

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
Τμήμα Πληροφορικής



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Κινητή Εφαρμογή Εντοπισμού Κειμένου Φωτογραφίας Προς Μετάφραση

Αλέξανδρος Μιχαλάρης, Π17196

Επιβλέπων: Αναπληρωτής Καθηγητής Ε. Σακκόπουλος

Πειραιάς, Νοέμβριος 2023



Επιτελική Σύνοψη

Η παρούσα διπλωματική εργασία παρουσιάζει το σχεδιασμό και την ανάπτυξη μιας εφαρμογής για κινητά που έχει κατασκευαστεί για Android ¹ συσκευές κάνοντας χρήση της προγραμματιστικής γλώσσας Kotlin². Η ανάπτυξη της εφαρμογής έγινε στο προγραμματιστικό περιβάλλον του Android Studio³ αξιοποιώντας τις δυνατότητες του Google ML Kit SDK⁴ για εξαγωγή γραμμάτων από εικόνες και στη συνέχεια για την μετάφραση του κειμένου σε διάφορες γλώσσες. Οι πρωταρχικοί στόχοι αυτής της έρευνας ήταν η διερεύνηση των δυνατοτήτων της μηχανικής μάθησης και επεξεργασίας εικόνας σε εφαρμογές για φορητές συσκευές, η ενίσχυση της γλωσσικής επικοινωνίας μέσω μετάφρασης σε πραγματικό χρόνο και η αξιολόγηση της ακρίβειας και της χρηστικότητας της αναπτυγμένης λύσης.

Το επίκεντρο της ανάπτυξης είναι η δημιουργία της Android εφαρμογής που ενσωματώνεται άψογα με το Google ML Kit SDK, ένα ισχυρό σύνολο εργαλείων μηχανικής εκμάθησης για εφαρμογές κινητών συσκευών. Η εφαρμογή χρησιμοποιεί αυτό το SDK για τη λήψη και επεξεργασία εικόνων, την εξαγωγή γραμμάτων και λέξεων και την αναγνώριση του κειμένου που περιέχεται στις εικόνες. Οι δυνατότητες OCR⁵ του Google ML Kit επιτρέπουν την ακριβή εξαγωγή κειμένου, ακόμη και από πολύπλοκες ή χειρόγραφες πηγές. Μόλις εξαχθούν με επιτυχία τα γράμματα, η εφαρμογή χρησιμοποιεί τη δυνατότητα μετάφρασης του ίδιου SDK για να μετατρέψει το κείμενο σε διάφορες γλώσσες, καθιστώντας το ένα ευέλικτο εργαλείο για πολύγλωσση επικοινωνία. Επιπλέον ο χρήστης έχει τη δυνατότητα αντιγραφής του κειμένου για την χρήση του εκτός της εφαρμογής.

Η διαδικασία ανάπτυξης είναι λεπτομερής, συμπεριλαμβανομένης της επιλογής εργαλείων, βιβλιοθηκών και πλαισίων, καθώς και ο αρχιτεκτονικός σχεδιασμός και η υλοποίηση της εφαρμογής αυτής. Η πτυχιακή εμβαθύνει επίσης στις προκλήσεις που αντιμετωπίζονται κατά την ανάπτυξη και συζητά τις στρατηγικές που χρησιμοποιούνται για την αντιμετώπισή τους, διασφαλίζοντας μια φιλική προς τον χρήστη και αποτελεσματική εφαρμογή.

Η εφαρμογή έχει τη δυνατότητα να ωφελήσει τους χρήστες σε διαφορετικά περιβάλλοντα, όπως ταξίδια, εκπαίδευση και επιχειρήσεις, καταρρίπτοντας τα γλωσσικά εμπόδια και προωθώντας την απρόσκοπτη επικοινωνία μεταξύ των γλωσσών.

¹ <https://www.android.com/what-is-android/>

² <https://kotlinlang.org/>

³ <https://developer.android.com/studio>

⁴ <https://developers.google.com/ml-kit>

⁵ <https://www.necc.mass.edu/wp-content/uploads/accessible-media-necc/uncategorized/resources/What-is-OCR.pdf>



Ευχαριστίες

Στο ταξίδι ολοκλήρωσης αυτής της διπλωματικής εργασίας, είμαι βαθύτατα ευγνώμων για την καθοδήγηση και την υποστήριξη του επίκουρο καθηγητή κ. Ευάγγελου Σακκόπουλου, του οποίου η τεχνογνωσία και η αμέριστη υποστήριξη συνέβαλαν καθοριστικά στη διαμόρφωση της βάσης αυτής της εργασίας. Επιπλέον, εκφράζω τις ευχαριστίες μου σε όλους τους άλλους αξιότιμους καθηγητές που συνέβαλαν στην ακαδημαϊκή μου ανάπτυξη. Η συλλογική τους σοφία, οι διαφορετικές προοπτικές και η αφοσίωσή τους στην αναζήτηση της γνώσης έχουν εμπλουτίσει την εκπαιδευτική μου εμπειρία και διεύρυνε τους ορίζοντές μου. Κάθε καθηγητής άφησε ένα ανεξίτηλο σημάδι στο πνευματικό μου ταξίδι, και γι' αυτό, είμαι ειλικρινά ευγνώμων. Επίσης, θα ήθελα να ευχαριστήσω τους γονείς μου για τη συνεχή στήριξη που μου δίνουν.

**Σεπτέμβριος 2023,
Αλέξανδρος Μιχαλάρης.**



ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Επιτελική Σύνοψη	2
Ευχαριστίες	3
Κεφάλαιο 1 ^ο	7
1 Εισαγωγή	7
1.1 Πρόλογος.....	7
1.2 Περιγραφή του υπό μελέτη προβλήματος	8
1.3 Σκοπός και στόχοι της εργασίας	8
1.4 Βασικοί ορισμοί, εργαλεία και τεχνολογίες που χρησιμοποιούνται	9
1.4.1 SDK	9
1.4.2 Kotlin Programming Language	9
1.4.3 Android Studio.....	10
1.4.4 Google ML Kit	10
1.4.5 OCR.....	10
1.4.6 Jetpack Compose.....	11
1.5 Παραδοτέα εργασίας	11
1.6 Περιεχόμενο επόμενων κεφαλαίων	11
Κεφάλαιο 2 ^ο	12
2 Επισκόπηση του χώρου	12
2.1 Google Translate	12
2.2 Abbyy TextGrabber	12
2.3 Microsoft Translator	13
2.4 Σύγκριση Τεχνολογιών	13
Κεφάλαιο 3 ^ο	14
3 Σχεδίαση της εφαρμογής	14
3.1 UML Use Case Diagram	14
3.2 Περιπτώσεις χρήσης Εφαρμογής.....	16
Τίτλος: «Εξαγωγή λέξεων φωτογραφίας και μετάφραση»	16
3.3 Sequence Diagram	18
Κεφάλαιο 4 ^ο	19
4 Περιγραφή βασικών στοιχείων υλοποίησης και κώδικα	19
4.1 MVVM Αρχιτεκτονική.....	19
4.2 Αναφορά βασικότερων σημείων κώδικα	21



4.2.1	Gradle Dependencies	21
4.2.2	ChooseOrTakePhotoView	22
4.2.3	CameraView	23
4.2.4	CropImageView	24
4.2.5	AppContentView	26
4.2.6	AppContentViewModel.....	27
	Κεφάλαιο 5°	29
5	Εγχειρίδιο Χρήσης Εφαρμογής.....	29
5.1	Εγκατάσταση εφαρμογής	29
5.2	Οδηγίες Χρήσης	31
5.2.1	Pick Image From Gallery.....	32
5.2.2	Take Photo.....	33
5.2.3	Εξαγωγή κειμένου	34
5.2.4	Μετάφραση εξαγόμενου κειμένου.....	38
	Κεφάλαιο 6°	42
6	Συμπεράσματα και προτάσεις για μελλοντικές επεκτάσεις	42
	Κεφάλαιο 7°	44
7	Βιβλιογραφικές πηγές	44



ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: UML Διάγραμμα εφαρμογής με τις περιπτώσεις χρήσης.....	14
Εικόνα 2: Sequence Διάγραμμα της εφαρμογής με την ροή χρήσης.	18
Εικόνα 3: Gradle Dependencies	21
Εικόνα 4: ChooseOrTakePhotoView	22
Εικόνα 5: Take Photo Function	23
Εικόνα 6: CropImageView Bitmap	24
Εικόνα 7: uriToBitmap και rotateBitmap functions	25
Εικόνα 8: AppContentView UI Components	26
Εικόνα 9: Συνάρτηση για το extraction των λέξεων.	27
Εικόνα 10: Συνάρτηση για την μετάφραση της γλώσσας.	28
Εικόνα 11: Αρχικό παράθυρο εφαρμογής κατά την εκκίνηση.	31
Εικόνα 12: Gallery κινητού για επιλογή φωτογραφίας.	32
Εικόνα 13: Σελίδα όπου τραβάμε την φωτογραφία.....	33
Εικόνα 14: Παράθυρο επεξεργασίας εικόνας.....	34
Εικόνα 15: Δεξιά περιστροφή εικόνας.	34
Εικόνα 16: Αριστερή περιστροφή εικόνας.	34
Εικόνα 17: Περικοπή εικόνας.....	35
Εικόνα 18: Αντιγραφή αρχικού κειμένου.....	36
Εικόνα 19: Επικόλληση αρχικού κειμένου.	36
Εικόνα 20: Αρνητική αναγνώριση χαρακτήρων από την εικόνα.	37
Εικόνα 21: Επιλογή γλώσσας προς μετάφραση.	38
Εικόνα 22: Μετάφραση κειμένου.....	38
Εικόνα 23: Αντιγραφή μεταφρασμένου κειμένου.....	39
Εικόνα 24: Επικόλληση μεταφρασμένου κειμένου.....	39
Εικόνα 25: Αναγνώριση Ισπανικής Γλώσσας	40
Εικόνα 26: Μετάφραση Ισπανικών σε Αγγλικά.....	40

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1. Διάγραμμα σύγκρισης των συστημάτων που ερευνήσαμε και του δικού μας.....	13
---	----



Κεφάλαιο 1^ο

1 Εισαγωγή

1.1 Πρόλογος

Στην σύγχρονη ψηφιακή εποχή, ο κόσμος μας έχει γίνει τεράστιο μείγμα γλωσσών και πολιτισμών όπου συνδέονται όλες μέσω του Διαδικτύου. Ο πολλαπλασιασμός των εικόνων στην καθημερινή μας ζωή, οι οποίες μοιράζονται μέσα από διάφορες πλατφόρμες, έχει ξεπεράσει τα γλωσσικά όρια, φέρνοντας τους ανθρώπους όλο και πιο κοντά. Ωστόσο, αυτή η έκρηξη συνδεσιμότητας, έχει παρουσιάσει μια σημαντική πρόκληση: πως μπορεί να γεφυρωθεί το χάσμα της γλώσσας που συχνά μας χωρίζει όταν ερχόμαστε αντιμέτωποι με εικόνες που περιέχουν ξενόγλωσσο κείμενο.

Καθώς πορευόμαστε σε αυτόν τον παγκοσμιοποιημένο κόσμο, λαμβάνει χώρα ένας αξιοσημείωτος μετασχηματισμός, όπου καθοδηγείται από τη σύγκλιση της επιστήμης των υπολογιστών και της τεχνητής νοημοσύνης. Στην προσπάθεια να αξιοποιήσουμε τη δύναμη της τεχνολογίας για να βελτιώσουμε την επαφή μας με τον άνθρωπο, αναδύθηκε μία νέα γέφυρα, όπου οι εικόνες μπορούν να μεταφραστούν αβίαστα, ξεπερνώντας τα όρια της γλώσσας.

Η βάση αυτής της εργασίας είναι μια εφαρμογή μετάφρασης εικόνων που έχει σχεδιαστεί για να ανιχνεύει και να μεταφράζει κείμενο από εικόνες. Η εφαρμογή είναι ένα πρακτικό παράδειγμα του τρόπου με τον οποίο η σύγχρονη τεχνολογία μπορεί να χρησιμοποιηθεί για να ξεπεραστούν τα γλωσσικά εμπόδια. Στο επίκεντρο της ανάπτυξης της εφαρμογής βρίσκεται η χρήση ενός υπάρχοντος εργαλείου ανάπτυξης λογισμικού (SDK) που ονομάζεται ML Kit, το οποίο ενσωματώνει δυνατότητες μηχανικής εκμάθησης και όρασης υπολογιστή από την Google, έναν από τους γίγαντες στον κλάδο της τεχνολογίας.

Αυτή η δουλειά είναι κάτι περισσότερο από ένα απλό εργαλείο. Είναι μια απόδειξη των δυνατοτήτων του ψηφιακού μας κόσμου. Αυτή είναι η απόδειξη ότι η πληροφορική συνεχίζει να εξελίσσεται και ότι τα όρια, όπου κάποτε θεωρούνταν αδύνατα, ωθούνται και ξεπερνιούνται συνεχώς. Μέσω της καινοτομίας και της αφοσίωσης, προσπαθούμε να ανοίξουμε το δρόμο για ένα μέλλον όπου η γλώσσα δεν θα είναι πλέον εμπόδιο, αλλά μια γέφυρα, που συνδέει άτομα και κοινότητες σε όλο τον κόσμο.



1.2 Περιγραφή του υπό μελέτη προβλήματος

Στην καθημερινή του ζωή, ο άνθρωπος έρχεται σε επαφή με ένα τεράστιο πλήθος εικόνων, είτε διαδικτυακά, είτε στην πραγματική ζωή.

Σε επίπεδο διαδικτύου, ένας άνθρωπος μπορεί να έρθει αντιμέτωπος με ένα κείμενο εικόνας στο οποίο μπορεί να θέλει να το αντιγράψει και να το μεταφέρει σε άλλα μέσα. Το κείμενο αυτό δε μπορεί να εξαχθεί όμως από την εικόνα καθώς είναι σε μορφή ψηφιακής φωτογραφίας. Σε δεύτερο χρόνο, το κείμενο αυτό μπορεί να είναι σε μια γλώσσα με την οποία ο χρήστης να μην είναι οικείος. Το ίδιο σενάριο μπορεί να εμφανιστεί και στην καθημερινή μας ζωή. Ένα κείμενο σε ένα poster, σε μια εφημερίδα, σε ένα εστιατόριο, ακόμα και σε μία σήμανση στο δρόμο μπορεί να είναι η πηγή ενός κειμένου που να θέλουμε να αντιγράψουμε (σε ψηφιακή μορφή) και να μεταφράσουμε σε μία άλλη γλώσσα.

