Ανάπτυξη Εφαρμογών Κινητών Συσκευών

Εαρινό Εξάμηνο 2021-2022

Ιορδάνης Κοκκινίδης - 3412

Ιωάννης Οικονομίδης - ΧΧΧΧ

Δημήτρης Παραθύρας - ΧΧΧΧ

**MovieNight**

## Σκοπός

Η εφαρμογή MovieNight διευκολύνει τον χρήστη στην εύρεση ταινιών. Ο χρήστης μπορεί να ψάξει για κάποια ταινία χρησιμοποιώντας τον τίτλο της καθώς και να χρησιμοποιήσει φίλτρα όπως την κατηγορία (genre). Η λειτουργία αυτή μπορεί εύκολα να επεκταθεί και στη χρήση άλλων φίλτρων, όπως η βαθμολογία, η ημερομηνία κυκλοφορίας κλπ. Ο χρήστης μπορεί να επιλέξει ταινίες για να τις αποθηκεύσει στη λίστα με τις αγαπημένες του, καθώς και να απορρίψει ταινίες που έχει ήδη δει, ή που του είναι αδιάφορες.

Χρήστης αυτής της εφαρμογής θα μπορούσε να είναι ο καθένας που δυσκολεύεται να επιλέξει κάποια ταινία για να δει, καθώς και κάποιος που θέλει γρήγορα να ψάξει τις αγαπημένες του ταινίες, ή να βρει τις δημοφιλέστερες ταινίες δράσης.

## Χρήστες

Οι χρήστες που χρησιμοποιούν για πρώτη φορά την εφαρμογή θα συναντήσουν το LoginRegistrationActivity, το οποίο περιέχει τα fragments loginFragment και registerFragment, στα οποία ο χρήστης μπορεί να μετακινηθεί μέσω ενός Bottom Navigation Menu. Στο Login o χρηστης πρεπει να χρησιμοποιήσει το email και τον κωδικό του για να συνδεθεί, ενώ στο registration χρησιμοποιεί email, username και password για να δημιουργήσει τον λογαριασμό του. Το Authentication γίνεται μέσω Firebase Auth ενώ κάθε χρήστης αποθηκεύεται σε μια Firebase Real Time Database (με email και username). H αποθήκευση και πρόσβαση στην Firebase Real Time Database γίνεται μέσω της βοηθητικής κλάσης DAOUser (π.χ. daoUser.add(user) ).

## Βάση Δεδομένων

Η βάση δεδομένων της εφαρμογής είναι μια SQLite βάση δεδομένων, όπου βρίσκονται αποθηκευμένες οι ταινίες, καθώς και πίνακες για τις αγαπημένες ταινίες του χρήστη και για αυτές που επιθυμεί να αγνοήσει. Η πρόσβαση στη βάση δεδομένων γίνεται μέσω της κλάσης DbHandler, και των μεθόδων της. Η κλήση αυτών των μεθόδων γίνεται από τις κλάσεις SearchUtill και DisFavUtil, στις οποίες καθορίζεται με βάση το input του χρήστη, ποιό θα είναι το είδος του ερωτήματος στη βάση δεδομένων.

Οι μέθοδοι findMovieByTitle και findMovieByTitleAndGenre της SearchUtil επιστρέφουν ένα ArrayList με τις ταινίες που επιστράφηκαν μετά από ερώτημα στη βάση δεδομένων. Συγκεκριμένα, το ArrayList περιέχει αντικείμενα της κλάσης MovieModel, η οποία είναι μια κλάση για την μοντελοποίηση των ταινιών που αποθηκεύονται στη βάση δεδομένων και εμφανίζονται στο User Interface, Κάθε MovieModel περιέχει πεδία για id, τιτλο, ημερομηνια κυκλοφορίας, βαθμολογία, περιγραφή ταινίας, το path για το αρχείο του poster και ArrayList με τις κατηγορίες στις οποίες ανήκει η ταινία. Αντίστοιχα δηλαδή με τις στήλες του πίνακα movies στη Βάση Δεδομένων.

Οι μέθοδος getFavorites της DisFavUtil επιστρέφει ένα Arraylist με τις αγαπημένες ταινίες του χρήστη με id userID. Οι υπόλοιπες μέθοδοι της κλάσης αυτής εκτελούν τη λειτουργία που περιγράφουν στο όνομά τους με βάση το userID του τωρινού χρήστη και το id της ταινίας movieID αν χρειάζεται χρησιμοποιώντας τη κλάση DBHandler και τις αντίστοιχες μεθόδους της.

## Περιγραφή fragments και των αντίστοιχων κλάσεων που τα προσθέτον προγραμματιστικά.

Η εφαρμογή αποτελείται από 3 Activities. To MainActivity, το LoginRegistrationActivity και το SettingsActivity. Στο MainActivity περιέχει ένα FragmentContainerView μεσα στο οποίο περιέχονται δύο fragments, το Home (στο UI αντί για Home γράφει Favourites) και το Search ενώ υπάρχει και η επιλογή Settings που οδηγεί στο τρίτο Activity SettingsActivity . Ο χρήστης μπορεί να μετακινηθεί από και προς το Home (Favourites στο UI) και το Search μέσω ενός BottomNavigationMenu, ενώ αν πάει στα Settings, μπορεί να πάει πίσω με αντίστοιχο κουμπί που υπάρχει σε εκείνο το Activity. Αντίστοιχα, στο LoginRegistrationActivity, περιέχεται ένα FragmentContainerView, μέσα στο οποίο περιέχονται τα loginFragment και registerFragent στα οποία μετακινείται ο χρήστης μέσω ενός BottomNavigationMenu.

