



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

| | |
|-----------------------|--|
| Τίτλος Διατριβής | Εφαρμογή Android για αναζήτηση ταινιών γραμμένη σε kotlin An Android app for movie search written in kotlin |
| Όνοματεπώνυμο Φοιτητή | Γκίνης Δημήτριος |
| Πατρώνυμο | Παναγιώτης |
| Αριθμός Μητρώου | ΜΠΠΛ/ 17008 |
| Επιβλέπων | Αλέξης Ευθύμιος, Αναπληρωτής Καθηγητής |

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Αλέπης Ευθύμιος
Αναπληρωτής Καθηγητής

(υπογραφή)

Βίβρου Μαρία
Καθηγήτρια

(υπογραφή)

Τσιχριντζής Γεώργιος
Καθηγητής

Περιεχόμενα

| | |
|---|-----------|
| Ευχαριστίες | 5 |
| Περίληψη | 5 |
| Abstract | 6 |
| 1.Εισαγωγή | 6 |
| 1.1 Χώρος του Προβλήματος | 6 |
| 1.2 Στόχοι της εργασίας | 7 |
| 1.3 Παρόμοια Συστήματα | 7 |
| 1.3.1 IMDB | 7 |
| 1.3.2 Movies Hub | 8 |
| 1.3.3 Moviebase: Discover Movies & Track TV Shows | 8 |
| 1.3.4 Σύγκριση Εφαρμογών | 9 |
| 2. Φορητές συσκευές | 9 |
| 2.1 Έξυπνα τηλέφωνα(Smartphones) | 10 |
| 2.2 Υπολογιστής ταμπλέτα(Tablet) | 11 |
| 2.3 Λειτουργικά Συστήματα για Έξυπνα Τηλέφωνα | 11 |
| 2.3.1 Android | 12 |
| 2.3.1 iOS | 12 |
| 2.3.3 Windows Phone/Mobile | 13 |
| 2.4 Android και iOS | 14 |
| 2.4.1 Θετικά του Android | 14 |
| 2.4.2 Θετικά του iOS | 14 |
| 2.4.3 Αρνητικά του Android | 14 |
| 2.4.4 Αρνητικά του iOS | 15 |
| 3. Εφαρμογές Έξυπνων τηλεφώνων | 15 |
| 3.1 Εργαλεία Ανάπτυξης Εφαρμογών | 15 |
| 3.2 Τύποι εφαρμογών κινητών συσκευών | 16 |
| 3.2.1 Εγγενείς Εφαρμογές (Native App) | 16 |
| 3.2.2 Εφαρμογές Ιστού (Web App) | 16 |
| 3.2.3 Υβριδικές Εφαρμογές (Hybrid applications) | 16 |
| 3.2.4 Επιλογή κατάλληλης μεθόδου | 17 |
| 4. Λειτουργικό Σύστημα Android και Τεχνολογίες | 17 |
| 4.1 Αρχιτεκτονική Συστήματος Android | 17 |
| 4.1.1 Linux Kernel | 18 |
| 4.1.2 Hardware Abstraction Layer (HAL) | 18 |
| 4.1.3 Android Runtime | 18 |
| 4.1.4 Native Libraries | 18 |
| Εφαρμογή Android για αναζήτηση ταινιών γραμμένη σε kotlin | 3 |

| | |
|---|-----------|
| 4.1.5 Java API Framework | 19 |
| 4.1.6 System Apps | 19 |
| 4.2 Δομικά Στοιχεία Μιας Εφαρμογής | 19 |
| 4.2.1 Δραστηριότητα (Activity) | 19 |
| 4.2.2 Υπηρεσίες (Services) | 19 |
| 4.2.3 Προθέσεις (Intents) | 19 |
| 4.2.4 Πάροχοι Περιεχομένου (Content Providers) | 20 |
| 4.2.5 Δέκτες μηνυμάτων Broadcast(Broadcast Receivers) | 20 |
| 4.3 Κύκλος ζωής των δραστηριοτήτων | 20 |
| 4.3.1 onCreate | 21 |
| 4.3.2 onStart | 21 |
| 4.3.3 onResume | 21 |
| 4.3.4 onPause | 21 |
| 4.3.5 onStop | 21 |
| 4.3.6 onDestroy | 21 |
| 4.4 Διεπαφή χρήστη-User Interface | 21 |
| 4.5 Material Design | 22 |
| 4.6 Kotlin | 23 |
| 4.7 Δικαιώματα | 23 |
| 4.8 RecyclerView και CardView | 24 |
| 4.9 Google Firebase | 25 |
| 4.9.1 Realtime Database | 25 |
| 4.9.2 Firebase Authentication | 25 |
| 4.9.3 ML Kit | 26 |
| 5. Ανάλυση των λειτουργιών της Εφαρμογής | 26 |
| 5.1 Ανάλυση Απαιτήσεων | 26 |
| 5.2 Σχεδιασμός | 27 |
| 5.2.1 Δυνατότητες Λειτουργιών των Χρηστών | 27 |
| 5.2.2 Σχεδιασμός της Διαδικασίας Σύνδεσης και Εγγραφής του Χρήστη | 28 |
| 5.2.3 Σχεδιασμός της λειτουργίας “Λεπτομέρειες για την ταινία” | 30 |
| 5.2.4 Σχεδιασμός της Firebase Βάσης Δεδομένων | 30 |
| 6. Παρουσίαση της Εφαρμογής | 33 |
| 6.1 Αρχική Διεπαφή | 33 |
| 6.2 Οθόνη Εγγραφής Χρήστη | 34 |
| 6.3 Home Page | 35 |
| 6.4 Στοιχεία Ταινίας | 35 |
| 6.5 Στοιχεία Ηθοποιού | 36 |
| 6.6 Αναζήτηση ταινίας | 37 |
| 6.7 Ανακάλυψη ταινίας | 37 |
| 6.8 Αναζήτηση ταινίας μέσω κάμερας | 38 |
| 6.9 Το προφιλ μου | 39 |
| 6.10 Λίστα Αγαπημένων | 40 |
| Εφαρμογή Android για αναζήτηση ταινιών γραμμένη σε kotlin | 4 |

| | | |
|-----------|---|-----------|
| 6.11 | Λίστα Παρακολούθησης | 40 |
| 6.12 | Chat | 41 |
| 6.12 | Στατιστικά | 41 |
| 6.13 | Ρυθμίσεις | 42 |
| 6.14 | Αλλαγή Κωδικού | 43 |
| 7. | Επεξήγηση κώδικα | 43 |
| 7.1 | Δημιουργία Base Activity | 43 |
| 7.2 | Dark Mode | 45 |
| 7.3 | SafeClickListener | 46 |
| 7.4 | Σύνδεση Χρήστη | 47 |
| 7.5 | Retrofit | 48 |
| 8. | Συμπεράσματα - Μελλοντικές Προτάσεις | 50 |
| 8.1 | Συμπεράσματα | 50 |
| 8.2 | Μελλοντικές Επεκτάσεις | 51 |
| | Βιβλιογραφία | 51 |

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον κ. Αλέπη για την ανάθεση της μεταπτυχιακής μου διατριβής, καθώς και τους συναδέλφους μου Ρογκακο Γεώργιο, Καραμπίνη Μπράϊαν και Χασάπη Παντελή για την επίβλεψη και την επιστημονικά άρτια καθοδήγησή τους κατά τη διάρκεια του σχεδιασμού και της ανάπτυξης αυτής της εργασίας, αλλά και για τις καίριες συμβουλές που μου έδωσαν για τη βελτίωση της εφαρμογής που αναπτύχθηκε.

Τέλος, ευχαριστίες οφείλω στην οικογένειά μου και στους φίλους μου, για την ηθική υποστήριξη και βοήθεια όλα τα χρόνια των σπουδών, όπως επίσης για την κατανόηση και την ενθάρρυνση κατά τη διάρκεια εκπόνησης της μεταπτυχιακής διατριβής.

Περίληψη

Στόχος της παρούσας Μεταπτυχιακής Διατριβής αποτελεί η ανάπτυξη εφαρμογής για λειτουργικό σύστημα Android για την αναζήτηση μέσω λέξεων, φίλτρων και κάμερας για την εύρεση και προβολή στοιχείων ταινιών. Αρχικά παρατίθενται συνοπτικά εισαγωγικές πληροφορίες για το λειτουργικό σύστημα Android, καθώς και τα αποτελέσματα της έρευνας για τέτοιου είδους εφαρμογές. Στην συνέχεια στα πλαίσια της εργασίας, είναι η υλοποίηση μιας παρόμοιας εφαρμογής. Η εφαρμογή χρησιμοποιώντας την cloud βάση της firebase θα επιτρέπει στον χρήστη να δημιουργεί και να εισέρχεται στην εφαρμογή με τον προσωπικό του λογαριασμό, μέσω των APIs του tmdb.org ο χρήστης θα μπορεί να ψάξει και να δει λεπτομερείς για την ταινία και τους ηθοποιούς της, να μπορέσει να την αποθηκεύσει και να την μοιραστεί. Θα έχει την δυνατότητα να επεξεργαστεί τον προσωπικό του λογαριασμό και να αλλάξει τον τρόπο εμφάνισης της εφαρμογής. Επίσης θα έχει την δυνατότητα να παρακολουθήσει τα στατιστικά της χρήσης του. Τέλος περιγράφονται λεπτομερώς και αναλύονται όλα τα στάδια της δημιουργίας της εφαρμογής, από τη μελέτη των λειτουργικών απαιτήσεων, την αρχιτεκτονική του συστήματος και την τεχνική υλοποίηση, μέχρι την παρουσίαση των συμπερασμάτων και των μελλοντικών προοπτικών της.

Abstract

The purpose of this master thesis is to develop a digital application for the Android platform. The app will find and display movies based on keywords, filters and the camera. Initially a summary of introductory information about the Android operating system and similar apps will be presented. Continuing the development of the app will be presented. The app using the cloud storage of the firebase will allow the user to create a personal account and log in to the app using it, through the APIs of the tmdb.org the user will be able to search and find details about the movie and its cast, he will also be able to save and share movies. He will have the ability to modify his personal account and change the way the application looks. Also available to the user are the application usage stats. Finally, the steps of creating the application are described, from the functional requirements, the apps architecture and the app development, to the presentation of the final thoughts and possible future improvements.

1.Εισαγωγή

Στη σημερινή εποχή η ραγδαία αύξηση των κινητών συσκευών και η ανάγκη για υπερπληροφόρηση, έχει οδηγήσει στην αυξανόμενη εμφάνιση σύγχρονων ηλεκτρονικών εφαρμογών και υπηρεσιών, που έχουν ως στόχο να διευκολύνουν την απομακρυσμένη επικοινωνία και να παρέχουν στους χρήστες τη δυνατότητα να πλοηγηθούν και να λάβουν την απαραίτητη πληροφορία που επιθυμούν ταχύτατα. Σε αυτό έχει συμβάλει ιδιαίτερα το διαδίκτυο και η άνθιση της υπολογιστικής νέφους (cloud computing) τα τελευταία χρόνια, τεχνολογίες οι οποίες έχουν δώσει τη δυνατότητα στο τελικό χρήστη σε οποιοδήποτε σημείο του κόσμου βρίσκεται να έχει άμεσα αποτελέσματα και επαρκή πληροφόρηση.

Όσο αναφορά τις ταινίες για τις οποίες θα αναφερθούμε, τα τελευταία χρόνια έχουν δημιουργήσει τους κατάλληλους ιστότοπους όπου οι επισκέπτες μπορούν να βρουν τις κατάλληλες πληροφορίες που τους ενδιαφέρουν αναφορικά με το περιεχόμενο των ταινιών. Όμως είτε δεν έχουν δημιουργηθεί οι πλέον «φιλικοί» ιστότοποι είτε το περιεχόμενο που περιέχουν δεν ανταποκρίνεται στα σημερινά πρότυπα, όπου χαρακτηρίζεται από την αμεσότητα και την επαρκή πληροφορία.

Στην παρούσα μεταπτυχιακή διατριβή θα παρουσιαστεί η ανάπτυξη της εφαρμογής σε περιβάλλον Android για την αναζήτηση ταινιών και πώς οι νέες τεχνολογίες θα βοηθήσουν το τελικό χρήστη στην καλύτερη ενημέρωση και περιήγηση του.

Στο παρόν κεφάλαιο θα αναφερθούμε στο σκοπό εκπόνησης της διπλωματικής εργασίας, στο χώρο προβλήματος και σε παρόμοιες εφαρμογές αναζήτησης ταινιών.

1.1 Χώρος του Προβλήματος

Τις προηγούμενες δεκαετίες όταν ακόμα η τεχνολογία δεν είχε παρεισφρήσει στη καθημερινότητά μας, τα μουσεία δεν κατείχαν τα κατάλληλα μέσα για την απαραίτητη ενημέρωση του σχετικά με το περιεχόμενο και την ποιότητα μια ταινίας, ο οποίος μπορούσε να ενημερωθεί μόνο από περιοδικά, από αφίσες ή μέσω φίλων και γνωστών. Τα επόμενα χρόνια η τηλεόραση βοήθησε σημαντικά στην ενημέρωση των θεατών σχετικά με τις ταινίες που τους ενδιέφεραν και τους προσέφερε επιπλέον προτάσεις με ταινίες που δεν είχαν στο μυαλό τους.

Στη συνέχεια το διαδίκτυο ήρθε για να εκσυγχρονίσει τα προαναφερθέντα, σταδιακά άρχισαν οι δημιουργίες ιστοσελίδων και με την εξέλιξη των φορητών συσκευών δημιουργήθηκαν οι κατάλληλες εφαρμογές σε περιβάλλον Android/iOS/Windows Phone όπως τις γνωρίζουμε σήμερα. Μέσω των ιστοσελίδων ο χρήστης έχει πρόσβαση σε μια τεράστια συλλογή πληροφοριών, κριτικών και προτάσεων οι οποίες θα τον βοηθήσουν στην τελική του επιλογή και θα του προσφέρουν εναλλακτικές επιλογές βάση των προτιμήσεων του. Με την άνοδο των smartphones οι πληροφορίες είναι άμεσα προσβάσιμες στον χρήστη καθώς δεν χρειάζεται πλέον υπολογιστής απλά μια φορητή συσκευή που έχει πρόσβαση στο διαδίκτυο.

1.2 Στόχοι της εργασίας

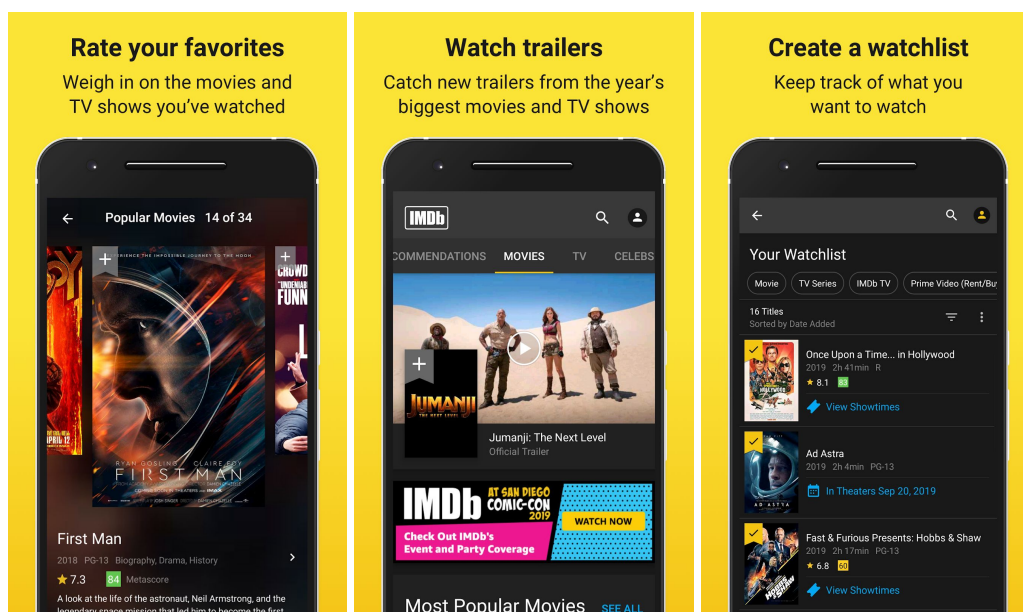
Αντικείμενο της παρούσας μεταπτυχιακής διατριβής αποτελεί η μελέτη και ανάπτυξη της εφαρμογής αναζήτησης ταινιών με λειτουργικό σύστημα Android με τους πλέον ενδεδειγμένους τρόπους σχεδίασης και συγχρόνων λειτουργιών. Το βασικό χαρακτηριστικό της εφαρμογής είναι να είναι εύχρηστη και φιλική στο τελικό χρήστη, να κάνει χρήση των κατάλληλων πόρων ώστε να μην επιβαρύνει τις επιδόσεις της συσκευής, να προσαρμόζεται σε κάθε τύπου συσκευή (tablet/smartphone) ανάλογα με τις διαστάσεις της οθόνης, να παρέχει τη κατάλληλη πληροφόρηση και να αξιοποιούνται νέες τεχνολογίες της υπολογιστικής νέφους(Cloud).

1.3 Παρόμοια Συστήματα

Σε αυτό το κεφάλαιο θα παρουσιαστούν αντίστοιχες εφαρμογές, θα αναφερθούν οι καινοτομίες και οι παροχές τους. Όλες οι εφαρμογές αφορούν το λειτουργικό σύστημα Android.

1.3.1 IMDB

Το Imdb είναι η πιο γνωστή και περιεκτική βάση ταινιών στον κόσμο με πληροφορίες για ηθοποιούς, ταινίες, τηλεοπτικά προγράμματα, παρουσιαστές της τηλεόρασης, βιντεοπαιχνίδια και συντελεστές παραγωγής ταινιών ή προγραμμάτων. Το IMDB δημιουργήθηκε το 1990, όταν δεν υπήρχε ακόμη ο Παγκόσμιος Ιστός, τον Ιούνιο του 2010 κυκλοφόρησε η Android εφαρμογή στο Play Store.



Εικόνα 1. Στιγμιότυπα του IMDB (<https://play.google.com/>)

Η εφαρμογή έχει εγκατασταθεί σε παραπάνω από 100.000.000 Android συσκευές αποσπώντας βαθμολογία 4,7/5 σε ένα σύνολο από 673.084 ψήφους.

Η εφαρμογή προσφέρει τις παρακάτω δυνατότητες στον χρήστη:

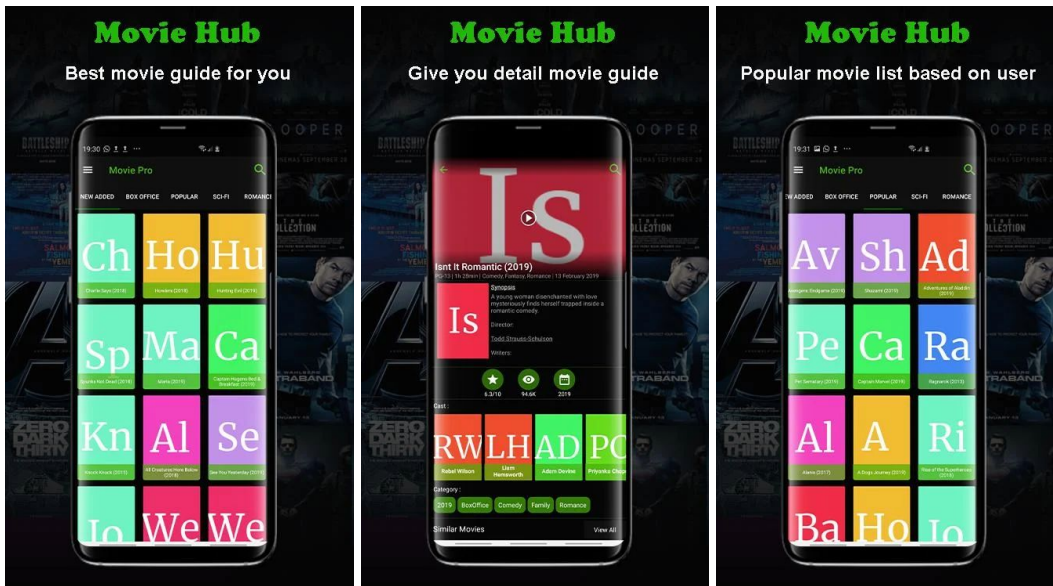
- Δημιουργία προφίλ και σύνδεση
- Αναζήτηση ταινιών / σειρών / ηθοποιών
- Δημιουργία Watchlist
- Βαθμολογία ταινιών / σειρών

Εφαρμογή Android για αναζήτηση ταινιών γραμμένη σε kotlin

- Προσωποποιημένες προτάσεις ταινιών / σειρών
- Τμήμα με νέα και ειδήσεις
- Κριτικές για ανάγνωση από τον θεατή

1.3.2 Movies Hub

Το Movies Hub είναι μια εφαρμογή η οποία έχει φτιαχτεί από μια μικρή ομάδα developers και όχι από μια μεγάλη εταιρεία η οποία κυκλοφόρησε πρόσφατα, συγκεκριμένα τον Μάιο του 2019. Η εφαρμογή είναι μικρή σε μέγεθος και δεν καταναλώνει πολύ τους πόρους της συσκευής.



Εικόνα 2. Στιγμιότυπα του Movies Hub (<https://play.google.com/>)

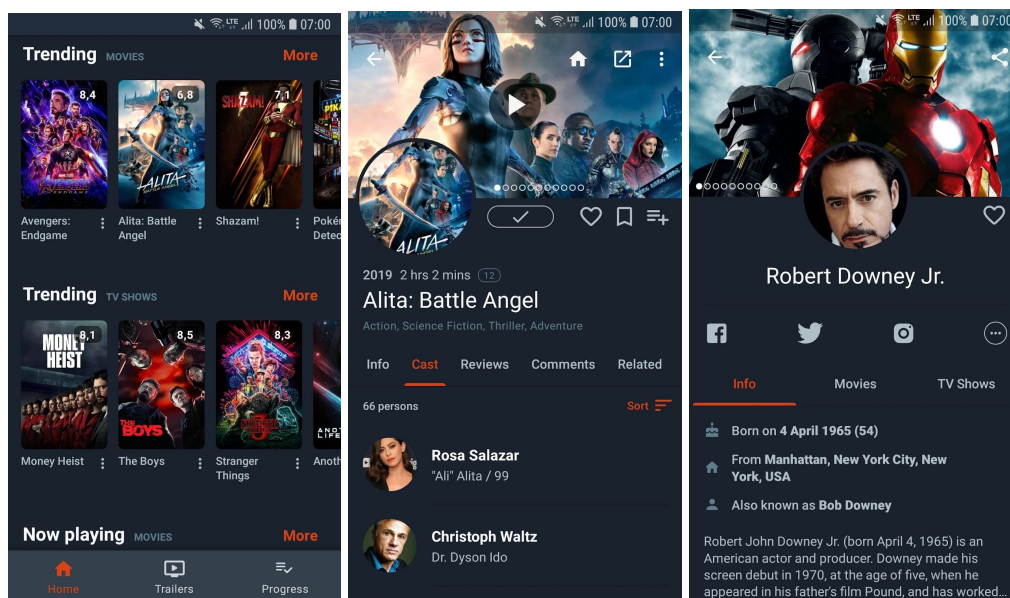
Η εφαρμογή έχει εγκατασταθεί σε παραπάνω από 1.000.000 Android συσκευές αποσπώντας βαθμολογία 4,5/5 σε ένα σύνολο από 121.216 ψήφους.