Πριν την άνοδο της τεχνολογίας, οι λύσεις για αυτά τα προβλήματα ήταν η χειροκίνητη αντιγραφή κειμένου σε εφαρμογές μετάφρασης ή ακόμα και η χρήση λεξικών βιβλίων. Οι επιλογές πριν την εμφάνιση των αντίστοιχων τεχνολογικών λύσεων περιορίζονταν με την χρήση της συγκεκριμένης εφαρμογής, ένας άνθρωπος μπορεί τραβώντας μια φωτογραφία να εξερευνήσει ευκολότερα το περιβάλλον του.

1.3 Σκοπός και στόχοι της εργασίας

Ο σκοπός αυτής της πτυχιακής εργασίας είναι η ανάπτυξη, η σχεδίαση και η αξιολόγηση μιας Android εφαρμογής η οποία αξιοποιεί εργαλεία μηχανικής μάθησης και αναγνώρισης κειμένου επιτρέποντας σε έναν χρήστη την ανίχνευση και μετάφραση ενός κειμένου. Ο αρχικός στόχος είναι να δημιουργηθεί ένα φιλικό περιβάλλον εφαρμογής συνδυάζοντας της ακρίβεια και την αποτελεσματικότητα ενός εργαλείου που να μπορεί να βοηθήσει τους χρήστες του να ξεπεράσουν γλωσσικά εμπόδια στην καθημερινή τους ζωή. Το αποτέλεσμα θα είναι λειτουργικό μόνο σε Android συσκευές.

Οι στόχοι της διπλωματικής εργασίας είναι:

1. **Σχεδιασμός και ανάπτυξη μιας Android εφαρμογής:** Η δημιουργία μιας user--friendly εφαρμογής η οποία να δίνει την δυνατότητα στους χρήστες να καταγράφουν ή να επιλέγουν εικόνες που περιέχουν κείμενο.
2. **Ενσωμάτωση του ML Kit SDK εργαλείου της Google:** Ένα βασικό βήμα για την λειτουργικότητα της εφαρμογής είναι η ενσωμάτωση του εργαλείου ML Kit. Μέσω αυτής θα διερευνήσουμε και θα εφαρμόσουμε βέλτιστες πρακτικές για την εξαγωγή κειμένου.
3. **Ανάπτυξη εξαγωγής κειμένου:** Θα δημιουργήσουμε ένα σύστημα εξαγωγής κειμένου από εικόνα με την χρήση του παραπάνω εργαλείου.
4. **Ανάπτυξη μετάφρασης:** Θα δημιουργήσουμε μια ισχυρή μονάδα μετάφρασης αξιοποιώντας το ML Kit. Θα διερευνήσουμε επιπλέον την επιλογή μετάφρασης χωρίς προσβασιμότητα για περιοχές με περιορισμένο δίκτυο.



5. **Διερεύνηση Τεχνικών απόδοσης:** Διερεύνηση τεχνικών βελτιστοποίησης της εφαρμογής συμπεριλαμβανομένου και των πόρων χρήσης και του χρόνου απόκρισης προκειμένου να προσφέρουμε στον χρήστη μια ομαλή και ανταποκρινόμενη εμπειρία
6. **Σύγκριση με υπάρχουσες λύσεις:** Θα συγκρίνουμε την χρηστικότητα και τις δυνατότητες τις εφαρμογής, με υπάρχουσες εφαρμογές που υπάρχουν στο εμπόριο επισημαίνοντας τα πλεονεκτήματα και τις συνεισφορές της.

1.4 Βασικοί ορισμοί, εργαλεία και τεχνολογίες που χρησιμοποιούνται

Σε αυτή την ενότητα θα αναφέρουμε και θα αναλύσουμε του βασικότερους ορισμούς μεθόδων και τεχνολογιών που χρησιμοποιήθηκαν για την ανάπτυξης αυτής της εφαρμογής.

1.4.1 SDK

Ένα SDK, ή Κιτ ανάπτυξης λογισμικού, είναι ένα σύνολο εργαλείων λογισμικού και προγραμμάτων που χρησιμοποιούνται από προγραμματιστές για τη δημιουργία εφαρμογών για συγκεκριμένες πλατφόρμες (όπως για παράδειγμα το Android Studio). Τα εργαλεία SDK μπορούν να περιλαμβάνουν μια σειρά πραγμάτων, όπως βιβλιοθήκες, τεκμηρίωση, δείγματα κώδικα, διαδικασίες και οδηγούς που οι προγραμματιστές μπορούν να χρησιμοποιήσουν και να ενσωματώσουν στις δικές τους εφαρμογές. Τα SDK παρέχονται συχνά από τον κατασκευαστή μιας πλατφόρμας υλικού, ενός λειτουργικού συστήματος (OS) ή μιας γλώσσας προγραμματισμού. Βοηθούν τους προγραμματιστές λογισμικού να δημιουργήσουν εφαρμογές για τη συγκεκριμένη πλατφόρμα, σύστημα ή γλώσσα προγραμματισμού⁴. Ένα τυπικό SDK μπορεί να περιλαμβάνει πόρους όπως μεταγλωττιστή (που μεταφράζεται από μια γλώσσα προγραμματισμού σε αυτήν στην οποία θα εργαστείτε), δείγματα κώδικα, βιβλιοθήκες κώδικα, εργαλεία δοκιμών και ανάλυσης, τεκμηρίωση και προγράμματα εντοπισμού σφαλμάτων.

1.4.2 Kotlin Programming Language

Η Kotlin είναι μία σύγχρονη γλώσσα προγραμματισμού υψηλού επιπέδου και γενικής χρήσης, όπου απέκτησε μεγάλη δημοτικότητα τα τελευταία χρόνια. Δημιουργήθηκε από την JetBrains το 2011 και έκτοτε έγινε επίσημη γλώσσα για την ανάπτυξη εφαρμογών Android, παράλληλα με την Java. Αυτό που ξεχωρίζει το Kotlin είναι δυνατότητα λειτουργίας της σε συνδυασμό με την Java, διευκολύνοντας τους προγραμματιστές να μεταβούν από Java στο Kotlin και να αξιοποιήσουν τις υπάρχουσες βιβλιοθήκες Java. Ενσωματώνει πολλές δυνατότητες από άλλες γλώσσες προγραμματισμού, προσφέροντας έναν ισχυρό και συνοπτικό τρόπο για τη σύνταξη κώδικα για ένα ευρύ φάσμα εφαρμογών, από εφαρμογές για κινητά μέχρι ανάπτυξη ιστού και πολλά άλλα. Είναι πιο ελαφρύ, πιο καθαρό και πιο συνοπτικό από την Java, ιδιαίτερα όταν πρόκειται για τη σύνταξη κλάσεων. Είναι επίσης πιο ασφαλές, με την ενσωματωμένη μηδενική ασφάλεια ένα κύριο χαρακτηριστικό.



1.4.3 Android Studio

Το Android Studio είναι ένα ισχυρό και ευρέως χρησιμοποιούμενο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) ειδικά σχεδιασμένο για τη δημιουργία εφαρμογών Android. Αναπτύχθηκε από την Google, παρέχει στους προγραμματιστές ένα ολοκληρωμένο σύνολο εργαλείων και λειτουργιών για τις διαδικασίες ανάπτυξης εφαρμογών. Το Android Studio προσφέρει μια πλούσια και φιλική προς το χρήστη διεπαφή, καθιστώντας εύκολη τη σύνταξη, τη δοκιμή και τον εντοπισμό σφαλμάτων κώδικα για εφαρμογές Android. Περιλαμβάνει έναν ισχυρό εξομοιωτή για τη δοκιμή εφαρμογών σε εικονικές συσκευές και εκτενείς βιβλιοθήκες. Βασισμένο στο λογισμικό της JetBrains' IntelliJ IDEA, το Android Studio σχεδιάστηκε αποκλειστικά για προγραμματισμό Android. Είναι διαθέσιμο για Windows, Mac OS X και Linux, και αντικατέστησε τα Eclipse Android Development Tools (ADT) ως το κύριο IDE της Google για ανάπτυξη εφαρμογών Android.

1.4.4 Google ML Kit

Το ML Kit, ο πρωταγωνιστής αυτής της πτυχιακής, το οποίο αναπτύχθηκε από την Google, είναι ένα ευέλικτο εργαλείο SDK (Kit Ανάπτυξης Λογισμικού) για κινητά που έχει σχεδιαστεί με σκοπό να φέρει τη δύναμη της μηχανικής μάθησης σε εφαρμογές Android και iOS. Το ML Kit απλοποιεί την ενσωμάτωση των δυνατοτήτων μηχανικής μάθησης σε εφαρμογές για κινητά, καθιστώντας το «προσβάσιμο» σε ένα ευρύτερο φάσμα προγραμματιστών, ακόμη και σε εκείνους που δεν έχουν μεγάλη εμπειρία στη μηχανική εκμάθηση.

Το ML Kit περιέχει εκπαιδευμένα μοντέλα όπου καλύπτουν ένα μεγάλο φάσμα εργασιών, όπως η αναγνώριση εικόνας, η αναγνώριση κειμένου (OCR), η αναγνώριση QR Code, η ανίχνευση προσώπου, η μετάφραση γλώσσας κ.α. Αυτά τα μοντέλα είναι βελτιστοποιημένα για φορητές συσκευές, εξασφαλίζοντας αποτελεσματική και γρήγορη επεξεργασία, παρέχοντας παράλληλα ακριβή αποτελέσματα. Αυτό επιτρέπει στους προγραμματιστές να ενσωματώνουν προηγμένες λειτουργίες όπως ανίχνευση κειμένου σε πραγματικό χρόνο, απρόσκοπτα στις εφαρμογές τους χωρίς να χρειάζεται να αναπτύξουν πολύπλοκους αλγόριθμους μηχανικής εκμάθησης από την αρχή.

1.4.5 OCR

OCR σημαίνει «Optical Character Recognition». Αναφέρεται στην τεχνολογία όπου αναγνωρίζει ένα κείμενο μέσα από μία ψηφιακή εικόνα. Όπως θα δούμε στην συνέχεια, ένα τέτοιο λογισμικό χρησιμοποιείται για την μετατροπή ενός φυσικού εγγράφου (για παράδειγμα σε ένα χαρτί ή μια εικόνα) σε μια προσβάσιμη ηλεκτρονική μορφή. Για παράδειγμα, εάν σαρώσουμε ένα έγγραφο σε χαρτί ή μια φωτογραφία με έναν εκτυπωτή, ο εκτυπωτής πιθανότατα θα δημιουργήσει ένα αρχείο με ψηφιακή εικόνα. Το αρχείο θα μπορούσε να είναι JPG/TIFF ή PDF, αλλά το νέο ηλεκτρονικό αρχείο μπορεί να εξακολουθεί να είναι μόνο μια εικόνα του αρχικού εγγράφου. Στη συνέχεια, μπορούμε να το φορτώσουμε το σαρωμένο ηλεκτρονικό έγγραφο που δημιούργησε, το οποίο περιέχει την εικόνα, σε ένα πρόγραμμα OCR. Το OCR πρόγραμμα που θα αναγνωρίζει το κείμενο και θα μετατρέπει το έγγραφο σε επεξεργάσιμο αρχείο κειμένου όπου στην περίπτωση μας θα είναι η εφαρμογή Image Translator.



1.4.6 Jetpack Compose

Το Jetpack Compose είναι μια πρωτοποριακή και σύγχρονη εργαλειοθήκη Android UI που αναπτύχθηκε από την Google. Μειώνει την ποσότητα του κώδικα όπου χρειάζεται να γράψουμε πράγμα που σημαίνει ότι όχι μόνο είναι πιο γρήγορο να γράψουμε κώδικα αλλά είμαστε επιρρεπείς και σε λιγότερα σφάλματα. Το Jetpack Compose ενσωματώνεται με τα υπάρχοντα στοιχεία Android, επιτρέποντας στους προγραμματιστές να αξιοποιήσουν πλήρως την ισχύ του Android Studio, ενώ υιοθετούν έναν πιο διαισθητικό και αποτελεσματικό τρόπο σχεδίασης UI. Επιπλέον αξιοποιεί όλη την δύναμη της Kotlin καθιστώντας την ιδανική για ανάπτυξη UI σε Android εφαρμογές.

1.5 Παραδοτέα εργασίας

1. Το έντυπο κείμενο της πτυχιακής εργασίας.
2. Ο κώδικας με όλα τα αρχεία του project Image Translator
3. Μια σειρά από οδηγίες χρήσης όπου αναγράφονται στο έντυπο κείμενο της πτυχιακής.
4. Μια συλλογή πηγών και σχετικής βιβλιογραφίας με την οποία δημιουργήθηκε μια βάση γνώσης.

1.6 Περιεχόμενο επόμενων κεφαλαίων

Στα κεφάλαια που ακολουθούν θα αναλύσουμε τις λειτουργίες και τις δυνατότητες της εφαρμογής μας, συγκρίνοντας τες με παρόμοιες εφαρμογές του εμπορίου. Θα εμβαθύνουμε πολύ στην χρησιμότητα του ML Kit εργαλείου της Google καθώς επίσης και στις πρακτικές εφαρμογές αυτού σε συνδυασμό με το Android Studio. Στην συνέχεια θα αναλύσουμε τις περιπτώσεις χρήστη της εφαρμογής αναλύοντας τις λειτουργίες και τις απαιτήσεις της. Για την καλύτερη κατανόηση του αναγνώστη θα παρουσιαστούν και διαγράμματα UML. Έπειτα θα δούμε σημαντικά σημεία του κώδικα της εφαρμογής (συναρτήσεις, κλάσεις κ.ο.κ.) όπου αξίζει να σημειωθούν και να περιγραφούν. Τέλος θα δείξουμε και μια αναπαράσταση του τελικού αποτελέσματος της εφαρμογής μας απεικονίζοντας το πως γίνεται η εκτέλεση των λειτουργιών.



Κεφάλαιο 2^ο

2 Επισκόπηση του χώρου

Η δημοτικότητα των εφαρμογών μετάφρασης εικόνας έχει ανέβει ραγδαία τα τελευταία χρόνια χάρις την εξέλιξη της τεχνίτης νοημοσύνης και της τεχνολογίας των κινητών συσκευών. Οι εφαρμογές αυτές προσφέρουν ένα μεγάλο φάσμα δυνατοτήτων το οποίο θα αναλύσουμε στις επόμενες ενότητες του κεφαλαίου αυτού.

