανοιχτά και τέλος πόσα πρόσωπα χαμογελούσαν. Σε όλα τα στατιστικά αυτά εκτός από το πλήθος αναγράφεται και το ποσοστό επί της εκατό.

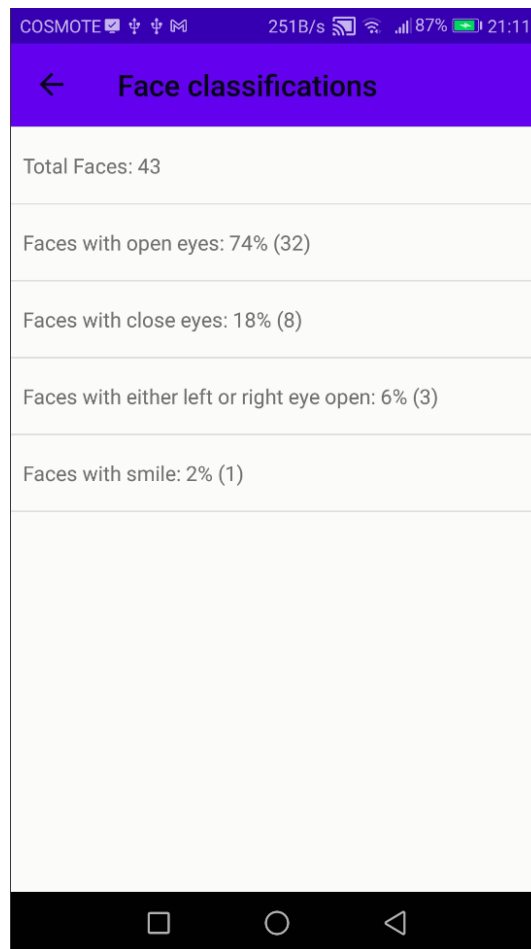


Figure 23: Οθόνη Χαρακτηριστικά Προσώπου (Γενικές Αναφορές)

Οθόνη: Αντικείμενα ανά 3 έτη

Επιλέγοντας ο χρήστης στην επιλογή «Εντοπισμένα Αντικείμενα», θα μεταβεί στην οθόνη «Αντικείμενα ανά 3 έτη». Στην οθόνη παρέχετε στον χρήστη να δει πληροφορίες σχετικά με τα εντοπισμένα αντικείμενα για τα 2 χρόνια πριν, το προηγούμενο έτος καθώς και το τρέχον έτος. Επιλέγοντας ο χρήστης μια επιλογή που δεν περιέχει δεδομένα, θα του εμφανιστεί ένα pop-up με το αντίστοιχο μήνυμα. Ενώ αν επιλεγθεί μια επιλογή που περιέχει δεδομένα, τότε ο χρήστης θα μεταβεί στην οθόνη που αναφέρονται ανά μήνα (του αντίστοιχου έτους που επιλέχθηκε) τόσα αντικείμενα ανιχνεύτηκαν.