Η εφαρμογή προσφέρει τις παρακάτω δυνατότητες στον χρήστη:

- Αναζήτηση ταινιών
- Δημιουργία Watchlist
- Προβολή παρόμοιων ταινιών

1.3.3 Moviebase: Discover Movies & Track TV Shows

Το Moviebase: Discover Movies & Track TV Shows είναι μια εφαρμογή η οποία χρησιμοποιεί την The Movie Database (TMDB) σαν βάση ταινιών, την ίδια που χρησιμοποιήθηκε για την δημιουργία της παρούσας εφαρμογής. Η εφαρμογή κυκλοφόρησε τον Οκτώβρη του 2017 και υποστηρίζει συνολικά 39 γλώσσες.



Εικόνα 3. Στιγμιότυπα του Moviebase: Discover Movies & Track TV Shows (<https://play.google.com/>)

Η εφαρμογή έχει εγκατασταθεί σε παραπάνω από 100.000 Android συσκευές αποσπώντας βαθμολογία 4,2/5 σε ένα σύνολο από 2.380 ψήφους.

Η εφαρμογή προσφέρει τις παρακάτω δυνατότητες στον χρήστη:

- Δημιουργία προφίλ και σύνδεση
- Αναζήτηση ταινιών / σειρών / ηθοποιών
- Αναζήτηση ταινιών βάση φίλτρων
- Δημιουργία Watchlist
- Βαθμολογία ταινιών / σειρών
- Προσωποποιημένες προτάσεις ταινιών / σειρών
- Ημερολόγιο κυκλοφορίας νέων ταινιών / σειρών

1.3.4 Σύγκριση Εφαρμογών

Θα προσπαθήσουμε να συγκρίνουμε τις εφαρμογές που παρουσιάστηκαν. Αρχικά η εφαρμογή που ξεχωρίζει είναι το Imdb καθώς είναι η πιο γνωστή, παλιά και έχει την μεγαλύτερη βάση. Είναι μια εφαρμογή η οποία πλέον θεωρείται ώριμη από άποψη κώδικα κάνοντάς την πιο stable στην λειτουργία της. Παρόλα αυτά η εφαρμογή είναι η μεγαλύτερη σε μέγεθος από τις 3 (συνολικά 66MB) έχοντας ως αποτέλεσμα να μην ενδείκνυται για παλαιότερες συσκευές. Το Movies Hub ακολουθεί μια άλλη φιλοσοφία, είναι μια ελαφριά εφαρμογή η οποία προσφέρει βασικές λειτουργίες στον χρήστη χωρίς να έχει από πίσω την φήμη του Imdb και την μεγάλη του βάση ταινιών. Ενδείκνυται για παλαιότερες συσκευές και για χρήστες οι οποίοι δεν έχουν ανάγκη για έξτρα λειτουργίες. Το Moviebase: Discover Movies & Track TV Shows είναι μια φιλόδοξη εφαρμογή η οποία προσπαθεί να μιμηθεί τις λειτουργίες του Imdb προσφέροντας πληθώρα δυνατοτήτων στους χρήστες της, η εφαρμογή έχει την μικρότερη βαθμολογία συγκριτικά με τις άλλες 2, γεγονός που δείχνει ότι πιθανόν να μην είναι τόσο stable όσο οι άλλες 2. Ενδείκνυται για χρήστες τους οποίους τους αρέσει το Imdb αλλά θα ήθελαν να δοκιμάσουν και κάτι διαφορετικό.

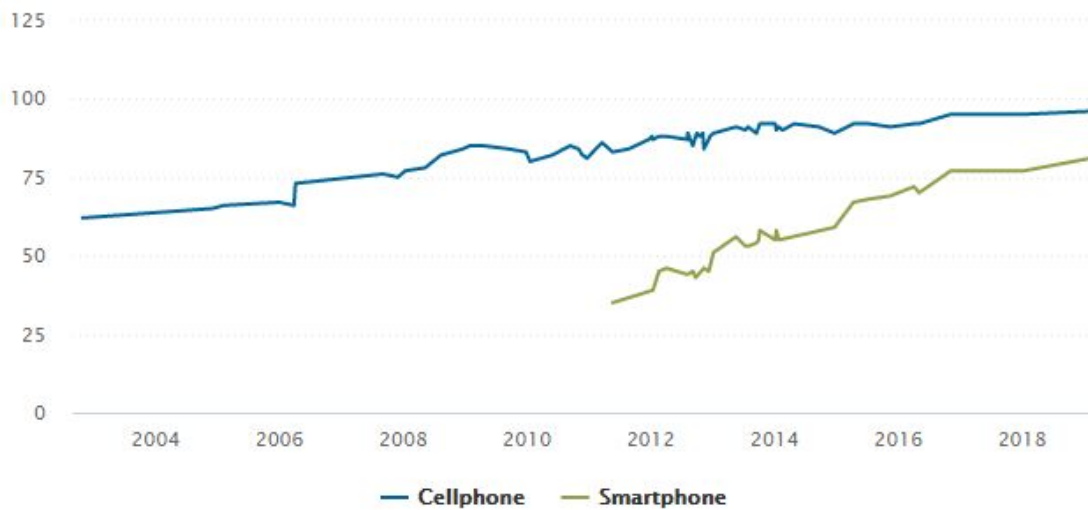
2. Φορητές συσκευές

Οι φορητές συσκευές αποτελούν ένα μεγάλο σύνολο. Στην ενότητα αυτή παρουσιάζονται οι δύο επικρατέστερες φορητές συσκευές (Smartphones, tablet) με τα χαρακτηριστικά τους καθώς και οι ανάγκες που επιχειρούν να ικανοποιήσουν, οι οποίες ταυτίζονται αλλά και ποικίλουν σε μεγάλο βαθμό.

Εφαρμογή Android για αναζήτηση ταινιών γραμμένη σε kotlin

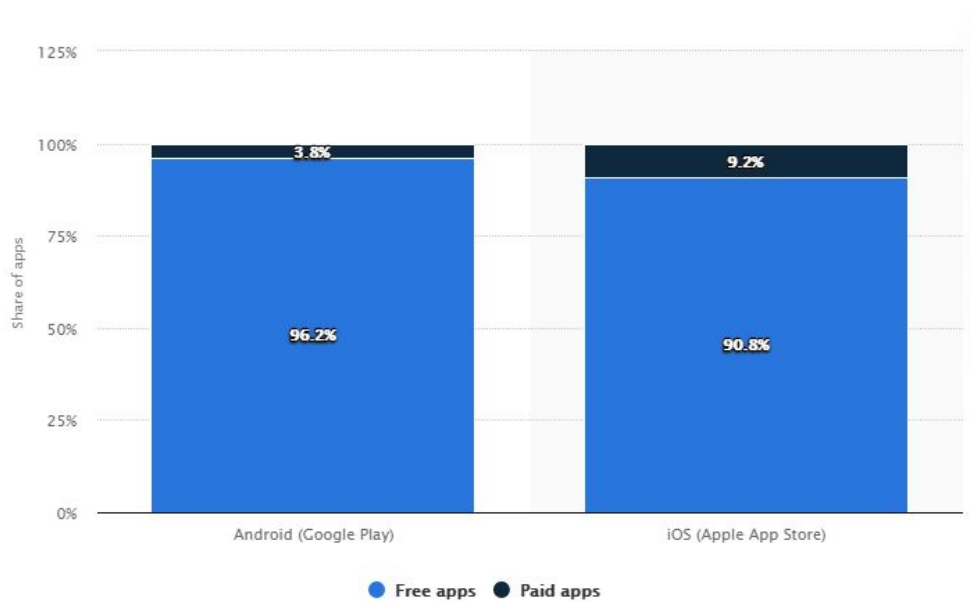
2.1 Έξυπνα τηλέφωνα(Smartphones)

Τα πρώτες έξυπνες συσκευές έκαναν την εμφάνισή τους το 1984 με την μορφή των PDA, είχαν βασικές λειτουργίες ηλεκτρονικού υπολογιστή χωρίς όμως την δυνατότητα να συνδεθούν σε κάποιο δίκτυο. Το πρώτο κινητό τηλέφωνο με δυνατότητες PDA ήταν ένα πρωτότυπο IBM που αναπτύχθηκε το 1992 και διατέθηκε στο εμπόριο για τους καταναλωτές το 1994, ήταν η πρώτη συσκευή που μπορεί να αναφέρεται ως smartphone, έστω και αν αυτός ο όρος δεν είχε ακόμη επινοηθεί. Είχε την ικανότητά του να πραγματοποιεί και να δέχεται κλήσεις καθώς ήταν και σε θέση να αποστέλλει και να λαμβάνει φαξ και e-mail. Ο όρος smartphone δεν εμφανίστηκε μέχρι το 1997, όταν η Ericsson περιέγραψε το GS 88 ως Smart Phone (Έξυπνο Τηλέφωνο). Τα Smartphone έγιναν δημοφιλείς με την κυκλοφορία των λειτουργικών συστημάτων iOS από την Apple το 2007 και του Android από την Google το 2008 και πλέον κατέχουν το μεγαλύτερο μερίδιο αγοράς στις κινητές συσκευές.



Διάγραμμα 1. Χρήστες Smartphone(<https://www.pewresearch.org/internet/fact-sheet/mobile/>)

Κύριο χαρακτηριστικό των Smartphones είναι ότι έχουν πρόσβαση σε λειτουργίες και προγράμματα που μέχρι να εμφανιστούν τα Smartphones είχαν μόνο οι υπολογιστές. Πλέον διαθέτουν οθόνες αφής για ευκολότερη χρήση, υποστήριξη 4G/5G τεχνολογίας σύνδεσης, ενσωματωμένο Wi-Fi για ασύρματη συνδεσιμότητα με το διαδίκτυο, αισθητήρες θέσης και κίνησης καθώς και φωτογραφική μηχανή. Μεγάλο πλεονέκτημα που έχουν τα Smartphones, που τελικά τα έκανε να κυριαρχήσουν στην αγορά, είναι ότι διαθέτουν ηλεκτρονικά καταστήματα τα οποία προσφέρουν πληθώρα εφαρμογών, πολλές από τις οποίες διατίθενται δωρεάν, στους χρήστες. Ολοένα και περισσότερες εφαρμογές πλέον διατίθενται δωρεάν αλλά για να μπορέσει ο χρήστης να έχει πρόσβαση σε όλες τις λειτουργίες της εφαρμογής θα πρέπει να προβεί σε αγορά αυτών ή να παρακολουθήσει διαφημίσεις. Ο παρακάτω πίνακας είναι μέρος έρευνας μέχρι τον Δεκέμβριο του 2019 όπου παρατηρούμε των αριθμών των εφαρμογών οι οποίες είναι δωρεάν στα εκάστοτε ηλεκτρονικά καταστήματα.



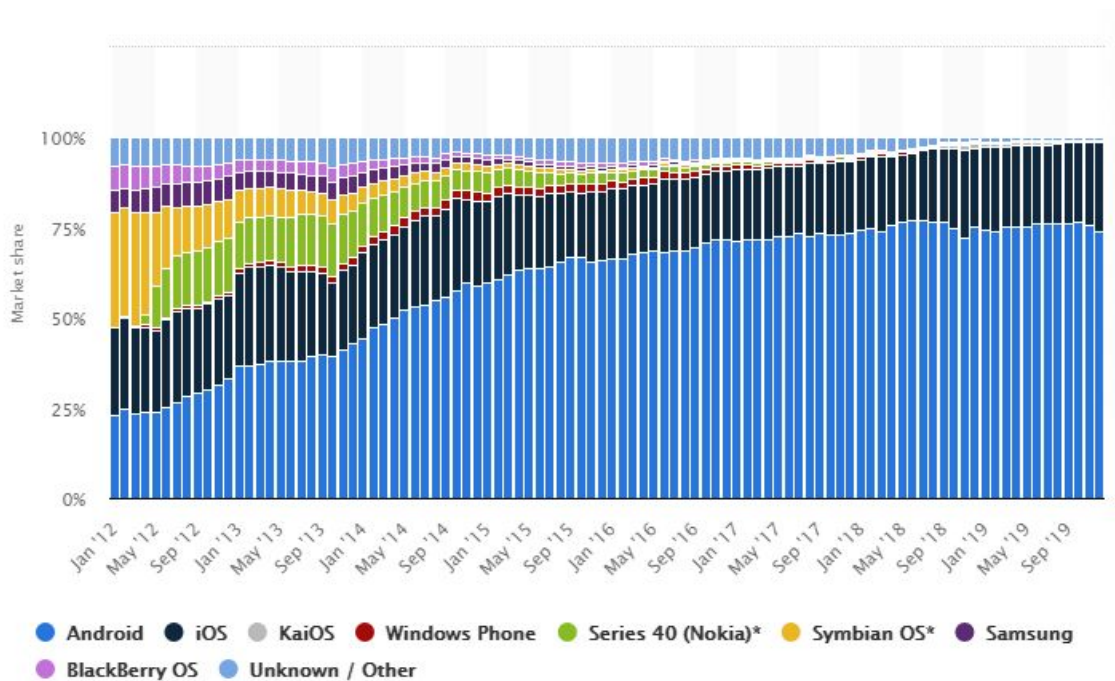
Διάγραμμα 2. Distribution of free and paid apps in the Apple App Store and Google Play as of December 2019 (<https://www.statista.com/>)

2.2 Υπολογιστής ταμπλέτα (Tablet)

Ένας υπολογιστής ταμπλέτα ή tablet διαθέτει τις λειτουργίες και δυνατότητες των Smartphones έχοντας μεγαλύτερη οθόνη από αυτά αλλά μικρότερη οθόνη από τους προσωπικούς υπολογιστές. Τα tablets υπερτερούν των Smartphones καθώς λόγω των μεγαλύτερων διαστάσεων διαθέτουν μεγαλύτερη μπαταρία αρα και καλύτερη αυτονομία, διαθέτουν περισσότερες θύρες επέκτασης και είναι πιο βολικά σε παράλληλες εργασίες λόγω της μεγαλύτερης οθόνης. Τα λειτουργικά συστήματα στις ταμπλέτες είναι συνήθως Android και iOS. Τα τελευταία χρόνια, λόγω της εξέλιξης της τεχνολογίας, οι ταμπλέτες είναι πλέον σε θέση να υποστηρίζουν πλήρως το λειτουργικό σύστημα Windows 10 της Microsoft και με την χρήση ενός πληκτρολογίου μετατρέπεται σε έναν φορητό υπολογιστή ο οποίος μπορεί να έχει και επαγγελματική χρήση.

2.3 Λειτουργικά Συστήματα για Έξυπνα Τηλέφωνα

Τα λειτουργικά συστήματα για έξυπνα τηλέφωνα συνδυάζουν δυνατότητες τις οποίες έχουν οι ηλεκτρονικοί υπολογιστές με δυνατότητες οι οποίες είναι χρήσιμες στις σύγχρονες κινητές συσκευές. Την παρούσα χρονική περίοδο οι επιλογές λειτουργικών συστημάτων είναι περιορισμένες, τα προηγούμενα χρόνια, όταν τα Smartphones πρωτοεμφανίστηκαν, ο κάθε κατασκευαστής κινητών συσκευών είχε το δικό του λειτουργικό σύστημα στην συσκευή του. Τα λειτουργικά συστήματα που κυριάρχησαν είναι το Android, iOS και το Windows Phone με το τελευταία να μην μπορέσει να ανταγωνιστεί τα υπολοιπα συστήματα με αποτέλεσμα να σταματήσει η κυκλοφορία του τον Οκτώβριο του 2017. Εώς τον Δεκέμβριο του 2019 το Android κατέχει το 74% της αγοράς και το iOS το 25%.



Διάγραμμα 3. Mobile operating systems' market share worldwide from January 2012 to December 2019 (<https://www.statista.com/>)

Το γεγονός ότι η πλατφόρμα του Android έχει το συντριπτικό μερίδιο της αγοράς καθιστά την απασχόληση και την ανάπτυξη εφαρμογών σε αυτό το λειτουργικό ιδιαίτερα ασφαλής επιλογή (δεδομένο το ότι λόγω της χρήσης της) και με αρκετές προοπτικές για το μέλλον. Παρακάτω θα γίνει μια μικρή αναφορά στα λειτουργικά συστήματα που υπάρχουν στην αγορά αυτή την περίοδο καθώς και του Windows Phone που πλέον η ανάπτυξη για αυτό το λειτουργικό έχει σταματήσει.

2.3.1 Android

Android είναι το λειτουργικό σύστημα το οποίο έχει αναπτύξει η Google και το οποίο θα μας απασχολήσει στη παρούσα μεταπτυχιακή εργασία. Όπως παρατηρούμε στο παραπάνω διάγραμμα, είναι η πιο γρήγορη αναπτυσσόμενη πλατφόρμα και χρησιμοποιείται από πολλούς κατασκευαστές, στην αγορά διατίθενται πολλές διαφορετικές συσκευές σε όλες τις κατηγορίες τιμών, ανάλογα με τις απαιτήσεις του χρήστη. Εκτενέστερη αναφορά θα γίνει σε επόμενο κεφάλαιο.

2.3.1 iOS

Το 2007 η Apple παρουσίασε το πρώτο iPhone φέρνοντας επανάσταση και σηματοδότησε την αρχή μιας νέας γενιάς έξυπνων τηλεφώνων. Αυτό που έκανε την διαφορά και έδωσε και την ώθηση στην εταιρεία να ξεχωρίσει ήταν η κατάργηση του φυσικού πληκτρολογίου και αντί αυτού η χρήση οθόνης αφής και πολλών λειτουργιών που έκανε ευχάριστη και φιλική την ενασχόληση με τα κινητά τηλέφωνα στο τελικό χρήστη.

Οι συσκευές της Apple έχουν το δικό της λειτουργικό σύστημα, το iOS. Σήμερα, το iOS βρίσκεται στην έκδοσή 13.2, και έχει εκδώσει ειδικά τροποποιημένο λογισμικό για να τρέχει στην ταμπλέτα της το iPad. Το λειτουργικό βασίζεται στο macOS, ο προγραμματισμός γίνεται σε Objective-C και Swift. Το λειτουργικό αυτό σύστημα είναι κλειστού κώδικα και οι συσκευές που διατίθενται στην αγορά, κατασκευάζονται αποκλειστικά από την ίδια την Apple.



Εικόνα 4. Το iOS και οι συσκευές της Apple.(<https://www.apple.com>)

2.3.3 Windows Phone/Mobile

Στη προσπάθεια της Microsoft να αποσπάσει ένα κομμάτι της αγοράς από την Google και την Apple, το 2010 παρουσίασε το λειτουργικό σύστημα Windows Phone. Είναι η εξέλιξη των Windows Mobile που μέχρι τότε χρησιμοποιούνταν από υπολογιστές παλάμης. Η ανάπτυξη εφαρμογών σε αυτό το λογισμικό γίνονταν με την χρήση C#, Visual Basic, και C++. Η Microsoft μέσω των Windows Phone, προσπάθησε να φέρει όσο πιο κοντά γίνεται το γραφικό περιβάλλον που χρησιμοποιεί στους σταθερούς υπολογιστές στα έξυπνα κινητά της, έτσι ώστε οι χρήστες της να έχουν μεγαλύτερη οικειότητα με το λειτουργικό της σύστημα. Το 2015 η Microsoft ανακοίνωσε την τελευταία έκδοση του λειτουργικού, το Windows Phone 10. Ο μικρός αριθμός εφαρμογών και ο υψηλός ανταγωνισμός ήταν η αιτία που τα Windows Phone δεν κατάφεραν να κερδίσουν ένα καλό μερίδιο της αγοράς. Τέλος, τον Οκτώβριο του 2017 ανακοίνωσε επίσημα ότι θα σταματήσει την περαιτέρω εξέλιξη των Windows Mobile 10



Εικόνα 5. Το Windows Phone και οι συσκευές του.(<https://www.theverge.com>)

2.4 Android και iOS

Η Google με την εξέλιξη του Android έχει καταφέρει να ξεπεράσει σε πωλήσεις την Apple. Η μεγάλη εξάπλωση του Android έγινε εφικτή επειδή η Google παρέχει το λειτουργικό της σύστημα δωρεάν. Αυτό στη συνέχεια είχε ως αποτέλεσμα να κυκλοφορούν ολοένα και περισσότερες συσκευές, με τις τιμές να κυμαίνονται από τα 50€ έως τα 1300€, με αυτό τον τρόπο ο τελικός καταναλωτής έχει πληθώρα επιλογών για την αγορά νέου κινητού. Σε αυτό έχει συμβάλει και η τιμολογιακή πολιτική της Apple δίνοντας μεγαλύτερη ώθηση στην αγορά των κινητών Android, καθώς με λιγότερα χρήματα είχαν εφάμιλλες επιδόσεις και χαρακτηριστικά. Στο παρόν υποκεφάλαιο θα αναφερθούμε στα θετικά και αρνητικά του κάθε λειτουργικού.

2.4.1 Θετικά του Android

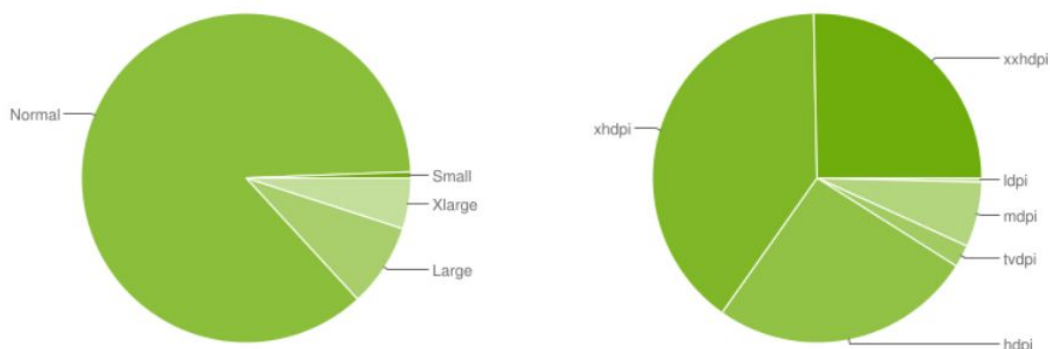
Το πιο σημαντικό θετικό χαρακτηριστικό του Android είναι οι επιλογές που δίνει στον τελικό καταναλωτή. Έχοντας πληθώρα συσκευών ο καταναλωτής μπορεί να αποκτήσει μια συσκευή στα μέτρα του επικεντρώνοντας στο χαρακτηριστικό που έχει πιο πολύ σημασία για αυτόν (κάμερα, οθόνη, μπαταρία κτλ). Το Android επίσης προσφέρει στον τελικό καταναλωτή μια τεράστια γκάμα εφαρμογών για να επιλέξει (η πλειοψηφία των οποίων είναι δωρεάν). Από την πλευρά των προγραμματιστών, η ανάπτυξη εφαρμογών είναι ευκολότερη καθώς μπορούν να προγραμματιστούν από οποιοδήποτε ηλεκτρονικό υπολογιστή (οι εφαρμογές στο iOS απαιτούν Mac για την ανάπτυξή τους) και μπορούν να γραφτούν σε Java ή σε Kotlin με την πρώτη να αποτελεί μια από τις πιο διαδεδομένες γλώσσες προγραμματισμού.