2.1 Google Translate⁶

Ίσως η δημοφιλέστερη εφαρμογή μετάφρασεων, το Google Translate είναι μία ευρέως γνωστή εφαρμογή μετάφρασης εικόνων. Για πολλούς ανθρώπους είναι το κύριο εργαλείο μετάφρασης της επιλογής τους κάνοντας το ένα από τα πιο σημαντικά μεταφραστικά εργαλεία στον κόσμο. Προσφέρει μετάφραση κειμένου σε πραγματικό χρόνο από εικόνες όπου μπορούν να τραβηχτούν από την κάμερα ενός smartphone καθώς επίσης διαθέτει και μια λειτουργία όπου την ονομάζει «Conversation Mode» όπου στην ουσία μεταφράζει συνομιλίες σε πραγματικό χρόνο. Όταν ο χρήστης είναι συνδεδεμένος με το διαδίκτυο, η εφαρμογή μπορεί να μεταφράσει και ολόκληρα έγγραφα συγκεκριμένου μεγέθους.

Παρόλο που η τεχνολογία αυτή είναι εντυπωσιακή, έχει μερικές αξιοσημείωτες αδυναμίες. Μία από αυτές είναι η ακρίβεια της αναγνώρισης χαρακτήρων όταν η εικόνα περιλαμβάνει σύνθετες γραμματοσειρές ή αν το κείμενο είναι χειρόγραφο, οδηγώντας έτσι σε λάθη και παρεξηγήσεις. Επιπλέον η μετάφραση εικόνας της Google εστιάζει κυρίως σε μετάφραση μεμονωμένων λέξεων και σύντομων φράσεων με αποτέλεσμα η τελική μετάφραση να στερείτε το νόημα.

2.2 Abby TextGrabber⁷

Το Abby TextGrabber είναι μία γνωστή εφαρμογή κινητών συσκευών όπου συνδυάζει την OCR τεχνολογία και την μετάφραση. Το TextGrabber εντοπίζει αυτόματα εάν η συσκευή έχει πρόσβαση σε σταθερή σύνδεση στο Διαδίκτυο και, εάν ναι, τη χρησιμοποιεί για να παρέχει μετάφραση πλήρους κειμένου σε πραγματικό χρόνο. Εάν η συσκευή δεν έχει πρόσβαση στο Διαδίκτυο, η εφαρμογή μεταβαίνει σε μετάφραση εκτός σύνδεσης, η οποία είναι απαραίτητη για ταξίδια και καθημερινές καταστάσεις, όπως η κατανόηση μενού, πινακίδες, ετικέτες και άλλα. Η μετάφραση εκτός σύνδεσης λειτουργεί για 10 γλώσσες, συμπεριλαμβανομένων των Αγγλικών, Ισπανικών, Γαλλικών, Γερμανικών, Κινέζων και Ιαπωνικών. Ο χρήστης έχει επίσης την επιλογή να μεταβεί στη λειτουργία εκτός σύνδεσης χειροκίνητα στις ρυθμίσεις.

⁶ <https://translate.google.com/?sl=auto&tl=en&op=images>

⁷ <https://www.abbyy.com/company/news/abbyy-textgrabber-for-android-adds-real-time-translation/>



Η εφαρμογή αυτή δεν είναι δωρεάν προς το κοινό έχοντας πακέτα για μηνιαία συνδρομή καθώς επίσης για one-time αγορά. Αυτό το κόστος είναι ένα μεγάλο μειονέκτημα για χρήστες με χαμηλό «μπατζετ». Επιπλέον υπάρχουν πολυάριθμες αναφορές χρηστών όπου υποστηρίζουν ότι το περιβάλλον χρήσης της εφαρμογής είναι λίγο πολύπλοκο με αποτέλεσμα να χρειάζεται κάποιος χρόνος να τη συνηθίσει.

2.3 Microsoft Translator ⁸

Αυτή η δωρεάν εφαρμογή της Microsoft προσφέρει μετάφραση κειμένου από εικόνα σε πάνω από 70 γλώσσες. Σε αντίθεση με το λογισμικό του Google Translate, η εφαρμογή αυτή δεν προσφέρει μετάφραση κειμένου σε πραγματικό χρόνο. Παρόλα αυτά λειτουργεί με εικόνες, είτε σε κατακόρυφη, είτε σε οριζόντια μορφή καθιστώντας το εύχρηστο για καταγραφή πιο εκτεταμένου γραπτού περιεχομένου. Ένα πλεονέκτημα επίσης είναι ότι προσφέρει την δυνατότητα στο χρήστη να κατεβάσει και να εγκαταστήσει γλώσσες για χρήση εκτός σύνδεσης.

Ωστόσο οι κριτικές των χρηστών αναφέρουν ότι η ακρίβεια και η διαθεσιμότητα των μεταφράσεων ποικίλει σημαντικά ανάλογα την γλώσσα στην οποία θέλουμε να μεταφράσουμε. Δηλαδή μία λιγότερο ομιλούμενη γλώσσα δεν θα έχει τη ίδια υποστήριξη με μία άλλη η οποία χρησιμοποιείτε από περισσότερες χώρες.

2.4 Σύγκριση Τεχνολογιών

Παρακάτω ακολουθεί ένα διάγραμμα με τις δυνατότητες κάθε συστήματος που ερευνήσαμε.

Όνομα	Free	Αντιγραφή Αρχικού Κειμένου	Επιλογή από Gallery	Φωτογραφία μέσω κάμερας	Offline Mode
Google Translate	✓	✗	✓	✓	✓
Abbyy TextGrabber	✗	✗	✓	✓	✓
Microsoft Translator	✓	✗	✗	✓	✗
Image Translator	✓	✓	✓	✓	✓

Πίνακας 1. Διάγραμμα σύγκρισης των συστημάτων που ερευνήσαμε και του δικού μας

⁸ <https://www.microsoft.com/en-us/translator/>

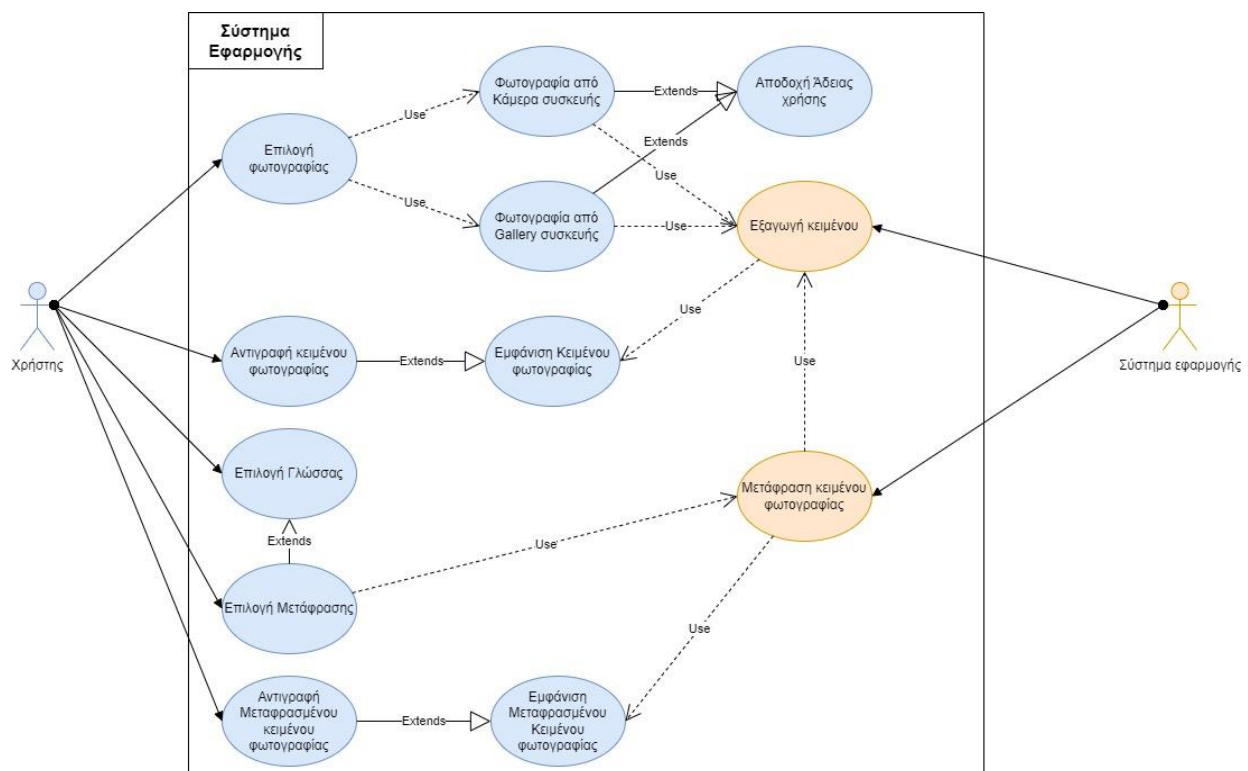
Κεφάλαιο 3^ο

3 Σχεδίαση της εφαρμογής

Σε αυτό το κεφάλαιο της διπλωματικής θα αναλύσουμε τον τρόπο με τον οποίο σχεδιάστηκε η παρούσα εφαρμογή Image Translator καθώς επίσης και ποιες ενέργειες κατείχαν πρωταγωνιστικό ρόλο για την αποδοτική και ολοκληρωμένη υλοποίηση της. Η ανάλυση των απαιτήσεων είναι ένας σύγχρονος και απαραίτητος τρόπος για να μειωθεί η πολυπλοκότητα της σχεδίασης της εφαρμογής. Για την ανάλυση θα δημιουργήσουμε ένα UML Use Case Diagram, ένα Sequence Diagram καθώς επίσης και την ανάλυση των περιπτώσεων χρήσης της εφαρμογής. Οι παρακάτω ενέργειες δημιουργήθηκαν πριν την υλοποίηση της εφαρμογής.

3.1 UML Use Case Diagram

Ένα διάγραμμα περίπτωσης χρήσης UML απεικονίζει την κύρια μορφή των απαιτήσεων ενός συστήματος λογισμικού. Οι περιπτώσεις αυτές καθορίζουν την αναμενόμενη συμπεριφορά και όχι την ακριβή μέθοδο πραγματοποίησής της. Αφού καθοριστούν, μπορούν να υποδηλώσουν τόσο την συγγραφική όσο και την οπτική αναπαράσταση των λειτουργιών της εφαρμογής. Μια βασική ιδέα της μοντελοποίησης αυτής είναι ότι μας βοηθάει να σχεδιάσουμε ένα σύστημα από την οπτική γωνία ενός τελικού χρήστη. Παρακάτω έχουμε σχεδιάσει το διάγραμμα περίπτωσης χρήσης:



Εικόνα 1: UML Διάγραμμα εφαρμογής με τις περιπτώσεις χρήσης.



Στο παραπάνω διάγραμμα φαίνεται η ροή ενεργειών που μπορεί να πραγματοποιήσει ένας χρήστης κατά την περιήγησή του στην εφαρμογή.

Πιο συγκεκριμένα στα αριστερά έχουμε τον Χρήστη Α όπου απεικονίζει το άτομο όπου χρησιμοποιεί την εφαρμογή προκειμένου να πετύχει έναν σκοπό. Στα δεξιά έχουμε τον Χρήστη Β όπου είναι το σύστημα εφαρμογής. Αυτός ο χρήστης παρέχει τις πληροφορίες και τις ενέργειες όπου κάνουν το σύστημα λειτουργικό. Οι χρήστες αυτοί αλληλοεπιδρούν με μία ροή στοιχείων ώστε να επιτύχουν ένα στόχο. Οι συνδέσεις μεταξύ των στοιχείων αφορούν την διαμοίραση πληροφορίας μεταξύ των περιπτώσεων χρήσης ή την χρονική συσχέτιση μεταξύ τους.

Οπότε, σύμφωνα με το σχήμα, το πρώτο βήμα είναι η επιλογή μεθόδου ανάκτησης φωτογραφίας από την συσκευή. Όπως έχουμε αναφέρει οι επιλογές είναι δύο, από την κάμερα ή από την Gallery της συσκευής. Προαπαιτούμενο αυτής της ενέργειας είναι η αποδοχή της άδειας χρήσης των μέσων της συσκευής από το σύστημα της εφαρμογής. Στην συνέχεια πραγματοποιείται η εξαγωγή κειμένου από το σύστημα της εφαρμογής δίνοντας την δυνατότητα στον χειριστή να αντιγράψει το εξαγόμενο κείμενο.

Στην περίπτωση όπου ο χρήστης προχωρήσει σε μετάφραση του κειμένου αυτού, απαιτείτε να επιλέξει μια από τις γλώσσες που του εμφανίζεται στην οθόνη μέσω μια λίστας (DropDown). Το σύστημα μεταφράζει το κείμενο με την χρήση του ML Kit εργαλείου εμφανίζοντας του το μεταφρασμένο κείμενο και δίνοντας του την δυνατότητα να το αντιγράψει και το χρησιμοποιήσει εκτός της εφαρμογής.

Στην επόμενη ενότητα θα αναφέρουμε αναλυτικά τις περιπτώσεις χρήσης.



3.2 Περιπτώσεις χρήσης Εφαρμογής

Τίτλος: «Εξαγωγή λέξεων φωτογραφίας και μετάφραση»

Χειριστές

1. Κύριος χειριστής – Χρήστης
2. Δευτερεύων χειριστής – Σύστημα Εφαρμογής

Περιγραφή

Η περίπτωση χρήσης αναφέρεται στην εξαγωγή του κειμένου μιας εικόνας όπου παραθέτει ο χρήστης στο Σύστημα Εφαρμογής καθώς επίσης την μετάφραση του κειμένου αυτού.

Βασική ροή

1. ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ ΦΩΤΟΓΡΑΦΙΑΣ

Το σύστημα, μετά την είσοδο του χρήστη στην εφαρμογή, τον παραπέμπει να επιλέξει φωτογραφία είτε τραβώντας φωτογραφία μέσω της εφαρμογής, είτε επιλέγοντας μία φωτογραφία από την συσκευή.

2. ΥΠΟΒΟΛΗ ΑΔΕΙΑΣ ΧΡΗΣΤΗ

Το σύστημα ζητάει την άδεια του χρήστη να χρησιμοποιεί την κάμερα του ή την άδεια του για πρόσβαση στις φωτογραφίες της συσκευής κατά την χρήση της εφαρμογής. Ο χρήστης αποδέχεται. Αν ο χρήστης έχει ήδη συγκαταθέσει, η περίπτωση χρήσης πάει κατευθείαν στο βήμα ΕΞΑΓΩΓΗ ΚΕΙΜΕΝΟΥ.

