



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

Σχολή Τεχνολογίας

Τμήμα Ψηφιακών Συστημάτων

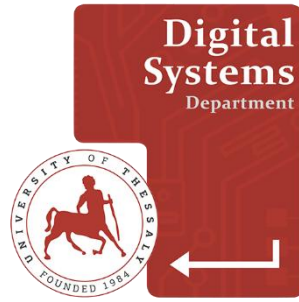
**FLIC: Εφαρμογή αλληλεπίδρασης και
δικτύωσης φίλων του κινηματογράφου**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Μαργαρίτα Χριστίδου (ΑΜ: 3121089)

Επιβλέπων: Βασιλική Κουτσονικόλα, ΕΔΙΠ

ΛΑΡΙΣΑ, Σεπτέμβριος 2025



UNIVERSITY OF THESSALY

School of Technology

Digital Systems Department

FLIC - Film Lovers Interacting and Connecting

BACHELOR THESIS

Margarita Christidou (RN: 3121089)

Supervisor: *Vasiliki Koutsonikola, EDIP*

LARISSA, September 2025

Υπεύθυνη Δήλωση περί Ακαδημαϊκής Δεοντολογίας και Πνευματικών Δικαιωμάτων

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα πτυχιακή εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον, και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο/Η Δηλών/ούσα

(Υπογραφή)

Μαργαρίτα Χριστίδου

09/09/2025

Εγκρίνεται από την Επιτροπή Εξέτασης:

Επιβλέπων/πουσα	Ονοματεπώνυμο Επιβλέποντα Βαθμίδα/ιδιότητα επιβλέποντα, Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Θεσσαλίας
Μέλος	Ονοματεπώνυμο Μέλους 1 Βαθμίδα/ιδιότητα μέλους 1, Τμήμα/Ιδρυμα μέλους 1
Μέλος	Ονοματεπώνυμο Μέλους 2 Βαθμίδα/ιδιότητα μέλους 2, Τμήμα/Ιδρυμα μέλους 2

Ημερομηνία έγκρισης: dd-mm-yyyy

Περίληψη

Στόχος αυτής της πτυχιακής εργασίας ήταν η δημιουργία μιας διαδικτυακής πλατφόρμας για φίλους του κινηματογράφου, όπου οι χρήστες μπορούν να καταγράφουν τις ταινίες που παρακολουθούν, να τις αξιολογούν, να γράφουν κριτικές, να δημιουργούν λίστες και να αλληλοεπιδρούν με άλλους χρήστες, προσφέροντας ένα ολοκληρωμένο περιβάλλον.

Για την επιτυχή υλοποίηση των παραπάνω λειτουργιών, έγινε αξιοποίηση Front-End (HTML5, CSS3, JavaScript, Bootstrap5) και Back-End (PHP, MySQL μέσω WAMP Server) τεχνολογιών. Παράλληλα, έγινε ενσωμάτωση του OMDb API για την άντληση πληροφοριών σχετικά με ταινίες. Η βάση δεδομένων σχεδιάστηκε με τη χρήση διαγραμμάτων οντοτήτων-συσχετίσεων (ERD), ώστε να υποστηρίζει λειτουργίες όπως λίστες, κριτικές, προτιμήσεις ειδών και σχέσεις φιλίας μεταξύ χρηστών. Επιπλέον, η αρχιτεκτονική client-server εξασφάλισε ευκολία στη συντήρηση και δυνατότητα μελλοντικής επέκτασης.

Ως αποτέλεσμα, αναπτύχθηκε μια πλήρως λειτουργική εφαρμογή, η οποία επιτρέπει στους χρήστες να δημιουργούν το δικό τους προφίλ, να μοιράζονται απόψεις και περιεχόμενο με άλλους και να λαμβάνουν προτάσεις ταινιών σύμφωνα με τις προτιμήσεις τους. Παρότι η εφαρμογή ολοκληρώθηκε σε λειτουργικό επίπεδο, υπάρχουν αρκετά περιθώρια βελτίωσης, όπως η αυτόματη ενημέρωση της βάσης δεδομένων μέσω API και η ενσωμάτωση συστήματος συστάσεων με τεχνητή νοημοσύνη. Συνεπώς, το έργο πέτυχε τους αρχικούς στόχους του, ενώ παράλληλα άνοιξε προοπτικές για μελλοντική εξέλιξη.

Abstract

The aim of this thesis was to create an online platform for film lovers, where users can keep track of the films they watch, rate them, write reviews, create lists, and interact with other users, offering a comprehensive environment.

To successfully implement the functionalities mentioned above, Front-End (HTML5, CSS3, JavaScript, Bootstrap5) and Back-End (PHP, MySQL via WAMP Server) technologies were used. At the same time, the OMDb API was integrated to extract information about movies. The database was designed using Entity-Relationship Diagrams (ERD) to support functions such as lists, reviews, genre preferences, and friendships between users. In addition, the client-server architecture ensured ease of maintenance and the possibility of future expansion.

As a result, a fully functional application was developed, allowing users to create their own profiles, share opinions and content with others, and receive movie recommendations based on their preferences. Although the application was completed at a functional level, there is room for improvement, such as automatic database updates via API and the integration of an artificial intelligence recommendation system. Therefore, the project achieved its initial goals while opening up prospects for future development.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω όλους όσους ήταν δίπλα μου αυτά τα τέσσερα χρόνια των σπουδών μου. Πρώτα απ' όλα τους γονείς μου για τη στήριξη και τις συμβουλές τους, τους καθηγητές μου για τη βοήθεια και τις γνώσεις που μου έδωσαν, αλλά και τους φίλους μου που έκαναν την καθημερινότητα καλύτερη με την αγάπη και την στήριξη τους όταν τη χρειαζόμουν. Ιδιαίτερα, θα ήθελα να πω ένα μεγάλο ευχαριστώ στην κα Κουτσονικόλα, που ήταν πάντα πρόθυμη να με καθοδηγήσει και να με ενθαρρύνει κατά την διάρκεια της γραφής της πτυχιακής μου εργασίας. Χωρίς την δική σας βοήθεια, δεν θα τα κατάφερνα το ίδιο. Σας ευχαριστώ πολύ όλους.

Μαργαρίτα Χριστίδου

09/09/2025

Περιεχόμενα

ΠΕΡΙΛΗΨΗ	VII
ABSTRACT	IX
ΕΥΧΑΡΙΣΤΙΕΣ.....	XI
ΠΕΡΙΕΧΟΜΕΝΑ.....	XIII
1 ΕΙΣΑΓΩΓΗ	1
2 ΤΡΕΧΟΥΣΑ ΚΑΤΑΣΤΑΣΗ.....	3
2.1 LETTERBOXD	3
2.2 IMDb	4
2.3 ROTTEN TOMATOES.....	5
2.4 ΣΥΓΚΡΙΣΗ ΜΕ ΤΗΝ FLIC	6
3 ΤΕΧΝΟΛΟΓΙΕΣ	9
3.1 FRONT-END ΤΕΧΝΟΛΟΓΙΕΣ	9
3.1.1 HTML5	9
3.1.2 CSS3.....	9
3.1.3 Javascript	10
3.1.4 Bootstrap5	10
3.2 BACK-END ΤΕΧΝΟΛΟΓΙΕΣ.....	10
3.2.1 PHP (Hypertext Preprocessor)	10
3.2.2 Ενσωμάτωση Εξωτερικού API (OMDb)	11
3.3 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ	11
3.3.1 Σχεδιασμός και Υλοποίηση	11
3.3.2 Τοπικό Περιβάλλον Ανάπτυξης.....	12
3.3.3 Διάγραμμα Οντοτήτων-Συσχετίσεων (ERD).....	12
3.4 ΆΛΛΕΣ ΠΑΡΑΤΗΡΗΣΕΙΣ.....	14
4 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΕΦΑΡΜΟΓΗΣ	17
4.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ CLIENT-SERVER.....	17

4.1.1	Ανάλυση της Αρχιτεκτονικής	17
4.1.2	Πλεονεκτήματα της Αρχιτεκτονικής	18
4.2	ΛΕΙΤΟΥΡΓΙΑ ΟΘΟΝΩΝ ΕΦΑΡΜΟΓΗΣ.....	18
4.2.1	Αρχική Σελίδα (Homepage).....	19
4.2.2	Εγγραφή (Sign Up).....	19
4.2.3	Σύνδεση (Log In) και Αποσύνδεση (Log out)	19
4.2.4	Σελίδα Ταινιών (Films)	19
4.2.5	Σελίδα Κριτικών (Reviews)	19
4.2.6	Σελίδα Λιστών (Lists).....	19
4.2.7	Αναζήτηση (Search)	20
4.2.8	Προφίλ Χρήστη (My Profile)	20
4.2.9	Δημόσιο Προφίλ (Public Profile)	20
5	Η ΕΦΑΡΜΟΓΗ.....	23
6	ΣΥΜΠΕΡΑΣΜΑΤΑ	41
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	43
	ΠΑΡΑΡΤΗΜΑ	45

1 Εισαγωγή

Η ραγδαία ανάπτυξη του διαδικτύου και των τεχνολογιών πληροφορικής έχει επιφέρει σημαντικές αλλαγές στον τρόπο με τον οποίο οι άνθρωποι επικοινωνούν, μοιράζονται πληροφορίες και καταναλώνουν ψυχαγωγικό περιεχόμενο. Ο κινηματογράφος, ως μία από τις πιο δημοφιλείς μορφές τέχνης και ψυχαγωγίας, αποτελεί αντικείμενο ενδιαφέροντος για εκατομμύριο ανθρώπους παγκοσμίως. Παράλληλα, οι διαδικτυακές πλατφόρμες κοινωνικής δικτύωσης δίνουν την δυνατότητα στους χρήστες τους να ανταλλάσσουν απόψεις, να αξιολογούν περιεχόμενο και να λαμβάνουν εξατομικευμένες προτάσεις ανάλογα με τις προτιμήσεις τους.

Σκοπός της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη μιας διαδικτυακής εφαρμογής που θα λειτουργεί ως κοινωνική πλατφόρμα για τους φίλους του κινηματογράφου. Κάθε εγγεγραμμένος χρήστης θα μπορεί να δημιουργεί το προσωπικό του προφίλ, να προσθέτει τις ταινίες που έχει παρακολουθήσει, να τις βαθμολογεί και να δηλώνει τις προτιμήσεις του σε συγκεκριμένα είδη, όπως δράμα, κωμωδία, περιπέτεια κ.ά. Με βάση αυτά τα δεδομένα, το σύστημα θα παρέχει εξατομικευμένες προτάσεις ταινιών με στόχο τη βελτίωση της εμπειρίας εύρεσης και επιλογής περιεχομένου.

Η εφαρμογή θα προσφέρει, επιπλέον, δυνατότητες κοινωνικής αλληλεπίδρασης μεταξύ των χρηστών μέσω αξιολόγησης ταινιών που έχουν προτείνει άλλοι, ανάγνωσης και σχολιασμού κριτικών, καθώς και δημιουργίας λιστών με ταινίες που οι χρήστες επιθυμούν να παρακολουθήσουν στο μέλλον ή τους έχουν κεντρίσει το ενδιαφέρον. Με αυτόν τον τρόπο, η πλατφόρμα στοχεύει όχι μόνο στην παροχή χρήσιμων πληροφοριών, αλλά και στην ενίσχυση της κοινότητας των φίλων του κινηματογράφου.

Προτεινόμενη Διάρθρωση Πτυχιακής

Στη συνέχεια, γίνεται σύγκριση της εφαρμογής με ήδη υπάρχουσες εφαρμογές στην αγορά, αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν, η αρχιτεκτονική της εφαρμογής, η χρήση της και τέλος, τα συμπεράσματα και οι μελλοντικές επεκτάσεις της.

2 Τρέχουσα κατάσταση

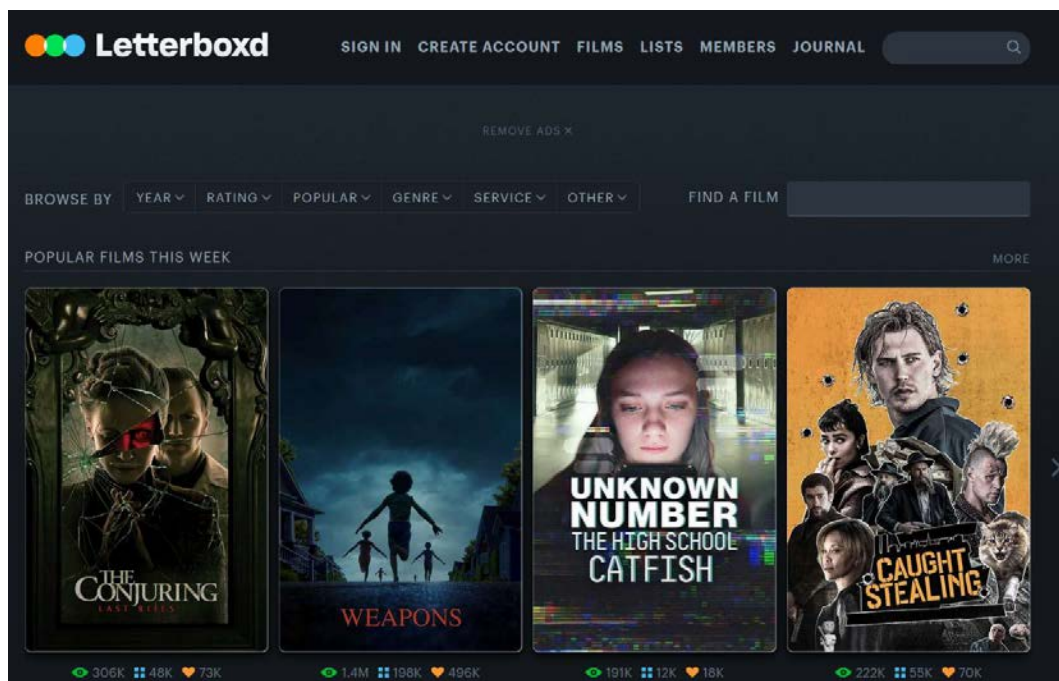
Σήμερα, υπάρχουν αρκετές γνωστές πλατφόρμες σχεδιασμένες για τους φίλους του κινηματογράφου, που προσφέρουν παρόμοιες λειτουργίες με την εφαρμογή FLIC. Κάποιες από τις πιο δημοφιλείς και χρήσιμες πλατφόρμες είναι το *Letterboxd*, το *IMDb* και το *Rotten Tomatoes*. Όλες αυτές δίνουν την δυνατότητα στους χρήστες να καταγράφουν ποιες ταινίες έχουν παρακολουθήσει, να τις βαθμολογούν, να δημιουργούν λίστες και να μοιράζονται τις απόψεις ή τις κριτικές τους, με άλλους που έχουν το ίδιο πάθος για τον κινηματογράφο.