2.4.2 Θετικά του iOS

Το πιο σημαντικό θετικό χαρακτηριστικό του iOS είναι ότι είναι πιο stable σαν λειτουργικό καθώς το λογισμικό είναι φτιαγμένο για να δουλεύει σε έναν μικρό αριθμό συσκευών. Επίσης το λογισμικό τους είναι σχεδόν πάντα up to date καθώς η Apple έχει δεσμευτεί να προσφέρει ενημερώσεις για τουλάχιστον 3 χρόνια. Επιπλέον πολλοί χρήστες έχουν αγαπήσει το οικοσύστημα που τους προσφέρει η Apple, το οποίο στην συνέχεια τους κάνει πιστούς με αποτέλεσμα να αυξάνονται οι πωλήσεις των Apple προϊόντων στο σύνολο. Τέλος οι Apple συσκευές είναι πάντα προσεγμένες τόσο στην εμφάνιση τους όσο και στα χαρακτηριστικά που προσφέρουν στον τελικό καταναλωτή.

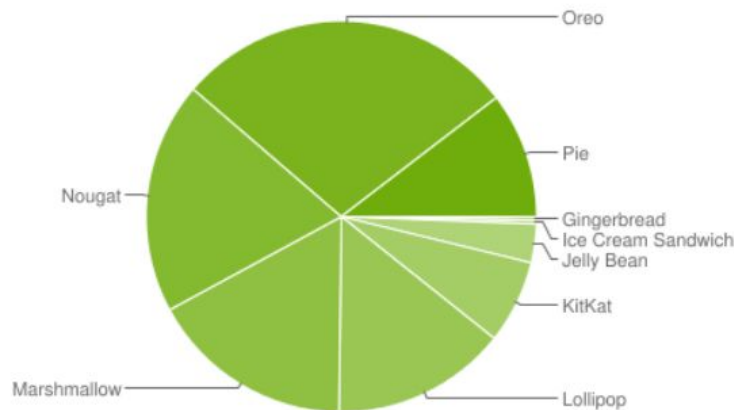
2.4.3 Αρνητικά του Android

Στο Android υπάρχει πλήθος συσκευών με διαφορετικά χαρακτηριστικά όπως για παράδειγμα οθόνες διαφορετικών διαστάσεων και αναλύσεων, και διαφορετικοί αισθητήρες, τα οποία οι προγραμματιστές καλούνται να λαμβάνουν υπόψη κατά την ανάπτυξη των εφαρμογών τους ώστε αυτές να τρέχουν αποτελεσματικά και με ομοιόμορφο τρόπο σε όλες τις συσκευές. Αυτό προϋποθέτει επιπλέον δουλειά από την πλευρά του προγραμματιστή.



Διάγραμμα 4. Screen sizes and densities(<https://developer.android.com/about/dashboards>)

Το μεγαλύτερο αρνητικό του Android σε σχέση με το iOS είναι το πρόβλημα του κατακερματισμού (Fragmentation). Ο κατακερματισμός στα mobile λειτουργικά συστήματα αφορά τις περιπτώσεις στις οποίες οι χρήστες τρέχουν στις συσκευές τους προγενέστερες εκδόσεις του λειτουργικού συστήματος, ενώ άλλοι τρέχουν νεότερες εκδόσεις. Το χάσμα αυτό δημιουργείται διότι οι κατασκευαστές των συσκευών σταματάνε να ενημερώνουν τις συσκευές με νεότερες εκδόσεις λόγω περιορισμών στα χαρακτηριστικά των συσκευών ή λόγω της παλαιότητας τους. Επιπλέον η εξέλιξη του λειτουργικού φέρνει ορισμένες δυνατότητες οι οποίες δεν μπορούν να εφαρμοστούν σε παλαιότερες εκδόσεις με αποτέλεσμα οι προγραμματιστές να μην υποστηρίζουν συσκευές με παλαιότερες εκδόσεις. Αντίθετα στο iOS, η Apple υποστηρίζει για τουλάχιστον 3 χρόνια τις συσκευές της και προβαίνει σε αναβαθμίσεις.



Διάγραμμα 5. Platform versions(<https://developer.android.com/about/dashboards>)

2.4.4 Αρνητικά του iOS

Το iOS από την δική του μεριά, λόγω της κλειστού λογισμικού έχει αρκετούς περιορισμούς που δεν υπάρχουν στο Android. Ο κυριότερος περιορισμός βρίσκεται στη συνδεσιμότητα των συσκευών. Για παράδειγμα, ένας χρήστης Android μπορεί να αποστείλει ένα αρχείο μέσω τεχνολογίας Bluetooth σε ένα άλλο χρήστη, στο iOS αυτή η δυνατότητα είναι κλειδωμένη και αναγκαστικά η αποστολή θα πρέπει να γίνει μέσω cloud εφαρμογών. Επιπλέον, το Android έχει υποστήριξη για τη σύνδεση σκληρών δίσκων για τη μεταφορά δεδομένων χωρίς να είναι απαραίτητη η χρήση υπολογιστή. Ο χρήστης για τη μεταφορά αρχείων, φωτογραφιών και οτιδήποτε άλλου αρχείου θα πρέπει να χρησιμοποιεί το Apple iTunes.

Ταυτόχρονα, ο χρήστης iOS δεν έχει την δυνατότητα να παραμετροποιήσει το γραφικό περιβάλλον. Ο χρήστης περιορίζεται στο να αλλάξει τη σειρά των εφαρμογών και ταπετσαρία, ενώ στο Android υπάρχει η δυνατότητα να τροποποιήσει τη συσκευή του όπως ακριβώς επιθυμεί.

3. Εφαρμογές Ξυπνων τηλεφώνων

Τα smartphones έχουν εισέλθει δυναμικά στην αγορά έχοντας μια σταθερά ανοδική πορεία πωλήσεων με εξαιρετικές προοπτικές επιπλέον ανάπτυξης. Αναπόσπαστο κομμάτι των τηλεφώνων αποτελούν οι εφαρμογές τους (mobile applications) οι οποίες μπορούν να καλύψουν μία μεγάλη γκάμα αναγκών.

Πολλές εταιρίες / προγραμματιστές έχουν ήδη αναπτύξει τις δικές τους εφαρμογές για κινητές συσκευές, προσφέροντας προστιθέμενη αξία στους καταναλωτές, καθώς μια κινητή εφαρμογή μπορεί να εξυπηρετήσει προσωπικές, επαγγελματικές και ενδο-επιχειρηματικές ανάγκες.

3.1 Εργαλεία Ανάπτυξης Εφαρμογών

Η ανάπτυξη εφαρμογών στο περιβάλλον του Android έχει φτάσει πλέον σε ένα ώριμο στάδιο. Αρχικά ο προγραμματιστής έπρεπε να χρησιμοποιήσει το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) Eclipse και κάποιων επιπρόσθετων add-ons για να αναπτύξει εφαρμογές στο περιβάλλον του Android. Στην συνέχεια

μέσω της συνεργασίας της Google με την JetBrains, η οποία έχει κυκλοφορήσει το επιτυχημένο IDE IntelliJ IDEA, κυκλοφόρησε τον Δεκέμβριο του 2014 το Android Studio και αποτελεί μέχρι και σήμερα το κύριο IDE της Google για ανάπτυξη εφαρμογών Android. Η κύρια γλώσσα προγραμματισμού για Android είναι η Java, γεγονός το οποίο βοήθησε το Android δεδομένο το ότι αρκετοί προγραμματιστές ξέρουν να προγραμματίζουν σε αυτή. Θέλοντας η Google να κάνει την γλώσσα προγραμματισμού για το Android πιο ευέλικτη τον Οκτώβριο του 2017 παρουσίασε την Kotlin ως εναλλακτική και τον Μάιο του 2019 ανακοίνωσε πως πλέον η Kotlin είναι η βασική γλώσσα ανάπτυξης σε Android. Όποτε, με λίγα λόγια, ένας προγραμματιστής μπορεί να αναπτύξει εφαρμογές Android σε έκδοση λογισμικού 4.0 έως 10, χρησιμοποιώντας το Android Studio ως IDE, γράφοντας σε Java & Kotlin και διανέμοντας αυτές σε Google Play Store & 3rd Party Stores.

3.2 Τύποι εφαρμογών κινητών συσκευών

Υπάρχουν τρία είδη εφαρμογών για κινητές συσκευές, οι εγγενείς εφαρμογές (Native applications), οι διαδικτυακές εφαρμογές (Web applications) και οι υβριδικές εφαρμογές (Hybrid applications). Κάθε ένας από τους τρόπους έχει τα δικά του μειονεκτήματα και πλεονεκτήματα στα οποία θα αναφερθούμε.

3.2.1 Εγγενείς Εφαρμογές (Native App)

Οι εγγενείς εφαρμογές είναι ο πιο συνηθισμένος τύπος εφαρμογής. Αναπτύσσονται για συγκεκριμένες πλατφόρμες και είναι γραμμένες σε γλώσσες προγραμματισμού που υποστηρίζει η πλατφόρμα. Τόσο η Apple όσο και η Google παρέχουν στους προγραμματιστές τα δικά τους εργαλεία ανάπτυξης, τα στοιχεία διεπαφής και το SDK. Οι περισσότερες εταιρείες επενδύουν σε ανάπτυξη εγγενών εφαρμογών για κινητά λόγω των πλεονεκτημάτων που προσφέρονται σε σύγκριση με άλλους τύπους εφαρμογών.

Μερικά από τα πλεονεκτήματα τους είναι η ταχύτητα και η πολύ καλή απόκριση επειδή είναι κατασκευασμένες για τη συγκεκριμένη πλατφόρμα, η αξιοποίηση διεπαφών προγραμματισμού εφαρμογών (API) και προσθήκη βιβλιοθηκών, οι προγραμματιστές έχουν πρόσβαση στο πλήρες σύνολο των λειτουργιών της πλατφόρμας και δεν απαιτείται συνεχής σύνδεση στο διαδίκτυο.

Το κυρίως αρνητικό που έχουν είναι ότι είναι πιο δύσκολη η ανάπτυξη τους και δεν μπορούν να χρησιμοποιηθούν σε άλλη πλατφόρμα.

3.2.2 Εφαρμογές Ιστού (Web App)

Οι εφαρμογές ιστού για smartphones είναι διαδικτυακές εφαρμογές ειδικά σχεδιασμένες για χρήση σε κινητά τηλέφωνα ή ταμπλέτες και ανοίγουν μέσω του περιηγητή της συσκευής με την επίσκεψή στην αντίστοιχη ηλεκτρονική διεύθυνση (URL) της εφαρμογής. Ωστόσο, οι εφαρμογές ιστού δεν χρειάζονται λήψη από καταστήματα εφαρμογών, όπως οι εγγενείς εφαρμογές. Μια εφαρμογή ιστού δεν αποθηκεύεται στην συσκευή του χρήστη και δεν δεσμεύει χώρο αποθήκευσης. Οι εφαρμογές ιστού δίνουν λύση στον κατακερματισμό, διότι δημιουργούνται για να προσαρμόζονται σε όλα τα είδη οθονών χάρις σε κάποια πλαίσια (framework) και τρέχουν σε διαφορετικά λειτουργικά συστήματα, ανεξάρτητα από τα τεχνικά χαρακτηριστικά της εκάστοτε συσκευής.

Οι εφαρμογές ιστών έχουν αρκετά πλεονεκτήματα κυρίως της εύκολης κατασκευής, μικρότερο κόστος κατασκευής και συντήρησης σε σχέση με τις εγγενείς εφαρμογές. Επίσης δημιουργείται μια εφαρμογή για πολλαπλά λειτουργικά συστήματα, αρκεί η συσκευή να διαθέτει πρόγραμμα περιήγησης.

Οι εφαρμογές ιστού διαθέτουν μια σειρά μειονεκτημάτων. Οι χρήστες θα πρέπει να πληκτρολογήσουν τη διεύθυνση (URL) της εφαρμογής, με την μερία του αυτό ισοδυναμεί με κακή εμπειρία χρήσης. Επιπλέον, οι εφαρμογές ιστού είναι πολύ πιο αργές, λιγότερο διαδραστικές και διαισθητικές συγκριτικά με τις εγγενείς εφαρμογές. Τέλος, παρατηρείται η μη αξιοποίηση βοηθητικού υλικού (hardware) των συσκευών και μικρότερο επίπεδο ασφάλειας.

3.2.3 Υβριδικές Εφαρμογές (Hybrid applications)

Οι υβριδικές εφαρμογές λειτουργούν σε διάφορες πλατφόρμες και συμπεριφέρονται σαν τις εγγενείς εφαρμογές. Πρόκειται για ένα συνδυασμό των εγγενών και των εφαρμογών ιστού. Οι χρήστες μπορούν να τις εγκαταστήσουν στη συσκευή τους όπως σε μια εγγενή εφαρμογή, αλλά στη πραγματικότητα πρόκειται για μια εφαρμογή ιστού.

Αυτές οι εφαρμογές βασίζονται σε JavaScript, HTML και CSS και τρέχουν σε ένα WebView. Μια υβριδική εφαρμογή αποτελείται από δύο μέρη. Το πρώτο κομμάτι είναι το “back-end” που δημιουργείται χρησιμοποιώντας γλώσσες όπως οι HTML, CSS και JavaScript. Το δεύτερο κομμάτι είναι ένα εγγενές κέλυφος το οποίο γίνεται λήψη και φορτώνει τον κώδικα χρησιμοποιώντας το WebView.

Τα πλεονεκτήματα αυτών των εφαρμογών είναι η γρήγορη ανάπτυξη τους, η φθηνότερη υλοποίηση σε σχέση με μια εγγενή εφαρμογή, έχουν πρόσβαση στα εσωτερικά API της συσκευής και το ότι δεν απαιτείται πρόγραμμα περιήγησης σε αντίθεση με μια εφαρμογή ιστού.

Στα μειονεκτήματα τους, οι υβριδικές εφαρμογές είναι πιο αργές και λιγότερο διαδραστικές από τις εγγενείς εφαρμογές, έχουν μειωμένη δυνατότητα παραμετροποίησης σε σχέση με τις εγγενείς εφαρμογές και είναι πιο δύσκολες στην ανάπτυξη τους σε σχέση με τις εφαρμογές ιστού.

3.2.4 Επιλογή κατάλληλης μεθόδου

Η απόφαση για την επιλογή της κατάλληλης μεθόδου ανάπτυξης βασίζεται στο επιχειρηματικό μοντέλο. Εξετάζοντας παράγοντες όπως ο προϋπολογισμός, η πολυπλοκότητα των χαρακτηριστικών που χρειάζεται, η επιθυμητή ποιότητα εμπειρίας χρήσης (User Experience) και ταχύτητα που θα έχει η εφαρμογή.

Οποιαδήποτε μέθοδος και να επιλεγεί, η τελική εφαρμογή θα πρέπει να είναι γρήγορη και αξιόπιστη. Καθώς οι χρήστες με το πέρασμα των χρόνων γίνονται περισσότερο απαιτητικοί από τις εφαρμογές. Τέλος, είναι σημαντικό να υποστηρίζουν μελλοντικές αλλαγές εφόσον χρειαστούν για να ανταπεξέλθουν στον ανταγωνισμό.

4. Λειτουργικό Σύστημα Android και Τεχνολογίες

Το Android είναι ένα λειτουργικό σύστημα ανοιχτού κώδικα βασισμένο σε μια τροποποιημένη έκδοση του πυρήνα του λειτουργικού Linux, το οποίο αρχικά μπορούσε να υποστηρίξει κινητές συσκευές με οθόνη αφής. Πλέον, έχει διευρυνθεί σε πολλούς τομείς και μπορούμε να το βρούμε σε έξυπνα ρολόγια/τηλεοράσεις/tablets/αυτοκίνητα.

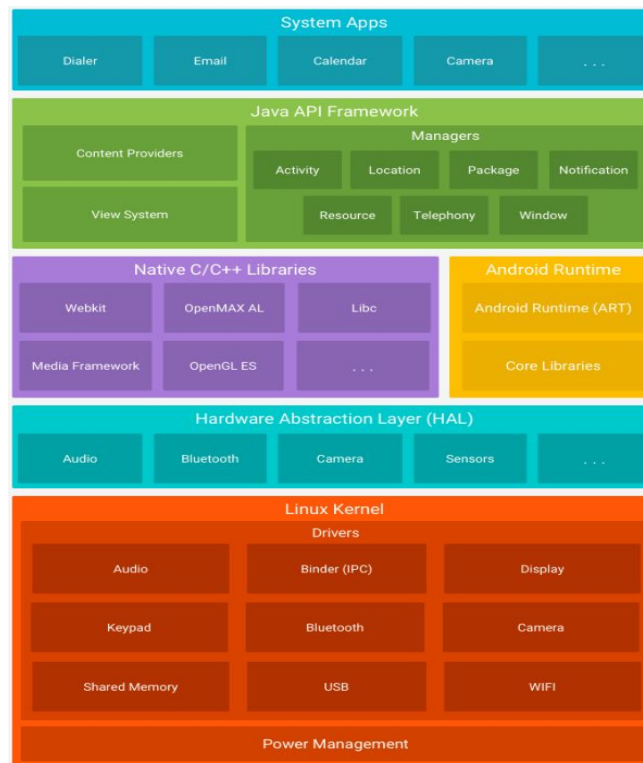
Αρχικά το λειτουργικό είχε αναπτυχθεί από την Android Inc. Το 2007 η Apple παρουσιάζοντας το πρώτο iPhone έδωσε την ώθηση στην Google ώστε να ακολουθήσει τις εξελίξεις της εποχής και προχώρησε στην εξαγορά της Android Inc. Αρχικά αναπτύχθηκε από την Google και αργότερα συνεχίστηκε σε συνεργασία με την Open Handset Alliance η οποία είναι μια κοινοπραξία εταιρειών λογισμικού, κατασκευής hardware και τηλεπικοινωνιών. Η πρώτη επίσημη παρουσίαση του λειτουργικού πραγματοποιήθηκε στις 5 Νοεμβρίου 2007, όπου ανακοινώθηκε η ίδρυση του οργανισμού Open Handset Alliance και η Google έδωσε το μεγαλύτερο κομμάτι του κώδικα στους προγραμματιστές.

Η πρώτη συσκευή που παρουσιάστηκε και στην οποία έγινε διαθέσιμο το λειτουργικό Android ήταν το HTC Dream στις 22 Οκτωβρίου του 2008. Το λειτουργικό είδε μεγάλη άνθηση ιδιαίτερος λόγω των πολλών κατασκευαστών παγκοσμίως και του ανοιχτού λογισμικού. Δίνοντας τη δυνατότητα σε μικρότερους κατασκευαστές να παράγουν χαμηλού κόστους συσκευές με λογισμικό ανοιχτού κώδικα, εύκολα προσαρμόσιμο στα μέτρα τους και το οποίο ενθάρρυνε αρκετούς προγραμματιστές να στραφούν σε αυτό το λειτουργικό.

Το λειτουργικό αξιοποιεί τη γλώσσα προγραμματισμού Java/Kotlin. Μέσω του IDE Android Studio και του λογισμικού ανάπτυξης (Android SDK) ο προγραμματιστής έχει τα απαραίτητα εφόδια για τη δημιουργία σύγχρονων εφαρμογών. Η εξέλιξη του λειτουργικού είναι ραγδαία. Σήμερα είναι στην δέκατη έκδοσή του με σημαντικές βελτιώσεις και προσθήκες λειτουργιών όλα αυτά τα χρόνια.

4.1 Αρχιτεκτονική Συστήματος Android

Το λειτουργικό σύστημα Android χωρίζεται σε ορισμένες συνιστώσες λογισμικού οι οποίες συνθέτουν ένα ολοκληρωμένο σύστημα. Καθένα από αυτά τα επίπεδα μπορεί να αποτελείται από άλλα υποστοιχεία και χρησιμοποιεί τις υπηρεσίες των επιπέδων τα οποία βρίσκονται από κάτω του. Στην παρακάτω εικόνα απεικονίζεται η αρχιτεκτονική του android και φαίνονται ευδιάκριτα τα επίπεδα.



Εικόνα 6. Αρχιτεκτονική του Android.(<https://developer.android.com/>)

4.1.1 Linux Kernel

Το Android βασίζεται στον πυρήνα του Linux και βρίσκεται στο χαμηλότερο επίπεδο της αρχιτεκτονικής του λειτουργικού συστήματος. Η Google το έχει τροποποιήσει κατάλληλα ώστε να διαχειρίζεται θέματα ενέργειας, μνήμης, διεργασιών, drivers. Γενικά ελέγχει ότι σχετίζεται μεταξύ software και hardware ώστε να αλληλεπιδρούν μεταξύ τους.

4.1.2 Hardware Abstraction Layer (HAL)

Το Android έχει ένα επίπεδο μεταξύ του υλικού και του λογισμικού το HAL, μέσω αυτού οι κατασκευαστές υλικού μπορούν να προσθέσουν νέες λειτουργίες χωρίς να επηρεαστούν ή να τροποποιηθούν υψηλότερα επίπεδα του συστήματος. Ένα από τα σημαντικότερα προβλήματα του Android είναι η ενημέρωση των κινητών συσκευών στην τελευταία έκδοση του λογισμικού, διότι υπάρχουν χιλιάδες συσκευές με διαφορετικό λογισμικό και υλικό. Αυτή η διεπαφή θα δώσει τη δυνατότητα στους κατασκευαστές να αναβαθμίσουν ευκολότερα, πιο γρήγορα και χωρίς κόστος τις συσκευές τους στη τελευταία έκδοση του λογισμικού.

4.1.3 Android Runtime

Στο επίπεδο εκτέλεσης περιλαμβάνονται οι βιβλιοθήκες του πυρήνα της Java και του Android ART. Το ART τρέχει σε εκδόσεις μεγαλύτερες του Android 5.0, χρησιμοποιείται για να εκτελέσει εφαρμογές και είναι βελτιστοποιημένο ώστε να λειτουργεί με χαμηλή επεξεργαστική ισχύ και σε συσκευές με μικρή μνήμη RAM. Αναλαμβάνει τη μεταγλώττιση και την δημιουργία εκτελέσιμων αρχείων τύπου .dex τα οποία αναγνωρίζονται και εκτελούνται από το λειτουργικό σύστημα. Η εκτέλεσή τους γίνεται παράλληλα από διαφορετικές μηχανές /νήματα. Σε περίπτωση που παρουσιαστεί σφάλμα σε κάποιο, τα υπόλοιπα συνεχίζουν να εκτελούνται κανονικά.

4.1.4 Native Libraries

Είναι το επίπεδο το οποίο επιτρέπει στη συσκευή να διαχειρίζεται διαφορετικούς τύπους δεδομένων. Οι βιβλιοθήκες είναι γραμμένες σε C και C++, η πρόσβαση γίνεται μέσω του κατάλληλου interface, και

έχουν τροποποιηθεί κατάλληλα διότι προορίζονται για συσκευές με μικρή μνήμη και χαμηλής ενέργειας επεξεργαστές.