3. ΕΞΑΓΩΓΗ ΚΕΙΜΕΝΟΥ

Το σύστημα εφαρμογής εξαγει το κείμενο που βρίσκεται στην εικόνα συνδεδεμένος με τους εξυπηρετητές της google με την χρήση του ML Kit εργαλείου.

4. ΕΜΦΑΝΙΣΗ ΚΑΙ ΑΝΤΙΓΡΑΦΗ ΛΕΞΕΩΝ

Το σύστημα εμφανίζει τις λέξεις στον χρήστη σε μορφή κειμένου, δίνοντας του την επιλογή να το αντιγράψει στο πρόχειρο (clipboard) της συσκευής.

5. ΜΕΤΑΦΡΑΣΗ ΚΕΙΜΕΝΟΥ

Ο χρήστης επιλέγει προαιρετικά μια γλώσσα από τις προτεινόμενες γλώσσες του συστήματος για την μετάφραση του κειμένου. Το σύστημα εφαρμογής συνδεδεμένος με τους εξυπηρετητές της google και με την χρήση του ML Kit εργαλείου αντικαθιστά το κείμενο μεταφρασμένο.

- Η περίπτωση χρήσης τελειώνει με επιτυχία.



Εναλλακτικές ροές

- ΑΡΝΗΤΙΚΗ ΣΥΓΚΑΤ'ΑΘΕΣΗ ΤΟΥ ΧΡΗΣΤΗ

Στο βήμα ΥΠΟΒΟΛΗ ΑΔΕΙΑΣ ΧΡΗΣΤΗ της βασικής ροής, ο χρήστης αρνείται να δώσει την άδεια στην εφαρμογή να χρησιμοποιήσει τη κάμερα ή τις φωτογραφίες της συσκευής. Το σύστημα του εμφανίζει ένα αρνητικό μήνυμα και η περίπτωση χρήσης επιστρέφει στην βασική ροή στο βήμα ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ ΦΩΤΟΓΡΑΦΙΑΣ τελειώνοντας την περίπτωση χρήσης με αποτυχία.

- ΑΝΑΚΤΗΣΗ ΕΙΚΟΝΑΣ ΜΕΣΩ ΚΑΜΕΡΑΣ

Στο βήμα ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ ΦΩΤΟΓΡΑΦΙΑΣ της βασικής ροής, ο χρήστης επιλέγει την προσθήκη εικόνας μέσω κάμερας. Ο χρήστης έχει ήδη συγκαταθέσει την άδεια του στο σύστημα για την χρήση της κάμερας του και τον παραπέμπει να τραβήξει φωτογραφία το επιθυμητό κείμενο. Η περίπτωση χρήσης επιστρέφει στην βασική ροή στο βήμα ΕΞΑΓΩΓΗ ΚΕΙΜΕΝΟΥ.

- ΑΝΑΚΤΗΣΗ ΕΙΚΟΝΑΣ ΑΠΟ ΤΗΝ ΣΥΣΚΕΥΗ

Στο βήμα ΕΠΙΛΟΓΗ ΜΕΘΟΔΟΥ ΑΝΑΚΤΗΣΗΣ ΦΩΤΟΓΡΑΦΙΑΣ της βασικής ροής, ο χρήστης επιλέγει την προσθήκη εικόνας από τις φωτογραφίες που υπάρχουν στην συσκευή. Ο χρήστης έχει ήδη συγκαταθέσει την άδεια του στο σύστημα για την πρόσβαση στις φωτογραφίες και τον παραπέμπει να επιλέξει μία από αυτές μέσω της εφαρμογής. Η περίπτωση χρήσης επιστρέφει στην βασική ροή στο βήμα ΕΞΑΓΩΓΗ ΚΕΙΜΕΝΟΥ.

- ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ ΧΩΡΙΣ ΚΕΙΜΕΝΟ

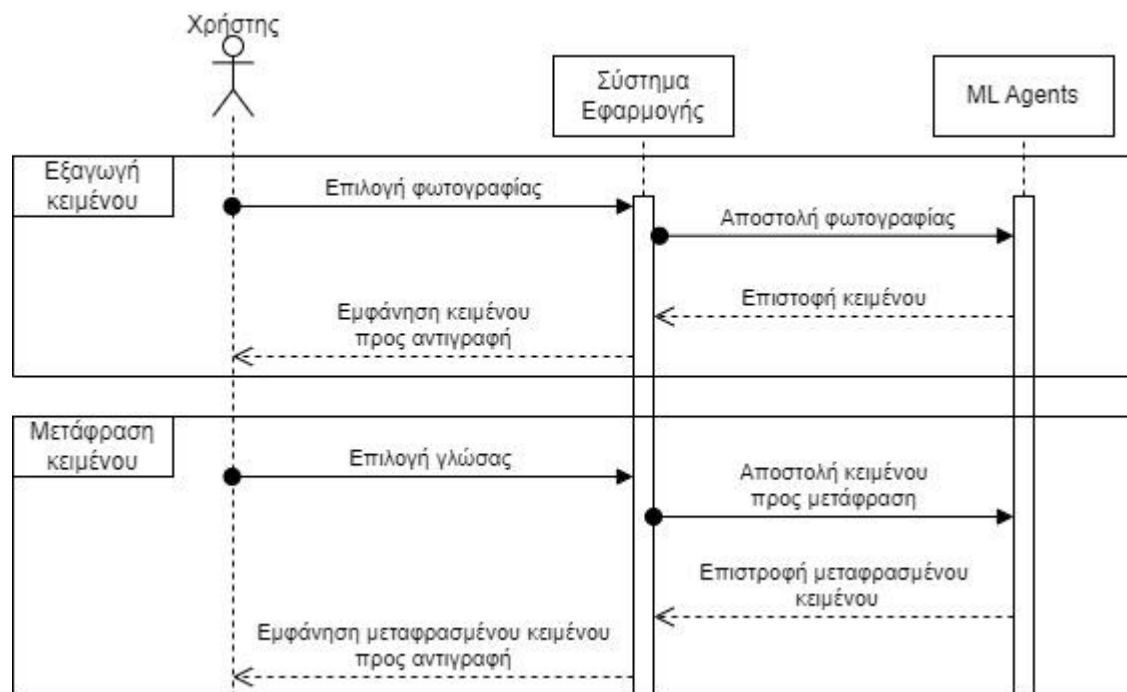
Στο βήμα ΕΞΑΓΩΓΗ ΚΕΙΜΕΝΟΥ της βασικής ροής, το σύστημα δεν εντοπίζει κανένα είδος κειμένου στην εικόνα (σε αυτή την περίπτωση εντάσσονται και κείμενα τα οποία δεν αναγνωρίζονται από το σύστημα όπως Κινέζικα, Κορεάτικα, Ελληνικά κ.ο.κ). Στην περίπτωση αυτή η εφαρμογή θα εμφανίσει ένα κενό κείμενο και δεν θα είναι δυνατή η αντιγραφή κειμένου ούτε η μετάφραση. Η περίπτωση χρήσης επιστρέφει στην ΕΜΦΑΝΙΣΗ ΚΑΙ ΑΝΤΙΓΡΑΦΗ ΛΕΞΕΩΝ (με κενά στοιχεία στην οθόνη).

3.3 Sequence Diagram

Ένα Διάγραμμα Ακολουθίας (Sequence Diagram) είναι ένα μία οπτική αναπαράσταση όπου χρησιμοποιείται για να απεικονίσει τις αλληλεπιδράσεις και κυρίως τις αλληλουχίες επικοινωνίας μεταξύ διαφόρων αντικειμένων και στοιχείων μέσα σε ένα σύστημα. Παρέχει μια χρονολογική απεικόνιση των ενεργειών και των γεγονότων όπου ανταλλάσσονται για μια χρονική περίοδο. Ένα τέτοιο διάγραμμα είναι πολύ σημαντικό για την ορθή κατανόηση και μοντελοποίηση της συμπεριφοράς ενός συστήματος. Χρησιμοποιούνται ευρέως στα στάδια σχεδιασμού και ανάλυσης ενός λογισμικού για την σαφή απεικόνιση της συνεργασίας των διάφορων στοιχείων μια εφαρμογής.

Τα διαγράμματα αυτά εστιάζουν στην χρονική σειρά των αλληλεπιδράσεων μεταξύ των περιπτώσεων χρήσης. Οι κατακόρυφοι άξονες αναπαριστούν τους χρήστες και τα βέλη τα μηνύματα τα οποία ανταλλάσσουν. Η χρονική ροή ξεκινάει από πάνω προς τα κάτω, και από αριστερά προς τα δεξιά.

Παρακάτω έχουμε σχεδιάσει το διάγραμμα ακολουθίας της εφαρμογής.



Εικόνα 2: Sequence Διάγραμμα της εφαρμογής με την ροή χρήσης.

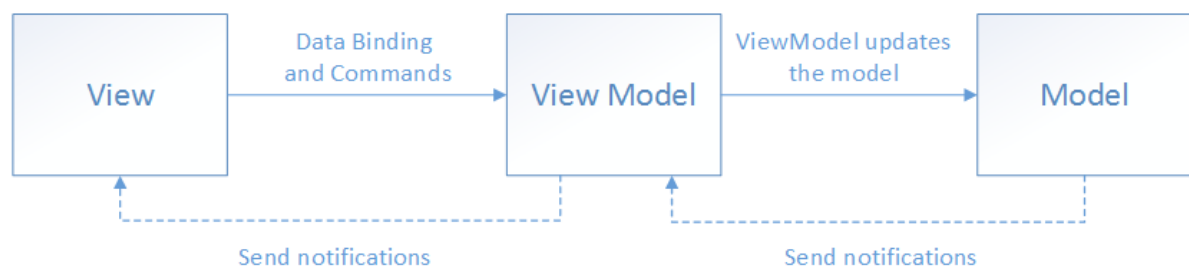
Κεφάλαιο 4^ο

4 Περιγραφή βασικών στοιχείων υλοποίησης και κώδικα

Σε αυτό το κεφάλαιο θα αναλύσουμε τις βασικές λειτουργίες και αρχιτεκτονικές της εφαρμογής σε τεχνικό επίπεδο.

4.1 MVVM Αρχιτεκτονική

Για την εργασία έχουμε χρησιμοποιήσει αρχιτεκτονική MVVM. Η αρχιτεκτονική αυτή είναι ένας δημοφιλής και αποτελεσματικός τρόπος σχεδιασμού και ανάπτυξης εφαρμογών που διαθέτουν GUI. Αυτό το Design Pattern αποτελείται από τρία κύρια στοιχεία, το Model, το View και το View Model. Κάθε στοιχείο έχει ένα συγκεκριμένο ρόλο στην εφαρμογή.



Το Model αντιπροσωπεύει τα δεδομένα και τους κανόνες της εφαρμογής. Είναι ανεξάρτητο από το View και το View Model και δεν περιέχει καμία διεπαφή με τον χρήστη. Το Model επικοινωνεί με το View Model μέσω data binding, events, ή messages.

Το View είναι το GUI της εφαρμογής. Εμφανίζει δεδομένα από το Model και επιτρέπει την διεπαφή των πληροφοριών με τον χρήστη. Το View δεν γνωρίζει το Model και το View Model και περιέχει ελάχιστη λογική όπου αφορά την οπτική συμπεριφορά των στοιχείων. Το View επικοινωνεί με το Model μέσω data binding, commands, ή events.

Το View Model είναι η γέφυρα μεταξύ του View και του Model. Κάνει expose διάφορα properties και εντολές όπου μπορεί να χρησιμοποιήσει το View για να εμφανίσει και να χειριστεί δεδομένα από το Model. Το View Model γνωρίζει τι συμβαίνει στο Model αλλά όχι στο View. Το View Model μπορεί να διαχειριστεί ενέργειες όπως επικύρωση, πλοήγηση και διάφορες λογικές εφαρμογών.

Στην εφαρμογή μας, χρησιμοποιούμε κυρίως το View Model και το View καθώς δεν έχουμε να διαχειριστούμε κάποιο ιδιαίτερο όγκο πληροφοριών.



Τα κύρια οφέλη από τη χρήση της αρχιτεκτονικής MVVM είναι:

- Επιτρέπει έναν άμεσο διαχωρισμό μεταξύ GUI και της «λογικής» της εφαρμογής, γεγονός που καθιστά την εφαρμογή πιο εύκολη στη συντήρηση και την δοκιμή.
- Επιτρέπει μία καλύτερη συνεργασία μεταξύ των προγραμματιστών καθώς ο καθένας μπορεί να εργαστεί στο δικό του στοιχείο, ανεξάρτητα και ταυτόχρονα.
- Δημιουργείται ένα code base το οποίο μπορεί να επαναχρησιμοποιηθεί πολύ πιο εύκολα καθώς το ίδιο View Model μπορεί να χρησιμοποιηθεί με διαφορετικά Views ή το ίδιο View μπορεί να χρησιμοποιηθεί με διαφορετικά View Models.