### Home Fragment

Το homeFragment περιέχει ένα ConstraintLayout μέσα στο οποίο υπάρχει ένα RecyclerVIew στο οποίο προστίθονται οι κάρτες των ταινιών με τη συνάρτηση findFavourites του αρχείου homeFragment.java

### Search Fragment

Το searchFragment περιέχει ένα EditText ώστε ο χρήστης να μπορεί να ψάξει κάποια ταινία με βάση τον τίτλο της. Επίσης, υπάρχει ένα ChipGroup το οποίο περιέχει 9 Chips, τα οποία ο χρήστης μπορεί να επιλέξει για να φιλτράρει την αναζήτησή του σε συγκεκριμένες κατηγορίες ταινιών. Τέλος, υπάρχει το SearchButton, το οποίο είναι το κουμπί που πατάει ο χρήστης όταν θέλει να κάνει αναζήτηση, το οποίο στέλνει το input του χρήστη σε ένα νέο searchResultsFragment (δηλαδή τίτλο και επιλεγμένες κατηγορίες) μέσω ενός Bundle και στην συνέχεια αλλάζει το εμφανιζόμενο fragment σε searchResultsFragment.

### Login Fragment

To loginFragment είναι ένα από τα δύο fragments του LoginRegistrationActivity και περιέχει ουσιαστικά μια φόρμα για input από τον χρήστη. Περιέχει EditText για το email και τον κωδικό του χρήστη, καθώς και ένα κουμπί για να ξεκινήσει η διαδικασία του login, όπου χρησιμοποιείται το input του χρήστη στα EditText για να γίνει το Authentication μέσω Firebase.

### Registration Fragment

To registerFragment είναι το δεύτερο από τα δύο fragments του LoginRegistrationActivity και περιέχει επίσης μια φόρμα για input από τον χρήστη. Περιέχει EditText για το email, το username, τον κωδικό και για την επιβεβαίωση κωδικού. Τέλος περιέχει και αυτό ένα κουμπί για να ξεκινήσει η διαδικασία του registration, όπου και εδώ χρησιμοποιείται το input του χρήστη για να γίνει το registration μέσω Firebase.

### SearchResults Fragment

To SearchResults είναι αντίστοιχο του HomeFragment με τη διαφορά πως έχει και ένα TextView πάνω στο οποίο εμφανίζεται το κλειδί που έψαξε ο χρήστης για να βρει ταινίες. Μετά από κάτω του εμφανίζονται τα αποτελέσματα.

### MovieCard Fragment

To MovieCard είναι το fragment που παρουσιάζει κάθε ταινία ως μια κάρτα που θα μπει μέσα σε RecyclerView. Έχει ImageView για το poster της ταινίας, TextViews για το τίτλο, το release date, το rating, τους σκηνοθέτες και τη περίληψη της ταινίες.

Επίσης έχει τρία κουμπιά, το More Info που όταν πατιέται οδηγεί στη σελίδα της ταινίας στο imdb, το Favourite/Unfavourite που αντίστοιχα προσθέτει τη ταινία στη λίστα των αγαπημένων του χρήστη ή την αφαιρεί και το Dismiss που κάνει τη ταινία να μην εμφανίζεται για το συγκεκριμένο χρήστη μέχρι να διαγράψει την εφαρμογή ή να πατήσει το Reset Dismissed Movies από τα settings.

### Settings Activity

Αυτό το Activity(δηλαδή το .xml αρχείο του) έχει επιλογή για αλλαγή του email και password του χρήστη καθώς και κουμπιά για reset favourite movies, reset dismissed movies, back to home και log out. Αυτό το fragment ενεργοποιείται από τη κλάση SettingsActivity και προστίθενται οι απαραίτητοι listeners για το χειρισμό των κουμπιών που μπορεί να πατήσει ο χρήστης.

### MovieListAdapter

Αυτή είναι μια RecyclerView adapter κλάση που χρησιμοποιήτε για να περνάει δεδομένα από ένα ArrayList από MovieModels στο αντίστοιχο RecyclerView. Έχει και μια εσωτερική κλάση MovieViewHolder ή οποία δέχεται ένα View και βάζει σε αυτό τα χαρακτηριστικά του MovieModel.

Αυτό γίνεται με την bindMovie η οποία αφού δώσει στο View το περιεχόμενο που πρέπει να δείχνει, προσθέτει δυναμικά όσα genres πρέπει να δείχνει η αντίστοιχη ταινία και επίσης προσθέτει στα buttons του fragmentMovieCard αντίστοιχους OnClickListeners που να χειρίζονται το πάτημα των κουμπιών.

### WatchTogether κλάση

Αυτή είναι μια βοηθητική κλάση που επιτρέπει το setting και getting του id του user ώστε να είναι διαθέσιμο οποιαδήποτε στιγμή.

## Ρυθμίσεις

## Τυπική Χρήση