4.1.5 Java API Framework

Το Java Framework έχει όλες τις λειτουργίες που είναι απαραίτητες ώστε οι προγραμματιστές να αξιοποιούν πλήρως το υλικό τη συσκευής, να έχουν πρόσβαση σε υπηρεσίες εντοπισμού θέσης, να τρέξουν υπηρεσίες σε νήματα χωρίς να επηρεάζουν τις επιδόσεις της συσκευής, να εμφανίζουν ειδοποιήσεις, να διαχειρίζονται φωνητικές κλήσεις και πολλά ακόμα. Επιπλέον, δίνεται η δυνατότητα σε κάθε εφαρμογή, εφόσον το επιθυμεί ο προγραμματιστής της, να χρησιμοποιούνται δυνατότητες από άλλες εφαρμογές.

4.1.6 System Apps

Το επίπεδο εφαρμογών είναι το ψηλότερο επίπεδο της αρχιτεκτονικής Android. Σε αυτό ακουμπάνε οι εφαρμογές που έρχονται προεγκατεστημένες στη συσκευή όπως οι dialer, η εφαρμογή SMS, ο email client, ο browser και πολλές άλλες. Ο χρήστης έχει τη δυνατότητα να εγκαταστήσει μέσω του εικονικού καταστήματος της Google Play εφαρμογές σύμφωνα με τις ανάγκες του.

4.2 Δομικά Στοιχεία Μιας Εφαρμογής

Κάθε εφαρμογή χρησιμοποιεί πέντε τμήματα κατασκευής ή τουλάχιστον ένα συνδυασμό αυτών. Την τελική απόφαση για το ποια τμήματα θα χρησιμοποιηθούν θα την πάρει ο προγραμματιστής και θα πρέπει να την ορίσει στο αρχείο AndroidManifest.xml. Είναι ένα αρχείο όπου δηλώνονται όλες οι δραστηριότητες, τα δικαιώματα, οι υπηρεσίες, οι προθέσεις και οι πάροχοι περιεχομένου καθώς και ο τρόπος με τον οποίο θα χρησιμοποιηθούν κατά τη διάρκεια ζωής της εφαρμογής. Ακολουθεί η ανάλυση αυτών των τμημάτων.

4.2.1 Δραστηριότητα (Activity)

Αποτελεί το βασικό στοιχείο της εφαρμογής. Είναι μια οθόνη διεπαφής χρήστη, η οποία προβάλλει πληροφορίες. Αποτελείται από κουμπιά, κείμενα, εικόνες και άλλα στοιχεία και ανταποκρίνεται σε γεγονότα. Σε μια εφαρμογή υπάρχουν τόσες δραστηριότητες, όσες και οι διαφορετικές οθόνες που προβάλλονται στο χρήστη. Οι δραστηριότητες συνεργάζονται μεταξύ τους έτσι ώστε να υπάρχει αλληλουχία στη δομή του μενού της εφαρμογής. Ενδεικτικά, η μετάβαση από οθόνη σε οθόνη υλοποιείται με την έναρξη μιας νέας δραστηριότητας, η οποία μπορεί να επιστρέφει κάποια τιμή σε μια προηγούμενη, όπως για παράδειγμα μια δραστηριότητα που εμφανίζει τις επαφές του τηλεφώνου, και επιστρέφει το προφίλ της επιλεγμένης επαφής στην δραστηριότητα που την κάλεσε.

4.2.2 Υπηρεσίες (Services)

Αφορά λειτουργίες που είναι σχεδιασμένες για την επεξεργασία δεδομένων στο παρασκήνιο για μεγάλο χρονικό διάστημα χωρίς κάποια διεπαφή χρήστη. Έχουν την ικανότητα να εμφανίζουν αποτελέσματα ακόμη και όταν η εφαρμογή δεν βρίσκεται στο προσκήνιο. Για παράδειγμα, σε εφαρμογή που ενημερώνει το χρήστη για τα email του, όταν αυτή δεν βρίσκεται στο προσκήνιο, μέσω μιας υπηρεσίας μπορεί να ελέγξει και να ενημερώσει τον χρήστη για το αν δέχτηκε κάποιο email. Η διάρκεια καθώς και ο κύκλος ζωής μιας υπηρεσίας διαφέρει από αυτούς της δραστηριότητας επειδή θα συνεχίσει να τρέχει μέχρι να τη σταματήσει ο χρήστης ή μέχρι να τη σταματήσει το σύστημα για να εξοικονομήσει μνήμη.

4.2.3 Προθέσεις (Intents)

Είναι ο συνδετικός κρίκος μεταξύ των δραστηριοτήτων. Σκοπός τους είναι η επικοινωνία και η μετάβαση από μια δραστηριότητα σε κάποια άλλη. Εξασφαλίζουν επίσης την μεταφορά δεδομένων είτε μεταξύ των δραστηριοτήτων είτε μεταξύ των εφαρμογών. Δεν έχουν την ικανότητα να προβάλλουν μια διεπαφή χρήστη, έχουν την ικανότητα να ενημερώσουν τον χρήστη με την εμφάνιση κάποιας ειδοποίησης για κάποιο γεγονός.

4.2.4 Πάροχοι Περιεχομένου (Content Providers)

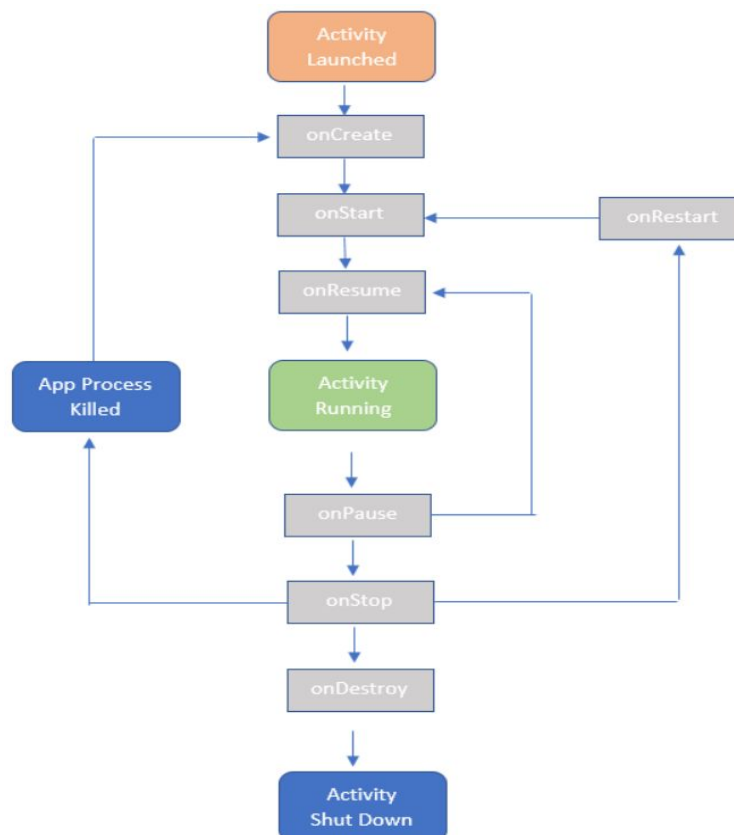
Πρόκειται για ένα στοιχείο του λειτουργικού συστήματος το οποίο διαχειρίζεται αποθηκευτικούς χώρους για δεδομένα τα οποία μπορεί να χρησιμοποιήσει οποιαδήποτε εφαρμογή. Αποτελούν κλάσεις που υλοποιούν ένα σύνολο από μεθόδους και επιτρέπουν στις εφαρμογές να επεξεργαστούν δεδομένα του συγκεκριμένου τύπου. Τα πιο συνηθισμένα δεδομένα τα οποία μοιράζονται είναι οι ενσωματωμένες βάσεις δεδομένων της SQLite.

4.2.5 Δέκτες μηνυμάτων Broadcast(Broadcast Receivers)

Πρόκειται για ένα στοιχείο του λειτουργικού συστήματος το οποίο παρέχει στον χρήστη κατάλληλες πληροφορίες. Για παράδειγμα, τυχόν ειδοποιήσεις από εφαρμογές, το επίπεδο της μπαταρίας, ότι έχει ολοκληρωθεί η λήψη ενός αρχείου κτλ. Από την πλευρά του ο προγραμματιστής έχει και αυτός τη δυνατότητα να παρέχει πληροφορίες στο χρήστη. Επίσης για να αποφύγει να καταναλώνει πόρους από το κύριο νήμα η εφαρμογή, μέσω των Broadcast, μπορεί να τρέχει μια λειτουργία όπως η καταγραφή στίγματος μέσω GPS σε δευτερεύον νήμα και με την ολοκλήρωσή του να επιστρέφει τις συντεταγμένες στην κύρια δραστηριότητα.

4.3 Κύκλος ζωής των δραστηριοτήτων

Η δραστηριότητα (activity) είναι συστατικό μιας εφαρμογής το οποίο παρέχει τις κατάλληλες πληροφορίες σε μια οθόνη, μέσω της οποίας ο χρήστης αλληλεπιδρά. Κάθε φορά στην οθόνη της εφαρμογής εμφανίζεται μία δραστηριότητα. Μια εφαρμογή μπορεί να έχει πολλαπλές δραστηριότητες που είναι συνδεδεμένες μεταξύ τους. Στα προγράμματα που αναπτύσσονται συνήθως υπάρχει η κεντρική δραστηριότητα (main activity) που εμφανίζεται κατά την εκκίνηση της εφαρμογής και μέσω της οποίας ο χρήστης περιηγείται στις άλλες δραστηριότητες. Παρακάτω παρουσιάζονται οι μέθοδοι που καλούνται κατά την διάρκεια ζωής μιας δραστηριότητας.



Εικόνα 7. Κύκλος ζωής δραστηριότητας(<https://www.homeandlearn.co.uk/>)

4.3.1 onCreate

Η μέθοδος καλείται όταν η δραστηριότητα εκτελείται για πρώτη φορά. Σε αυτό το στάδιο αρχικοποιούνται οι βασικές μεταβλητές της δραστηριότητας και οι μέθοδοι που θα κληθούν μια φορά και ό,τι χρειάζεται για να δημιουργηθεί το γραφικό περιβάλλον.

4.3.2 onStart

Αυτή η μέθοδος καλείται πάντα από το σύστημα μετά από την onCreate. Οι δραστηριότητες μπορούν να υπερφορτώσουν αυτήν τη μέθοδο εάν θέλουν να εκτελέσουν συγκεκριμένες εργασίες λίγο πριν γίνει ορατή η δραστηριότητα.

4.3.3 onResume

Σε περίπτωση που η δραστηριότητα «παγώσει» διότι μπορεί να δεχτεί μια εισερχόμενη κλήση, μέσω της onResume όταν τελειώσει η κλήση θα επιστρέψει στη δραστηριότητα και θα επαναφέρει κρίσιμες πληροφορίες που δεν θέλουμε να χάσουμε για παράδειγμα την πρόοδο ενός παιχνιδιού.

4.3.4 onPause

Μέσω της onPause δηλώνουμε στο λειτουργικό σύστημα ότι η δραστηριότητα θα πρέπει να «παγώσει» και θα πρέπει να μπει στο παρασκήνιο ώστε να πάρει τη θέση της εφαρμογής κάποια άλλη και να γίνει αόρατη. Επίσης, μπορούμε να αποθηκεύσουμε στοιχεία ώστε όταν ο χρήστης επιστρέψει αυτά να είναι διαθέσιμα.

4.3.5 onStop

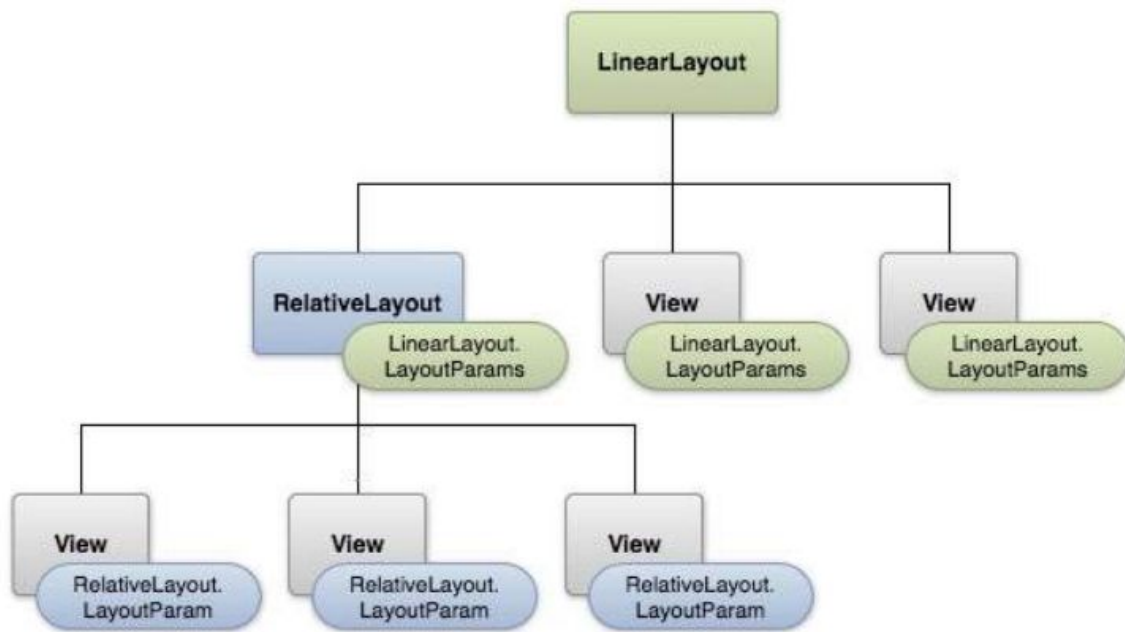
Καλείται όταν η δραστηριότητα δεν είναι πλέον ορατή στον χρήστη. Αυτό μπορεί να συμβεί είτε διότι καταστρέφεται είτε επειδή μια άλλη δραστηριότητα έχει επαναληφθεί και την καλύπτει. Ακολουθεί η μέθοδος onStart () αν η δραστηριότητα επιστρέφει για να αλληλοεπιδράσει με το χρήστη ή ακολουθεί η μέθοδος onDestroy () αν πρόκειται να τερματιστεί.

4.3.6 onDestroy

Αυτή είναι η τελική μέθοδος που καλείται σε ένα κύκλο ζωής μιας δραστηριότητας προτού καταστραφεί και αποδεσμεύσει τη μνήμη. Σε ακραίες καταστάσεις το λειτουργικό μπορεί να σκοτώσει τη διεργασία, η οποία θα οδηγήσει στη μη κλήση της onDestroy.

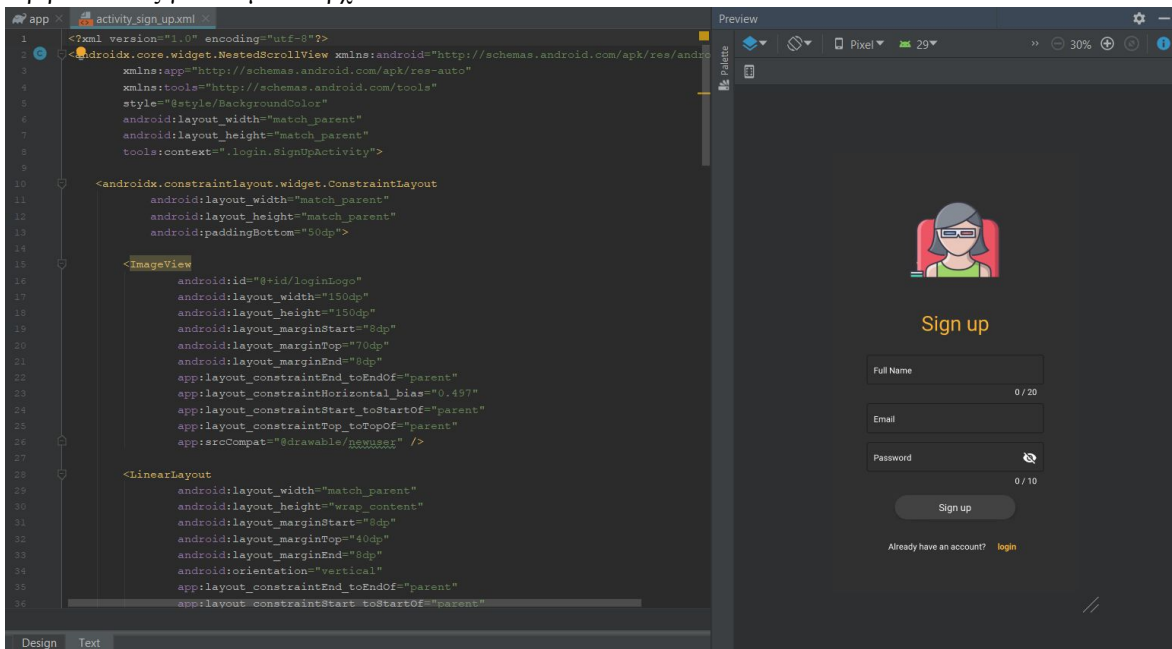
4.4 Διεπαφή χρήστη-User Interface

Η διεπαφή χρήστη περιέχει όλα τα υλικά τα οποία χρησιμοποιούνται από τον προγραμματιστή ώστε να δημιουργήσει το γραφικό περιβάλλον με το οποίο ο χρήστης θα αλληλεπιδρά με την εφαρμογή. Σε αυτό περιέχονται τα αντικείμενα προβολών και οι ομάδες προβολών. Προβολή είναι ένα αντικείμενο το οποίο ζωγραφίζει στοιχεία όπου μπορούμε να αλληλεπιδρούμε όπως ένα κουμπί. Οι ομάδες προβολής είναι ένα αντικείμενο το οποίο μπορεί να κρατάει μεμονωμένες προβολές ή ακόμα και ομάδες προβολής ώστε να δημιουργηθεί το πλαίσιο που θα περιέχει όλα αυτά τα στοιχεία.



Εικόνα 8. Ιεραρχία διεπαφών. (<https://www.tutorialspoint.com/>)

Η διεπαφή χρήστη για κάθε στοιχείο της εφαρμογής έχει οριστεί χρησιμοποιώντας μια ιεραρχία από προβολές και ομάδες προβολών, όπως φαίνεται στην παραπάνω εικόνα. Κάθε ομάδα προβολής είναι ένα αόρατο δοχείο που οργανώνει τις προβολές. Το δέντρο ιεραρχίας μπορεί να είναι τόσο απλό ή σύνθετο στη σχεδίαση, όμως η απλότητα βελτιώνει τις επιδόσεις της εφαρμογής. Η δημιουργία του γραφικού περιβάλλοντος γίνεται μέσω αρχείων XML.



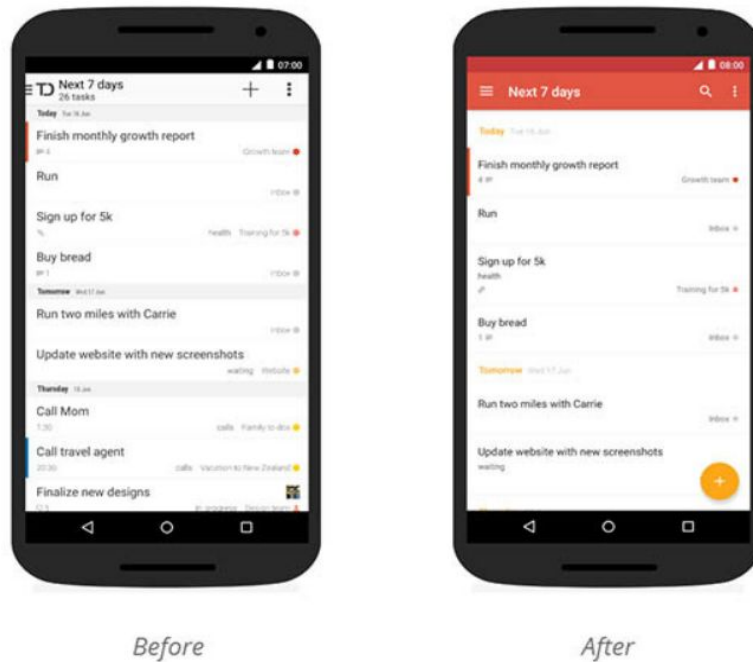
Εικόνα 9. Παράδειγμα XML κώδικα και το οπτικό αποτέλεσμα

4.5 Material Design

Τον Ιούνιο του 2014 η Google ανακοίνωσε το Material Design. Πρόκειται για κανόνες σχετικά με το UI, που κατοχύρωσε η ίδια ώστε οι προγραμματιστές να παρέχουν ένα πιο εύχρηστο και φιλικό περιβάλλον

στο τελικό χρήστη, ακολουθώντας συγκεκριμένα βήματα. Αποτελεί επέκταση του μοντέλου των "καρτών" που είναι σχεδιασμός με βασικό συστατικό στοιχείο τη διάταξη πίνακα (grid layout).

Βασικό χαρακτηριστικό του Material design είναι η ανταπόκριση των γραφικών στις κινήσεις των χρηστών με τη χρησιμοποιώντας διάφορα εφέ, όπως αιώρηση αντικειμένων, σκιές, μεταβαλλόμενη φωτεινότητα, μεταβάσεις γραφικών, αίσθηση βάθους στην οθόνη, έντονα χρώματα, με σκοπό να υπάρχει μια φυσική και λογική αλληλουχία γεγονότων σε τρεις διαστάσεις. Το Material Design έκανε εμφάνιση με την έκδοση Android 5.0.1 (Lollipop).



Εικόνα 10. Gmail App, Πριν και Μετά το Material Design.(<https://cdn57.androidauthority.net/>)

4.6 Kotlin

Η Kotlin είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που σχεδιάστηκε από την ομάδα JetBrains.

Η γλώσσα Kotlin τρέχει πάνω στην Εικονική Μηχανή της Java (JVM). Έχει επηρεαστεί από τις γλώσσες Java, Scala, C#, Groovy, είναι συμβατή με την Java και Android, έχει σχεδιαστεί για να τρέχει σαν εγγενής κώδικας στα iOS και macOS.

Από τον Φεβρουάριο του 2012 η Kotlin αναπτύσσεται ως γλώσσα ανοιχτού κώδικα. Η πρώτη επίσημη έκδοση 1.0 δημοσιεύτηκε τον Φεβρουάριο του 2016. Τον Μάιο του 2017 η Google ανακοίνωσε ότι η Kotlin θα είναι επίσημη γλώσσα του Android.

Η Kotlin αξιοποιεί τις υπάρχουσες βιβλιοθήκες για την Εικονική Μηχανή της Java (JVM), για Android και web browser. Ένα πρόγραμμα Kotlin μπορεί να αναπτυχθεί σε ολοκληρωμένο περιβάλλον ανάπτυξης IntelliJ IDEA, Android Studio, Eclipse, NetBeans, Visual Studio Code.