4.2 Αναφορά βασικότερων σημείων κώδικα

4.2.1 Gradle Dependencies

Για την χρήση όλων των προαναφερόμενων εργαλείων όπως το ml-kit καθώς επίσης και την χρήση των UI components, χρειάζεται να ορίσουμε κάποια dependencies στο project μέσω του Gradle αρχείου. Παρακάτω αναφέρονται τα Gradle dependencies που χρησιμοποιήσαμε.

```
//UI
implementation("androidx.activity:activity-compose:1.8.0")
implementation("androidx.lifecycle:lifecycle-viewmodel-compose:2.7.0-alpha02")
implementation(platform("androidx.compose:compose-bom:2023.03.00"))
implementation("androidx.compose.ui:ui")
implementation("androidx.compose.ui:ui-graphics")
implementation("androidx.compose.ui:ui-tooling-preview")
androidTestImplementation(platform("androidx.compose:compose-bom:2023.03.00"))
androidTestImplementation("androidx.compose.ui:ui-test-junit4")
debugImplementation("androidx.compose.ui:ui-tooling")
debugImplementation("androidx.compose.ui:ui-test-manifest")
implementation("androidx.compose.material3:material3")
//ML-KIT To recognize Latin script
implementation("com.google.mlkit:text-recognition:16.0.0")
//ML-KIT To recognize Chinese script
implementation("com.google.mlkit:text-recognition-chinese:16.0.0")
//ML-KIT To recognize Devanagari script
implementation("com.google.mlkit:text-recognition-devanagari:16.0.0")
//ML-KIT To recognize Japanese script
implementation("com.google.mlkit:text-recognition-japanese:16.0.0")
//ML-KIT To recognize Korean script
implementation("com.google.mlkit:text-recognition-korean:16.0.0")
implementation("com.google.mlkit:language-id:17.0.4")
implementation("com.google.mlkit:translate:17.0.1")
implementation("androidx.navigation:navigation-compose:2.7.4")
implementation("androidx.datastore:datastore-preferences:1.0.0")
implementation("io.coil-kt:coil-compose:1.3.2")
implementation("androidx.camera:camera-lifecycle:1.2.3")
implementation("androidx.camera:camera-core:1.2.3")
implementation("androidx.camera:camera-camera2:1.2.3")
implementation("androidx.camera:camera-view:1.2.3")
implementation("com.google.ar:core:1.39.0")
```

Εικόνα 3: Gradle Dependencies

4.2.2 ChooseOrTakePhotoView

Το αρχικό View της εφαρμογής όπου εμφανίζει τα κουμπιά “Take Photo” και “Pick Image From Gallery”.

```
@Composable
fun ChooseOrTakePhotoView(
    choosePhotoFromLibraryAction: () -> Unit,
    takePhotoAction: () -> Unit
) {
    Row(
        horizontalArrangement = Arrangement.spacedBy(10.dp),
        verticalAlignment = Alignment.CenterVertically,
        modifier = Modifier
            .fillMaxWidth()
            .height(IntrinsicSize.Max)
            .padding(top = 16.dp)
            .padding(horizontal = 10.dp)
    ) { this: RowScope
        Button(
            modifier = Modifier.fillMaxHeight(),
            onClick = { choosePhotoFromLibraryAction() },
        ) { this: RowScope
            Text(
                text = "Pick Image \nFrom Gallery",
                textAlign = TextAlign.Center
            )
        }

        Spacer(modifier = Modifier.weight(1f))

        Button(
            modifier = Modifier.fillMaxHeight(),
            onClick = { takePhotoAction() },
        ) { this: RowScope
            Text(
                text = "Take Photo",
                textAlign = TextAlign.Center
            )
        }
    }
}
```

Εικόνα 4: ChooseOrTakePhotoView

Όταν πατηθούν εκτελούν δύο διαφορετικά Actions.

4.2.3 CameraView

- **Take Photo Function**

Παρακάτω αναφέρεται ο κώδικας της συνάρτησης Take Photo όπου καλείται όποτε ο χρήστης τραβήξει φωτογραφία μέσω της κάμερας του. Η συνάρτηση θα καλεστεί σαν Action από το UI όταν ο χρήστης πατήσει το κουμπί “Take Photo”

```
private fun takePhoto(
    context: Context,
    filenameFormat: String,
    imageCapture: ImageCapture,
    outputDirectory: File,
    executor: Executor,
    onImageCaptured: (Bitmap) -> Unit,
    onError: (ImageCaptureException) -> Unit
) {

    val photoFile = File(
        outputDirectory,
        child: SimpleDateFormat(filenameFormat, Locale.US).format(System.currentTimeMillis()) + ".jpg"
    )

    val outputOptions = ImageCapture.OutputFileOptions.Builder(photoFile).build()

    imageCapture.takePicture(outputOptions, executor, object: ImageCapture.OnImageSavedCallback {
        override fun onError(exception: ImageCaptureException) {
            Log.e( tag: "kilo", msg: "Take photo error:", exception)
            onError(exception)
        }

        override fun onImageSaved(outputFileResults: ImageCapture.OutputFileResults) {
            val savedUri = Uri.fromFile(photoFile)
            val bitmapResult = uriToBitmap(context = context, uri = savedUri)
            if(bitmapResult != null) {
                onImageCaptured(bitmapResult)
            }
        }
    })
}
```

Εικόνα 5: Take Photo Function

4.2.4 CropImageView

Παρακάτω αναφέρονται τα UI components τα οποία περιέχουν την επεξεργασία της εικόνας μετά την επιλογή. Αυτό το view θα εμφανιστεί σε οποιαδήποτε από τις δύο περιπτώσεις ανάκτησης φωτογραφίας. Η φωτογραφία η οποία θα επιλεγεί, γίνεται convert σε bitmap.

```
@Composable
fun CropImageView(
    bitmap: Bitmap,
    bitmapResult: (Bitmap) -> Unit
) {
    var bitmap by remember { mutableStateOf(bitmap) }
    val state = rememberCropifyState()

    Column(
        modifier = Modifier
            .fillMaxSize()
            .fillMaxWidth()
            .background(Color.Black)
    ) { this: ColumnScope
        Cropify(
            bitmap = bitmap.asImageBitmap(),
            state = state,
            onImageCropped = { it: ImageBitmap
                bitmapResult(it.asAndroidBitmap())
            },
            modifier = Modifier
                .fillMaxWidth()
                .weight(4f),
        )
    }
    Row(
```

Εικόνα 6: CropImageView Bitmap

Παρακάτω αναγράφονται οι συναρτήσεις `uriToBitmap` και η `rotateBitmap` οι οποίες καλούνται από εξωτερικά Views.

```
// Function to convert a Uri to a Bitmap
fun uriToBitmap(context: Context, uri: Uri): Bitmap? {
    return try {
        val contentResolver = context.contentResolver
        val inputStream: InputStream? = contentResolver.openInputStream(uri)
        BitmapFactory.decodeStream(inputStream)
    } catch (e: Exception) {
        e.printStackTrace()
        null
    }
}

fun rotateBitmap(source: Bitmap, degrees: Float): Bitmap {
    val matrix = Matrix()
    matrix.postRotate(degrees)
    return Bitmap.createBitmap(source, 0, 0, source.width, source.height, matrix, filter: true)
}
```

Εικόνα 7: `uriToBitmap` και `rotateBitmap` functions

4.2.5 AppContentView

Αυτό το View περιέχει τα components πριν την μετάφραση και μετά το extraction των λέξεων. Παρακάτω αναγράφονται τα UI Components όποτε περιέχουν το extracted text, το κουμπί translate και την επιλογή της γλώσσας μετάφρασης.

```
ChooseOrTakePhotoView(  
    choosePhotoFromLibraryAction = { galleryLauncher.launch( input: "image/*" ) },  
    takePhotoAction = {  
        takePhotoAction(state = cameraPermissionState) { it: Boolean  
            shouldShowCamera = it  
        }  
    },  
)  
if (bitmap != null) {  
    Image(  
        bitmap = bitmap!!.asImageBitmap(),  
        contentScale = ContentScale.FillBounds,  
        contentDescription = null,  
        modifier = Modifier  
            .padding(16.dp, 8.dp)  
            .size(150.dp)  
            .clip(RoundedCornerShape(20.dp))  
    )  
    when (mlKitStatus) {  
        MLKitStatus.IN_PROGRESS -> CircularProgressIndicator()  
        else -> {  
            ExtractedTextView(  
                text = extractedTextFromMLKitImage,  
                mlKitStatus = mlKitStatus,  
                failedText = "Failed to extract text."  
            )  
            if (languageFromTextFromML != null && languageRecognitionStatus == MLKitStatus.SUCCESS) {  
                LanguagePicker(  
                    currentLanguageOfText = languageFromTextFromML!!.displayLocale,  
                    languageList = viewModel.languageList,  
                    selectedLanguageToTranslateTo = { it: String  
                        viewModel.setSelectedLanguageToTranslateTo(it)  
                    }  
                )  
                Row { this: RowScope  
                    Button(  
                        enabled = (selectedLanguageToTranslateTo != null),  
                        onClick = { viewModel.translateText() } ) { this: RowScope  
                            Text(text = "Translate")  
                        }  
                }  
                when (translationStatus) {  
                    MLKitStatus.IN_PROGRESS -> CircularProgressIndicator()  
                    else -> {  
                        if (translatedText.isNotEmpty() && translationStatus == MLKitStatus.SUCCESS) {  
                            ExtractedTextView(  
                                text = translatedText,  
                                mlKitStatus = translationStatus,  
                                failedText = "Failed to translate"  
                            )  
                        }  
                    }  
                }  
            } else if (languageRecognitionStatus == MLKitStatus.FAILED) {  
                Text(text = "Cannot Identify Language")  
            }  
        }  
    }  
}
```

Εικόνα 8: AppContentView UI Components

4.2.6 AppCompatActivity

Στο AppCompatActivity εκτελείται το μεγαλύτερο μέρος της λογικής της εφαρμογής.

- **inputImageProcessingWithMLKIT function**

Η συνάρτηση αυτή παίρνει ένα input image και με την χρήση του ML Kit κάνει extract τις λέξεις της φωτογραφίας και αναγνωρίζει την γλώσσα στην οποία είναι γραμμένο το κείμενο.

```
init {
    getAllTranslateLanguages()
}

fun inputImageProcessingWithMLKIT(inputImage: InputImage) {
    resetValues()

    this._textImageProcessingStatus.value = IN_PROGRESS
    recognizer.process(inputImage)
        .addOnSuccessListener { visionText ->
            processResultText(visionText = visionText)
        }
        .addOnFailureListener { e ->
            this._textImageProcessingStatus.value = FAILED
        }
}

private fun processResultText(visionText: Text) {
    _textFromML.value = visionText.text
    if (visionText.text.isEmpty()) {
        this._textImageProcessingStatus.value = FAILED
    } else {
        this._textImageProcessingStatus.value = SUCCESS
    }
    identifyTextLanguage(text = visionText.text)
}

private fun identifyTextLanguage(text: String) {
    val languageIdentifier = LanguageIdentification.getClient()
    this._languageIdentificationProcessingStatus.value = IN_PROGRESS
    languageIdentifier.identifyLanguage(text)
        .addOnSuccessListener { languageCode ->
            if (languageCode == "und") {
                this._languageIdentificationProcessingStatus.value = FAILED
            } else {
                Log.i(tag: "LanguageIdentification", msg: "Language: $languageCode")
                val currentLocale = Locale(languageCode)
                this._languageFromTextFromML.value = LanguageModel(locale = currentLocale, displayLocale = currentLocale.displayLanguage)
                this._languageIdentificationProcessingStatus.value = SUCCESS
            }
        }
        .addOnFailureListener { e: Exception
            this._languageIdentificationProcessingStatus.value = FAILED
        }
}

private fun getAllTranslateLanguages() {
    for (x in TranslateLanguage.getAllLanguages()) {
        val locale = Locale(x)
        _languageList.add(LanguageModel(locale = locale, displayLocale = locale.displayLanguage))
        Log.i(tag: "LanguageTranslatorOptionsBuilder", locale.displayLanguage)
    }
}
```

Εικόνα 9: Συνάρτηση για το extraction των λέξεων.



- **translateText function**

Η συνάρτηση αυτή κάνει translate το κείμενο της εικόνας με την χρήση του ML Kit λαμβάνοντας την γλώσσα που έχει αναγνωριστεί από το σύστημα και την γλώσσα στην οποία επέλεξε ο χρήστης. Στην συνέχεια αν δεν υπάρχει κατεβασμένο language pack της επιλεγμένης γλώσσας, πάει και το κατεβάζει μόνο αν η συσκευή είναι συνδεδεμένη στο wifi (καθώς το language pack είναι πολλά MB).

```
fun translateText(isFirstTime: Boolean = true) {  
    _textImageTranslationStatus.value = IN_PROGRESS  
    val options = TranslatorOptions.Builder()  
        .setSourceLanguage(_languageFromTextFromML.value?.locale.toString())  
        .setTargetLanguage(_selectedLanguageToTranslateTo.value?.locale.toString())  
        .build()  
    val translator = Translation.getClient(options)  
    val conditions = DownloadConditions.Builder()  
        .requireWifi()  
        .build()  
    translator.downloadModelIfNeeded(conditions)  
        .addOnSuccessListener { it: Void! }  
        {  
            translator.translate(_textFromML.value)  
                .addOnSuccessListener { translatedText ->  
                    this._translatedText.value = translatedText.toString()  
                    _textImageTranslationStatus.value = SUCCESS  
                }  
                .addOnFailureListener { exception ->  
                    _textImageTranslationStatus.value = FAILED  
                }  
        }  
        .addOnFailureListener { exception ->  
            if (isFirstTime) {  
                translateText(isFirstTime = false)  
            } else {  
                _textImageTranslationStatus.value = FAILED  
            }  
        }  
    }  
}
```

Εικόνα 10: Συνάρτηση για την μετάφραση της γλώσσας.



Κεφάλαιο 5^ο

5 Εγχειρίδιο Χρήσης Εφαρμογής

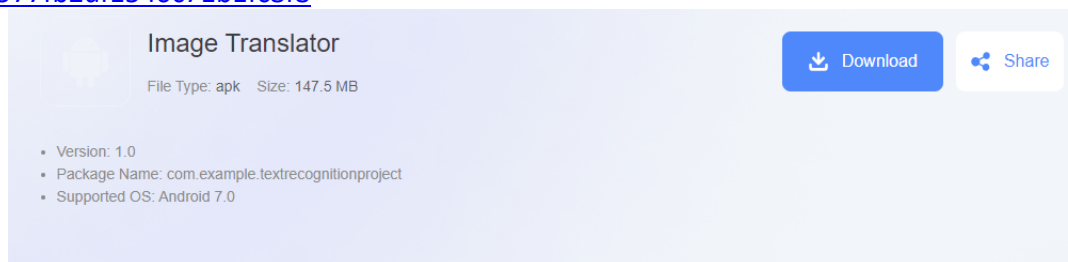
Σε αυτό το κεφάλαιο θα αναπαραστήσουμε λεπτομερώς τις λειτουργίες της εφαρμογής. Θα δείξουμε πως ο χρήστης μπορεί να χρησιμοποιήσει και να περιηγηθεί στο περιβάλλον της υλοποιημένης πλέον εφαρμογής γρήγορα και με άνεση.

5.1 Εγκατάσταση εφαρμογής

Η εφαρμογή έχει σχεδιαστεί για να λειτουργεί σε συσκευές που τρέχουν Android με έκδοση λειτουργικού συστήματος 8.0 ή νεότερη. Για αυτές τις συσκευές απαιτητέ από τον χρήστη να ενεργοποιήσει την εγκατάσταση εφαρμογών από άγνωστες πηγές. Αυτό είναι απαραίτητο επειδή η εφαρμογή δεν είναι διαθέσιμη στο επίσημο Google Play Store, αλλά σε διαφορετικό πλατφόρμα που φιλοξενεί εναλλακτικές εφαρμογές. Ο χρήστης επίσης πρέπει να έχει σταθερή σύνδεση στο διαδίκτυο και αρκετό αποθηκευτικό χώρο για να κατέβει και να εγκαταστήσει την εφαρμογή. Η εφαρμογή θα ζητήσει πρόσβαση σε ορισμένα δικαιώματα όπως πρόσβαση στη κάμερα και στις φωτογραφίες της συσκευής. Μπορεί να επιλέξει να αποδεχτεί ή να αρνηθεί τις άδειες ανάλογα με την κρίση του. Σε περίπτωση που αρνηθεί, βασικές λειτουργίες της εφαρμογής όπως η ανάκτηση εικόνας μέσω κάμερα.