2.1 Letterboxd

Το Letterboxd αποτελεί μια κοινωνική πλατφόρμα που δίνει μεγάλη σημασία στην αλληλεπίδραση μεταξύ των χρηστών. Επιτρέπει την καταγραφή των ταινιών που βλέπει κάποιος, τη βαθμολόγησή τους, καθώς και τη συγγραφή προσωπικών κριτικών. Παράλληλα, οι χρήστες μπορούν να δημιουργούν και να μοιράζονται λίστες ταινιών, ενώ υπάρχει η δυνατότητα αλληλοεπίδρασης μεταξύ τους, μέσω δημιουργίας σχολίων σε κριτικές και ακολουθίας άλλων χρηστών.

Στη συγκεκριμένη πλατφόρμα δίνεται έμφαση στη δημιουργία μιας ενεργής κοινότητας σινεφίλ, όπου οι χρήστες μοιράζονται συνεχώς τις απόψεις και τις προτάσεις τους.

[1]

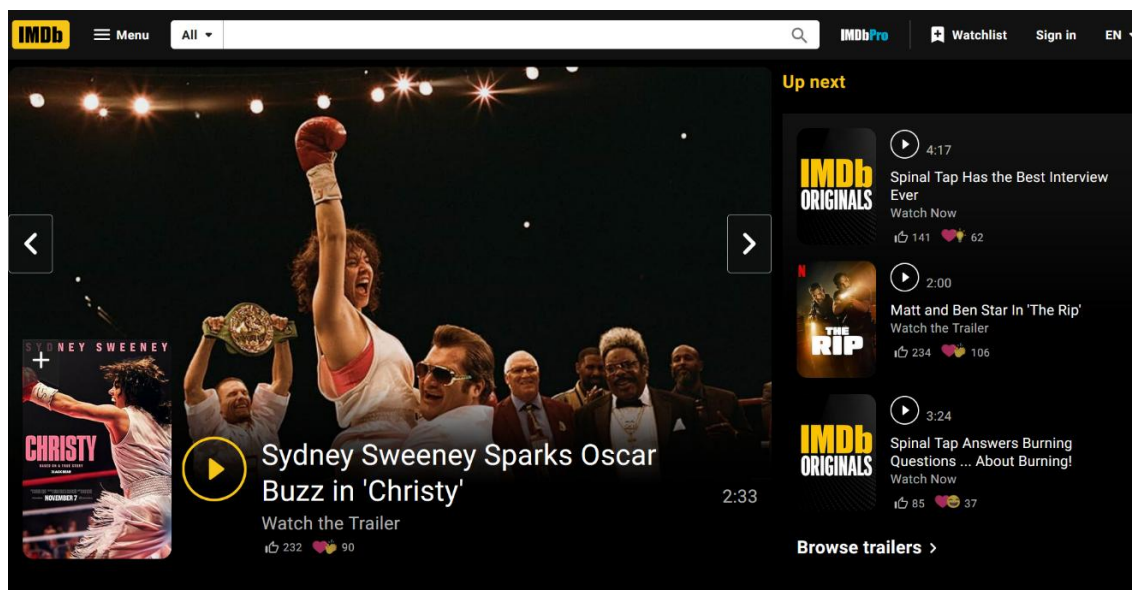


Εικόνα 1: Η πλατφόρμα Letterboxd

2.2 IMDb

Το IMDb είναι μια πλατφόρμα όπου κατά κύριο λόγο χρησιμοποιείται ως μια εκτεταμένη βάση δεδομένων ταινιών και σειρών, που παρέχει λεπτομερείς πληροφορίες για κάθε τίτλο, όπως τους ηθοποιούς, τους δημιουργούς και άλλα τεχνικά χαρακτηριστικά. Αν και προσφέρει δυνατότητες βαθμολόγησης τίτλων από χρήστες και δημιουργίας σχολίων, δεν χρησιμοποιείται ιδιαίτερα για τον συγκεκριμένο σκοπό.

Η βασική λειτουργία αυτής της πλατφόρμας είναι να παρέχει αξιόπιστες και λεπτομερείς πληροφορίες, καθιστώντας την ως ένα αξιόπιστο σημείο αναφοράς για τους χρήστες. [2]

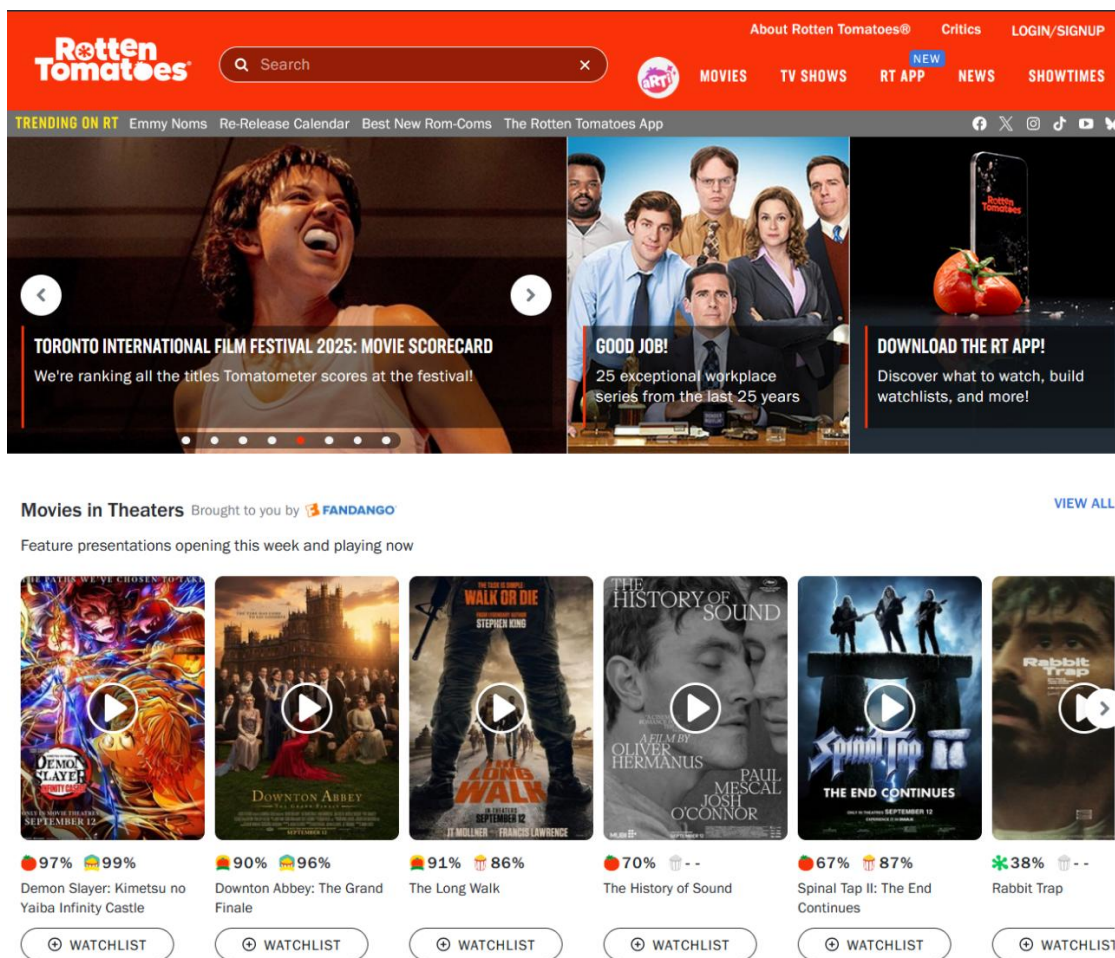


Εικόνα 2: Η πλατφόρμα IMDb

2.3 Rotten Tomatoes

Το Rotten Tomatoes είναι μια πλατφόρμα που λειτουργεί ως εργαλείο ενημέρωσης και αξιολόγησης ταινιών και σειρών. Συγκεντρώνει κριτικές από διάφορους επαγγελματίες κριτικούς και παρουσιάζει την συνολική αποδοχή κάθε ταινίας μέσω του συστήματος βαθμολόγησης Fresh/Rotten.

Αν και επιτρέπει σε απλούς χρήστες να αφήνουν αξιολογήσεις, δεν δίνει ιδιαίτερη έμφαση στην κοινωνική αλληλεπίδραση ή στη δημιουργία κοινοτήτων. Αντίθετα, αποτελεί έναν αξιόπιστο οδηγό ποιότητας που βοηθά το κοινό να επιλέγει τι να παρακολουθήσει, βασιζόμενο κυρίως στις απόψεις των ειδικών. [3]



Εικόνα 3: Η πλατφόρμα Rotten Tomatoes

2.4 Σύγκριση με την FLIC

Η εφαρμογή FLIC συγκεντρώνει τις βασικές λειτουργίες που βρίσκουμε σε όλες τις παραπάνω πλατφόρμες: καταγραφή και βαθμολόγηση ταινιών, δυνατότητα δημιουργίας λιστών και κοινωνική αλληλεπίδραση.

Αν και δεν προσφέρει κάποιον πρωτοποριακό ή ριζικά διαφορετικό μηχανισμό, δίνει έμφαση στο να προσφέρει μια απλή, φιλική και λειτουργική εμπειρία χρήστη. Η FLIC στοχεύει να καλύψει τις ανάγκες του μέσου σινεφίλ χρήστη, με όλα όσα θεωρούνται απαραίτητα, συγκεντρωμένα σε μια εύχρηστη και ολοκληρωμένη πλατφόρμα.

Στον παρακάτω Πίνακα 1, δίνεται μια συνοπτική σύγκριση μεταξύ όλων των εφαρμογών φίλων του κινηματογράφου που αναφέρθηκαν στις παραπάνω ενότητες.

Πίνακας 1: Συγκριτικός Πίνακας Εφαρμογών Φίλων του Κινηματογράφου

	Letterboxd	IMDb	Rotten Tomatoes	FLIC
Login	Ναι	Ναι	Ναι	Ναι
Λίστες	Ναι	Ναι	Περιορισμένες (όχι δυναμικές)	Ναι
Κριτικές-Σχόλια	Ναι	Ναι	Μόνο σύντομα σχόλια, κυρίως κριτικοί	Ναι
Αξιολόγηση ταινιών	Ναι (0.5-5)	Ναι (1-10)	Ναι (0.5-5)	Ναι (1-5)
Φίλοι	Ναι	Όχι	Όχι	Ναι
Προτάσεις ταινιών	Περιορισμένες (Βάσει φίλων και λιστών)	Ναι	Περιορισμένες (Δημοφιλείς)	Περιορισμένες (Ενδιαφέροντα)

3 Τεχνολογίες

Σε αυτή την ενότητα παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής. Αναλύονται ξεχωριστά οι τεχνολογίες που επιλέχθηκαν για το front-end, το back-end και τη βάση δεδομένων, ώστε να δοθεί πλήρης εικόνα της τεχνικής υποδομής της πλατφόρμας. Κάθε τεχνολογία επιλέχθηκε ώστε να καλύπτει όσο το δυνατόν καλύτερα τις ανάγκες και τις λειτουργίες της εφαρμογής.

3.1 Front-end Τεχνολογίες

Το front-end της εφαρμογής αφορά το κομμάτι που βλέπει και με το οποίο αλληλεπιδρά ο τελικός χρήστης. Περιλαμβάνει όλες τις τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για τη δημιουργία της εμφάνισης, της διάταξης των στοιχείων και της αλληλεπίδρασης σε πραγματικό χρόνο. Στόχος του είναι να προσφέρει ένα φιλικό, λειτουργικό και αισθητικά ευχάριστο περιβάλλον, το οποίο να προσαρμόζεται σε διαφορετικές συσκευές (υπολογιστές, κινητά, tablets) και να ανταποκρίνεται γρήγορα στις ενέργειες του χρήστη.