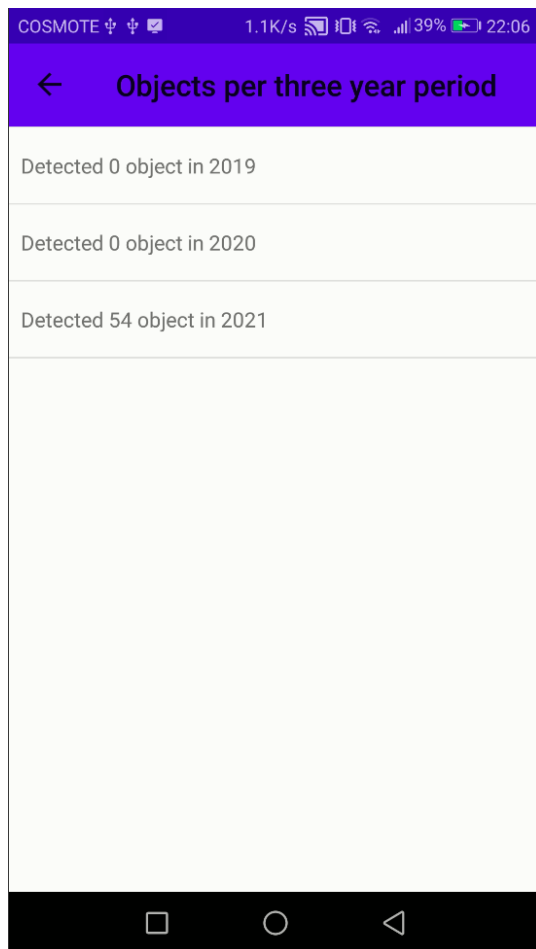


Figure 24: Οθόνη Αντικείμενα ανά 3 έτη

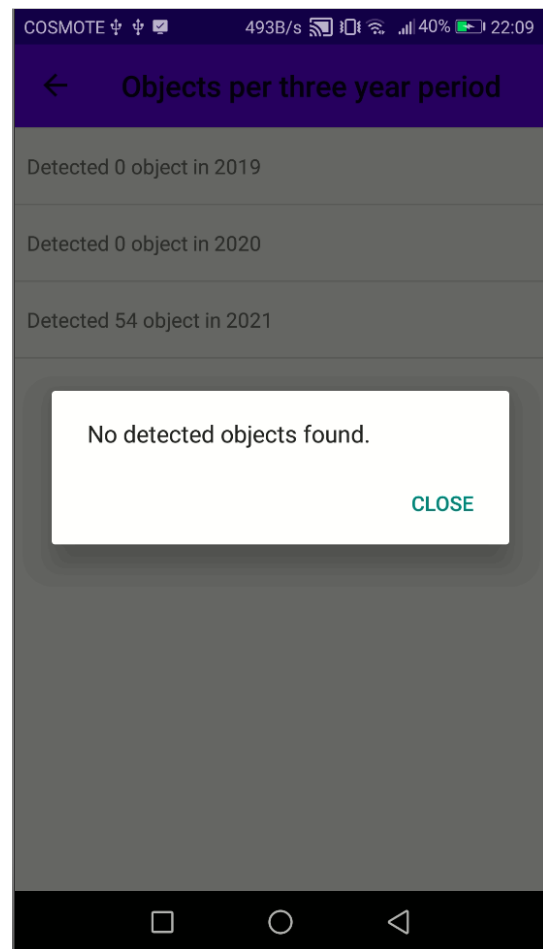


Figure 25: Δεν βρέθηκαν δεδομένα

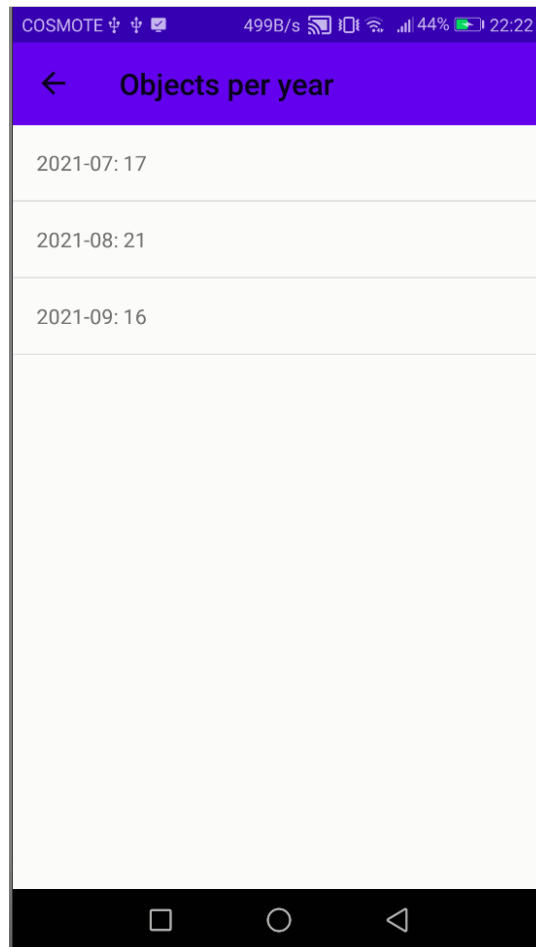


Figure 26: Οθόνη Αντικείμενα ανά μήνα (συγκεκριμένου έτους)