Αφού πληρούνται όλες οι παραπάνω προϋποθέσεις, πρέπει να κατεβάσει και να εγκαταστήσει το αρχείο .apk της εφαρμογής μέσω του παρακάτω link:

<https://upload.app/download/image-translator/com.example.textrecognitionproject/7dbdf144056286050b41064c2ae66cf66b857b3b30977fb2df1540c71b1fc8f8>



About Image Translator

Free and fast download Image Translator APK/XAPK V 1.0 without region limit.

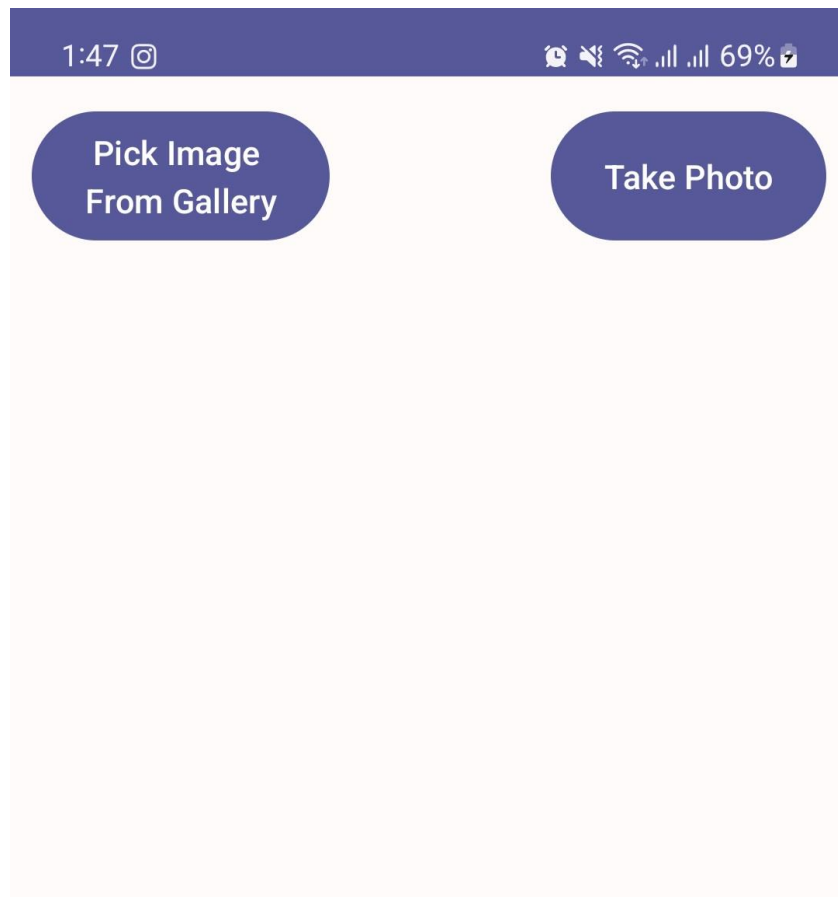
Package Name: com.example.textrecognitionproject	Version: 1.0(1)
Supported Architecture: arm64-v8a,armeabi-v7a,x86,x86_64	Supported OS: Android 7.0
Screen DPI: small,normal,large,xlarge	Signature: 018ad8a1477f5e9bb62e8868e80716e5b4e3f984
File SHA: 7dbdf144056286050b41064c2ae66cf66b857b3b30977fb2d...	Permissions: 4

Μετά την εγκατάσταση, το εικονίδιο της εφαρμογής θα εμφανιστεί στην λίστα εφαρμογών της συσκευής.



5.2 Οδηγίες Χρήσης

Πατάμε το εικονίδιο της εφαρμογής και εμφανίζεται μπροστά μας η αρχική σελίδα της εφαρμογής.

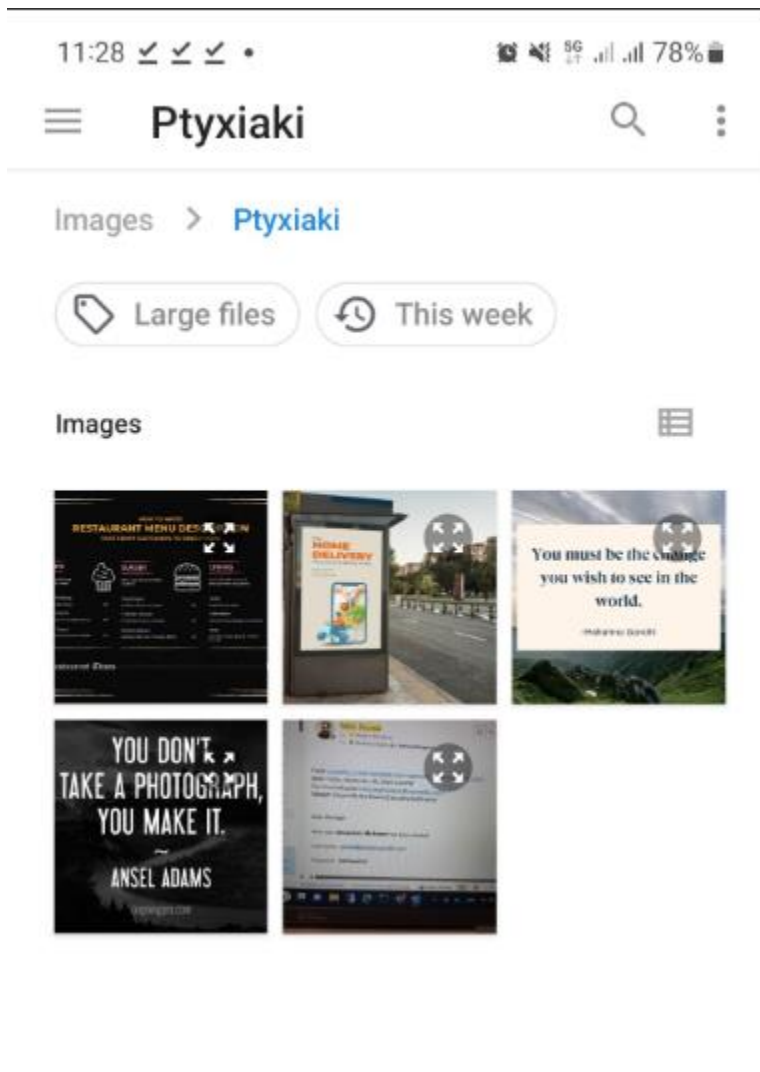


Εικόνα 11: Αρχικό παράθυρο εφαρμογής κατά την εκκίνηση.

Εκεί ο χρήστης μπορεί να επιλέξει μία από τις δύο μεθόδους ανάκτησης φωτογραφίας.

5.2.1 Pick Image From Gallery

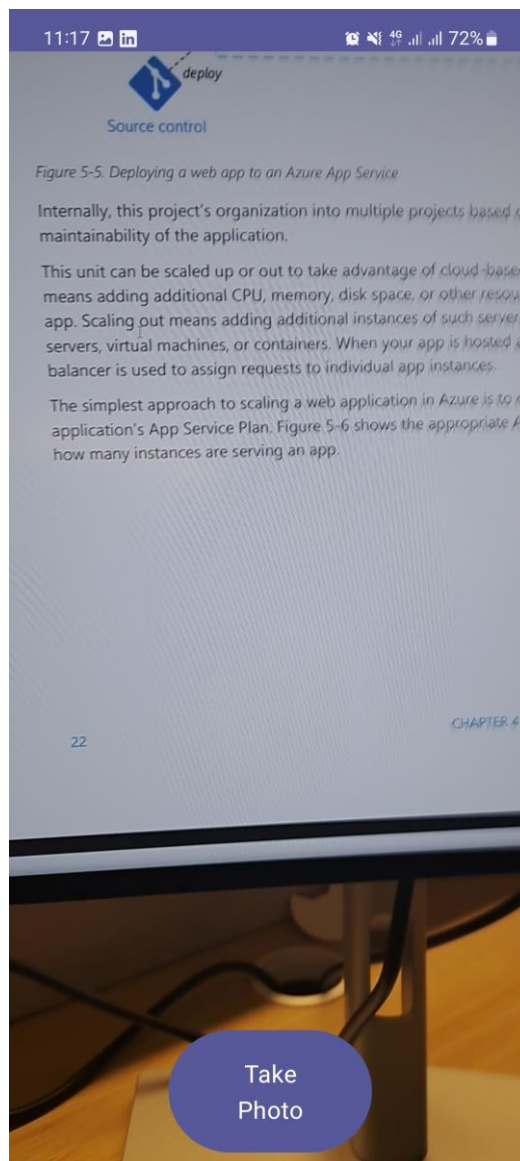
Όταν πατήσουμε το κουμπί “Pick Image From Gallery”, κατά την πρώτη χρήση, η συσκευή θα ζητήσει να δώσουμε άδεια στην εφαρμογή για πρόσβαση στις φωτογραφίες. Μετά την αποδοχή, η εφαρμογή θα ανοίξει την Gallery της συσκευής. Εκεί ο χρήστης μπορεί να επιλέξει την φωτογραφία που επιθυμεί να σκαναριστεί και να μεταφραστεί.



Εικόνα 12: Gallery κινητού για επιλογή φωτογραφίας.

5.2.2 Take Photo

Όταν πατήσουμε το κουμπί “Take photo”, κατά την πρώτη χρήση, η συσκευή θα ζητήσει να δώσουμε άδεια στην εφαρμογή για πρόσβαση στην κάμερα. Μετά την αποδοχή η εφαρμογή θα ανοίξει την κάμερα και εκεί ο χρήστης θα μπορέσει να τραβήξει σε πραγματικό χρόνο μία φωτογραφία προκειμένου να επεξεργαστεί από το σύστημα.



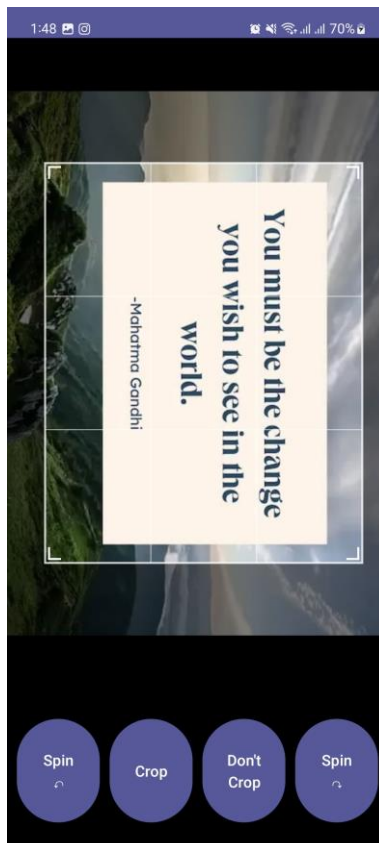
Εικόνα 13: Σελίδα όπου τραβάμε την φωτογραφία.

5.2.3 Εξαγωγή κειμένου

Μετά την επιλογή ή την αποτύπωση της εικόνας, η εφαρμογή θα ανοίξει την φωτογραφία και θα δώσει στον χρήστη τέσσερις επιλογές επεξεργασίας.



Εικόνα 14: Παράθυρο επεξεργασίας εικόνας.



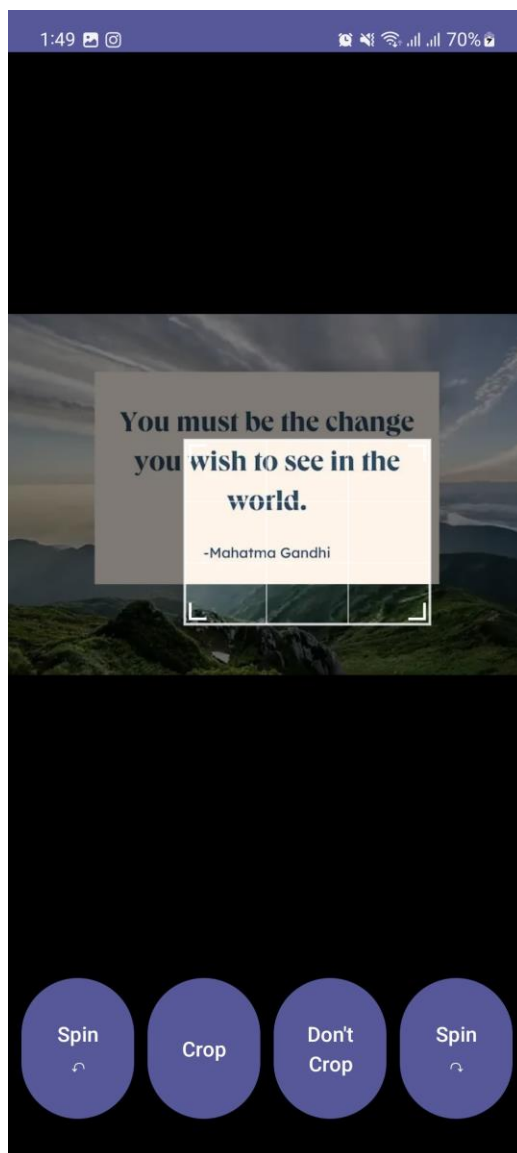
Εικόνα 15: Δεξιά περιστροφή εικόνας.



Εικόνα 16: Αριστερή περιστροφή εικόνας.