3.1.1 HTML5

Η HTML5 χρησιμοποιείται για τη δομή και την οργάνωση του περιεχομένου. Καθορίζει πώς θα τοποθετούνται και θα εμφανίζονται τα στοιχεία στην οθόνη του χρήστη, ενώ παράλληλα εξασφαλίζει συμβατότητα με όλους τους σύγχρονους browsers και διαφορετικές συσκευές. [4]

3.1.2 CSS3

Με την χρήση της CSS3, καθορίζεται η εμφάνιση και η αισθητική της εφαρμογής. Για παράδειγμα, καθορίζει τα χρώματα, τη τυπογραφία και τις αποστάσεις μεταξύ διάφορων στοιχείων που υπάρχουν σε κάθε σελίδα. Στην εφαρμογή, βοηθά ώστε όλες οι σελίδες να έχουν ομοιόμορφο και ελκυστικό σχεδιασμό, προσφέροντας ευχάριστη εμπειρία χρήσης. [5]

3.1.3 Javascript

Οι κύριοι λόγοι για τους οποίους χρησιμοποιείται η γλώσσα JavaScript στην ανάπτυξη της συγκεκριμένης εφαρμογής είναι οι εξής:

- Έλεγχο εγκυρότητας δεδομένων σε πραγματικό χρόνο πριν την αποστολή στον διακομιστή.
- Δυναμικές λειτουργίες και άμεση αλληλεπίδραση χωρίς πλήρη ανανέωση σελίδας.

Σε ορισμένα σημεία συνεργάζεται με την PHP, ενισχύοντας την ταχύτητα και την ευελιξία της εφαρμογής. [6]

3.1.4 Bootstrap5

Το Bootstrap είναι ένα framework που παρέχει έτοιμα στοιχεία σχεδίασης (όπως κουμπιά, φόρμες, μενού κτλ.) που προσαρμόζονται αυτόματα στις διαφορετικές διαστάσεις οθονών.

Χρησιμοποιώντας το Bootstrap, η εφαρμογή μπορεί:

- Να λειτουργεί και να εμφανίζεται σωστά σε υπολογιστές, κινητά και tablets.
- Να διατηρεί ομοιομορφία στον σχεδιασμό, ακόμα και όταν προστίθενται νέες σελίδες ή λειτουργίες.
- Να επιταχύνει την ανάπτυξη, αφού πολλά γραφικά στοιχεία δημιουργούνται με ελάχιστο κώδικα. [7]

3.2 Back-end Τεχνολογίες

Το back-end της εφαρμογής είναι το τμήμα που λειτουργεί «στο παρασκήνιο» και είναι υπεύθυνο για την επεξεργασία των δεδομένων, την εκτέλεση της επιχειρησιακής λογικής και την επικοινωνία με τη βάση δεδομένων και εξωτερικές υπηρεσίες. Μέσω του back-end διαχειρίζονται τα αιτήματα των χρηστών, υλοποιούνται οι λειτουργίες CRUD και πραγματοποιείται η διασύνδεση με το OMDb API για την άντληση επιπλέον πληροφοριών ταινιών. Στόχος του είναι να εξασφαλίζει γρήγορη, ασφαλή και αξιόπιστη λειτουργία της εφαρμογής.

3.2.1 PHP (Hypertext Preprocessor)

Η PHP είναι η βασική γλώσσα προγραμματισμού που τρέχει στον διακομιστή και υλοποιεί την επιχειρησιακή λογική της εφαρμογής. [8]

Στην συγκεκριμένη εφαρμογή:

- Διαχειρίζεται τις αιτήσεις που στέλνει ο χρήστης (requests) και επιστρέφει τα κατάλληλα αποτελέσματα ή σελίδες.
- Ελέγχει τις συνεδρίες (sessions), ώστε να γνωρίζει ποιος χρήστης είναι συνδεδεμένος κάθε φορά.
- Επεξεργάζεται δεδομένα από φόρμες, αναζητήσεις και αλληλεπιδράσεις.
- Επικοινωνεί με τη βάση δεδομένων MySQL για να ανακτά και να αποθηκεύει πληροφορίες (π.χ. λίστες ταινιών, αξιολογήσεις, προτιμήσεις).

Επιλέχθηκε λόγω της ευρείας χρήσης της στον χώρο της ανάπτυξης εφαρμογών διαδικτύου, της σταθερότητας της και της συμβατότητας της με μεγάλο πλήθος εργαλείων.

3.2.2 Ενσωμάτωση Εξωτερικού API (OMDb)

Το OMDb API (The Open Movie Database) είναι μια εξωτερική διαδικτυακή υπηρεσία που παρέχει αναλυτικά δεδομένα για ταινίες, όπως:

- Τίτλος, έτος κυκλοφορίας, είδος
- Περιγραφή/περίληψη
- Σκηνοθέτης, σεναριογράφος, ηθοποιοί
- Κριτικές και βαθμολογίες από διάφορες πηγές
- Αφίσα και εικόνες

Η εφαρμογή συνδέεται με το OMDb API κάθε φορά που θέλει να εμφανίσει λεπτομέρειες για μια ταινία που ενδιαφέρει τον χρήστη. Με αυτόν τον τρόπο, δεν χρειάζεται να αποθηκεύονται όλα τα στοιχεία στη δική μας βάση αλλά αντλούνται από μια αξιόπιστη πηγή. [9]

3.3 Βάση Δεδομένων

3.3.1 Σχεδιασμός και Υλοποίηση

Για την ανάπτυξη της βάσης δεδομένων της εφαρμογής, χρησιμοποιήθηκε η MySQL, ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Η MySQL είναι από τις πιο διαδεδομένες και αξιόπιστες επιλογές παγκοσμίως με βασικό της χαρακτηριστικό την οργάνωση των δεδομένων σε πίνακες (σχέσεις), οι οποίοι συνδέονται μεταξύ τους με

λογικές σχέσεις, καθιστώντας τη διαχείριση και την αναζήτηση των δεδομένων πιο εύκολη και αποτελεσματική. [10]

Η βάση δεδομένων υλοποιήθηκε σε τοπικό περιβάλλον ανάπτυξης και υποστηρίζει τις βασικές λειτουργίες CRUD (Create, Read, Update, Delete), οι οποίες εξυπηρετούν τις ανάγκες της εφαρμογής σε επίπεδο αποθήκευσης και επεξεργασίας πληροφορίας.

3.3.2 Τοπικό Περιβάλλον Ανάπτυξης

Για την τοπική εγκατάσταση και λειτουργία της MySQL χρησιμοποιήθηκε ο WAMP Server. Το WAMP (Windows, Apache, MySQL, PHP) είναι ένα περιβάλλον ανάπτυξης το οποίο εγκαθιστά και διαχειρίζεται όλα τα απαραίτητα στοιχεία για τη δημιουργία και λειτουργία διαδικτυακών εφαρμογών σε έναν τοπικό υπολογιστή. Συγκεκριμένα, περιλαμβάνει:

- Τον *Apache*, έναν δημοφιλή διακομιστή (server) ιστοσελίδων,
- Τη *MySQL* για τη διαχείριση της βάσης δεδομένων,
- Την *PHP* για την υλοποίηση της επιχειρησιακής λογικής στο back-end.

Με το WAMP, το back-end της εφαρμογής αποστέλλει εντολές CRUD (Create, Read, Update, Delete) προς τη MySQL μέσω του περιβάλλοντος που παρέχει ο WAMP Server, και λαμβάνει πίσω τα απαιτούμενα δεδομένα από τη βάση. Αυτό προσφέρει μια ολοκληρωμένη και εύχρηστη υποδομή για την ανάπτυξη, τον έλεγχο και τη συντήρηση της βάσης δεδομένων της εφαρμογής σε τοπικό επίπεδο πριν την ενδεχόμενη ανάπτυξη σε διαδικτυακό περιβάλλον. [11]

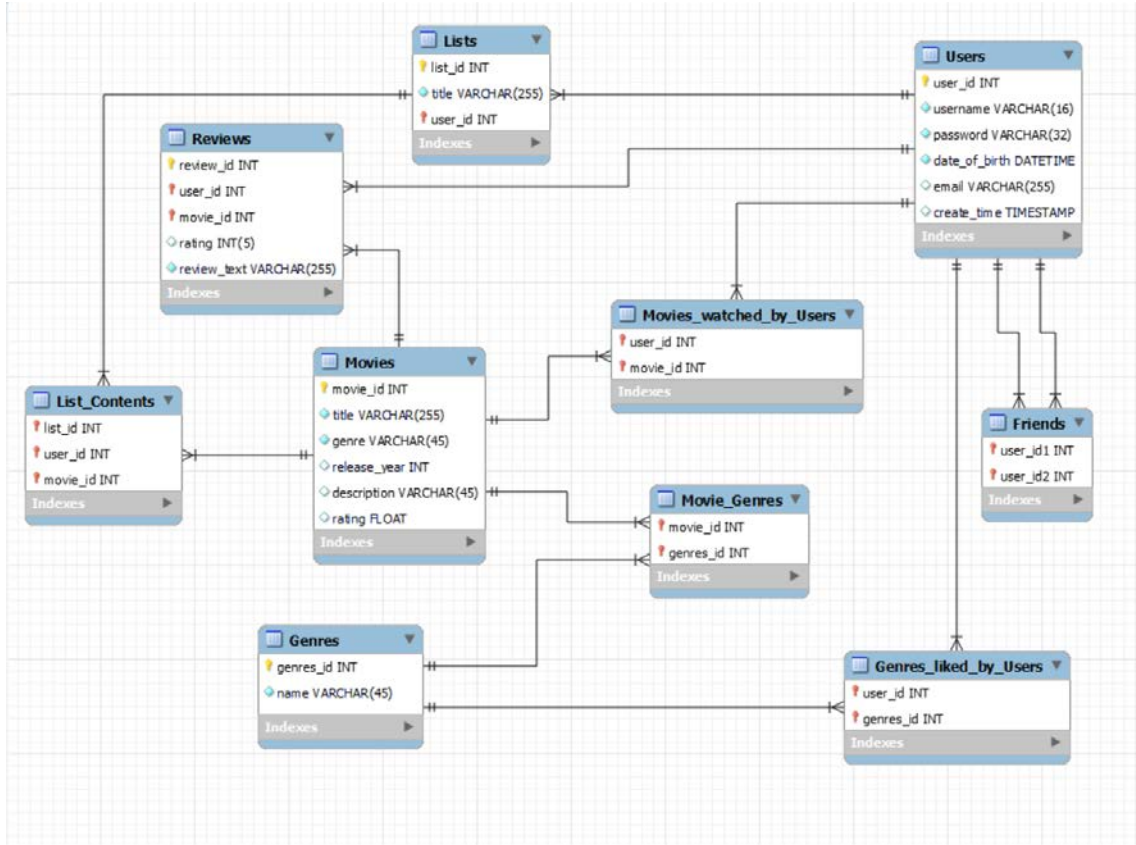
3.3.3 Διάγραμμα Οντοτήτων-Συσχετίσεων (ERD)

Το *Entity-Relationship Diagram (ERD)* αποτελεί ένα βασικό εργαλείο για τον σχεδιασμό και την απεικόνιση της δομής μιας βάσης δεδομένων. Μέσω αυτού, μπορούμε να αναπαραστήσουμε τις οντότητες (entities) της εφαρμογής, τις σχέσεις (relationships) μεταξύ τους και τα βασικά χαρακτηριστικά (attributes) κάθε οντότητας.

Ο σχεδιασμός του ERD της εφαρμογής έγινε σε δύο στάδια: τη δημιουργία ενός αρχικού διαγράμματος και την τελική του μορφή, η οποία προέκυψε μετά από αλλαγές και βελτιστοποιήσεις κατά την ανάπτυξη της.

3.3.3.1 Αρχικό Σχεδιάγραμμα

Το αρχικό ERD, που παρουσιάζεται στην Εικόνα 1, αποτελεί την πρώτη προσέγγιση για τη σχεδίαση της βάσης δεδομένων. Στόχος του ήταν η απλή και άμεση αναπαράσταση των βασικών λειτουργικών απαιτήσεων του συστήματος.



Εικόνα 4: Αρχικό Διάγραμμα Οντοτήτων-Συσχετίσεων (ERD)

3.3.3.2 Ανάλυση του Αρχικού Σχεδιασμού

Το αρχικό μοντέλο αποτελείται από τις ακόλουθες βασικές οντότητες και τις σχέσεις τους:

- **Users:** Περιλαμβάνει τις πληροφορίες των χρηστών της εφαρμογής. Τα βασικά χαρακτηριστικά (user_id, username, email) επιτρέπουν την ταυτοποίηση και την πρόσβαση στην πλατφόρμα.
- **Movies:** Αποθηκεύει τα δεδομένα των ταινιών, όπως τον τίτλο (title), την περιγραφή (description) και το έτος κυκλοφορίας (release_year).

- **Reviews:** Συνδέεται με τις οντότητες Users και Movies, επιτρέποντας στους χρήστες να καταγράφουν κριτικές για τις ταινίες. Περιέχει την βαθμολογία (rating) και το κείμενο της κριτικής (review_text).
- **Lists:** Επιτρέπει στους χρήστες να δημιουργούν προσαρμοσμένες λίστες ταινιών.
- **Genres:** Αποθηκεύει τις κατηγορίες (είδη) των ταινιών. Η σύνδεση με τον πίνακα Movies πραγματοποιείται μέσω του ενδιαμέσου πίνακα **Movie_Genres**, που επιτρέπει σε μια ταινία να ανήκει σε πολλά είδη (σχέση Many-to-Many).
- **Friends:** Δημιουργεί μια σχέση αυτοσυσχέτισης (self-referencing relationship) στον πίνακα Users, επιτρέποντας την καταγραφή των σχέσεων φιλίας μεταξύ των χρηστών.
- **Movies_watched_by_Users:** Πρόκειται για έναν βοηθητικό πίνακα που καταγράφει ποιες ταινίες έχει παρακολουθήσει κάθε χρήστης.
- **Genres_liked_by_Users:** Καταγράφει τις προτιμήσεις των χρηστών ως προς τα είδη ταινιών.