Ο κώδικας Kotlin μπορεί να περιέχει κώδικα Java και ο κώδικας Java να περιέχει Kotlin. Ο κώδικας Java μετατρέπεται σε κώδικα Kotlin.

Παράδειγμα κώδικα:

```
fun main(args: Array<String>) {
2   println("Hello World!")
3 }
```

4.7 Δικαιώματα

Τον Οκτώβριο του 2015 με την παρουσίαση της έκτης έκδοσης του λειτουργικού συστήματος Android η Google εισήγαγε ένα νέο μοντέλο εκχώρησης δικαιωμάτων στην εφαρμογή, που επιτρέπει στους χρήστες να κατανοήσουν καλύτερα τους λόγους για τους οποίους μια εφαρμογή θα ζητήσει συγκεκριμένα δικαιώματα. Στις προηγούμενες εκδόσεις του λογισμικού ο χρήστης κατά την εγκατάσταση της εφαρμογής αποδέχεται όλα τα δικαιώματα διαφορετικά δεν προχωρούσε η εγκατάσταση.

Πλέον ο χρήστης πράττοντας συγκεκριμένες ενέργειες στην εφαρμογή, όπως για παράδειγμα θα πρέπει να έχει πρόσβαση στην κάμερα η εφαρμογή, τότε θα εμφανιστεί ένα παράθυρο στο οποίο η εφαρμογή ζητά την άδεια από το χρήστη ώστε να μπορέσει να έχει πρόσβαση στη κάμερα της συσκευής του. Τα δικαιώματα χωρίζονται σε δύο κατηγορίες, στα κανονικά και επικίνδυνα δικαιώματα. Τα κανονικά δικαιώματα καλύπτουν τις περιοχές της εφαρμογής όπου χρειάζεται δεδομένα ή πόρους, των οποίων τον ρίσκο είναι μικρό, όπως η ημερομηνία, η κατάσταση του Wi-Fi και πολλά ακόμα.

Τα επικίνδυνα δικαιώματα καλύπτουν τις περιπτώσεις όπου η εφαρμογή χρειάζεται προσωπικά δεδομένα του χρήστη ή που ενδέχεται να επηρεάσουν τα αποθηκευμένα δεδομένα του χρήστη, ακόμα και τη λειτουργία άλλων εφαρμογών.

Λίστα Επικίνδυνων Δικαιωμάτων:

- Ανάγνωση/Εγγραφή στο Ημερολόγιο,
- Πρόσβαση στη Κάμερα.
- Ανάγνωση/Εγγραφή στις Επαφές.
- Πρόσβαση στη Τοποθεσία.
- Εγγραφή ήχων μέσω του μικροφώνου.
- Πρόσβαση στους αισθητήρες.
- Ανάγνωση/Εγγραφή στα Μηνύματα.
- Ανάγνωση/Εγγραφή στον αποθηκευτικό χώρο.

Τα δικαιώματα που χρησιμοποιεί η συγκεκριμένη εφαρμογή είναι η πρόσβαση στο διαδίκτυο και η χρήση της κάμερας (επικίνδυνο δικαίωμα) τα οποία έχουν δηλωθεί στο στο Android Manifest.

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

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.CAMERA" />
```

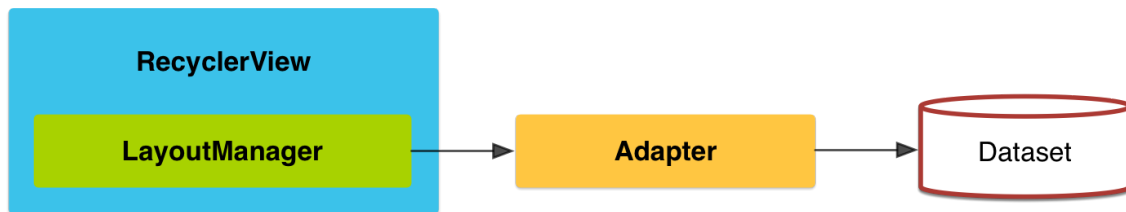
Εικόνα 11. Android Manifest και τα δικαιώματα

4.8 RecyclerView και CardView

Η ανάγκη για βελτίωση του γραφικού περιβάλλοντος χρήσης του λειτουργικού συστήματος, ώθησε τη Google στη δημιουργία ενός framework, που θα δίνει τη δυνατότητα στους προγραμματιστές να δημιουργούν περίπλοκες λίστες με μοντέρνα σχεδίαση. Αυτό επιτεύχθηκε μέσω του RecyclerView σε συνεργασία με το CardView.

RecyclerView

Το RecyclerView είναι μια λίστα με πιο προχωρημένες λειτουργίες και ευελιξία. Η συγκεκριμένη λίστα μπορεί να εμφανίσει ένα μεγάλο σύνολο δεδομένων που η κύλιση γίνεται ομαλά και μπορεί να προστεθούν τα κατάλληλα animations και σκιές. Η χρήση RecyclerView μαζί με CardView έχει αξιοποιηθεί στην παρούσα εφαρμογή όπου χρειάστηκε να δημιουργηθεί μια λίστα ή ένα δυναμικό γραφικό περιβάλλον.



Εικόνα 12. Μοντέλο ενός RecyclerView(<https://developer.android.com/>)

Για τη χρήση ενός RecyclerView απαιτούνται:

- RecyclerView.Adapter: Διαχειρίζεται τα δεδομένα και δεσμεύεται να τα εμφανίσει στη λίστα.
- Layout Manager: Βοηθάει στη τοποθέτηση των στοιχείων.
- ItemAnimator: Παρέχει τα animations για κοινές λειτουργίες όπως η προσθήκη ή η αφαίρεση στοιχείου.

Cardview

Το CardView προστέθηκε στην Πέμπτη έκδοση του Android μαζί με το Material Design. Πρόκειται για μια κάρτα που περιέχει πιο λεπτομερείς πληροφορίες. Μπορεί να περιέχει φωτογραφίες, κείμενα και πλήκτρα. Μπορεί να εμφανίζεται περιεχόμενο που περιέχει στοιχεία με διαφορετικό μέγεθος, όπως φωτογραφίες με λεζάντες μεταβλητού μήκους.

4.9 Google Firebase

Το Firebase είναι μια cloud υπηρεσία, μέσα από την οποία οι προγραμματιστές μπορούν προσθέσουν χαρακτηριστικά στις εφαρμογές ανάλογα με τις ανάγκες τους. Η εταιρεία Firebase ιδρύθηκε το 2011 στην Καλιφόρνια, το αρχικό προϊόν ήταν μια βάση δεδομένων πραγματικού χρόνου, που παρέχει ένα API το οποίο επιτρέπει στους προγραμματιστές να συγχρονίζουν δεδομένα μεταξύ πολλαπλών συσκευών. Η εταιρεία εξαγοράστηκε από την Google το 2014 και σήμερα έχει επεκταθεί στην ανάπτυξη μιας πλήρους σουίτας ανάπτυξης εφαρμογών. Στην εφαρμογή που δημιουργήθηκε έχει γίνει εκτεταμένη χρήση των λειτουργιών της Firebase. Κύριο χαρακτηριστικό της είναι η ενσωμάτωσή της στο λειτουργικό σύστημα Android, οι υπηρεσίες που προσφέρει αλλά και η δωρεάν διάθεσή της μέχρι ένα συγκεκριμένο όριο χρήσης.

4.9.1 Realtime Database

Είναι μία βάση δεδομένων NoSQL αποθηκευμένη στο Cloud σε μορφή JSON. Η υπηρεσία παρέχει στους developers ένα API μέσω αυτού, μπορούν να ενημερώσουν τη βάση δεδομένων σε οποιαδήποτε συσκευή που τρέχει την εφαρμογή σε πραγματικό χρόνο. Τα δεδομένα παραμένουν αποθηκευμένα ακόμα και εάν η εφαρμογή ξαφνικά χάσει την πρόσβαση της στο διαδίκτυο. Επίσης παρέχει βιβλιοθήκες που επιτρέπουν την ενσωμάτωση σε εφαρμογές Android, iOS, JavaScript, Java, Objective-C, swift και Node.js.

Η πρόσβαση στη βάση μπορεί να περιοριστεί από τον προγραμματιστή μέσα από κανόνες ασφάλειας. Τα δεδομένα αποθηκεύονται ως αντικείμενα JSON, δεν περιέχει δηλαδή πίνακες και εγγραφές όπως αναφέρθηκε και η διαχείρισή της γίνεται μέσω δέντρων JSON.