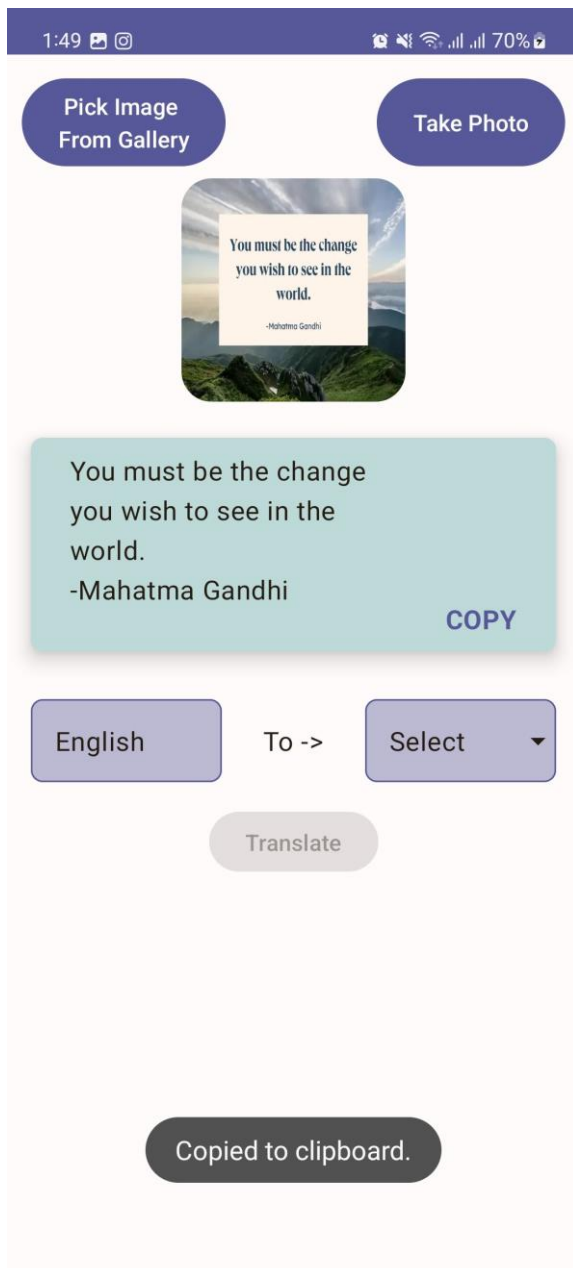
Οι δύο επιλογές είναι να περιστρέψει την εκάστοτε εικόνα αριστερόστροφα πατώντας το κουμπί “ Spin ↶” και δεξιόστροφα πατώντας το κουμπί “ Spin ↷”. Σε περίπτωση όπου η εικόνα δεν έχει τις λέξεις σε οριζόντια κανονική μορφή, ο χρήστης θα μπορεί να την περιστρέψει προκειμένου να αναγνωριστεί το κείμενο από το σύστημα.

Η τρίτη επιλογή είναι να κάνει αποκοπή σε κάποιο σημείο της εικόνας προκειμένου να απομονώσει τις λέξεις όπου θέλει να μεταφράσει σε περίπτωση που δεν χρειάζεται όλο το κείμενο της εικόνας πατώντας το κουμπί “Crop” που φαίνεται στην εικόνα. Τέλος έχει την επιλογή να προχωρήσει στο επόμενο στάδιο με την εικόνα στην αρχική της μορφή, όπως υπάρχει στη συσκευή πατώντας το κουμπί “Don’t Crop”.

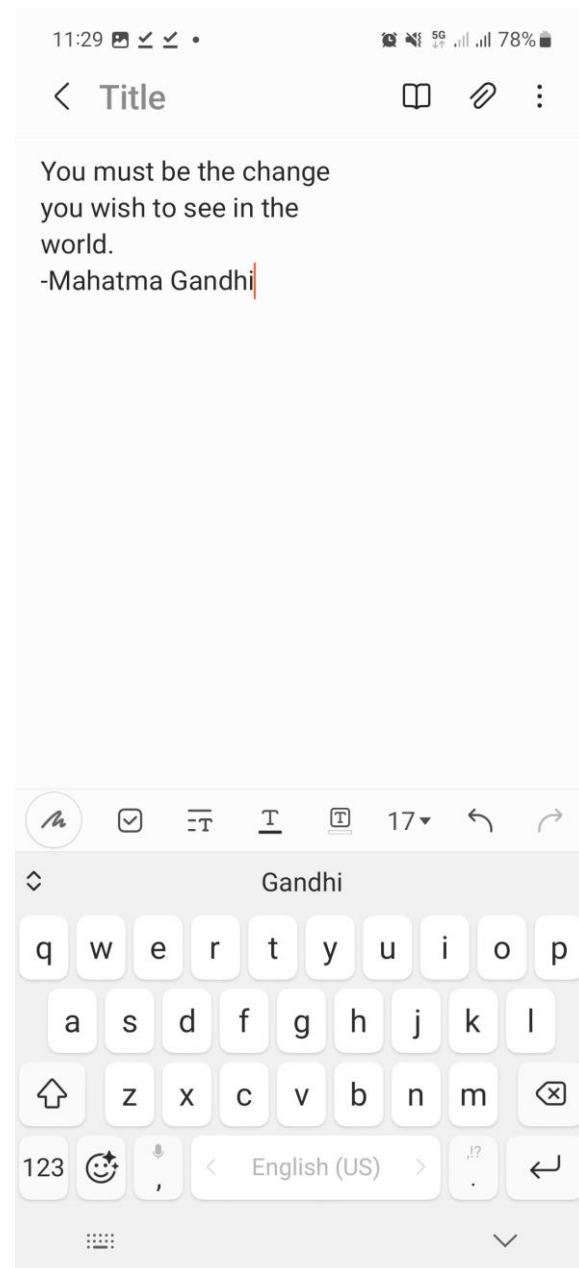


Εικόνα 17: Περικοπή εικόνας.

Στο επόμενο στάδιο το σύστημα αναγνωρίζει το κείμενο της φωτογραφίας και δίνει στον χρήστη την επιλογή να αντιγράψει το κείμενο πατώντας το κουμπί “Copy” και να το χρησιμοποιήσει εκτός της εφαρμογής. Επιπλέον η εφαρμογή έχει την δυνατότητα να ανιχνεύσει και την γλώσσα στην οποία είναι γραμμένο το κείμενο της εικόνας. **Σε αυτό το στάδιο υπενθυμίζουμε ότι το σύστημα μπορεί να ανιχνεύσει μόνο Λατινικούς Χαρακτήρες (και όχι για παράδειγμα Ελληνικούς).**



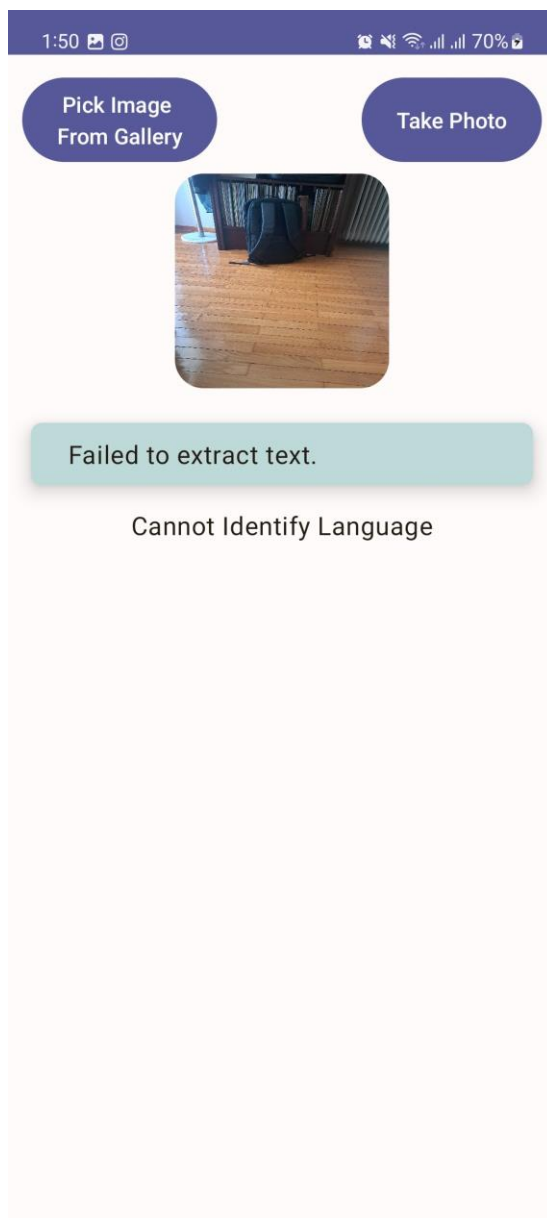
Εικόνα 18: Αντιγραφή αρχικού κειμένου.



Εικόνα 19: Επικόλληση αρχικού κειμένου.

Στην εικόνα 12 βλέπουμε ότι άμα δεν εντοπιστεί κείμενο, ή αν υπάρχει κείμενο με γράμματα μη λατινικών χαρακτήρων, εμφανίζει αντίστοιχο μήνυμα στην οθόνη της συσκευής αποτρέποντας τον χρήστη να προχωρήσει στην επόμενη λειτουργία, όπου είναι η μετάφραση.

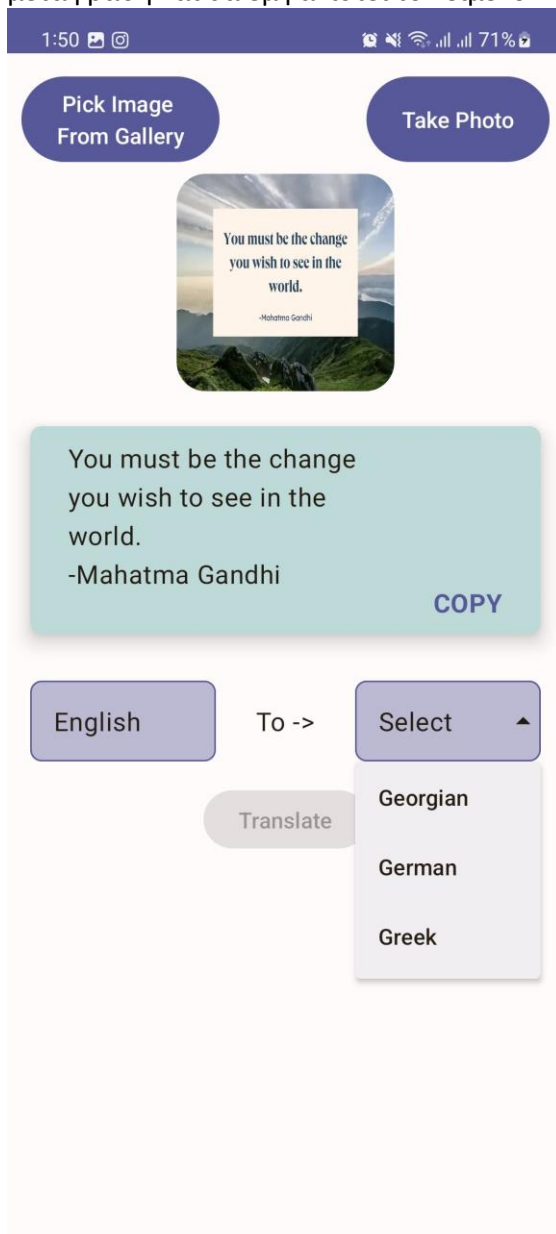
Προκειμένου να ξαναρχίσει την διαδικασία από την αρχή, αρκεί να πατήσει ένα από τα αρχικά κουμπιά προκειμένου να φορτωθεί καινούργια φωτογραφία.



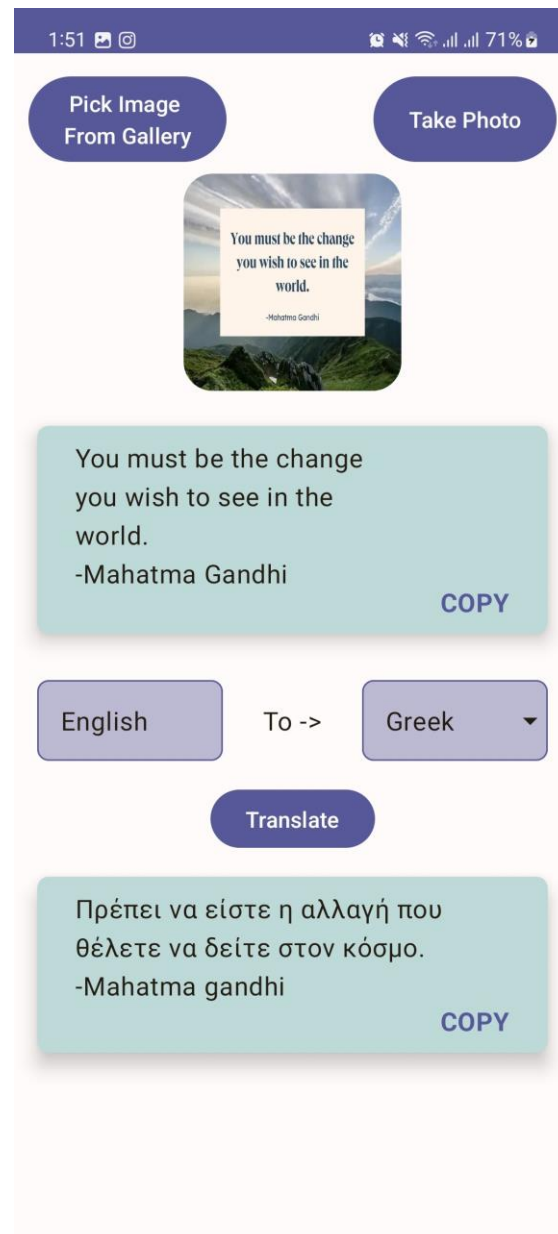
Εικόνα 20: Αρνητική αναγνώριση χαρακτήρων από την εικόνα.