3.3.3.3 Τελικές Τροποποιήσεις

Κατά τη διάρκεια της ανάπτυξης του FLIC, ο αρχικός σχεδιασμός υποβλήθηκε σε τροποποιήσεις για να ανταποκριθεί στις πραγματικές ανάγκες της εφαρμογής και να βελτιωθεί η λειτουργικότητά της. Οι σημαντικότερες αλλαγές περιλαμβάνουν την προσθήκη τριών νέων πινάκων στην βάση καθώς και την προσθήκη της στήλης «imdb_id» στον πίνακα Movies όπου βοηθά στην εμφάνιση των λεπτομερειών κάθε ταινίας που βρίσκεται στην βάση δεδομένων. Οι τρεις νέοι πίνακες είναι οι εξής:

- **List_likes:** Διαχειρίζεται τα likes των λιστών από τους χρήστες.
- **Review_likes:** Διαχειρίζεται τα likes των κριτικών από τους χρήστες.
- **Review_comments:** Αποθηκεύει τα σχόλια κάθε κριτικής που δημιουργούν οι χρήστες.

3.4 Άλλες Παρατηρήσεις

Για το γέμισμα του πίνακα Movies της βάσης δεδομένων χρησιμοποιήθηκε η τεχνητή νοημοσύνη (ChatGPT) για την δημιουργία των Insert εντολών. Επιπλέον, οι φωτογραφίες που χρησιμοποιήθηκαν στην Αρχική Σελίδα της εφαρμογής αντλήθηκαν από το διαδίκτυο, συγκεκριμένα από την ιστοσελίδα Wallpaper Abyss [12] για τις εικόνες του carou-

sel και την ιστοσελίδα FREEPIK [13] για τις υπόλοιπες φωτογραφίες και τις εικόνες προφίλ (avatars). Τέλος, αντλήθηκε κώδικας από το site του Bootstrap5 framework [7] για την υλοποίηση του carousel της αρχικής σελίδας, καθώς και για την υποβολή βαθμολογίας στις κριτικές με χρήση αστεριών από την σελίδα BootstrapExamples [14].

4 Αρχιτεκτονική Εφαρμογής

Η σωστή αρχιτεκτονική αποτελεί τη βάση για κάθε επιτυχημένη εφαρμογή, επηρεάζοντας τόσο τη λειτουργικότητα όσο και τη μελλοντική της εξέλιξη. Σε αυτή την ενότητα παρουσιάζεται η αρχιτεκτονική της εφαρμογής FLIC, με έμφαση στον τρόπο που διασφαλίζει την ευκολία χρήσης, την ασφάλεια και τη δυνατότητα μελλοντικής επέκτασης.

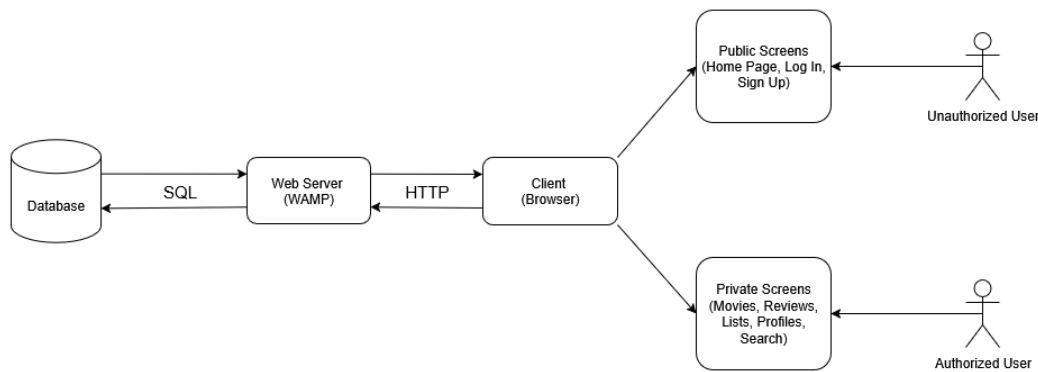
4.1 Αρχιτεκτονική Client-Server

Η αρχιτεκτονική client-server αποτελεί ένα από τα πιο διαδεδομένα πρότυπα ανάπτυξης λογισμικού για διαδικτυακές εφαρμογές. Στο μοντέλο αυτό, ο client (συνήθως ο browser του χρήστη) αναλαμβάνει την παρουσίαση και την αλληλεπίδραση με τον χρήστη, ενώ ο server διαχειρίζεται την επιχειρησιακή λογική, επεξεργάζεται τα αιτήματα και επικοινωνεί με τη βάση δεδομένων, όπου αποθηκεύονται όλες οι απαραίτητες πληροφορίες. Ο διαχωρισμός αυτός επιτρέπει μεγαλύτερη ευελιξία, καλύτερη απόδοση και ευκολότερη συντήρηση της εφαρμογής.

4.1.1 Ανάλυση της Αρχιτεκτονικής

Η εφαρμογή FLIC αξιοποιεί το μοντέλο client-server για την υλοποίησή της, ακολουθώντας τις βέλτιστες πρακτικές ανάπτυξης web εφαρμογών. Όπως βλέπουμε και στην Εικόνα 2, η εφαρμογή αποτελείται από τρία βασικά επιμέρους συστήματα:

- **Client (Browser):** Η διεπαφή με τον χρήστη, όπου εμφανίζονται όλες οι οθόνες της εφαρμογής ανάλογα με τα δικαιώματα του χρήστη (επισκέπτης ή εγγεγραμμένος).
- **Web Server (WAMP):** Υλοποιεί τη λογική, διαχειρίζεται τα αιτήματα και επικοινωνεί με τη βάση δεδομένων.
- **Database (MySQL):** Αποθηκεύει όλες τις πληροφορίες (χρήστες, ταινίες, λίστες, κριτικές).



Εικόνα 5: Σχεδιάγραμμα Αρχιτεκτονικής Client-Server της Εφαρμογής

4.1.2 Πλεονεκτήματα της Αρχιτεκτονικής

Η επικοινωνία μεταξύ των μερών γίνεται μέσω HTTP (client-server) και SQL (server-database). Ο διαχωρισμός των δημοσίων και ιδιωτικών οθονών και ο κεντρικός έλεγχος των δικαιωμάτων πρόσβασης εξασφαλίζουν απρόσκοπτη λειτουργία και ασφάλεια.

Η συγκεκριμένη αρχιτεκτονική επιλέχθηκε κυρίως για λόγους απλότητας και ευκολίας υλοποίησης της. Επειδή η εφαρμογή βρίσκεται σε αρχικό στάδιο ανάπτυξης και ο όγκος των χρηστών και των λειτουργιών δεν δικαιολογεί πιο σύνθετες λύσεις, το κλασικό μοντέλο client-server καλύπτει πλήρως τις ανάγκες του συστήματος. Με αυτόν τον τρόπο διευκολύνεται η συντήρηση, η γρήγορη επίλυση σφαλμάτων και η μελλοντική επέκταση της εφαρμογής, εάν αυτό είναι απαραίτητο.

Πέρα από την απλότητα, η αρχιτεκτονική αυτή διασφαλίζει την ασφάλεια, καθώς επιτρέπει τον έλεγχο των δικαιωμάτων πρόσβασης και την προστασία των δεδομένων μέσω του web server. Τέλος, προσφέρει επεκτασιμότητα, καθώς μπορεί να αναβαθμιστεί ή να προσαρμοστεί εύκολα αν αυξηθούν οι ανάγκες της εφαρμογής στο μέλλον.

4.2 Λειτουργία Οθονών Εφαρμογής

Η λειτουργία της εφαρμογής FLIC οργανώνεται γύρω από ένα σύνολο διασυνδεδεμένων οθονών, που επιτρέπουν την πρόσβαση σε περιεχόμενο (ταινίες, κριτικές, λίστες) και τη δικτύωση με άλλους χρήστες. Το παρακάτω σχεδιάγραμμα, στην Εικόνα 3, παρουσιάζει την αρχιτεκτονική πλοήγησης, ενώ στη συνέχεια περιγράφονται συνοπτικά οι κύριες οθόνες της εφαρμογής.

4.2.1 Αρχική Σελίδα (Homepage)

Η αρχική σελίδα αποτελεί το σημείο εκκίνησης της εφαρμογής. Από εδώ ο χρήστης μπορεί είτε να προχωρήσει σε *Εγγραφή (Sign Up)*, δημιουργώντας νέο λογαριασμό είτε να συνδεθεί με υπάρχοντα λογαριασμό μέσω της επιλογής *Σύνδεση (Log In)*.

4.2.2 Εγγραφή (Sign Up)

Η διαδικασία εγγραφής επιτρέπει τη δημιουργία νέου λογαριασμού με βασικά στοιχεία (email, κωδικός, όνομα χρήστη). Μετά την ολοκλήρωση, ο χρήστης αποκτά πρόσβαση στις βασικές λειτουργίες της εφαρμογής.

4.2.3 Σύνδεση (Log In) και Αποσύνδεση (Log out)

Η οθόνη σύνδεσης δίνει πρόσβαση στους εγγεγραμμένους χρήστες. Μετά την επιτυχή σύνδεση, ο χρήστης μπορεί να πλοηγηθεί σε όλες τις επιμέρους οθόνες. Ανά πάσα στιγμή υπάρχει η δυνατότητα *Αποσύνδεσης (Log out)*, που επιστρέφει στην Αρχική Σελίδα.

4.2.4 Σελίδα Ταινιών (Films)

Η οθόνη *Films* λειτουργεί ως κεντρικός κατάλογος ταινιών. Επιλέγοντας μια ταινία, ο χρήστης μεταφέρεται στη *Σελίδα Ταινίας (Movie Page)*, όπου προβάλλονται βασικές λεπτομέρειες για την ταινία που επιλέχθηκε, όπως η βαθμολογία της, η περιγραφή της και οι ηθοποιοί που συμμετείχαν σε αυτή.

4.2.5 Σελίδα Κριτικών (Reviews)

Η οθόνη *Reviews* συγκεντρώνει κριτικές που έχουν αναρτήσει οι χρήστες. Κάθε κριτική μπορεί να ανοιχτεί ξεχωριστά στην οθόνη *Open Review*, η οποία περιλαμβάνει το πλήρες περιεχόμενο της κριτικής. Από εκεί, ο χρήστης μπορεί να πλοηγηθεί στη σχετική Σελίδα Ταινίας (*Movie Page*).

4.2.6 Σελίδα Λιστών (Lists)

Στην οθόνη *Lists* προβάλλονται όλες οι διαθέσιμες λίστες. Επιλέγοντας μία, ο χρήστης μεταφέρεται στην οθόνη *Open List*, όπου φαίνονται οι ταινίες που περιλαμβάνονται, με δυνατότητα μετάβασης στη Σελίδα Ταινίας (*Movie Page*).

4.2.7 Αναζήτηση (Search)

Μέσω της οθόνης *Search*, ο χρήστης μπορεί να αναζητήσει ταινίες, λίστες, κριτικές ή άλλα προφίλ χρηστών. Τα αποτελέσματα εμφανίζονται στην οθόνη *Search Results*, από όπου ο χρήστης μπορεί να μεταβεί σε μία από τις παρακάτω οθόνες:

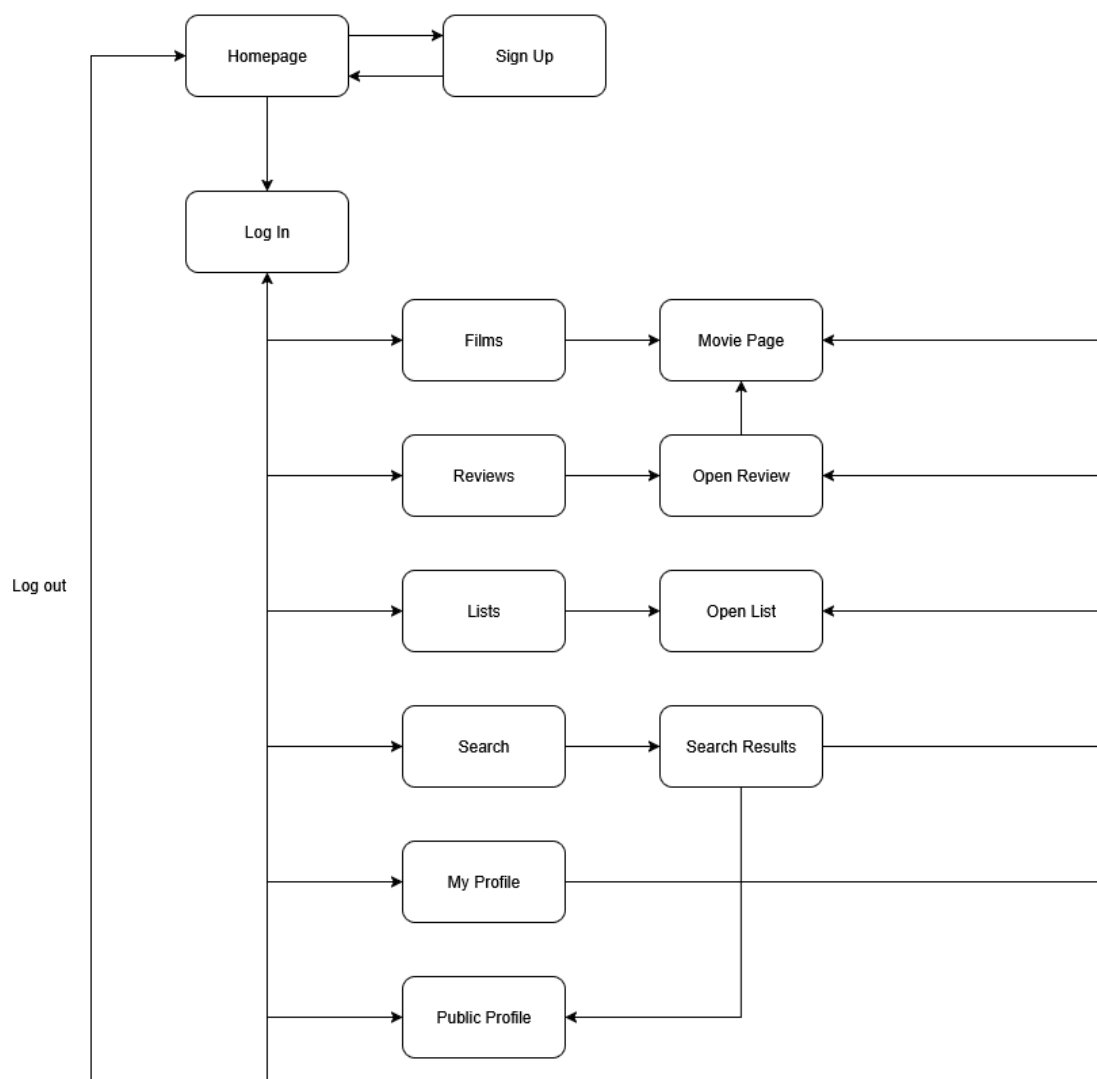
- Σε Σελίδα Ταινίας (Movie Page).
- Σε οθόνη λίστας (Open List).
- Σε Δημόσιο Προφίλ (Public Profile).

4.2.8 Προφίλ Χρήστη (My Profile)

Η οθόνη *My Profile* περιέχει προσωπικά στοιχεία του συνδεδεμένου χρήστη, τις κριτικές του και τις λίστες που έχει δημιουργήσει. Ο χρήστης μπορεί επίσης να επεξεργαστεί τα στοιχεία του προσωπικού του προφίλ.

4.2.9 Δημόσιο Προφίλ (Public Profile)

Η οθόνη *Public Profile* εμφανίζει τα στοιχεία και τη δραστηριότητα άλλων χρηστών (κριτικές, λίστες). Η πρόσβαση γίνεται μέσω των *Search Results* ή από τυχόν αλληλεπιδράσεις με περιεχόμενο άλλου χρήστη.

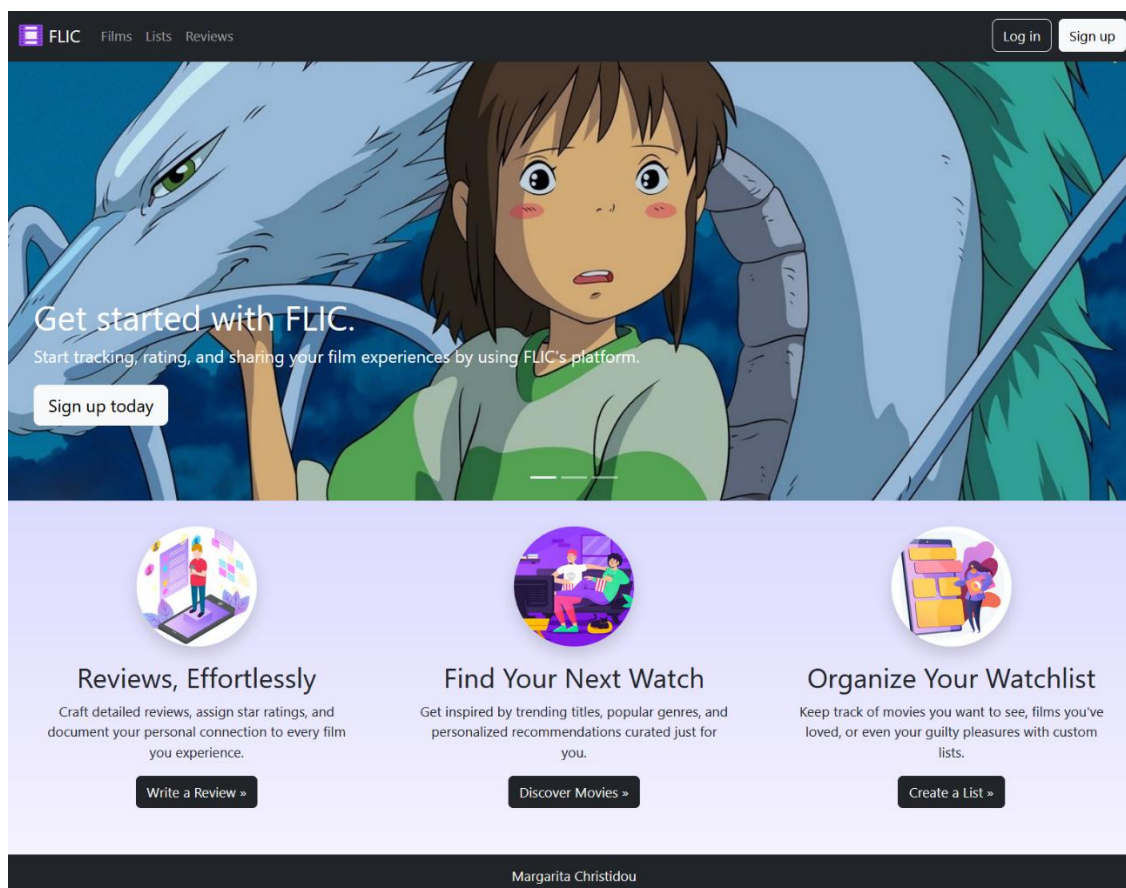


Εικόνα 6: Σχεδιάγραμμα Ανάλυσης Οθονών της Εφαρμογής

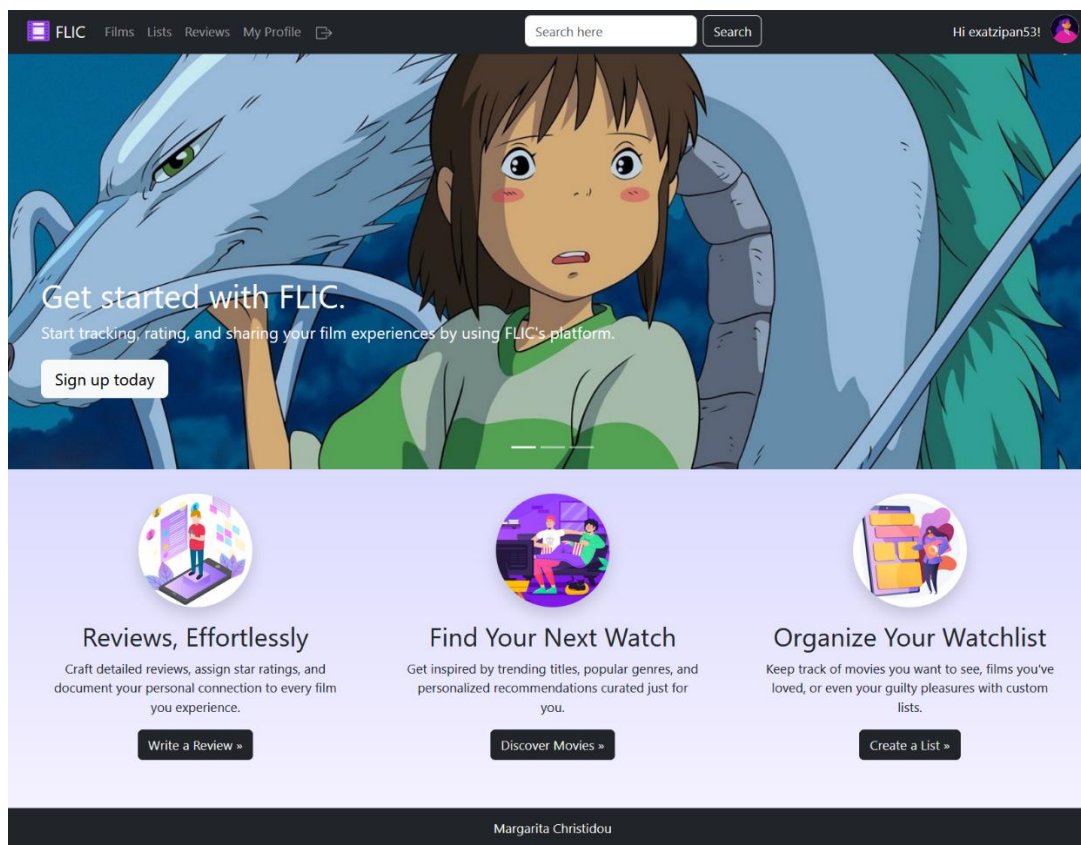
5 Η Εφαρμογή

Σε αυτό το κεφάλαιο, γίνεται ανάλυση της εφαρμογής και των λειτουργιών της, παρέχοντας εικόνες της.

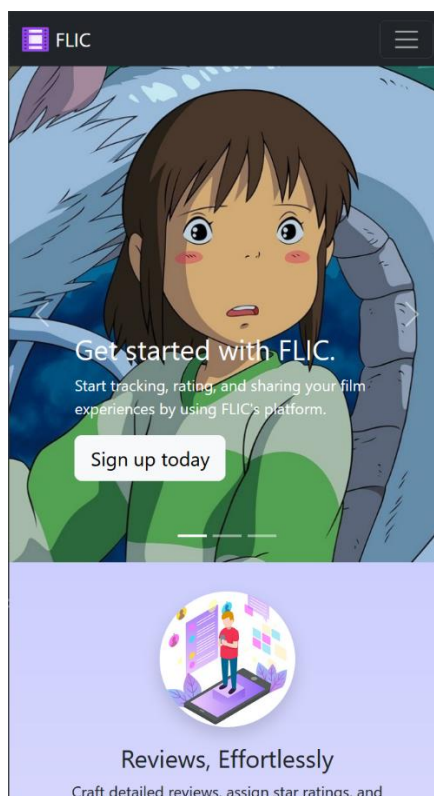
Στην παρακάτω εικόνα, βλέπουμε την Αρχική Σελίδα της εφαρμογής FLIC. Ένας χρήστης μπαίνοντας στην πλατφόρμα μπορεί είτε να συνδεθεί (Log In) είτε να εγγραφεί (Sign Up) σε αυτή. Για να έχει κάποιος πρόσβαση στην υπόλοιπη εφαρμογή, πρέπει να είναι εγγεγραμμένος χρήστης διαφορετικά, ζητείται από τον χρήστη να συνδεθεί.



Εικόνα 7: Αρχική Σελίδα - Homepage (Μη συνδεδεμένος χρήστης)



Εικόνα 8: Αρχική Σελίδα - Homepage (Συνδεδεμένος χρήστης)



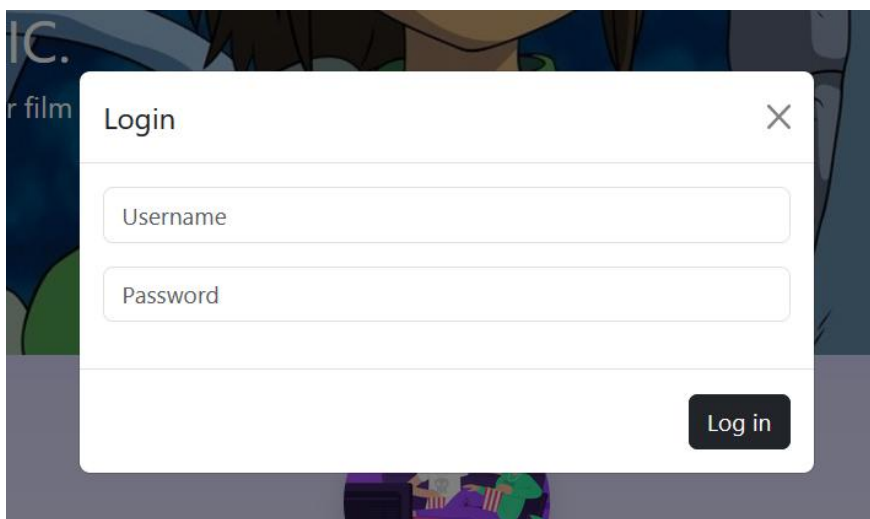
Εικόνα 9: Αρχική Σελίδα - Homepage (Responsive)

A login form titled "Login" is displayed on a light purple background. The form is a light gray rounded rectangle. At the top, the word "Login" is in a large, dark font. Below it, a purple message reads "Πρέπει να κάνεις login!". There are two white input fields: "Username" and "Password". At the bottom of the form is a dark gray button with the text "Log in" in white.