Οθόνη: Ετικέτες αντικειμένων ανά 3 έτη

Επιλέγοντας ο χρήστης στην επιλογή «Ετικέτες Εντοπισμένων Αντικειμένων», θα μεταβεί στην οθόνη «Ετικέτες Εντοπισμένων Αντικειμένων ανά 3 έτη». Στην οθόνη παρέχετε στον χρήστη να δει πληροφορίες σχετικά με τις ετικέτες των εντοπισμένων αντικειμένων για τα 2 χρόνια πριν, το προηγούμενο έτος καθώς και το τρέχον έτος. Επιλέγοντας ο χρήστης μια επιλογή που δεν περιέχει δεδομένα, θα του εμφανιστεί ένα pop-up με το αντίστοιχο μήνυμα. Ενώ αν επιλεγθεί μια επιλογή που περιέχει δεδομένα, τότε ο χρήστης θα μεταβεί στην οθόνη που αναφέρονται ανά μήνα (του αντίστοιχου έτους που επιλέχθηκε) πόσες ετικέτες αντικειμένων ανιχνεύτηκαν. Τέλος, όταν ο χρήστης επιλέξει και τον μήνα που επιθυμεί να δει τις ετικέτες, θα μεταβεί στην αντίστοιχη οθόνη ότι αναφέρονται οι ετικέτες και πόσες φορές ανιχνεύθηκε η κάθε μια ετικέτα.