```

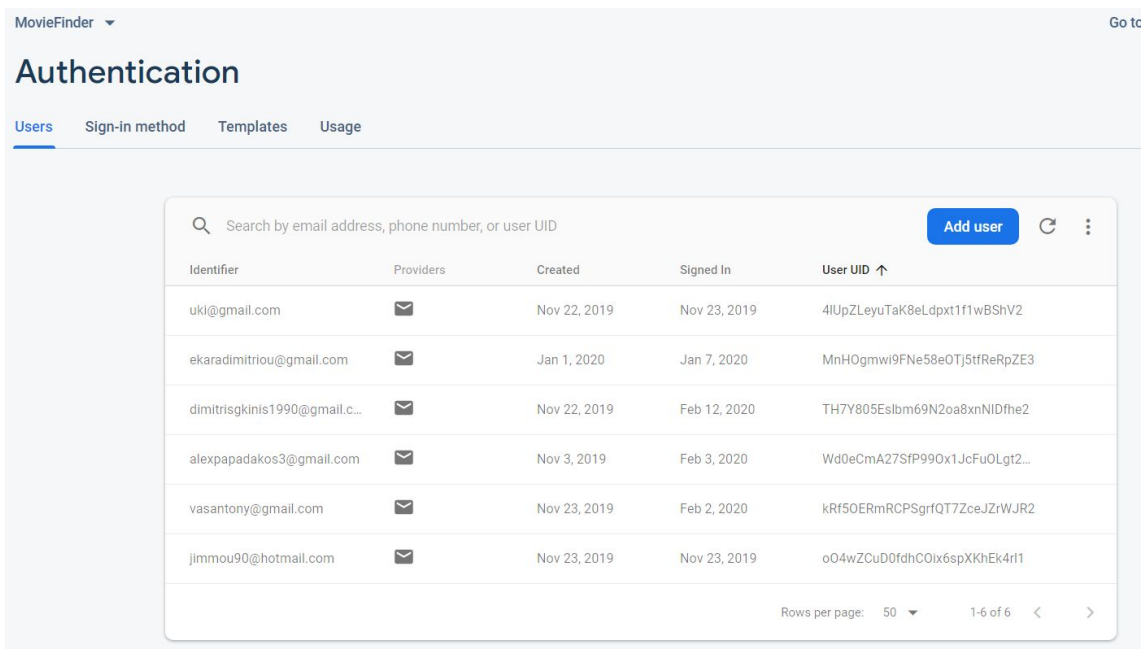
1  {
2  /* Visit https://firebase.google.com/docs/
3  "rules": {
4    ".read": true,
5    ".write": true
6  }
7  }

```

Εικόνα 13. Κανόνες Realtime Database

4.9.2 Firebase Authentication

Η Firebase Authentication είναι μια επέκταση της Firebase η οποία επιτρέπει στους προγραμματιστές να διαχειριστούν τους χρήστες της εφαρμογής τους. Εδώ ο προγραμματιστής μπορεί να δημιουργήσει καινούργιο χρήστη καθώς και να τον απενεργοποιήσει προσωρινά ή και μόνιμα. Μεσω του Firebase Authentication μπορούν να οριστούν τρόποι με του οποίους θα γίνεται επαλήθευση χρήστη, για παράδειγμα μέσω email, Google account, Facebook, κινητού τηλεφώνου και πολλών άλλων. Τέλος, εδώ καθορίζεται και ο τρόπος με τον οποίο θα γίνεται η επαναφορά κωδικού ή η επαλήθευση email και κινητού τηλεφώνου. Για την τρέχουσα εφαρμογή έχει χρησιμοποιηθεί το email ως επαλήθευση χρήστη και για επαναφορά κωδικού.



The screenshot shows the 'Users' tab in the Firebase Authentication console. It features a search bar at the top with the text 'Search by email address, phone number, or user UID' and an 'Add user' button. Below the search bar is a table with the following columns: Identifier, Providers, Created, Signed In, and User UID. The table contains six rows of user data.

| Identifier | Providers | Created | Signed In | User UID |
|-------------------------------|-----------|--------------|--------------|-------------------------------|
| uki@gmail.com | ✉ | Nov 22, 2019 | Nov 23, 2019 | 4lUpZLeyuTaK8eLdpXt1f1wBShV2 |
| ekaradimitriou@gmail.com | ✉ | Jan 1, 2020 | Jan 7, 2020 | MnHOgmwi9FNe58e0Tj5tfReRpZE3 |
| dimitrisgkinis1990@gmail.c... | ✉ | Nov 22, 2019 | Feb 12, 2020 | TH7Y805Eslbm69N2oa8xnNIDfhe2 |
| alexpapadacos3@gmail.com | ✉ | Nov 3, 2019 | Feb 3, 2020 | Wd0eCmA27SfP990x1JcFu0Lgt2... |
| vasantony@gmail.com | ✉ | Nov 23, 2019 | Feb 2, 2020 | kRf50ERmRCPsgrfQT7ZceJZrWJR2 |
| jimmou90@hotmail.com | ✉ | Nov 23, 2019 | Nov 23, 2019 | o04wZCuD0fdhCOix6spXKhEk4r1 |

At the bottom right of the table, there is a 'Rows per page' dropdown set to 50 and a pagination indicator showing '1-6 of 6'.

Εικόνα 14. Κονσόλα Firebase Authentication

4.9.3 ML Kit

Πρόκειται για μια συλλογή από βιβλιοθήκες της Firebase που παρουσιάστηκαν τον Μάιο του 2018 και παρέχουν έξυπνες λειτουργίες ΑΙ στους προγραμματιστές. Μερικά παραδείγματα λειτουργιών είναι η αναγνώριση κειμένου, η αναγνώριση προσώπου, η αναγνώριση αντικειμένων, αυτόματη μετάφραση,

σκανάρισμα Barcode και άλλα. Στην τρέχουσα εφαρμογή χρησιμοποιήθηκε η βιβλιοθήκη για την αναγνώριση κειμένου, έτσι ώστε ο χρήστης να μπορεί να κάνει αναζήτηση από ένα πόστερ ταινίας.

5. Ανάλυση των λειτουργιών της Εφαρμογής

Στο κεφάλαιο αρχικά θα αναφερθούμε στην ανάλυση απαιτήσεων, το σχεδιασμό της εφαρμογής και της βάσης δεδομένων. Ολοκληρώνοντας το κεφάλαιο με τον τρόπο υλοποίησης της.

5.1 Ανάλυση Απαιτήσεων

Στην παρούσα ενότητα γίνεται περιγραφή της ανάλυσης απαιτήσεων της εφαρμογής που αναπτύχθηκε. Συνοπτικά, μέσω της εφαρμογής ο χρήστης θα έχει τη δυνατότητα να αναζητήσει ταινίες που τον ενδιαφέρουν, να ενημερωθεί για λεπτομέρειες σχετικά με την ταινία, να αποθηκεύσει αγαπημένες ταινίες καθώς και ταινίες που επιθυμεί να δει στο μέλλον, να συνομιλήσει με άλλους χρήστες της εφαρμογής. Επίσης υπάρχει η δυνατότητα να παρακολουθήσει τα στατιστικά χρήσης του. Παρακάτω θα παραθέσουμε τις απαιτήσεις που συγκεντρώθηκαν.

Οι γενικότερες απαιτήσεις του συστήματος:

- Φιλικό και εύχρηστο γραφικό περιβάλλον στο χρήστη.
- Γρήγορη ταχύτητα εφαρμογής
- Αξιοπιστη εφαρμογή μέσω της επιδιόρθωσης των σφαλμάτων
- Χρήση cloud για μεγαλύτερη ευελιξία
- Δυνατότητα χρήσης εφαρμογής στα Ελληνικά και στα Αγγλικά
- Υποστήριξη διαφορετικών συσκευών ανάλογα με τα τεχνικά χαρακτηριστικά, την έκδοση του λογισμικού και το μέγεθος της οθόνης.
- Χρήση των πλέον σύγχρονων αρχιτεκτονικών και βιβλιοθηκών.
- Δημιουργία λογαριασμού χρήστη και επεξεργασία των στοιχείων.
- Προσθήκη των αγαπημένων ταινιών.
- Προσθήκη ταινιών προς παρακολούθηση.
- Δυνατότητα κοινοποίησης ταινιών και ηθοποιών.
- Ασφάλεια των δεδομένων
- Κατάλληλο optimization για δεδομένα
- Δημιουργία analytics για τους χρήστες
- Χρήση material components της Google

Η εφαρμογή έχει αναπτυχθεί ώστε να μπορεί εκτελεστεί από συσκευές με έκδοση μεγαλύτερη του λειτουργικού συστήματος Android 8.0. Έχει χρησιμοποιηθεί cloud βάση δεδομένων ώστε να αποθηκεύονται όλες οι απαραίτητες πληροφορίες. Επίσης η cloud βάση διαχειρίζεται όλη την διαδικασία της δημιουργίας και ταυτοποίησης των επιμέρους χρηστών. Επίσης αξιοποιούνται οι τεχνολογίες Retrofit και RxJava2 για να γίνει το fetch των δεδομένων. Έχει γίνει εκτενείς χρήση RecyclerView για την καλύτερη περιήγηση του χρήστη. Τέλος έχει γίνει εκτενείς χρήση της κληρονομικότητας και των κεντρικών styles επιτυγχάνοντας έτσι λιγότερη επανάληψη κώδικα και ευκολία στην παραμετροποίηση του.

5.2 Σχεδιασμός

Με βάση την ανάλυση απαιτήσεων που προαναφέρθηκε σχετικά με την εφαρμογή και τους χρήστες, θα συνεχίσουμε ενδεικτικά παραδείγματα από τον σχεδιασμό της εφαρμογής.

5.2.1 Δυνατότητες Λειτουργιών των Χρηστών

Η εφαρμογή προσφέρει μια πληθώρα δυνατοτήτων στους χρήστες. Σε αυτό το κεφάλαιο θα γίνει μια συνοπτική αναφορά αυτών.

Οι λειτουργίες που είναι διαθέσιμες είναι:

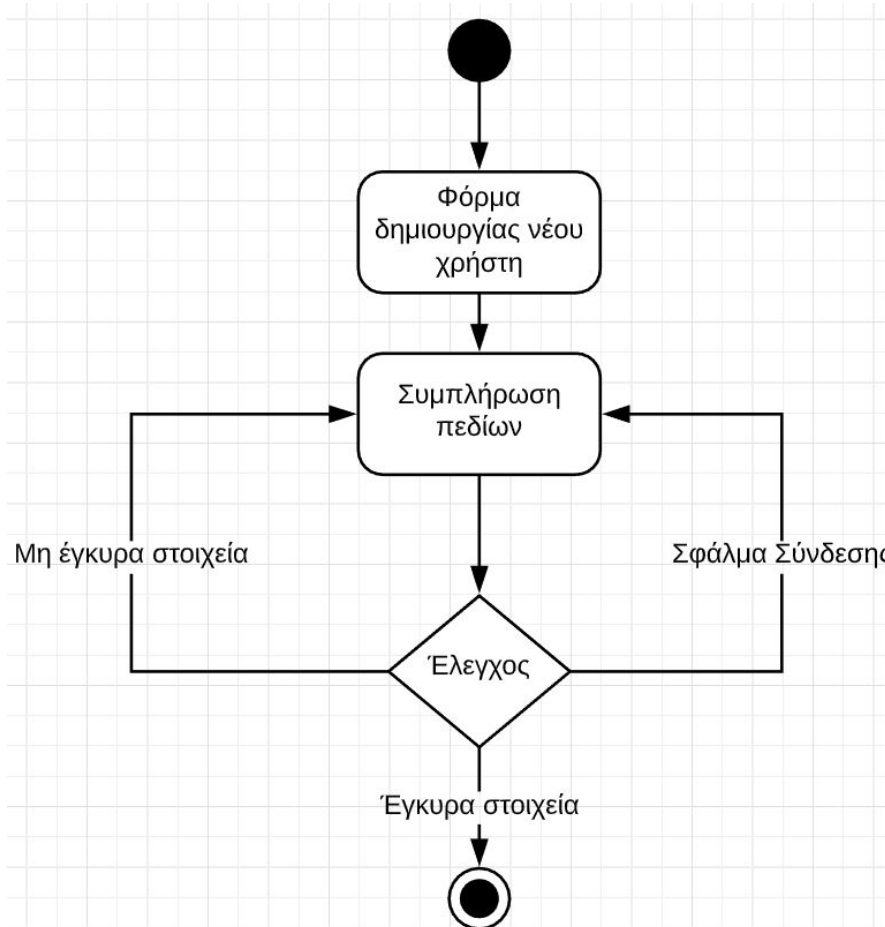
- Δημιουργία νέου χρήστη με την χρήση e-mail
- Οθόνη σύνδεσης
 - Λειτουργία “Να με θυμάσαι”
 - Λειτουργία ανάκτησης κωδικού
- Κεντρική οθόνη
 - Μήνυμα καλωσορίσματος
 - Δημοφιλείς, κορυφαίες, θα κυκλοφορήσουν σύντομα και παίζονται τώρα ταινίες οι οποίες επιλέγοντας μια ταινία μεταφέρει τον χρήστη στην σελίδα με τις λεπτομέρειες για την ταινία
 - floating action button
 - back to top button
- Λεπτομέρειες για την ταινία
 - Εμφάνιση λεπτομερειών για την ταινία
 - Εμφάνιση σχετικών ταινιών
 - Εμφάνιση ηθοποιών, η επιλογή του ηθοποιού μεταφέρει τον χρήστη στην σελίδα με τις λεπτομέρειες για τον ηθοποιό
 - Αποθήκευση ταινίας στην λίστα παρακολούθησης
 - Αποθήκευση ταινίας στα αγαπημένα
 - Κοινοποίηση ταινίας
- Λεπτομέρειες για τον ηθοποιό
 - Εμφάνιση λεπτομερειών για τον ηθοποιό
 - Εμφάνιση ταινιών για τις οποίες είναι γνωστός
 - Κοινοποίηση ηθοποιού
- Το floating action button έχει τις παρακάτω δυνατότητες
 - Αναζήτηση ταινίας
 - Ανακάλυψη ταινίας
 - Αναζήτηση ταινίας μέσω κάμερας
 - Το προφίλ μου
 - Αποσύνδεση
- Αναζήτηση ταινίας, αναζήτηση βάση του τίτλου
 - Τα αποτελέσματα εμφανίζονται σε RecyclerView
 - Υποστήριξη pagination για πολλές σελίδες αποτελεσμάτων
 - Η επιλογή της ταινίας μεταφέρει τον χρήστη στην σελίδα με τις λεπτομέρειες για την ταινία
- Ανακάλυψη ταινίας, αναζήτηση βάση είδους και χρονολογίας
 - Τα αποτελέσματα μπορούν να εμφανιστούν βάση διάφορων επιλογών (βάση βαθμολογίας, βάση δημοτικότητας κτλ)
 - Τα αποτελέσματα εμφανίζονται σε RecyclerView
 - Η επιλογή της ταινίας μεταφέρει τον χρήστη στην σελίδα με τις λεπτομέρειες για την ταινία
- Αναζήτηση ταινίας μέσω κάμερας
 - Η αναζήτηση γίνεται με αναγνώριση κειμένου από την κάμερα
 - Τα αποτελέσματα εμφανίζονται σε RecyclerView
 - Η επιλογή της ταινίας μεταφέρει τον χρήστη στην σελίδα με τις λεπτομέρειες για την ταινία
- Το προφίλ μου
 - Λίστα παρακολούθησης
 - Λίστα αγαπημένων
 - Στατιστικά χρήσης
 - Ρυθμίσεις
 - Συνομιλία
 - Σχετικά με την εφαρμογή

- Ρυθμίσεις
 - Λειτουργία αλλαγής ονόματος
 - Λειτουργία αλλαγής tag
 - Λειτουργία αλλαγής κωδικού
 - Λειτουργία αλλαγής φωτογραφίας χρήστη
 - Λειτουργία αλλαγής dark/light mode

5.2.2 Σχεδιασμός της Διαδικασίας Σύνδεσης και Εγγραφής του Χρήστη

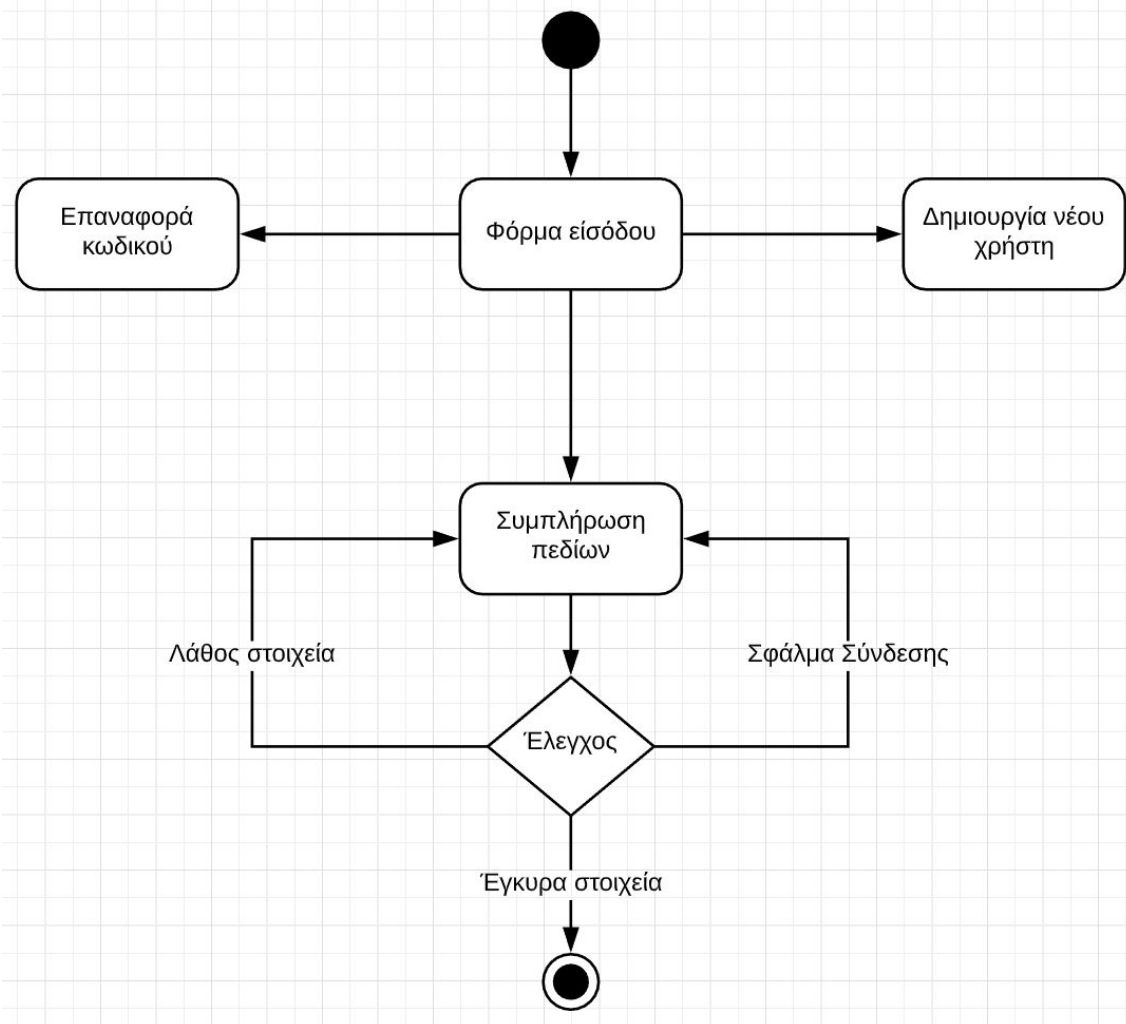
Στη ανάλυση απαιτήσεων είδαμε την αναγκαιότητα για εγγραφή και σύνδεση του στην εφαρμογή, είναι σκόπιμο να δοθεί προσοχή στον σχεδιασμό της διαδικασίας που θα ακολουθείται κατά τη σύνδεση του ώστε να αποφευχθούν τυχόν λάθη.

Η εγγραφή του χρήστη μπορεί να γίνει απευθείας μέσω της εφαρμογής. Τα στοιχεία που θα χρειαστεί να καταχωρήσει είναι το ηλεκτρονικό ταχυδρομείο, ένα όνομα και το κωδικός ασφαλείας. Όλα τα στοιχεία αποθηκεύονται στην cloud βάση δεδομένων, ο κωδικός είναι κρυπτογραφημένος ώστε να τον γνωρίζει ο χρήστης και μόνο. Στην εγγραφή ζητούνται τα παραπάνω στοιχεία, σε περίπτωση όπου ο χρήστης δεν έχει καταχωρήσει κάποιο πεδίο, ο χρήστης λαμβάνει κατάλληλο μήνυμα ώστε να συμπληρώσει το κενό πεδίο και να προχωρήσει την διαδικασία. Κατά τη διάρκεια της εγγραφής γίνεται έλεγχος εάν το ηλεκτρονικό ταχυδρομείο έχει τη σωστή μορφή και δεν το χρησιμοποιεί κάποιος άλλος χρήστης. Ειδάλλως εμφανίζεται ένα μήνυμα λάθους και τον ενημερώνει να αλλάξει τα στοιχεία με διαφορετικά ώστε να ολοκληρωθεί η εγγραφή. Στο παρακάτω διάγραμμα ακολουθίας περιγράφεται η διαδικασία εγγραφής.



Διάγραμμα 6. Διάγραμμα Δραστηριοτήτων Εγγραφής Χρήστη.

Έχοντας πλέον δημιουργήσει τον χρήστη μπορούμε πλέον να προχωρήσουμε στην σύνδεση του. Τα απαραίτητα πεδία είναι ο λογαριασμός του ηλεκτρονικού ταχυδρομείου καθώς και ο κωδικός του χρήστη. Υπάρχει η επιλογή επαναφοράς του κωδικού, στην περίπτωση που ο χρήστης τον ξεχάσει, καθώς και η λειτουργία “να με θυμάσαι” η οποία αποθηκεύει τον λογαριασμό του ηλεκτρονικού ταχυδρομείου, έτσι ώστε να μειωθεί ο χρόνος και τα βήματα που απαιτούνται για την σύνδεση. Σε περίπτωση όπου ο χρήστης δεν έχει καταχωρήσει κάποιο πεδίο, ο χρήστης λαμβάνει κατάλληλο μήνυμα ώστε να συμπληρώσει το κενό πεδίο και να προχωρήσει την διαδικασία. Στην συνέχεια γίνεται έλεγχος αν τα στοιχεία είναι έγκυρα και αν είναι ο χρήστης μεταφέρεται στην επόμενη οθόνη αλλιώς εμφανίζεται ένα μήνυμα λάθους. Στο παρακάτω διάγραμμα ακολουθίας περιγράφεται η διαδικασία σύνδεσης.



Διάγραμμα 7. Διάγραμμα Δραστηριοτήτων Εισόδου Χρήστη.

5.2.3 Σχεδιασμός της λειτουργίας “Λεπτομέρειες για την ταινία”

Η λειτουργία “Λεπτομέρειες για την ταινία” αποτελεί ίσως την πιο πολύπλοκη διαδικασία της εφαρμογής. Σε αυτήν την λειτουργία γίνονται κλήσεις τόσο στο TMDB όσο και στην Firebase.

Ενδεικτικά οι κλήσεις είναι:

- Έλεγχος αν η ταινία είναι στα αγαπημένα, μέσω Firebase
- Έλεγχος αν η ταινία είναι στη λίστα παρακολούθησης, μέσω Firebase
- Φόρτωση λεπτομερειών, μέσω TMDB
- Φόρτωση συνδέσμων, μέσω TMDB
- Φόρτωση ηθοποιών, μέσω TMDB
- Φόρτωση παρόμοιων ταινιών, μέσω TMDB

Η σχεδίαση των κλήσεων της FireBase είναι σχετικά απλή καθώς προσφέρει στον προγραμματιστή έτοιμες βιβλιοθήκες για αυτόν το σκοπό. Για κλήσεις εκτός FireBase έγινε χρήση των βιβλιοθηκών Retrofit και RxJava.

5.2.4 Σχεδιασμός της Firebase Βάσης Δεδομένων

Επιλέχθηκε η Firebase για να φιλοξενήσει τη βάση δεδομένων διότι πρόκειται για real time βάση, δίνοντας τη δυνατότητα για άμεσες και αποτελεσματικές αλλαγές, διαθέτει αρκετές μεθόδους υποβολής ερωτημάτων (querying), μεγάλη αξιοπιστία και ασφάλεια των δεδομένων με την υποστήριξη της Google. Σε κάθε καινούργιο χρήστη του δίνεται ένα unique id και μέσα σε αυτό αποθηκεύονται όλες οι απαραίτητες πληροφορίες.

Αναφορικά οι πληροφορίες που αποθηκεύονται για τον χρήστη είναι:

- Το ηλεκτρονικό ταχυδρομείο
- Η λίστα παρακολούθησης
- Η λίστα αγαπημένων
- Το url της φωτογραφίας του χρήστη
- Τα στατιστικά του
- Η ετικέτα που έχει επιλέξει
- Ο μοναδικός κωδικός
- Το όνομα

Η Firebase χρησιμοποιήθηκε επίσης και για την υλοποίηση της ομαδικής συνομιλίας, το γεγονός ότι είναι real time βάση την καθιστά κατάλληλη για αυτή την χρήση.

Αναφορικά οι πληροφορίες που αποθηκεύονται για την ομαδική συνομιλία είναι:

- Το ηλεκτρονικό ταχυδρομείο του αποστολέα
- Το μήνυμα
- Το timestamp του μηνύματος



Εικόνα 15. Παράδειγμα χρήστη στην κονσόλα της Firebase.

6. Παρουσίαση της Εφαρμογής

Σε αυτό το κεφάλαιο θα παρουσιάσουμε την λειτουργικότητα της εφαρμογής και θα δούμε στην πράξη τις δυνατότητες που έχει ο χρήστης. Βοηθητικά θα παρουσιαστούν στιγμιότυπα (screenshots) για μεγαλύτερη ευκολία στην κατανόηση, η παρουσίαση αυτή χρησιμεύει και ως εγχειρίδιο χρήστη (user manual).

Σκοπός της μεταπτυχιακής εργασίας είναι παροχή μιας εφαρμογής που θα βελτιώσει την ποιότητα αναζήτησης ταινιών, θα κάνει καλύτερη την αναζήτηση πληροφοριών, και θα δώσει τη δυνατότητα στον χρήστη να αποθηκεύσει τις αγαπημένες του ταινίες καθώς και αυτές που έχει σκοπό να δει. Επίσης με την εφαρμογή θα μπορέσει να συνομιλήσει με χρήστες, και όλοι μαζί θα μπορούν να ανταλλάξουν άμεσα ιδέες και απόψεις ταινίες της προτίμησής τους.

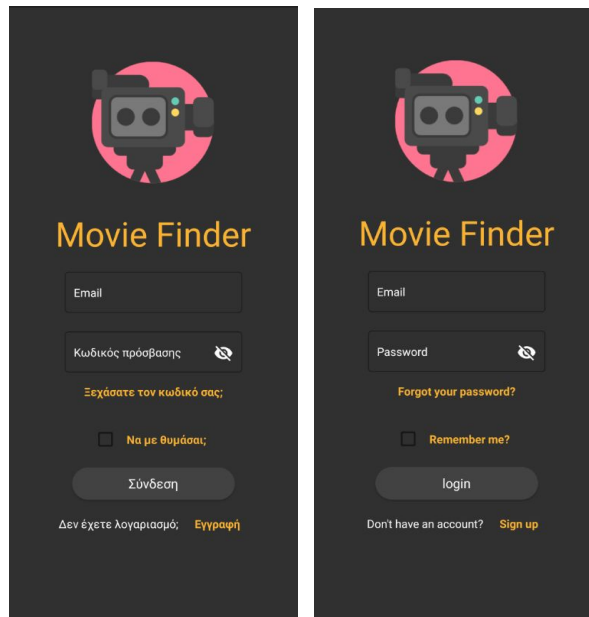
Όλα τα παραπάνω, πλέον, υλοποιούνται σε μια εφαρμογή χωρίς ο χρήστης να χρειάζεται να έχει πολλές διαφορετικές εφαρμογές εγκατεστημένες στη συσκευή του. Αυτό μειώνει τη πολυπλοκότητα. Μέσω μιας μοντέρνας και με απλή σχεδίαση εφαρμογής επιτυγχάνουμε την πλέον ευχάριστη και εύχρηστη εμπειρία στο τελικό χρήστη.

Πριν ξεκινήσουμε την παρουσίαση της εφαρμογής αξίζει να σημειωθεί ότι η εφαρμογή υποστηρίζει πλήρως Ελληνικά και Αγγλικά. Η παρουσίαση θα αποτελείται κυρίως από Screenshots της εφαρμογής στα Αγγλικά.

6.1 Αρχική Διεπαφή

Κατά την είσοδο του χρήστη στην εφαρμογή του, μέσω ενός animation εμφανίζεται η σελίδα σύνδεσης. Ο χρήστης εδώ έχει την δυνατότητα να συνδεθεί στην εφαρμογή, να ανακτήσει τον κωδικό του σε περίπτωση που τον έχει ξεχάσει καθώς και να δημιουργήσει χρήστη στην περίπτωση που δεν έχει. Τα πεδία “email” και “password” είναι υποχρεωτικά και σε περίπτωση που δεν συμπληρωθούν εμφανίζεται το σχετικό μήνυμα λάθους. Στην περίπτωση που ο χρήστης έχει ξεχάσει τον κωδικό του του δίνεται η δυνατότητα να ανακτήσει την πρόσβασή του μέσα από την λειτουργία “ξεχάσατε τον κωδικό σας” η οποία, μέσω της Firebase στέλνει ένα email στον χρήστη με έναν σύνδεσμο επαναφοράς κωδικού.

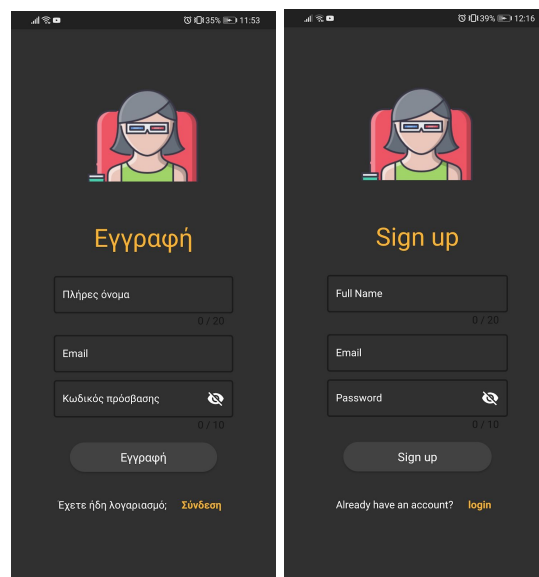
Δεδομένο ότι ο χρήστης συμπλήρωσε τα απαραίτητα πεδία και πατήσει το κουμπί “σύνδεση” η εφαρμογή επικοινωνεί με την Firebase για την ταυτοποίηση, σε περίπτωση που τα στοιχεία δεν είναι σωστά εμφανίζεται ένα μήνυμα λάθους, αν τα στοιχεία είναι σωστά ο χρήστης μεταβαίνει στην αρχική οθόνη (Home Page). Σε περίπτωση που ο χρήστης έχει επιλέξει το checkbox “να με θυμάσαι” πριν πατήσει το κουμπί “σύνδεση” θα θυμάται το email το χρήστη τις επόμενες φορές που θα ανοίξει την εφαρμογή. Σε περίπτωση που ο χρήστης δεν έχει κάνει εγγραφή με την επιλογή του συνδέσμου “Εγγραφή” μπορεί να δημιουργήσει έναν νέο χρήστη.