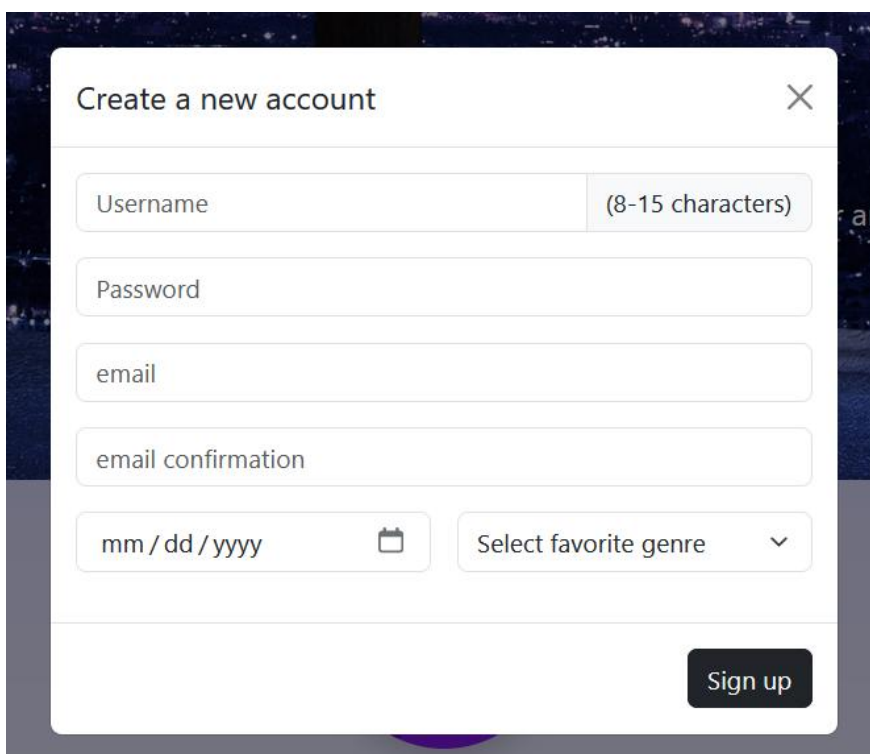
Εικόνα 10: Φόρμα σύνδεσης χρήστη και μήνυμα σφάλματος (Σε μη εγγεγραμμένους χρήστες)

A form titled "Create a new account" is displayed on a light purple background. The form is a light gray rounded rectangle. It contains several input fields: "Username" with a character count "(8-15 characters)", "Password", "email", and "email confirmation". Below these is a date field with the placeholder "mm / dd / yyyy" and a calendar icon, and a dropdown menu labeled "Select favorite genre" with a downward arrow. At the bottom are two dark gray buttons: "Clear" and "Sign up".

Εικόνα 11: Φόρμα εγγραφής χρήστη

A login modal window with a title bar 'Login' and a close button 'X'. It contains two input fields: 'Username' and 'Password'. A 'Log in' button is located at the bottom right.

Εικόνα 12: Φόρμα σύνδεσης χρήστη σε μορφή modal

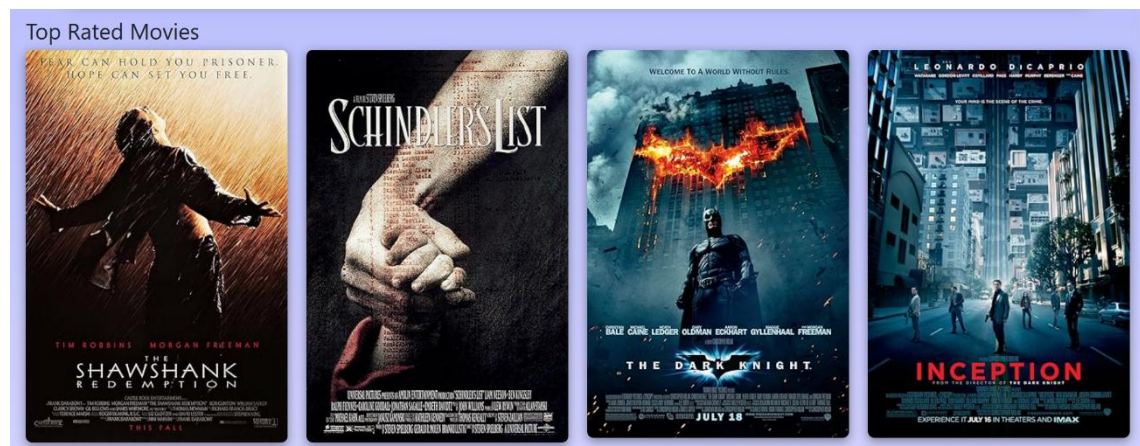
A 'Create a new account' modal window with a title bar and a close button 'X'. It contains four input fields: 'Username' (with a character count '(8-15 characters)'), 'Password', 'email', and 'email confirmation'. Below these are a date input field labeled 'mm / dd / yyyy' with a calendar icon, and a dropdown menu labeled 'Select favorite genre'. A 'Sign up' button is at the bottom right.

Εικόνα 13: Φόρμα εγγραφής χρήστη σε μορφή modal

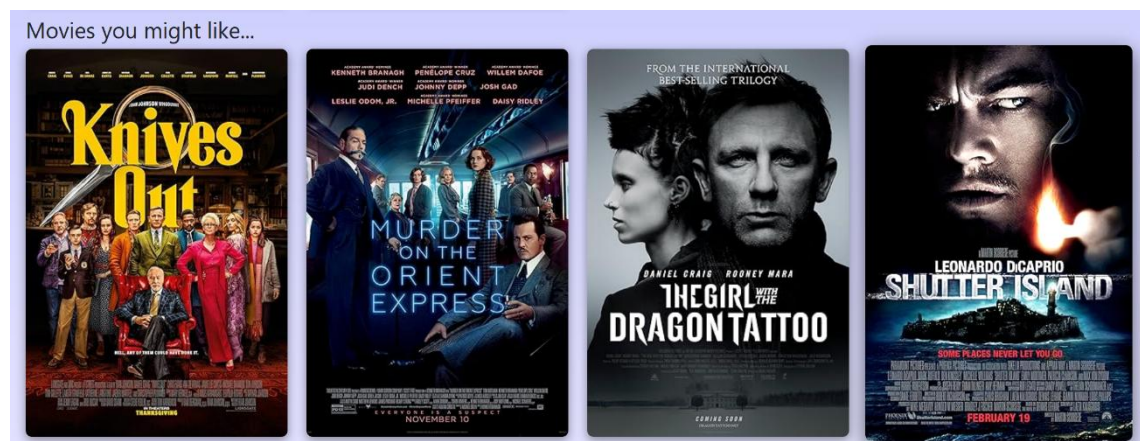
Αφού ο χρήστης συνδεθεί στην πλατφόρμα, έχει την δυνατότητα να μεταβεί σε όλες τι υπόλοιπες διαθέσιμες οθόνες, όπως Films, Lists, Reviews και My Profile. Επίσης, έχει την δυνατότητα να κάνει αναζήτηση ταινιών, χρηστών και λιστών, μέσω της μπάρας που βρίσκεται στο μενού, καθώς και να αποσυνδεθεί.

Η σελίδα Films εμφανίζει στον χρήστη όλες τις ταινίες που υπάρχουν στην βάση και του προτείνει μερικές. Οι ταινίες χωρίζονται σε τρεις κατηγορίες: Top Rated, Movies you might like και Other movies.

Στην Εικόνα 11, οι ταινίες έχουν επιλεγθεί με βάση τα στοιχεία που παίρνουμε από το API. (IMDb rating). Στην Εικόνα 12, οι ταινίες προτείνονται με βάση το προτεινόμενο είδος ταινίας που έδωσε ο χρήστης κατά την εγγραφή του στο σύστημα. Και τέλος, στην Εικόνα 13 εμφανίζονται όλες οι υπόλοιπες ταινίες που υπάρχουν στη βάση. Για την συγκεκριμένη περίπτωση, δημιουργήθηκε pagination με χρήση AJAX τεχνολογίας, ενώ οι προηγούμενες δύο γραμμές περιορίστηκαν στις πρώτες 4 ταινίες των αποτελεσμάτων.



Εικόνα 14: Γραμμή Top Rated Movies - Films

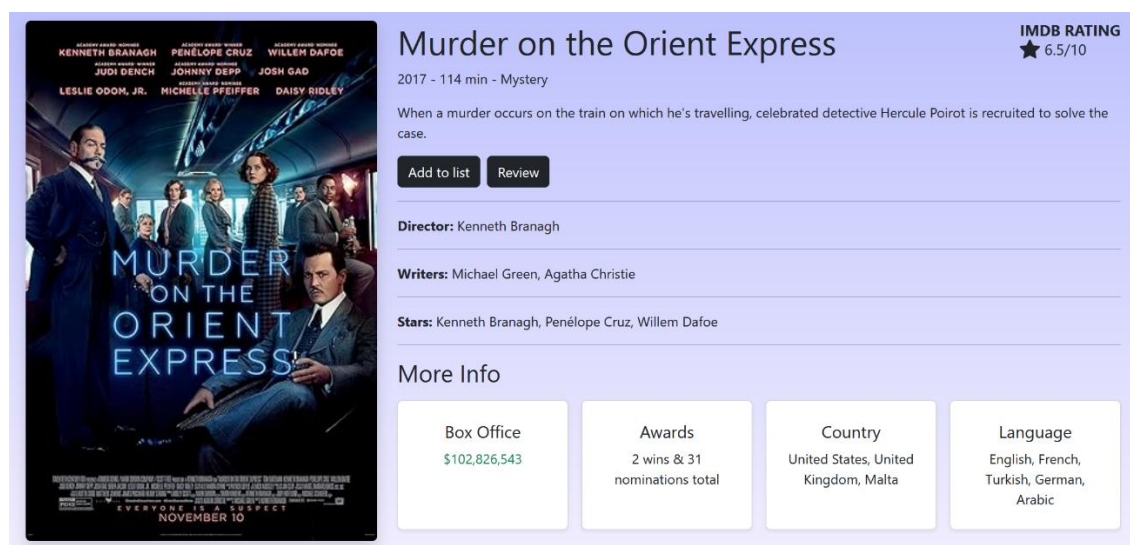


Εικόνα 15: Γραμμή Movies you might like - Films



Εικόνα 16: Γραμμή Other movies – Films

Κάνοντας κλικ σε οποιαδήποτε ταινία, ο χρήστης μεταφέρεται στην Σελίδα Ταινίας (Movie Page), που φαίνεται στην Εικόνα 14. Σε αυτήν, εμφανίζονται πληροφορίες σχετικά με την ταινία που επιλέχθηκε καθώς και δίνεται η επιλογή στον χρήστη να προσθέσει την ταινία σε κάποια λίστα που έχει δημιουργήσει ή να γράψει μία κριτική για αυτή.

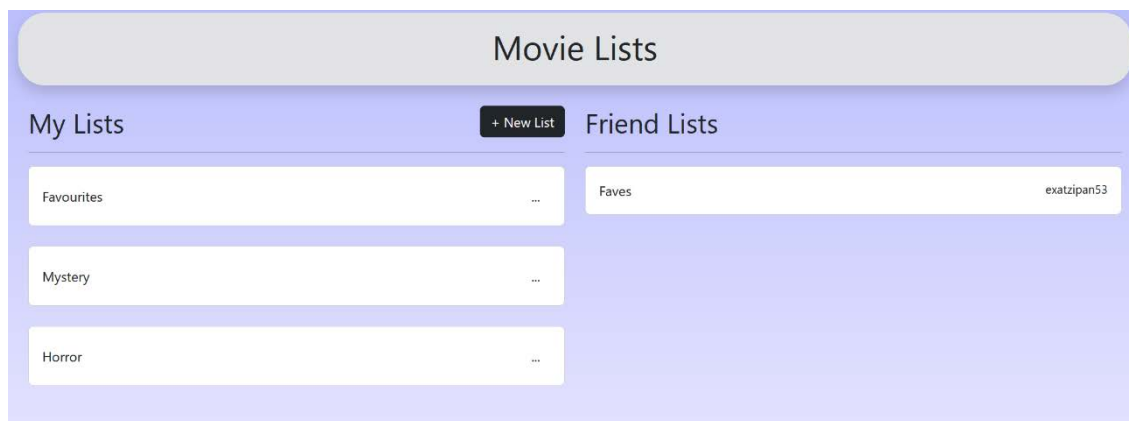


Εικόνα 17: Σελίδα Ταινίας (Movie Page)

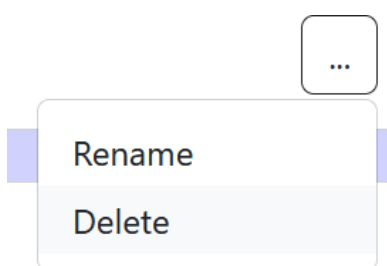
Εικόνα 18: Φόρμα κριτικής ταινίας - Movie Page

Εικόνα 19: Φόρμα προσθήκης ταινίας σε λίστα – Movie Page

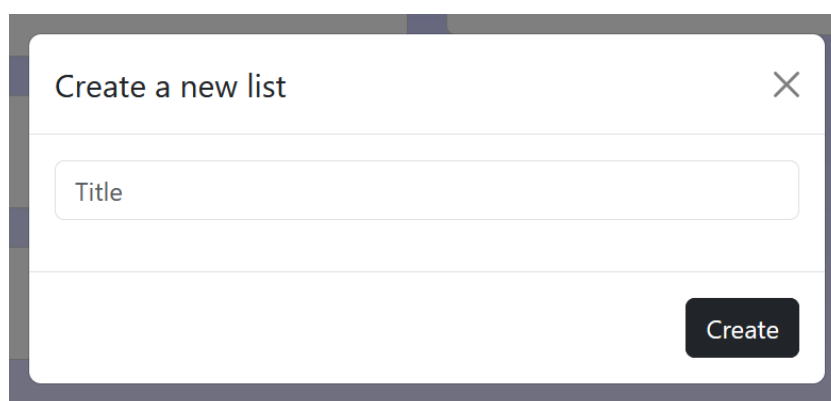
Η σελίδα Lists εμφανίζει στον χρήστη τις προσωπικές του λίστες καθώς και αυτές των φίλων του (αν υπάρχουν). Κάνοντας κλικ σε μία λίστα, ο χρήστης μεταφέρεται στην σελίδα Open List, όπου μπορεί να δει τα περιεχόμενα της και αν του αρέσει να κάνει “Like” σε αυτή. Παράλληλα, κάνοντας κλικ στο κουμπί «...» ο χρήστης έχει την επιλογή να μετονομάσει ή να διαγράψει την συγκεκριμένη λίστα, εφόσον αυτή έχει δημιουργηθεί από αυτόν. Επιπλέον, πατώντας το κουμπί «+ New List», ο χρήστης μπορεί να δημιουργήσει μία νέα λίστα, δίνοντας της ένα τίτλο. Στην Εικόνα 20, βλέπουμε το περιεχόμενο της λίστας Faves το οποίο είναι κενό.