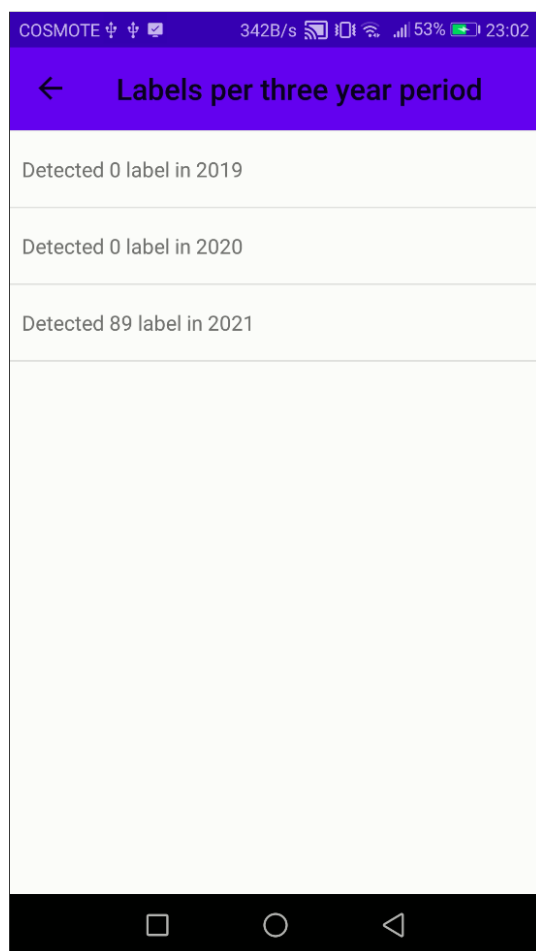


Figure 27: Οθόνη Ετικέτες ανά 3 έτη

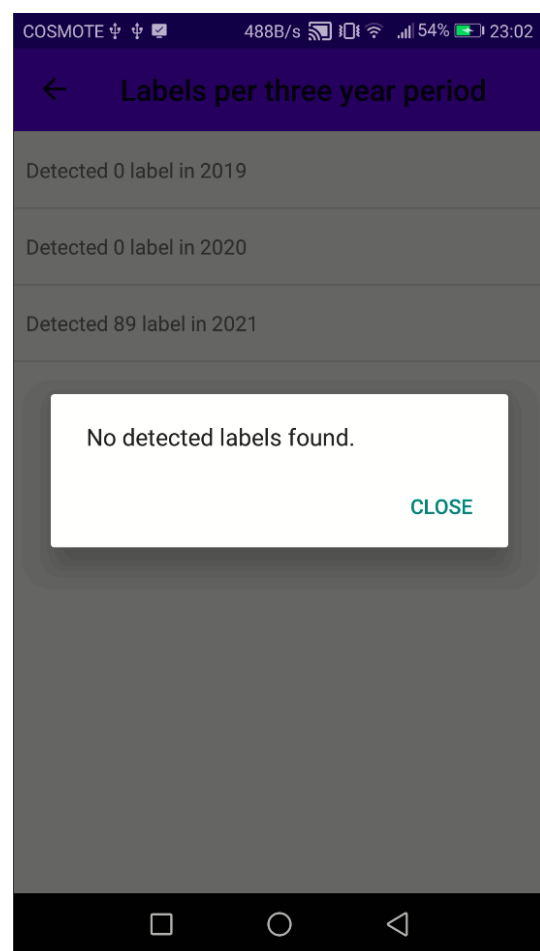


Figure 28: Δεν βρέθηκαν δεδομένα (Ετικέτες)

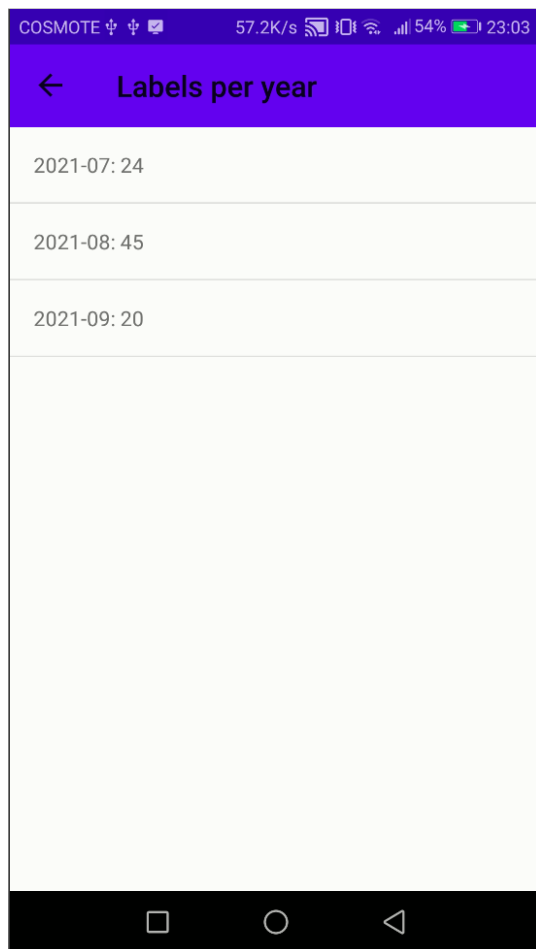


Figure 29: Οθόνη Πλήθος ετικετών ανά μήνα

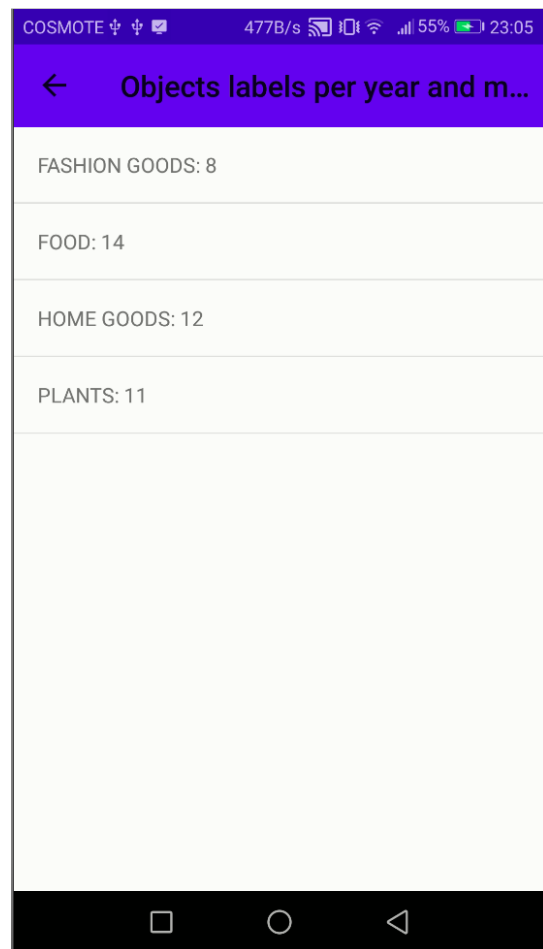


Figure 30: Οθόνη Ετικετών (συγκεκριμένου μήνα)