Εικόνα 16. Αρχική Διεπαφή σε Ελληνικά και Αγγλικά

6.2 Οθόνη Εγγραφής Χρήστη

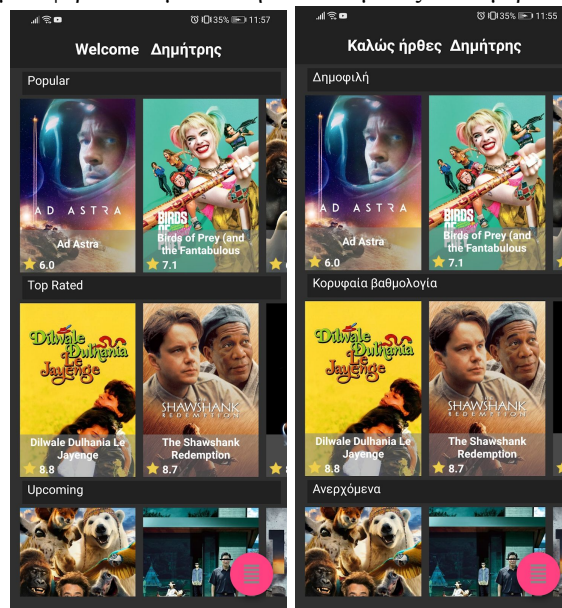
Σε αυτή την οθόνη μπορούμε να δημιουργήσουμε έναν καινούργιο χρήστη. Η οθόνη αυτή έχει τρία υποχρεωτικά πεδία και έναν σύνδεσμο για να γυρίσει ο χρήστης στην οθόνη Σύνδεσης. Τα τρία υποχρεωτικά πεδία είναι το όνομα χρήστη, το ηλεκτρονικό του ταχυδρομείο και ο προσωπικός του κωδικός. Σε αυτά τα πεδία γίνονται κάποιοι επιπλέον έλεγχοι, το όνομα θα πρέπει να είναι μέχρι είκοσι χαρακτήρες, το ηλεκτρονικό ταχυδρομείο θα πρέπει να έχει format email και να μην χρησιμοποιείται ήδη και ο προσωπικός κωδικός να είναι από έξι έως δέκα χαρακτήρες. Σε περίπτωση που κάποιο από αυτά τα κριτήρια δεν πληρούται θα εμφανίζεται ένα σχετικό μήνυμα λάθους. Σε περίπτωση που τα στοιχεία είναι εντάξει θα εμφανίζεται ένα μήνυμα επιτυχίας και ο χρήστης θα μεταφέρεται αυτόματα στην οθόνη Σύνδεσης.



Εικόνα 17. Οθόνη Εγγραφής Χρήστη

6.3 Home Page

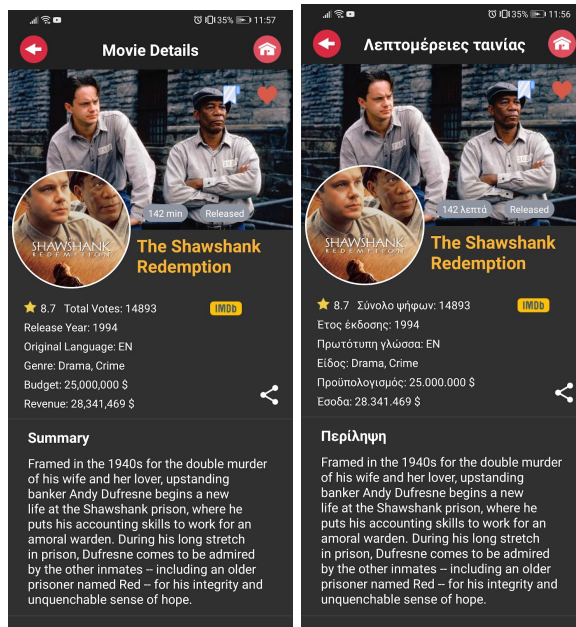
Η Home Page αποτελεί την κεντρική σελίδα της εφαρμογής. Στον χρήστη παρουσιάζεται ένα μήνυμα που τον καλωσορίζει, το οποίο αποτελείται από “Καλώς ήρθες” συν το όνομα που έχει δηλώσει ο χρήστης. Πιο κάτω παρουσιάζονται 4 κατηγορίες ταινιών δημοφιλείς, κορυφαίες, θα κυκλοφορήσουν σύντομα και παίζονται τώρα. Οι κατηγορίες εμφανίζονται σε RecyclerView οπότε ο χρήστης μπορεί να κάνει scroll για να δει τις υπόλοιπες. Στο κάτω μέρος της σελίδας υπάρχει το κουμπί “επιστροφή στην κορυφή” το οποίο επιστρέφει τον χρήστη στην κορυφή με ευκολία. Στο κάτω δεξιά μέρος της οθόνης υπάρχει ένα floating button το οποίο δίνει πρόσβαση στον χρήστη και στις άλλες σελίδες της εφαρμογής Αναφορικά αυτές οι σελίδες είναι αναζήτηση ταινίας, ανακάλυψη ταινίας, αναζήτηση ταινίας μέσω κάμερας, Το προφίλ μου και Αποσύνδεση. Ο χρήστης μπορεί να επιλέξει μια από τις ταινίες που φαίνονται στην Home Page, αυτό θα τον μεταφέρει αυτόματα στην σελίδα με τις λεπτομέρειες για αυτήν την ταινία.



Εικόνα 18. Home Page

6.4 Στοιχεία Ταινίας

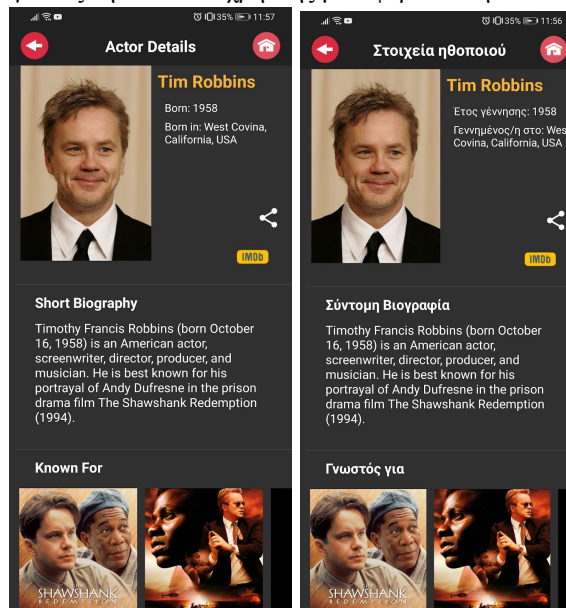
Σε αυτή την σελίδα παρουσιάζονται τα στοιχεία της ταινίας που έχει επιλεγεί. Ο χρήστης μπορεί να μεταβεί σε αυτήν την οθόνη από την Home Page, την αναζήτηση ταινίας, την ανακάλυψη ταινίας, την αναζήτηση ταινίας μέσω κάμερας, μέσα από τις λεπτομέρειες ηθοποιού, από την λίστα αγαπημένων και από την λίστα παρακολούθησης. Ξεκινώντας από το header, ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Movie Details) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα ή στην Home Page. Στην συνέχεια ο χρήστης βλέπει ένα πόστερ της ταινίας (αν είναι διαθέσιμο) στο οποίο παρουσιάζεται η κατάσταση της ταινίας (π.χ. έχει κυκλοφορήσει) και η συνολική διάρκεια της σε λεπτά, σε αυτό το σημείο ο χρήστης έχει την δυνατότητα να προσθέσει και να αφαιρέσει μια ταινία από την λίστα αγαπημένων και από την λίστα παρακολούθησης. Αμέσως πιο κάτω εμφανίζεται η κεντρική φωτογραφία της ταινίας (αν είναι διαθέσιμη) και στα δεξιά ο τίτλος της. Από κάτω εμφανίζεται μια λίστα πληροφοριών, πιο συγκεκριμένα η βαθμολογία της ταινίας και οι συνολικοί ψήφοι, το έτος κυκλοφορίας, η γλώσσα κυκλοφορίας, η κατηγορία της ταινίας, ο προϋπολογισμός, τα κέρδη, σύνδεσμοι για facebook και imdb και η επιλογή κοινοποίησης, όποια πληροφορία δεν είναι διαθέσιμη δεν εμφανίζεται. Συνεχίζοντας, ο χρήστης βλέπει μια περίληψη της ταινίας, μια λίστα με τους ηθοποιούς που παίζουν (σε RecyclerView) όπου επιλέγοντας τον ηθοποιό ο χρήστης μεταφέρετε στην σελίδα του ηθοποιού και μια λίστα με σχετικές ταινίες (σε RecyclerView) όπου επιλέγοντας την ταινία ο χρήστης μεταφέρετε στην σελίδα της ταινίας.



Εικόνα 19. Στοιχεία Ταινίας

6.5 Στοιχεία Ηθοποιού

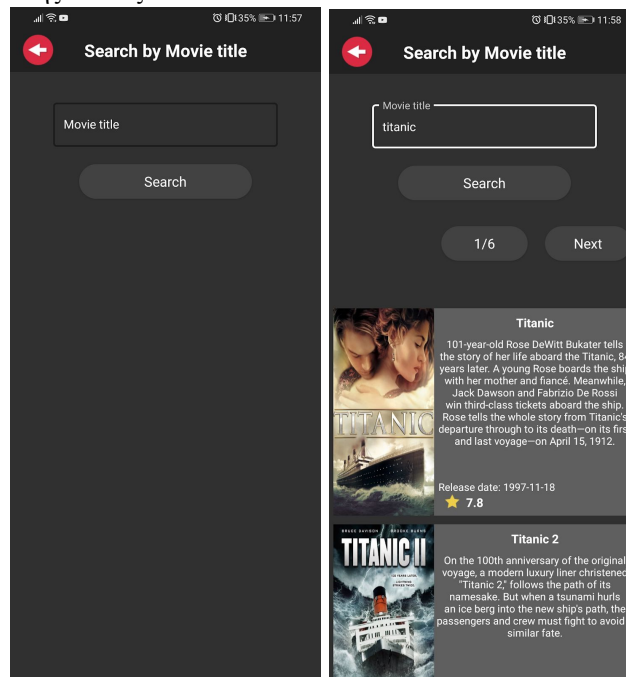
Η σελίδα στοιχεία ηθοποιού ακολουθεί την ίδια λογική με την σελίδα στοιχεία ταινίας, αυτό έγινε για λόγους ομοιομορφίας και για καλύτερη εμπειρία του χρήστη. Ξεκινώντας από το header, ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Actor Details) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα ή στην Home Page. Στην συνέχεια εμφανίζεται η φωτογραφία του ηθοποιού (αν είναι διαθέσιμη) και στα δεξιά το όνομα του ηθοποιού, το έτος γέννησής, ο τόπος γέννησης, η ημερομηνία θανάτου, ο σύνδεσμος imdb και Facebook και η επιλογή κοινοποίησης (αν είναι διαθέσιμα). Από κάτω εμφανίζεται ένα σύντομο βιογραφικό σημείωμα και οι ταινίες για τις οποίες είναι γνωστός (σε RecyclerView) όπου επιλέγοντας την ταινία ο χρήστης μεταφέρετε στην σελίδα της ταινίας.



Εικόνα 20. Στοιχεία Ηθοποιού

6.6 Αναζήτηση ταινίας

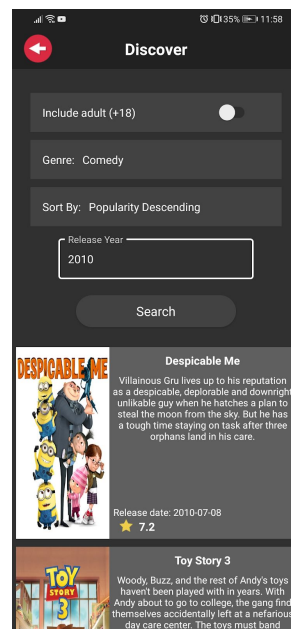
Η σελίδα αναζήτηση ταινίας επιτρέπει στον χρήστη να αναζητήσει μια ταινία βάσει του τίτλου της. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Search by Movie Title) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω έχει διαθέσιμο ένα πεδίο όπου μπορεί να συμπληρώσει την ταινία που θέλει να ψάξει καθώς και το κουμπί αναζήτησης. Αφότου συμπληρώσει το πεδίο και πατήσει το κουμπί αναζήτησης γίνεται αναζήτηση στο tmdb, σε περίπτωση που δεν εμφανιστούν αποτελέσματα εμφανίζεται σχετικό μήνυμα, αν βρεθούν αποτελέσματα εμφανίζεται μήνυμα με τον συνολικό αριθμό αποτελεσμάτων. Τα αποτελέσματα εμφανίζονται ανα είκοσι, οπότε στην περίπτωση που τα αποτελέσματα είναι παραπάνω από είκοσι υποστηρίζεται pagination ώστε να πλοηγηθεί ο χρήστης στα αποτελέσματα. Στα αποτελέσματα εμφανίζονται βασικές πληροφορίες για την κάθε ταινία όπως το πόστερ της, ο τίτλος, μια σύντομη περιγραφή, το έτος κυκλοφορίας και η βαθμολογία της ταινίας. Επιλέγοντας την ταινία ο χρήστης μεταφέρετε στην σελίδα της ταινίας.



Εικόνα 21. Αναζήτηση ταινίας

6.7 Ανακάλυψη ταινίας

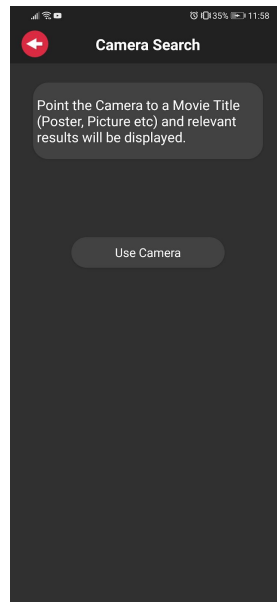
Η σελίδα ανακάλυψη ταινίας επιτρέπει στον χρήστη να ανακαλύψει ταινίες οι οποίες θα τον ενδιαφέρουν. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Discover) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω έχει διαθέσιμα κάποια φίλτρα αναζήτησης όπως αν η ταινία θα είναι για ενήλικες (επιλογή με διακόπτη), τι είδος ταινία ενδιαφέρει τον χρήστη (επιλογή από λίστα), πως θα ήθελε να γίνει η ταξινόμηση (επιλογή από λίστα) και το έτος κυκλοφορίας της ταινίας (σε πεδίο). Όλα τα πεδία εκτός από το αν θα είναι για ενήλικες είναι υποχρεωτικά. Τρέχοντας με το κουμπί αναζήτησης τα συγκεκριμένα φίλτρα θα γίνει αναζήτηση στο tmdb και θα εμφανιστούν έως είκοσι αποτελέσματα. Επιλέγοντας την ταινία ο χρήστης μεταφέρετε στην σελίδα της ταινίας.



Εικόνα 22. Ανακάλυψη ταινίας

6.8 Αναζήτηση ταινίας μέσω κάμερας

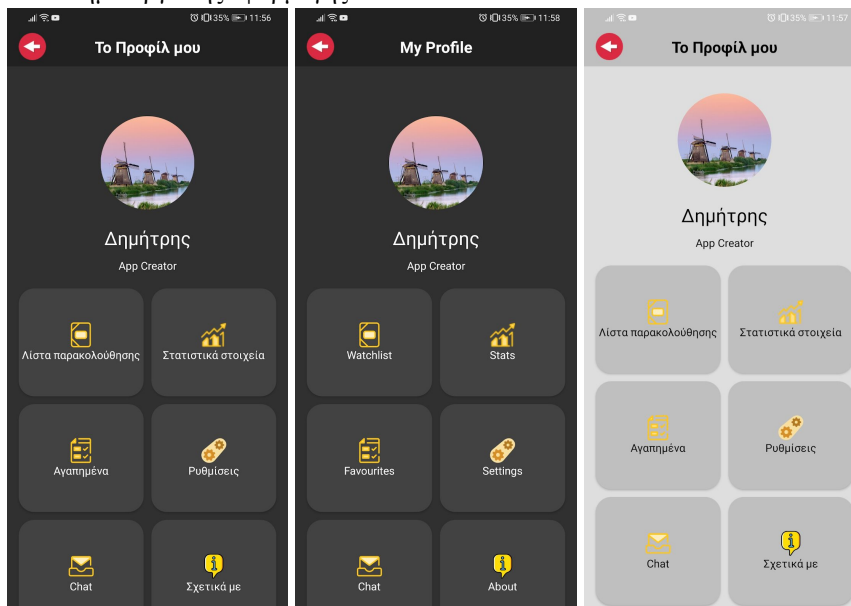
Η σελίδα αναζήτησης ταινίας μέσω κάμερας επιτρέπει στον χρήστη να αναζητήσει μια ταινία βάση του τίτλου της χρησιμοποιώντας την κάμερα της συσκευής του. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Camera Search) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω έχει πληροφορίες σχετικά με το πως θα πρέπει να χρησιμοποιήσει αυτή την λειτουργία και το κουμπί για την χρήση της κάμερας. Σε περίπτωση που ο χρήστης χρησιμοποιεί για πρώτη φορά την κάμερα στην εφαρμογή θα του ζητηθεί από το λειτουργικό να δώσει δικαιώματα χρήσης κάμερας στην εφαρμογή. Χρησιμοποιώντας την κάμερα ο χρήστης μπορεί να την εστιάσει σε ένα πόστερ ή σε οποιοδήποτε αντικείμενο έχει μορφή κειμένου. Αφού εγκρίνει την φωτογραφία το κείμενο που αναγνωρίστηκε μεταφέρεται σε ένα πεδίο το οποίο ο χρήστης μπορεί να κάνει βελτιώσεις. Πατώντας το κουμπί αναζήτησης γίνεται αναζήτηση στο tmdb, σε περίπτωση που δεν εμφανιστούν αποτελέσματα εμφανίζεται σχετικό μήνυμα, αν βρεθούν αποτελέσματα εμφανίζεται μήνυμα με τον συνολικό αριθμό αποτελεσμάτων. Τα αποτελέσματα εμφανίζονται ανα είκοσι, οπότε στην περίπτωση που τα αποτελέσματα είναι παραπάνω από είκοσι υποστηρίζεται pagination ώστε να πλοηγηθεί ο χρήστης στα αποτελέσματα. Στα αποτελέσματα εμφανίζονται βασικές πληροφορίες για την κάθε ταινία όπως το πόστερ της, ο τίτλος, μια σύντομη περιγραφή, το έτος κυκλοφορίας και η βαθμολογία της ταινίας. Επιλέγοντας την ταινία ο χρήστης μεταφέρετε στην σελίδα της ταινίας.



Εικόνα 23. Αναζήτηση ταινίας μέσω κάμερας

6.9 Το προφίλ μου

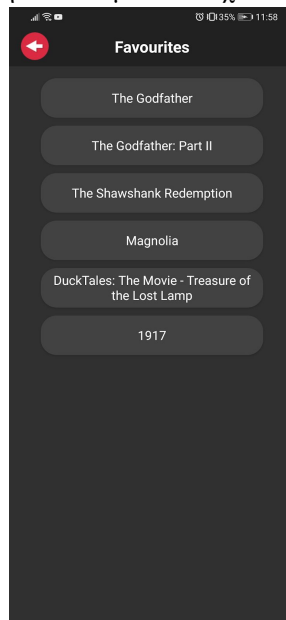
Η σελίδα προφίλ δίνει επιπλέον δυνατότητες στον χρήστη, επιτρέποντας του να δει τις αποθηκευμένες ταινίες του, να επεξεργαστεί το προφίλ του και να επικοινωνήσει με τους υπόλοιπους χρήστες της εφαρμογής. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (My Profile) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω έχει την φωτογραφία προφίλ του, το όνομα και την ετικέτα που έχει επιλέξει. Από κάτω ο χρήστης έχει τις εξής επιλογές, την λίστα αγαπημένων, την λίστα παρακολούθησης, το Chat, τα στατιστικά, οι ρυθμίσεις και τα “σχετικά με την εφαρμογή” όπου εκεί μπορεί να ενημερωθεί ο χρήστης για την τρέχουσα έκδοση καθώς και για τον δημιουργό της εφαρμογής.



Εικόνα 24. Το προφίλ μου

6.10 Λίστα Αγαπημένων

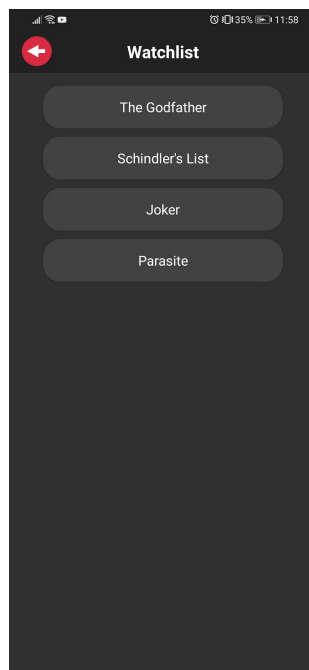
Η σελίδα λίστα αγαπημένων δίνει την δυνατότητα στον χρήστη να ανατρέξει στις ταινίες που έχει αποθηκεύσει ως αγαπημένες από την σελίδα στοιχεία ταινίας. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Favourites) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω είναι η λίστα η οποία αποτελείται από έναν RecyclerView ο οποίος προβάλλει ονόματα ταινιών. Τα δεδομένα έχουν αποθηκευτεί και ανακτούνται από την Firebase στην οποία είναι αποθηκευμένη μόνο ο τίτλος της ταινίας και το id της που είναι απαραίτητο για να μεταφερθεί ξανά στην σελίδα με τα στοιχεία της ταινίας.



Εικόνα 25. Λίστα Αγαπημένων

6.11 Λίστα Παρακολούθησης

Η σελίδα λίστα παρακολούθησης δίνει την δυνατότητα στον χρήστη να ανατρέξει στις ταινίες που έχει αποθηκεύσει, για να τις δει στο μέλλον, από την σελίδα στοιχεία ταινίας. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Watchlist) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω είναι η λίστα η οποία αποτελείται από έναν RecyclerView ο οποίος προβάλλει ονόματα ταινιών. Τα δεδομένα έχουν αποθηκευτεί και ανακτούνται από την Firebase στην οποία είναι αποθηκευμένη μόνο ο τίτλος της ταινίας και το id της που είναι απαραίτητο για να μεταφερθεί ξανά στην σελίδα με τα στοιχεία της ταινίας.



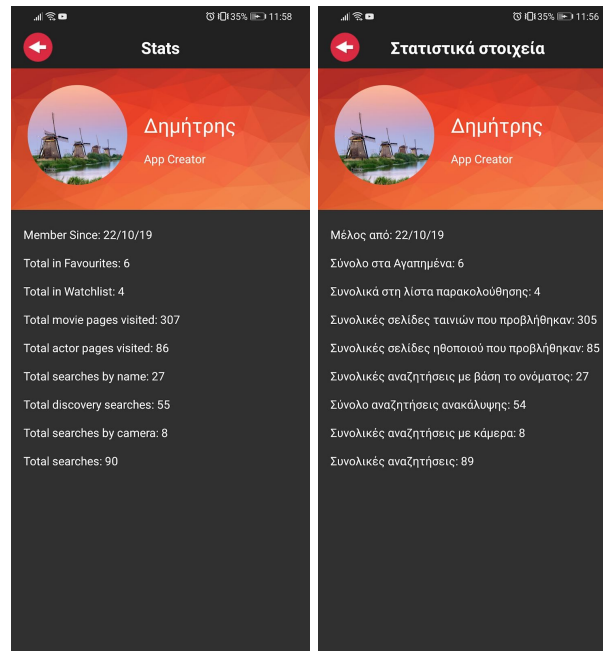
Εικόνα 26. Λίστα Παρακολούθησης

6.12 Chat

Η σελίδα αυτή δίνει την δυνατότητα στους χρήστες να συνομιλούν για οτιδήποτε αφορά τις ταινίες. Το Chat είναι ανοιχτό σε όλους και δεν απαιτείται κάποια επιπλέον ρύθμιση από τον χρήστη. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Chat) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω είναι η λίστα η οποία αποτελείται από έναν RecyclerView ο οποίος προβάλλει τα μηνύματα που έχουν ανταλλάξει οι χρήστες. Κατά την είσοδο του στην σελίδα ο χρήστης μεταφέρεται αυτόματα στο πιο πρόσφατο μήνυμα.

6.12 Στατιστικά

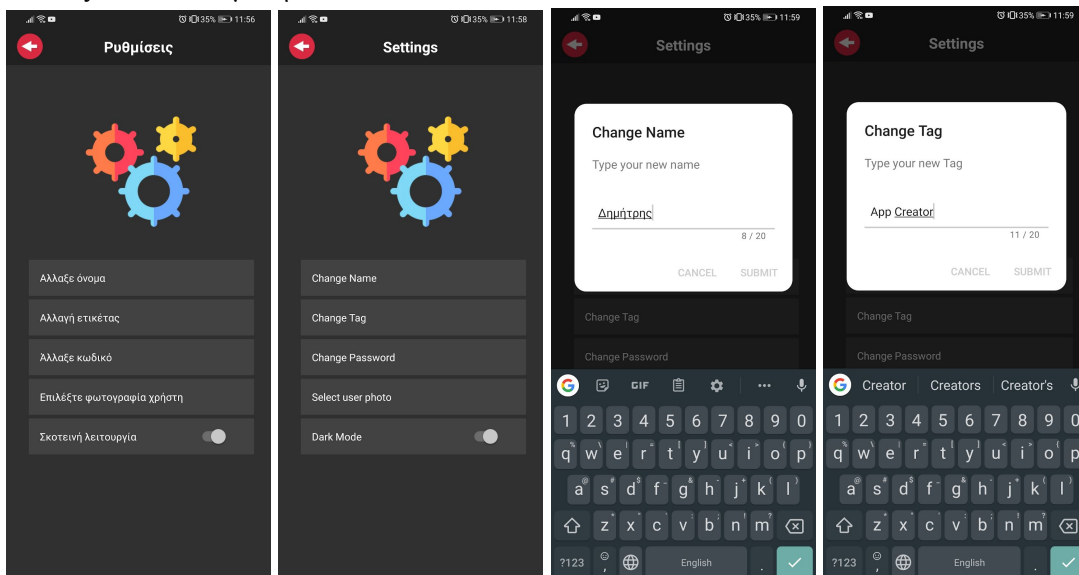
Η σελίδα αυτή δίνει την δυνατότητα στους χρήστες να βλέπουν την δραστηριότητα τους μέσα στην εφαρμογή. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Stats) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω τα εξής στοιχεία “Μέλος από” η ημερομηνία εγγραφής του χρήστη, “Σύνολο στα Αγαπημένα” συνολικές ταινίες αποθηκευμένες ως αγαπημένες, “Συνολικά στην λίστα παρακολούθησης” συνολικές ταινίες αποθηκευμένες στην λίστα παρακολούθησης, “Συνολικές σελίδες ταινιών που προβλήθηκαν” συνολικές επισκέψεις στην σελίδα Στοιχεία Ταινίας, “Συνολικές σελίδες ηθοποιού που προβλήθηκαν” συνολικές επισκέψεις στην σελίδα Στοιχεία Ηθοποιού, “Συνολικές αναζητήσεις με βάση το όνομα” συνολικές αναζητήσεις μέσω της σελίδας Αναζήτηση ταινίας, “Συνολικές αναζητήσεις ανακάλυψης” συνολικές αναζητήσεις μέσω της σελίδας Ανακάλυψη ταινίας, “Συνολικές αναζητήσεις με κάμερα” συνολικές αναζητήσεις μέσω της σελίδας Αναζήτηση ταινίας μέσω κάμερας, “Συνολικές αναζητήσεις” συνολικές αναζητήσεις από όλες τις σελίδες.



Εικόνα 27. Στατιστικά

6.13 Ρυθμίσεις

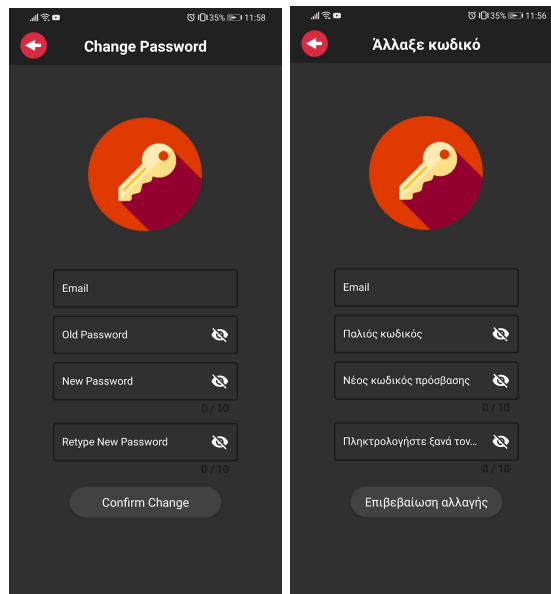
Η σελίδα αυτή δίνει την δυνατότητα στους χρήστες να διαχειριστούν τον λογαριασμό τους. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Settings) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω ο χρήστης έχει την επιλογή να αλλάξει το όνομα του, την ετικέτα του, να μεταβεί στην σελίδα “Αλλαγή Κωδικού”, να μεταβεί στην σελίδα “Αλλαγή Φωτογραφίας” και να αλλάξει το θέμα της εφαρμογής (να κάνει μετάβαση από light mode σε dark mode και αντίστροφα) μέσω ενός switcher. Η αλλαγή ονόματος και ετικέτας γίνεται με απευθείας επικοινωνία με την Firebase.



Εικόνα 28. Ρυθμίσεις

6.14 Αλλαγή Κωδικού

Η σελίδα αυτή δίνει την δυνατότητα στους χρήστες να αλλάξουν τον κωδικό του λογαριασμού τους. Αρχικά στο χρήστη εμφανίζεται το header, όπου ο χρήστης βλέπει σαν τίτλο το όνομα της σελίδας (Change Password) και έχει την δυνατότητα να επιστρέψει στην προηγούμενη σελίδα, από κάτω υπάρχουν τέσσερα υποχρεωτικά πεδία, το “Email”, ο “Παλιός Κωδικός”, ο “Καινούργιος Κωδικός” και το “Πληκτρολόγησε ξανά τον καινούργιο κωδικό” και το κουμπί επιβεβαίωσης. Τα πεδία κωδικών κρύβουν το περιεχόμενο τους έχοντας την επιλογή να το εμφανίσουν πατώντας το ματάκι. Πατώντας το κουμπί επιβεβαίωσης τρέχουν αυτόματα κάποιοι έλεγχοι, τόσο τοπικά όσο και στην Firebase. Οι έλεγχοι είναι ότι όλα τα πεδία είναι συμπληρωμένα, το “Email” και ο “Παλιός Κωδικός” είναι σωστοί, ο “Παλιός Κωδικός” δεν είναι ίδιος με τον “Καινούργιος Κωδικός”, ο “Καινούργιος Κωδικός” είναι τουλάχιστον έξι χαρακτήρες και ότι ο “Καινούργιος Κωδικός” και το “Πληκτρολόγησε ξανά τον καινούργιο κωδικό” είναι ίδιοι.



Εικόνα 29. Αλλαγή Κωδικού

7. Επεξήγηση κώδικα

Στο παρόν κεφάλαιο θα αναφερθούμε στα πιο σημαντικά σημεία του κώδικα της εφαρμογής αναφέροντας συνοπτικά τον τρόπο λειτουργίας του.

7.1 Δημιουργία Base Activity

Προκειμένου να εκμεταλλευτούμε την κληρονομικότητα που μας προσφέρει η Kotlin ώστε ο κώδικας να είναι όσο γίνεται πιο καθαρός και ευανάγνωστος, δημιουργήθηκε ένα αρχείο BaseActivity από το οποίο κληρονομούν όλα τα Activities.