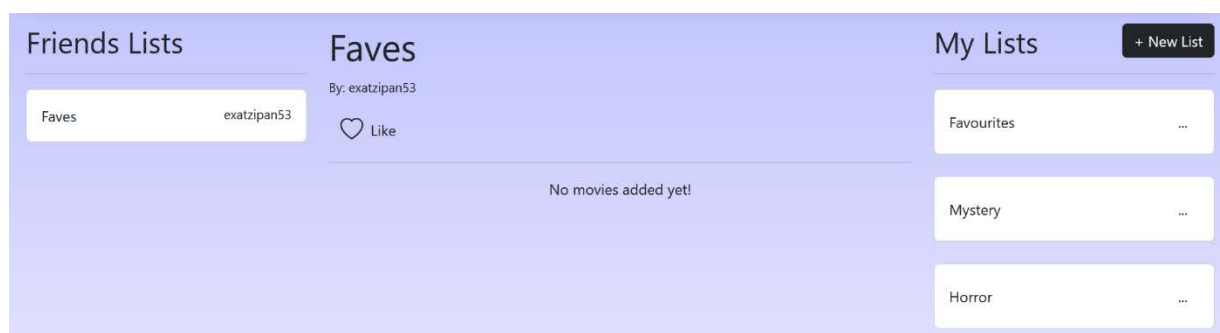
Εικόνα 20: Σελίδα Λιστών – Lists



Εικόνα 21: Κουμπί επιλογών "Rename" και "Delete" λίστας



Εικόνα 22: Φόρμα δημιουργίας νέας λίστας – Lists



Εικόνα 23: Περιεχόμενα σελίδας Open List - Λίστα Faves

Ανοίγοντας μία προσωπική λίστα, ο χρήστης έχει επίσης την δυνατότητα να κάνει “Like” στην λίστα του καθώς και να διαγράψει οποιαδήποτε ταινία ανήκει σε αυτή. Στην Εικόνα 21, βλέπουμε ένα τέτοιο παράδειγμα.



Εικόνα 24: Παράδειγμα προσωπικής λίστας χρήστη

Η σελίδα Reviews εμφανίζει στον χρήστη τις κριτικές που έχει δημιουργήσει καθώς και αυτές των φίλων του (αν υπάρχουν). Κάνοντας κλικ στο εικονίδιο της Εικόνας 22, ο χρήστης μεταφέρεται στην σελίδα Open Review όπου μπορεί, πέρα από το να δει τα περιεχόμενα της, να διαβάσει και να γράψει σχόλια σε αυτήν. Επίσης, πατώντας τον τίτλο της ταινίας μεταφέρεται στην Σελίδα Ταινίας που της αντιστοιχεί και πατώντας την καρδιά, Εικόνα 23, μπορεί να κάνει “Like” στην αντίστοιχη κριτική.

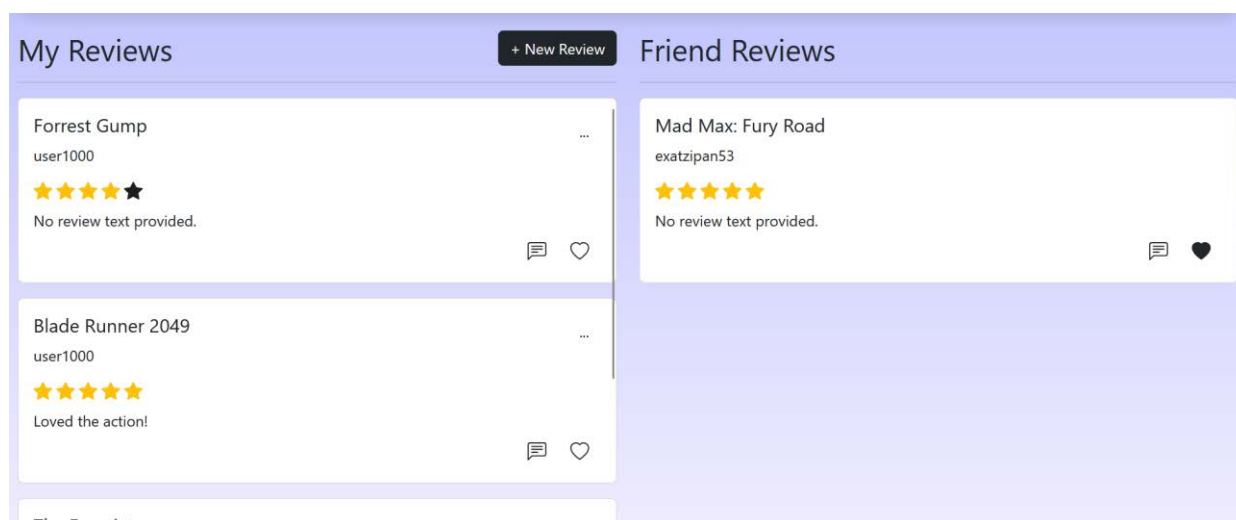


Εικόνα 25: Εικονίδιο Σχολίων



Εικόνα 26: Εικονίδιο "Like"

Παράλληλα, κάνοντας κλικ στο κουμπί «...» (ίδια λειτουργία με τις λίστες – βλέπε Εικόνα 18), ο χρήστης έχει την επιλογή να μετονομάσει ή να διαγράψει την συγκεκριμένη κριτική, εφόσον αυτή έχει δημιουργηθεί από αυτόν. Επιπλέον, πατώντας το κουμπί «+ New Review», ο χρήστης μπορεί να δημιουργήσει μία νέα κριτική.

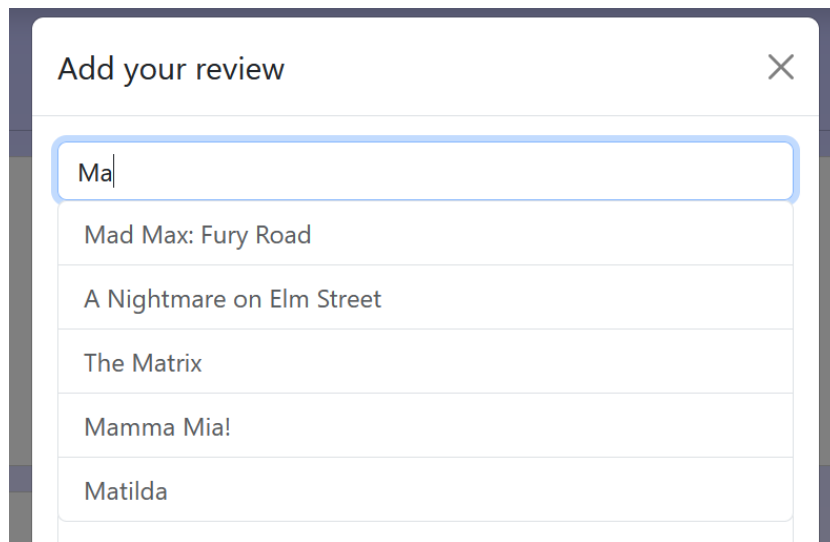


Εικόνα 27: Σελίδα κριτικών – Reviews

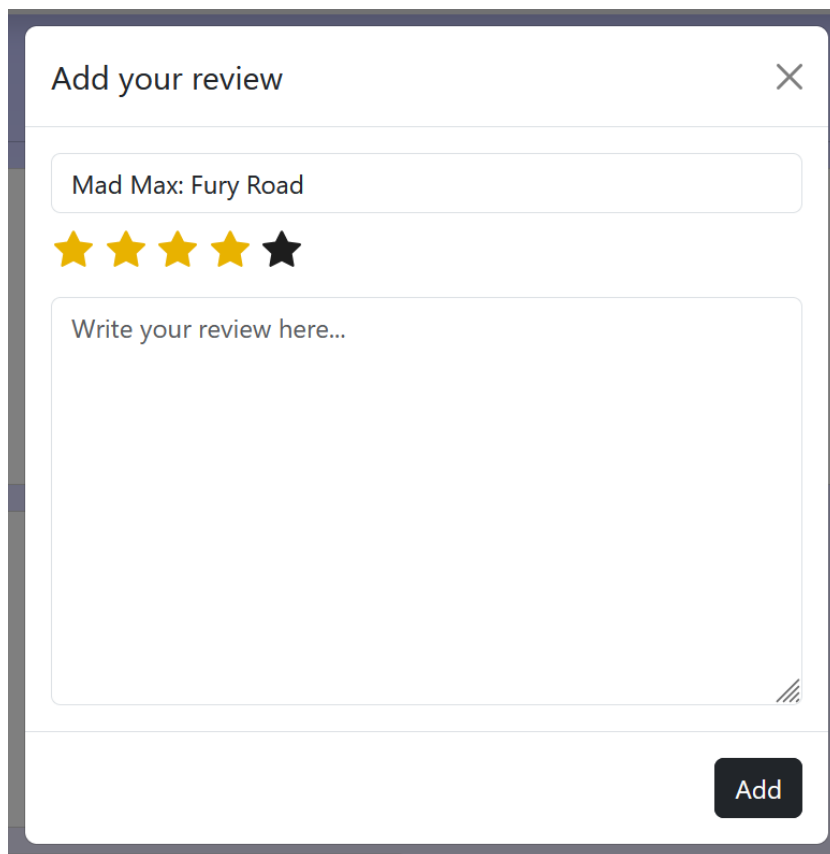
The screenshot shows a modal window titled 'Add your review'. It features a 'Search Movie' input field at the top. Below the input field is a 5-star rating. Underneath the rating is a large text area with the placeholder text 'Write your review here...'. At the bottom right of the modal is an 'Add' button.

Εικόνα 28: Φόρμα κριτικής ταινίας – Reviews

Στη συγκεκριμένη φόρμα-modal, δίνεται η δυνατότητα στον χρήστη να αναζητήσει την ταινία για την οποία θέλει να γράψει την κριτική του. Στην Εικόνα 26, βλέπουμε ένα τέτοιο παράδειγμα και στην Εικόνα 27, βλέπουμε το αποτέλεσμα αφού επιλεγθεί.

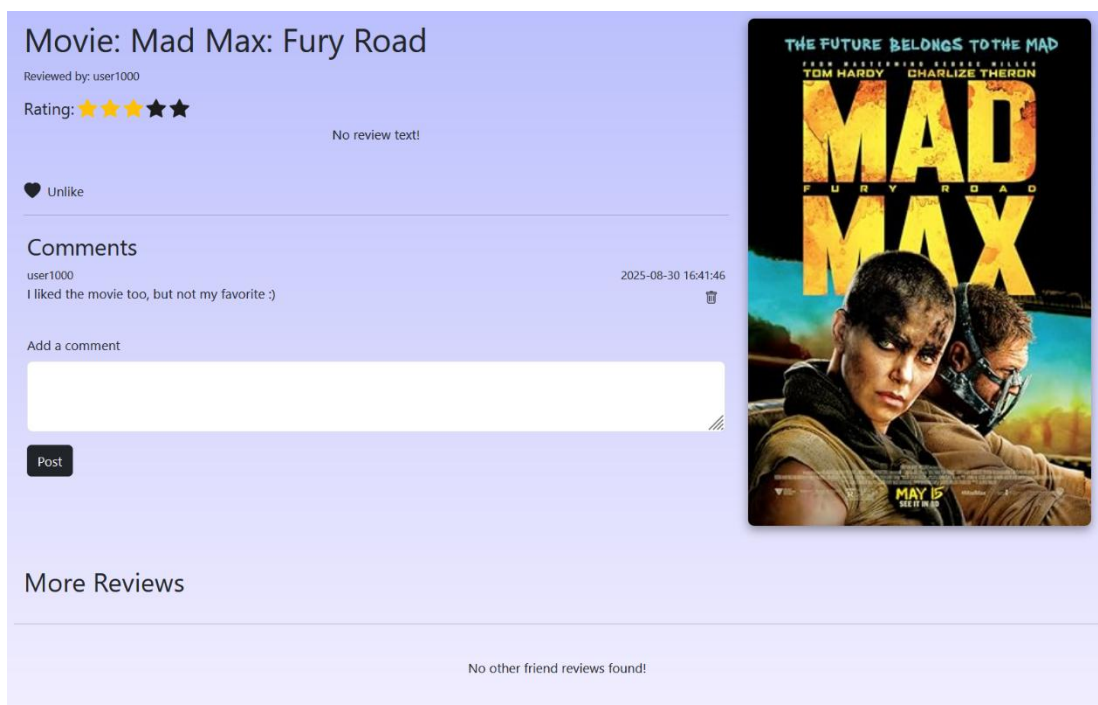


Εικόνα 29: Παράδειγμα αναζήτησης ταινίας - Modal «+New Review»



Εικόνα 30: Παράδειγμα προσθήκης νέας κριτικής με χρήση του modal «+New Review»

Στην Εικόνα 28, βλέπουμε ένα παράδειγμα της σελίδας Open Review όπου ο χρήστης μπορεί επιπλέον να αφήσει ένα σχόλιο στην συγκεκριμένη κριτική. Επιπλέον, ένα σχόλιο μπορεί να διαγραφεί μόνο από τον δημιουργό του ή από τον δημιουργό της κριτικής της οποίας ανήκει.