Συμπεράσματα

Συνοψίζοντας, στην σημερινή καθώς η τεχνολογία εξελίσσεται, βλέπουμε ότι έχουμε φτάσει σε ένα σημείο που η τεχνική νοημοσύνη έχει γίνει κομμάτι της ζωής μας (είτε για καλό σκοπό είτε για κακό σκοπό). Πέρα όμως από την τεχνική νοημοσύνη, βλέπουμε ότι και οι cloud υπηρεσίες έχουν γίνει ένα μέρος της ζωής μας, προσφέροντάς μας εκτός από υπηρεσίες και ολόκληρες υποδομές που δύσκολα θα μπορούσαν να υλοποιηθούν από μια εταιρία εξ 'αρχής.

Στα πλαίσια της διπλωματικής εργασίας προσπαθήσαμε να δείξουμε ένα μέρος από τις δυνατότητες που προσφέρονται από την Google τόσο σε επίπεδο υποδομής (όσο αφορά την cloud βάση δεδομένων) όσο και την υπηρεσία που αφορά την τεχνική νοημοσύνη, εφαρμόζοντάς τα σε μια εφαρμογή για κινητά που αναλύει σε πραγματικό χρόνο τα χαρακτηριστικά προσώπου καθώς και τα αντικείμενα που υπάρχουν στον περιβάλλον στις εικόνες που καταγράφονται από την κάμερα. Με τα δεδομένα αυτά είχαμε την δυνατότητα μέσω αναφορών να βγάλουμε κάποια στατιστικά τα οποία είναι χρήσιμα από κάποια εταιρία.

Μέσω των στατιστικών αυτών, θα μπορούσαν να δημιουργηθούν αναφορές σχετικά με το πότε είναι η πιο βέλτιστη ώρα για να γίνονται τα meetings, χρησιμοποιώντας τα στατιστικά που αφορούν τα χαρακτηριστικά προσώπου. Επιπρόσθετα, θα μπορούσε να δημιουργηθεί κάποια αναφορά που να αναφέρει τι αντικείμενα προτιμούν οι χρήστες να έχουν τριγύρω τους κατά την διάρκεια των meetings.

Βιβλιογραφία - Αναφορές

Google. (2020, 07 16). *AndroidX Overview*. Ανάκτηση από Android Developers:

<https://developer.android.com/jetpack/androidx>

Google. (2020, 12 17). *Pass data between destinations*. Ανάκτηση από Android Developers:

<https://developer.android.com/guide/navigation/navigation-pass-data#Safe-args>

Google. (2021, 04 14). *ARN*. Ανάκτηση από Android Developers:

<https://developer.android.com/topic/performance/vitals/anr>

Google. (2021, 03 19). *Camera X*. Ανάκτηση από Android Developers:

<https://developer.android.com/training/camerax>

Google. (2021, 06 01). *Firebase Authentication*. Ανάκτηση από Firebase:

<https://firebase.google.com/docs/auth>

Google. (2021, 06 01). *Firebase Machine Learning*. Ανάκτηση από Firebase:

https://firebase.google.com/docs/ml#cloud_vs_on-device

Google. (2021, 06 01). *Firebase Realtime Database*. Ανάκτηση από Firebase:

<https://firebase.google.com/docs/database>

Google. (2021, 05 20). *Getting started with Android Jetpack*. Ανάκτηση από Android

Developers: <https://developer.android.com/jetpack/getting-started>

Google. (2021, 06 02). *Lifecycle*. Ανάκτηση από Android Developers:

<https://developer.android.com/jetpack/androidx/releases/lifecycle#java>

Google. (2021, 06 01). *ML Kit Guides*. Ανάκτηση από Google Developer:

<https://developers.google.com/ml-kit/guides>

Google. (2021, 03 29). *Navigation*. Ανάκτηση από Android Developers:

<https://developer.android.com/guide/navigation>

Google. (2021, 05 20). *View Binding*. Ανάκτηση από Android Developers:

<https://developer.android.com/topic/libraries/view-binding>

Google. (2021, 04 28). *View Model Overview*. Ανάκτηση από Android Developers:

<https://developer.android.com/topic/libraries/architecture/viewmodel>

Google. (n.d.). *Machine learning for mobile developers*. Ανάκτηση από Google Developer:

<https://developers.google.com/ml-kit>

JetBrains. (n.d.). *Multiplatform Mobile*. Ανάκτηση από Kotlin: <https://kotlinlang.org/lp/mobile/>

LiveData Overview. (2021, 02 24). Ανάκτηση από Android Developer:

<https://developer.android.com/topic/libraries/architecture/livedata?hl=en>

Wikipedia. (2021, 05 10). *Android software development*. Ανάκτηση από Android software

development - Wikipedia: https://en.wikipedia.org/wiki/Android_software_development

Wikipedia. (2021, 05 28). *Java (programming language)*. Ανάκτηση από Wikipedia:

[https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))

Wikipedia. (2021, 05 30). *Kotlin (programming language)*. Ανάκτηση από Wikipedia:

[https://en.wikipedia.org/wiki/Kotlin_\(programming_language\)](https://en.wikipedia.org/wiki/Kotlin_(programming_language))

Πίνακας Εικόνων

Figure 1: nav_graph	16
Figure 2: Οθόνη Splash.....	17
Figure 3: Οθόνη δημιουργίας λογαριασμού	18
Figure 4: Οθόνη δημιουργίας λογαριασμού - Σφάλμα	18
Figure 5: Οθόνη Είσοδος	19
Figure 6: Οθόνη Είσοδος - Σφάλμα.....	19
Figure 7: Οθόνη Επαφές	21
Figure 8: Navigation Drawer - Προσθήκη Επαφής	22
Figure 9: Οθόνη Δημιουργίας Επαφής.....	23
Figure 10: Οθόνη Επαφές - Νέα Επαφή.....	23
Figure 11: Οθόνη Επαφές - Επεξεργασία Επαφής.....	24
Figure 12: Οθόνη Επεξεργασίας Επαφής.....	25
Figure 13: Οθόνη Επαφής - Ενημερωμένη Επαφή.....	25
Figure 14: Οθόνη Επαφές – Μενού Στατιστικά βιντεοκλήσεων	27
Figure 15: Οθόνη Στατιστικά Βιντεοκλήσεων	27
Figure 16: Οθόνη Αναφορές.....	28
Figure 17: Χαρακτηριστικά Προσώπου ανά Βιντεοκλήση	29
Figure 18: Οθόνη Αντικείμενα ανά Βιντεοκλήση	30
Figure 19: Οθόνη Ετικέτες Αντικειμένων ανά Βιντεοκλήση και Ώρα	31
Figure 20: Navigation Drawer - Αναφορές	32
Figure 21: Οθόνη Αναφορές.....	32
Figure 22: Οθόνη Βιντεοκλήσεις (Γενικές Αναφορές).....	33
Figure 23: Οθόνη Χαρακτηριστικά Προσώπου (Γενικές Αναφορές)	34
Figure 24: Οθόνη Αντικείμενα ανά 3 έτη	35
Figure 25: Δεν βρέθηκαν δεδομένα.....	35
Figure 26: Οθόνη Αντικείμενα ανά μήνα (συγκεκριμένου έτους)	36

Figure 27: Οθόνη Ετικέτες ανά 3 έτη.....	37
Figure 28: Δεν βρέθηκαν δεδομένα (Ετικέτες)	37
Figure 29: Οθόνη Πλήθος ετικετών ανά μήνα	38
Figure 30: Οθόνη Ετικετών (συγκεκριμένου μήνα)	38