5.2.4 Μετάφραση εξαγόμενου κειμένου

Όταν γίνει η εξαγωγή κειμένου, θα αναγνωριστεί η γλώσσα από το σύστημα. Στην συνέχεια, ο χρήστης πρέπει να επιλέξει από το DropDown “Select” την γλώσσα στην οποία θέλει να μεταφραστεί το κείμενο. Όπως έχουμε αναφέρει, η ImageTranslator μπορεί να μεταφράσει το κείμενο χωρίς να είναι απαραίτητη η σύνδεση της συσκευής στο διαδίκτυο. Παρόλα αυτά, όταν χρησιμοποιήσουμε την εκάστοτε γλώσσα για πρώτη φορά, θα πρέπει η συσκευή να είναι συνδεδεμένη σε Wi-Fi. Η εφαρμογή θα πρέπει να κατεβάσει στην συσκευή το γλωσσικό πακέτο, το οποίο αποθηκεύει τοπικά. Από εκεί και πέρα, για την συγκεκριμένη γλώσσα, δε χρειάζεται να ξανά υπάρξει σύνδεση. Όταν πατήσει το κουμπί “Translate” θα γίνει η μετάφραση και θα εμφανιστεί το κείμενο

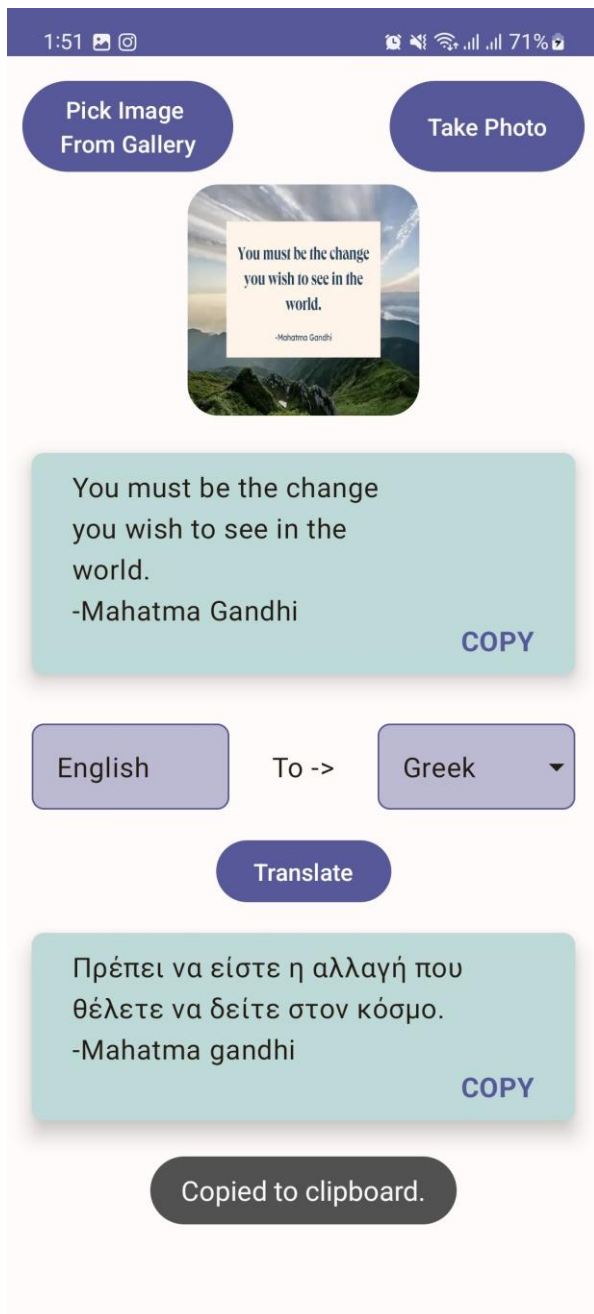


Εικόνα 21: Επιλογή γλώσσας προς μετάφραση.

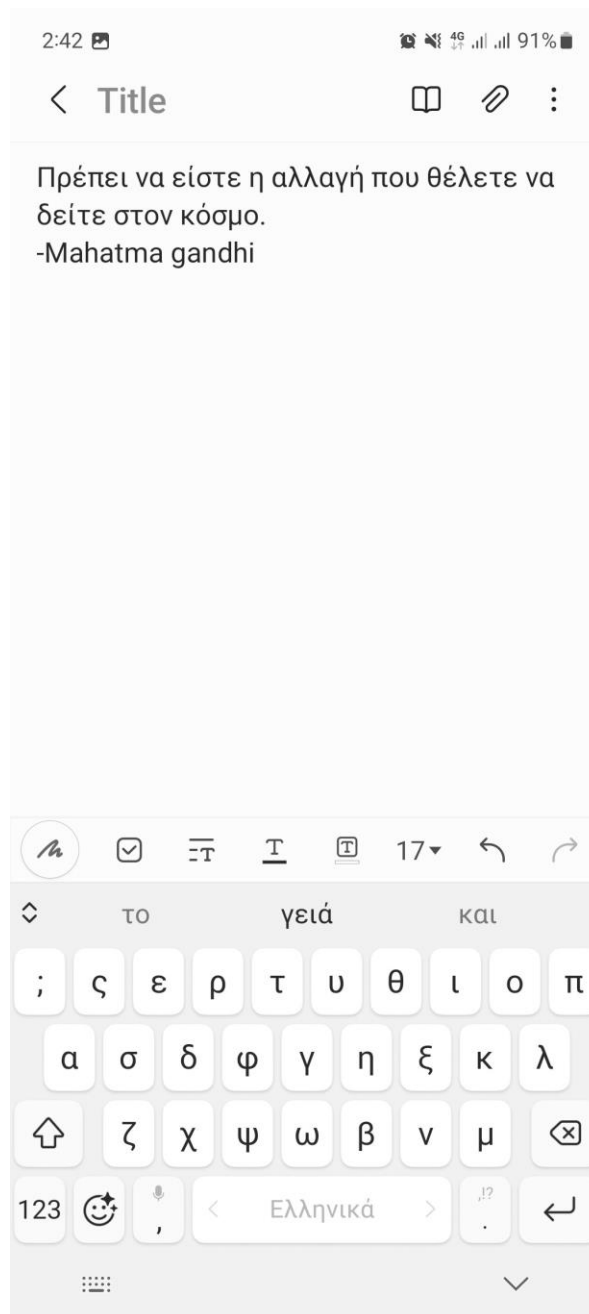


Εικόνα 22: Μετάφραση κειμένου.

Σε αυτό το σημείο μπορεί να αντιγραφεί στο clipboard το μεταφρασμένο κείμενο πατώντας το κουμπί “Copy” το οποίο βρίσκεται στην κάτω δεξιά γωνία του μεταφρασμένου κειμένου. Αν αλλάξουμε γλώσσα και πατήσουμε ξανά το κουμπί “Translate” θα γίνει μετάφραση του αρχικού κειμένου στην άλλη γλώσσα που επιλέξαμε.



Εικόνα 23: Αντιγραφή μεταφρασμένου κειμένου.



Εικόνα 24: Επικόλληση μεταφρασμένου κειμένου.

Όπως αναφέραμε και προηγούμενος, η γλώσσα του κειμένου αναγνωρίζεται αυτόματα από το σύστημα χωρίς να χρειάζεται ο χρήστης να την ενσωματώσει χειροκίνητα.



Εικόνα 25: Αναγνώριση Ισπανικής Γλώσσας



Εικόνα 26: Μετάφραση Ισπανικών σε Αγγλικά.



Το κείμενο μπορεί να μεταφραστεί σε 42 γλώσσες. :

- Αφρικάνικα
- Αλβανικά
- Αραβικά
- Λευκορωσικά
- Βουλγάρικα
- Καταλανικά
- Κινέζικα
- Κροατικά
- Τσέχικα
- Δανικά
- Αγγλικά
- Εσθονικά
- Φινλανδικά
- Γαλλικά
- Γεωργιανά
- Γερμανικά
- **Ελληνικά**
- Ινδικά
- Ουγγρικά
- Ισλανδικά
- Ινδονησιακά
- Ιρλανδικά
- Ιταλικά
- Γιαπωνέζικα
- Κορεάτικα
- Λιθουανικά
- Σκοπιανά
- Λετονικά
- Μαλαισιανά
- Νορβηγικά
- Περσικά
- Πολωνικά
- Πορτογαλικά
- Ρουμάνικα
- Ρωσικά
- Σλοβακικά
- Σλοβένικα
- Ισπανικά
- Σουηδικά
- Ταϊλανδικά
- Τουρκικά
- Ουκρανικά



Κεφάλαιο 6^ο

6 Συμπεράσματα και προτάσεις για μελλοντικές επεκτάσεις

Συμπερασματικά, η παρούσα πτυχιακή εργασία διερεύνησε την ανάπτυξη και την υλοποίηση μιας Android Εφαρμογής όπου έχει σχεδιαστεί για να αξιοποιεί το ML Toolkit της Google με σκοπό την εξαγωγή και την μετάφραση κειμένων σε εικόνες. Η εφαρμογή παρέχει επίσης την επιλογή προσωρινής αποθήκευσης του μεταφρασμένου κειμένου ή κοινής χρήσης της με άλλες εφαρμογές.

Η εφαρμογή δοκιμάστηκε σε διάφορες εικόνες με διαφορετικές γλώσσες, γραμματοσειρές, μεγέθη, προσανατολισμούς και φόντο. Τα αποτελέσματα έδειξαν ότι η εφαρμογή ήταν σε θέση να αναγνωρίσει και να μεταφράσει το κείμενο στις περισσότερες περιπτώσεις, με ορισμένες εξαιρέσεις όπου το κείμενο ήταν πολύ μικρό, θολό, παραμορφωμένο ή μερικώς αποφραγμένο. Η ακρίβεια και η ταχύτητα της μετάφρασης εξαρτιόταν από την ποιότητα της εικόνας και την πολυπλοκότητα του κειμένου.

Επιπλέον η εφαρμογή έδειξε τη δυνατότητα χρήσης μηχανικής μάθησης για μετάφραση κειμένου εικόνας, η οποία μπορεί να είναι χρήσιμη για διάφορους σκοπούς όπως η εκπαίδευση, ο τουρισμός, η επικοινωνία και η προσβασιμότητα. Η εφαρμογή έδειξε επίσης τις ευκολίες χρήσης του ML kit της Google, το οποίο παρέχει έτοιμα προς χρήση API για κοινές εργασίες μηχανικής εκμάθησης.

Μερικοί πιθανοί τρόποι βελτίωσης και επέκτασης της λειτουργικότητας της εφαρμογής είναι:

- Προσθήκη δυνατότητας εξόδου ομιλίας που μπορεί να διαβάσει δυνατά το μεταφρασμένο κείμενο χρησιμοποιώντας τεχνολογία μετατροπής κειμένου σε ομιλία. Αυτό μπορεί να γίνει χρησιμοποιώντας τα API επεξεργασίας φυσικής γλώσσας του ML Kit, τα οποία παρέχουν δυνατότητες σύνθεσης ομιλίας.
- Υποστήριξη περισσότερων γλωσσών και σεναρίων με την ενσωμάτωση άλλων μεταφραστικών υπηρεσιών ή API που μπορούν να χειριστούν γλώσσες που δεν υποστηρίζονται από το kit ML της Google. Αυτό μπορεί να γίνει χρησιμοποιώντας βιβλιοθήκες τρίτων ή υπηρεσίες web που προσφέρουν δυνατότητες μετάφρασης σε πολλές γλώσσες.
- Αναγνώριση Ελληνικών χαρακτήρων στην εξαγωγή κειμένου από την εικόνα καθώς το ML Kit δεν έχει τέτοια δυνατότητα. Αυτό μπορεί να γίνει χρησιμοποιώντας βιβλιοθήκες τρίτων ή υπηρεσίες web που προσφέρουν παρόμοιες λειτουργίες με το kit αλλά να αναγνωρίζουν τέτοιους χαρακτήρες.



Ενώ αυτό το έργο έχει επιτύχει τους στόχους του, είναι σημαντικό να αναγνωρίσουμε τους περιορισμούς του, συμπεριλαμβανομένης της ανάγκης για συνεχείς ενημερώσεις για τη διατήρηση της ακρίβειας των μεταφράσεων καθώς εξελίσσονται τα γλωσσικά μοντέλα. Επιπλέον, οι πτυχές του απορρήτου και της ασφάλειας του χειρισμού δεδομένων και των υπηρεσιών που βασίζονται στο cloud θα πρέπει να εξετάζονται προσεκτικά και να αντιμετωπίζονται για να διασφαλίζεται η προστασία των πληροφοριών των χρηστών.

Συνοπτικά, αυτή η διατριβή συνέβαλε στον τομέα της ανάπτυξης εφαρμογών για κινητά και της μηχανικής μάθησης δημιουργώντας μια εφαρμογή Android που χρησιμοποιεί το ML Toolkit της Google για μετάφραση κειμένου εικόνας, προωθώντας έτσι την προσβασιμότητα της γλώσσας και τη διαπολιτισμική επικοινωνία. Καθώς η τεχνολογία συνεχίζει να εξελίσσεται, είναι επιτακτική ανάγκη να αναγνωρίσουμε τις δυνατότητες περαιτέρω προόδου σε αυτόν τον τομέα. Αυτή η έρευνα παρέχει τη βάση για μελλοντικές εργασίες για τη βελτίωση και την επέκταση των δυνατοτήτων των μεταφραστικών εφαρμογών, συμβάλλοντας τελικά σε μια πιο συνδεδεμένη και πολύγλωσση παγκόσμια κοινωνία.



Κεφάλαιο 7^ο

7 Βιβλιογραφικές πηγές

1. **Ανάλυση και σχεδιασμός συστημάτων με τη UML 2.0**, Τρίτη αμερικανική έκδοση, A. Dennis, B. Wixom, D. Tegarden.
2. **Get started with Kotlin.** <https://kotlinlang.org/docs/getting-started.html>
3. **IBM Explore the UML sequence diagram.** <https://developer.ibm.com/articles/the-sequence-diagram/>
4. **Writing Effective Use Cases.** <https://www-public.imtbs-tsp.eu/~gibson/Teaching/Teaching-ReadingMaterial/Cockburn00.pdf>
5. **Android Studio Documentation.** <https://developer.android.com/docs>
6. **Identify the language of text with ML Kit on Android.** <https://developers.google.com/ml-kit/language/identification/android>
7. **Translate text with ML Kit on Android.** <https://developers.google.com/ml-kit/language/translation/android>
8. **Recognize text in images with ML Kit on Android.** <https://developers.google.com/ml-kit/vision/text-recognition/v2/android>
9. **Optical Character Recognition (OCR) Technology.** https://unstats.un.org/unsd/demographic/sources/census/wphc/dataCapture/docs/Data-Capture_ch06-ABS.pdf
10. **Android Top Image Cropper Libraries.** <https://medium.com/mindorks/android-top-image-cropper-libraries-3bc4a4f8f2df>
11. **Image capture.** <https://developer.android.com/training/camerax/take-photo>
12. **How to Select an Image from Gallery in Android.** <https://www.geeksforgeeks.org/how-to-select-an-image-from-gallery-in-android/>
13. **Get started with Jetpack Compose.** <https://developer.android.com/jetpack/compose/documentation>
14. **Compose-Cropper.** <https://github.com/SmartToolFactory/Compose-Cropper>
15. **Toast Widget.** <https://developer.android.com/reference/android/widget/Toast>
16. **When You Should Choose Firebase over Backend Development for Your Project.** <https://www.cypresssoft.com/post/choose-firebase-over-backend-development/>



17. **Image View Documentation.**
<https://developer.android.com/reference/android/widget/ImageView>

18. **MVVM Architecture in Android using Kotlin.**
<https://medium.com/@jecky999/mvvm-architecture-in-android-using-kotlin-a-practical-guide-73f8de1d9c58>

19. **ViewModel overview.**
<https://developer.android.com/topic/libraries/architecture/viewmodel>