```

1 package com.example.moviefinder.base
2
3 import ...
4
26
27 open class BaseActivity : AppCompatActivity() {
28
29     internal lateinit var sharedPref: SharedPreferences
30     internal lateinit var editor: SharedPreferences.Editor
31     var amountFormat = DecimalFormat( pattern: "#,###.##")
32
33     override fun onCreate(savedInstanceState: Bundle?) {
34         super.onCreate(savedInstanceState)
35         darkMode()
36     }
37
38     fun makeDialog(title: String, body: String, button: String, icon: Int) {
39         val dialog = MaterialDialog( windowContext: this)
40             .cornerRadius( literalDp: 16f)
41             .title(text = title)
42             .message(text = body)
43             .icon(icon)
44             .positiveButton(text = button) { dialog ->
45                 dialog.hide()
46             }
47         dialog.show()
48     }

```

Εικόνα 30. BaseActivity

```

1 package com.example.moviefinder.login
2
3 import ...
4
21
22 class LoginPageActivity : BaseActivity() {
23
24     override fun onCreate(savedInstanceState: Bundle?) {
25         super.onCreate(savedInstanceState)
26         loginAnimation()
27         setContentView(R.layout.activity_login_page)
28
29         initLayout()
30         rememberMeCheck()
31
32     }

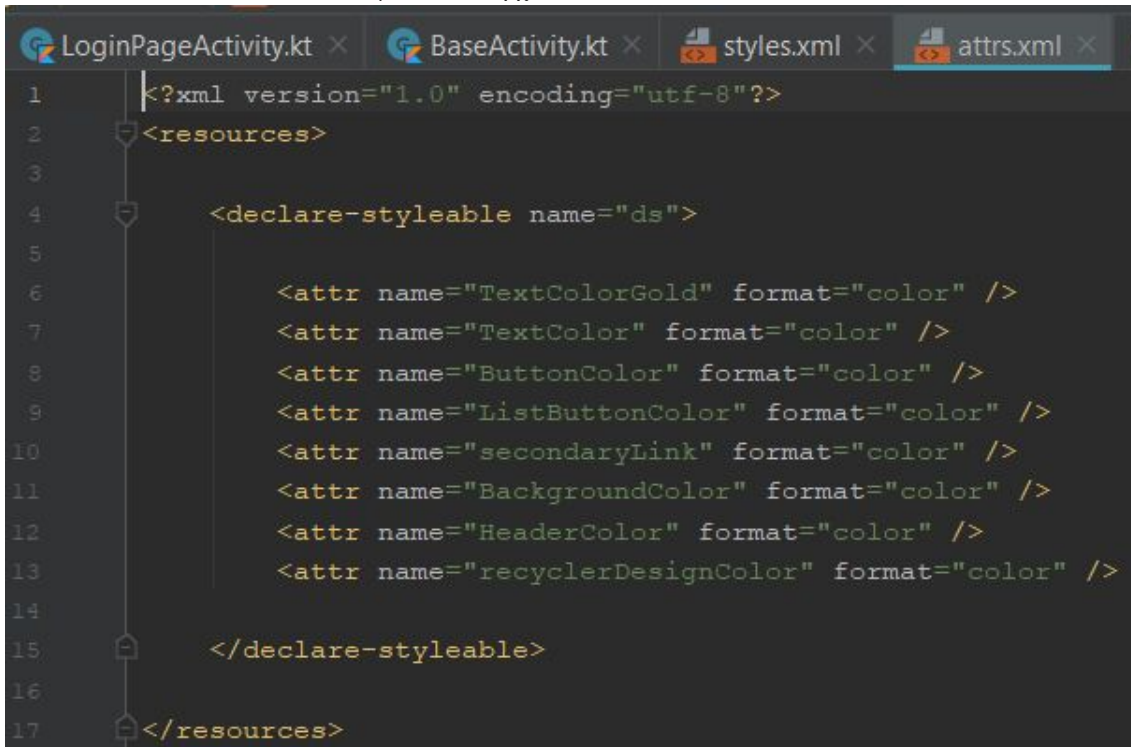
```

Εικόνα 31. Κληρονομικότητα της BaseActivity

Το Android έχει ως κανόνα τα activities να κληρονομούν από το AppCompatActivity. Στην δική μας περίπτωση όλα τα activities κληρονομούν το BaseActivity, ο κανόνας δεν παραβιάζεται καθώς το BaseActivity με την σειρά του κληρονομεί από το AppCompatActivity. Οπότε με λίγα λόγια η κληρονομικότητα στην εφαρμογή έχει την εξής ιεραρχία (Οποιοδήποτε activity → BaseActivity → AppCompatActivity). Στο BaseActivity έχουν υλοποιηθεί πολλά functions τα οποία μπορούν να χρησιμοποιηθούν σε οποιοδήποτε σημείο απλά καλώντας το function αποφεύγοντας έτσι την επανάληψη κώδικα.

7.2 Dark Mode

Η ύπαρξη δύο διαφορετικών mode εμφάνισης στις εφαρμογές είναι ένα χαρακτηριστικό το οποίο έχει γίνει αρκετά δημοφιλές τα τελευταία χρόνια. Για την υλοποίηση του απαιτούνται τρία στοιχεία, η δήλωση των χρωμάτων σαν styleable resources στο αρχείο attrs.xml, η δήλωση των δύο διαφορετικών modes στο αρχείο styles.xml και η κλήση μιας function σε κάθε activity. Η function, δεδομένο το ότι θα τρέχει σε κάθε activity είναι υλοποιημένη στο BaseActivity και τρέχει κάνοντας override την onCreate function. Ακολουθούν screenshots με τα xml αρχεία.



```
1 |<?xml version="1.0" encoding="utf-8"?>
2 |<resources>
3 |
4 |   <declare-styleable name="ds">
5 |
6 |     <attr name="TextColorGold" format="color" />
7 |     <attr name="TextColor" format="color" />
8 |     <attr name="ButtonColor" format="color" />
9 |     <attr name="ListButtonColor" format="color" />
10 |    <attr name="secondaryLink" format="color" />
11 |    <attr name="BackgroundColor" format="color" />
12 |    <attr name="HeaderColor" format="color" />
13 |    <attr name="recyclerDesignColor" format="color" />
14 |
15 |   </declare-styleable>
16 |
17 | </resources>
```

Εικόνα 32. attrs.xml

```

1 <resources>
2
3 <!-- Base application theme. -->
4 <style name="AppTheme" parent="Theme.MaterialComponents.DayNight.NoActionBar">
5 <!-- Dark Theme. -->
6 <item name="colorPrimary">#cccccc</item>
7 <item name="colorPrimaryDark">#212121</item>
8 <item name="colorAccent">#cccccc</item>
9 <item name="BackgroundColor">#303030</item>
10 <item name="TextColorGold">#FAB333</item>
11 <item name="TextColor">#fff</item>
12 <item name="ButtonColor">#424242</item>
13 <item name="ListButtonColor">#424242</item>
14 <item name="secondaryLink">#FAB333</item>
15 <item name="HeaderColor">#212121</item>
16 <item name="recyclerDesignColor">#8888</item>
17
18 </style>
19
20 <style name="lightTheme" parent="Theme.MaterialComponents.DayNight.NoActionBar">
21 <!-- Light Theme. -->
22 <item name="colorPrimary">#000000</item>
23 <item name="colorPrimaryDark">#bfbfbf</item>
24 <item name="colorAccent">#2e2e2e</item>
25 <item name="BackgroundColor">#dedede</item>
26 <item name="TextColorGold">#000000</item>
27 <item name="TextColor">#000000</item>
28 <item name="ButtonColor">#bfbfbf</item>
29 <item name="ListButtonColor">#bfbfbf</item>
30 <item name="secondaryLink">#000000</item>
31 <item name="HeaderColor">#bfbfbf</item>
32 <item name="recyclerDesignColor">#cccc</item>
33
34 </style>
35

```

Εικόνα 33. styles.xml

7.3 SafeClickListener

Η Kotlin προσφέρει στον προγραμματιστή τον `OnClickListener` για να αναγνωρίζει την διεπαφή του χρήστη με τα στοιχεία της εφαρμογής. Το αρνητικό με τον συγκεκριμένο Listener είναι ότι δεν περιορίζει τις διεπαφές του χρήστη με αποτέλεσμα να μπορεί να στείλει πολλές κλήσεις για το ίδιο αίτημα, που αυτό με την σειρά του δημιουργεί προβλήματα στην εφαρμογή. Για την επίλυση αυτού δημιουργήθηκε ο `SafeClickListener` ο οποίος κληρονομεί τον `OnClickListener` προσθέτοντας ένα επίπεδο ασφαλείας το οποίο δεν επιτρέπει στον χρήστη να κάνει πολλαπλές κλήσεις σε σύντομο χρονικό διάστημα. Ακολουθεί ο κώδικας του `SafeClickListener`.

```
package com.example.moviefinder.base

import android.os.SystemClock
import android.view.View

class SafeClickListener(
    private var defaultInterval: Int = 1000,
    private val onSafeClick: (View) -> Unit
) : View.OnClickListener {

    private var lastTimeClicked: Long = 0

    override fun onClick(v: View?) {

        if (SystemClock.elapsedRealtime() - lastTimeClicked < defaultInterval) {
            return
        }
        lastTimeClicked = SystemClock.elapsedRealtime()
        v?.let { onSafeClick(it) }
    }
}
```

Εικόνα 34. SafeClickListener

7.4 Σύνδεση Χρήστη

Ανοίγοντας την εφαρμογή ο χρήστης βλέπει την σελίδα σύνδεσης. Έχοντας συμπληρώσει όλα τα πεδία και πατώντας το κουμπί σύνδεσης γίνεται μια κλήση στην Firebase με σκοπό την επαλήθευση του χρήστη. Πιο συγκεκριμένα, αρχικά γίνεται ένας έλεγχος από την εφαρμογή ότι έχουν συμπληρωθεί τα απαραίτητα πεδία. Στην συνέχεια αυτά τα πεδία στέλνονται στην Firebase σε μορφή String με την μέθοδο `auth.signInWithEmailAndPassword`, η Firebase απαντάει είτε με `isSuccessful` είτε με `Failure`. Αμα απαντήσει `isSuccessful` ο χρήστης συνδέεται με επιτυχία στην εφαρμογή αλλιώς του εμφανίζεται ένα σχετικό μήνυμα λάθους. Ακολουθεί ο κώδικας για την επαλήθευση του χρήστη.

```

val auth: FirebaseAuth = FirebaseAuth.getInstance()
loginButtonText.text = ""
progressBar.visibility = View.VISIBLE
auth.signInWithEmailAndPassword(email.toString(), password.toString())
    .addOnCompleteListener(this) { it: Task<AuthResult!>
        if (it.isSuccessful) {
            goToHomePageAnimation()
            Handler().postDelayed({
                auth.currentUser
                rememberMeUpdate(email.toString())
                val intent =
                    Intent(packageContext, this@LoginPageActivity, HomePageActi
                intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK
                intent.flags = Intent.FLAG_ACTIVITY_CLEAR_TASK
                startActivity(intent)
                overridePendingTransition(
                    android.R.anim.fade_in,
                    android.R.anim.fade_out
                )
                this@LoginPageActivity.finish()
            }, delayMillis: 500)
        }
    }
}

.addOnFailureListener { it: Exception
    progressBar.visibility = View.GONE
    loginButtonText.setText("login")
    Toasty.error(
        baseContext,
        "Failed to log in: {it.message}",
        Toast.LENGTH_LONG,
        withIcon: true
    ).show()
}

```

Εικόνα 35. signInWithEmailAndPassword

7.5 Retrofit

Το Retrofit είναι υπεύθυνο και περιλαμβάνει όλες τις κλήσεις που γίνονται στο Tmdb. Απαιτείται ένα object αρχείο για την δημιουργία και τον ορισμό των παραμέτρων του RetrofitClient, ένα interface αρχείο για τον ορισμό όλων των κλήσεων που θα γίνουν και ο κώδικας στο εκάστοτε activity που γίνεται η κλήση. Ακολουθούν εικόνες για με τα απαραίτητα στοιχεία του Retrofit.


```

object RetrofitClient {

    private var ourInstance: Retrofit? = null
    var logging = HttpLoggingInterceptor()
        .setLevel(HttpLoggingInterceptor.Level.BODY)
    val httpClient = OkHttpClient.Builder()
        .addInterceptor(logging)

    val instance: Retrofit
    get() {
        if (ourInstance == null) {
            ourInstance = Retrofit.Builder()
                .baseUrl( baseUrl: "https://api.themoviedb.org/3/")
                .addConverterFactory(GsonConverterFactory.create())
                .addCallAdapterFactory(RxJava2CallAdapterFactory.create())
                .client(httpClient.build())
                .build()
        }
        return ourInstance!!
    }
}

```

Εικόνα 36. object RetrofitClient

```

interface IHomeRetrofit {

    @get:GET( value: "movie/popular?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    val movies: Observable<HomePageModel>

    @get:GET( value: "movie/top_rated?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    val topRatedMovies: Observable<HomePageModel>

    @get:GET( value: "movie/upcoming?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    val upcomingMovies: Observable<HomePageModel>

    @get:GET( value: "movie/now_playing?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    val nowPlayingMovies: Observable<HomePageModel>

    @GET( value: "search/movie?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    fun getMoviesBasedOnQuery(@Query( value: "query") q: String, @Query( value: "page") p: String): Observable<HomePageModel>

    @GET( value: "movie/{movie_id}?api_key=4df261f7b714f86c5d9a9950ff3bd257&language=en-US")
    fun getMovieDetails(@Path( value: "movie_id") movie_id: Int): Observable<DetailsModel>

    @GET( value: "movie/{movie_id}/credits?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    fun getMoviesCast(@Path( value: "movie_id") movie_id: Int): Observable<ActorsModel>

    @GET( value: "movie/{movie_id}/external_ids?api_key=4df261f7b714f86c5d9a9950ff3bd257")
    fun getExternalIds(@Path( value: "movie_id") movie_id: Int): Observable<ExternalIdsData>

    @GET( value: "movie/{movie_id}/similar?api_key=4df261f7b714f86c5d9a9950ff3bd257&language=en-US&page=1")
    fun getSimilarMovies(@Path( value: "movie_id") movie_id: Int): Observable<HomePageModel>

    @GET( value: "person/{person_id}?api_key=4df261f7b714f86c5d9a9950ff3bd257&language=en-US")
    fun getActorDetails(@Path( value: "person_id") person_id: Int): Observable<ActorDetailsModel>
}

```

Εικόνα 37. Retrofit calls

```

private fun displayKnownFor() {
    val retrofit = RetrofitClient.instance
    jsonApiPopular = retrofit.create(IHomeRetrofit::class.java)

    compositeDisposable.add(
        jsonApiPopular.getActorMovies(id)
            .subscribeOn(Schedulers.io())
            .observeOn(AndroidSchedulers.mainThread())
            .subscribeWith(object : DisposableObserver<KnownForModel>() {
                override fun onComplete() {

                }

                override fun onNext(t: KnownForModel) {
                    if (t.cast.isNullOrEmpty()) {
                        detailsSimilarTitle.visibility = View.GONE
                        separationLineActors.visibility = View.GONE
                    } else {
                        displaySimilarMovies(t)
                    }
                }

                override fun onError(e: Throwable) {
                    displayError(e)
                }
            })
    )
}

```

Εικόνα 38. Retrofit μέσα σε activity

8. Συμπεράσματα - Μελλοντικές Προτάσεις

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν τα συμπεράσματα που προέκυψαν από την υλοποίηση της παρούσας μεταπτυχιακής εργασίας. Επίσης θα γίνει αναφορά για μελλοντικές επεκτάσεις για την υπάρχουσα εφαρμογή.

8.1 Συμπεράσματα

Έχοντας ολοκληρώσει την ανάπτυξη της εφαρμογής, είμαστε σε θέση να εξάγουμε διάφορα χρήσιμα συμπεράσματα. Στη παρούσα μεταπτυχιακή εργασία σχεδιάστηκε και υλοποιήθηκε μια Android εφαρμογή για έξυπνα τηλέφωνα που αφορά την αναζήτηση ταινιών. Δίνοντας τη δυνατότητα στους χρήστες να γνωρίσουν καλύτερα τις ταινίες που τους ενδιαφέρουν με τις πλέον ενδεδειγμένες τεχνολογίες που αξιοποιούν τα έξυπνα τηλέφωνα. Αρχικά η εφαρμογή καταφέρνει τον σκοπό της χωρίς να έχουν παρατηρηθεί bugs ή κάποια καθυστέρηση στην απόδοση της. Η γρήγορη απόκριση της, το εύρος των δυνατοτήτων καθώς και η σχεδίαση της έχει αποσπάσει θετικά σχόλια από όποιον την έχει χρησιμοποιήσει.

Η όλη διαδικασία που απαιτήθηκε, από την έρευνα, τα πρώτα αρχεία κώδικα και τον σχεδιασμό της εφαρμογής, μέχρι και την ολοκλήρωσή της μπόρεσε να μας δώσει μια συνολική εικόνα για την ανάπτυξη εφαρμογών σε Android. Ο προγραμματισμός σε Android είναι μια ευχάριστη πρόκληση που δίνει πολλές δυνατότητες και επιλογές στον προγραμματιστή, του δίνει την αίσθηση της δημιουργίας και τον αναγκάζει να χρησιμοποιήσει διαφορετικά πολύ ενδιαφέροντα εργαλεία. Από την άλλη όμως ο προγραμματιστής έρχεται αντιμέτωπος με αρκετά προβλήματα που προκύπτουν κατά την ανάπτυξη. Συγκεκριμένα αν και φαινομενικά η συγκεκριμένη εφαρμογή δείχνει σχετικά απλή, στην πράξη υπήρξαν διάφορα τεχνικά προβλήματα. Η μεγαλύτερη δυσκολία προέρχεται από το γεγονός ότι η Kotlin είναι μια σχετικά καινούργια γλώσσα προγραμματισμού με αποτέλεσμα το documentation να είναι

ελλιπές, τα σχετικά forum είναι περιορισμένα και τα διαθέσιμα libraries λιγα σχετικά με αυτά που είναι διαθέσιμα στην Java.

8.2 Μελλοντικές Επεκτάσεις

Στην εποχή μας που ανάπτυξη των έξυπνων συσκευών και των υπηρεσιών που προσφέρουν είναι ραγδαία. Με κάθε καινούργια έκδοση Android οι εφαρμογές έχουν επιπλέον δυνατότητες και για να ανταπεξέλθει στον ανταγωνισμό κάθε εφαρμογή πρέπει να αναβαθμίζεται και να βελτιώνεται. Έτσι και οι δυνατότητες της εφαρμογής που υλοποιήσαμε μπορούν να επεκταθούν μελλοντικά προς αρκετές κατευθύνσεις. Μερικές ιδέες για μελλοντική επέκταση της εφαρμογής παρουσιάζονται παρακάτω:

- Βελτιστοποίηση των διεπαφών ως προς το γραφικό περιβάλλον(UI) και την εμπειρία χρήσης(UX).
- Προσθήκη “suggested results” στην αναζήτηση ταινίας
- Προσθήκη δυνατότητας προβολής τρέιλερ στις λεπτομέρειες ταινίας
- Δυνατότητα επιλογής εικόνας προφίλ από την κάμερα
- Δυνατότητα επιλογής εικόνας προφίλ από την FireBase Database
- Χρήση animation για μετάβαση από σελίδα σε σελίδα
- Αναζήτηση ηθοποιού
- Προσθήκη περισσότερων φίλτρων στην αναζήτηση ταινίας
- Προσθήκη τηλεοπτικών σειρών
- Μετάβαση σε αρχιτεκτονική MVVM, χρησιμοποιώντας έτσι την πλέον σύγχρονη αρχιτεκτονική την οποία προτείνει και η Google
- Διόρθωση οποιονδήποτε bugs προκύψουν

Βιβλιογραφία

1. <https://play.google.com/store>
2. <https://en.wikipedia.org/wiki/IMDb>
3. <https://developer.android.com/>
4. <https://kotlinlang.org/docs/tutorials/>
5. <https://www.tutorialspoint.com/kotlin/index.htm>
6. <https://superkotlin.com/>
7. Android Programming: The Big Nerd Ranch Guide (3rd Edition) by Bill Phillips, Chris Stewart, Kristin Marsicano
8. Professional Android 4th Edition by Reto Meier, Ian Lake
9. Android Studio 3.5 Development Essentials - Kotlin Edition: Developing Android 10 (Q) Apps Using Android Studio 3.5, Kotlin and Android Jetpack by Neil Smyth