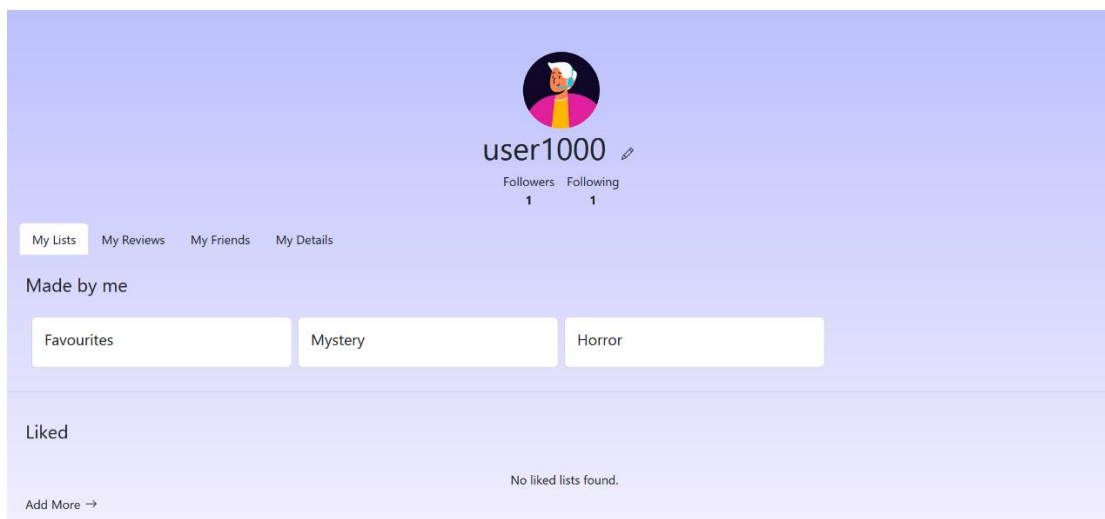


Εικόνα 31: Σελίδα Περιεχομένων Κριτικής - Open Review

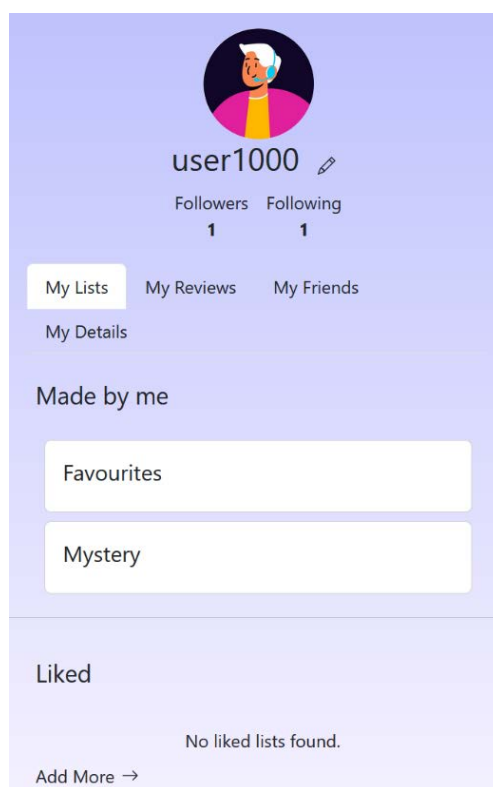
Η σελίδα My Profile εμφανίζει στον χρήστη το προσωπικό του προφίλ, το οποίο περιέχει όλες της πληροφορίες σχετικά με αυτόν. Παρακάτω, στην Εικόνα 29, βλέπουμε ότι ο κάθε χρήστης έχει την δυνατότητα να δει στο προφίλ του, τα εξής τέσσερα ξεχωριστά παράθυρα (tabs):

- My Lists : Περιέχει τις λίστες που έχει δημιουργήσει καθώς και αυτές τις οποίες έχει κάνει “Like”.
- My Reviews: Περιέχει τις κριτικές που έχει δημιουργήσει καθώς και αυτές τις οποίες έχει κάνει “Like”.
- My Friends: Εμφανίζει τα προφίλ των χρηστών με τους οποίους ο χρήστης είναι φίλος. Οι χρήστες θεωρούνται «Φίλοι» μόνο εάν ο ένας ακολουθάει τον άλλον.
- My Details: Εμφανίζει τις προσωπικές πληροφορίες που έδωσε ο χρήστης κατά την εγγραφή του και του δίνεται η δυνατότητα να τις τροποποιήσει.

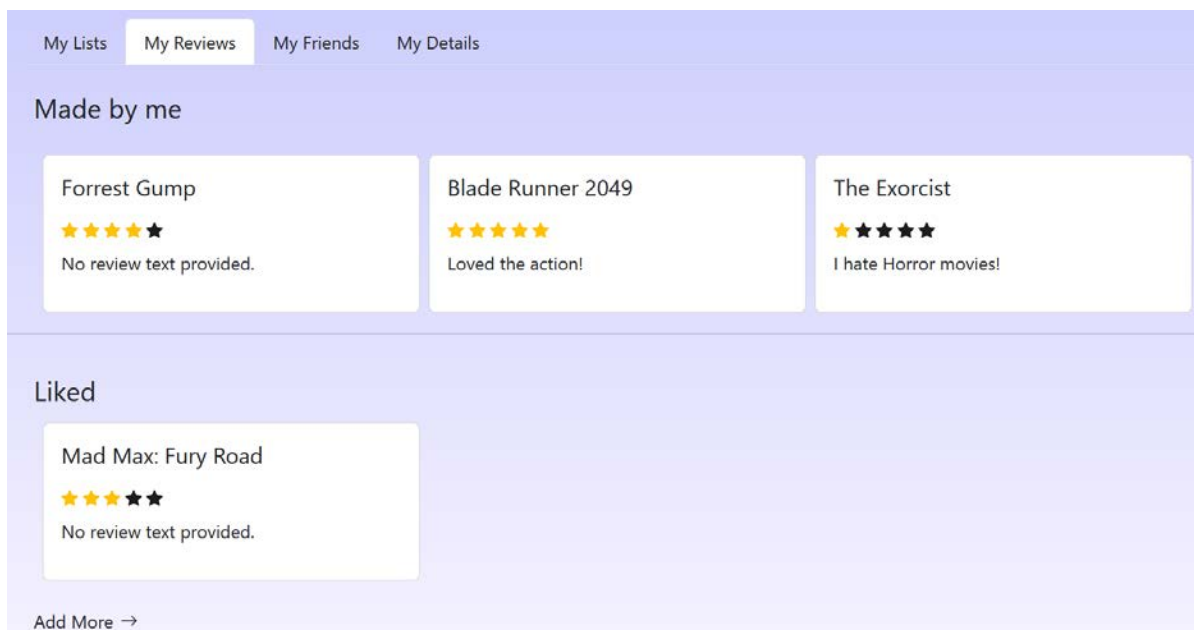
Επιπλέον, δίνεται η δυνατότητα στον χρήστη να αλλάξει το ψευδώνυμο του (username) και την εικόνα προφίλ του (avatar) επιλέγοντας μεταξύ τεσσάρων διαφορετικών φωτογραφιών. Οι φωτογραφίες άρθηκαν από την ιστοσελίδα Freepik.[]



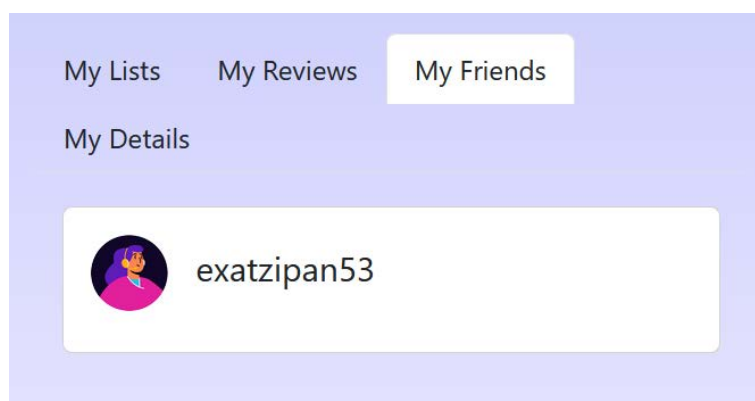
Εικόνα 32:Σελίδα Προσωπικού Προφίλ - My Profile



Εικόνα 33: Σελίδα Προσωπικού Προφίλ - My Profile (Responsive)



Εικόνα 34: Σελίδα Προσωπικού Προφίλ - My Reviews Tab



Εικόνα 35: Σελίδα Προσωπικού Προφίλ - My Friends Tab

My Lists My Reviews My Friends My Details

Username user1000 Change Email user1@uth.gr Change

Password Password Favourite Genre -- Mystery -- Change

Confirm Password Confirm Password Change

Εικόνα 36: Σελίδα Προσωπικού Προφίλ - My Details Tab

My Lists My Reviews My Friends

My Details

Username
user1000 Change

Email
user1@uth.gr Change

Password
.....

Confirm Password
 Change

Favourite Genre
-- Mystery -- Change

Εικόνα 37: Σελίδα Προσωπικού Προφίλ - My Details Tab (Responsive)

Choose your Avatar

Choose

Εικόνα 38: Επιλογή εικόνας προφίλ - Modal

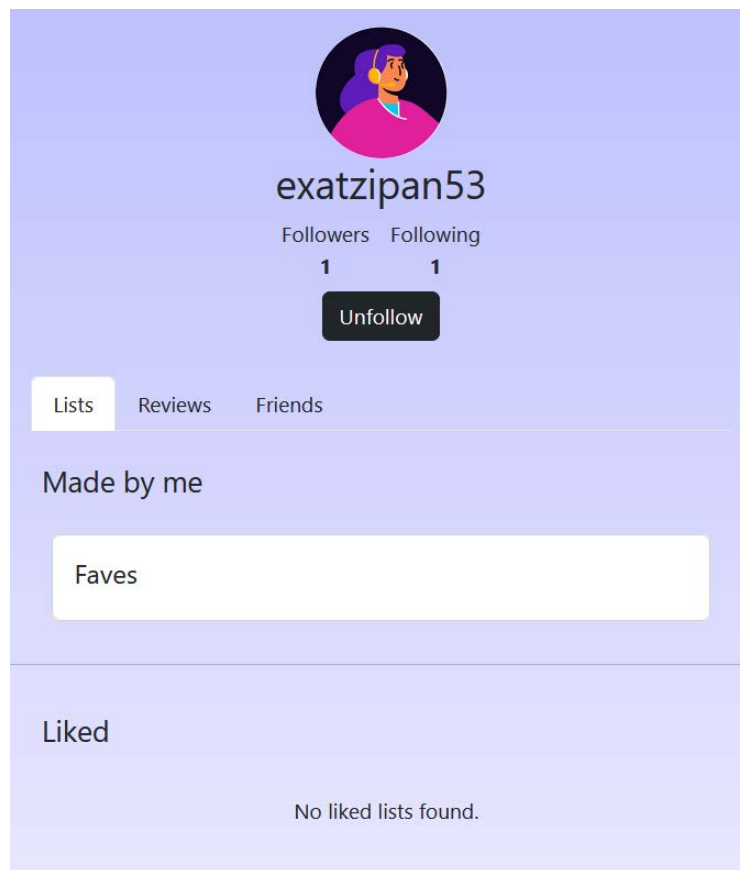
Choose your new username

Username (8-15 characters)

Submit

Εικόνα 39: Αλλαγή ψευδώνυμου (username) – Modal

Όπως μπορούμε να δούμε στην παρακάτω εικόνα, σε ένα δημόσιο προφίλ άλλου χρήστη κάποιος μπορεί να δει την δραστηριότητα του και να επιλέξει να τον «ακολουθήσει» πατώντας το κουμπί “Follow”. Αν ο χρήστης ξεκινήσει να ακολουθεί το επιλεγμένο προφίλ, τότε του δίνεται η δυνατότητα να σταματήσει να το ακολουθεί πατώντας το κουμπί “Unfollow”.



Εικόνα 40: Σελίδα Δημόσιου Προφίλ (Responsive)

Επιπρόσθετα, ένας χρήστης μπορεί να έχει πρόσβαση στα δημόσια προφίλ άλλων χρηστών με τους οποίους είναι ήδη φίλος ή που ανακάλυψε μέσω του search bar που βρίσκεται ψηλά στο μενού της σελίδας, όπως φαίνεται στην Εικόνα 38 .



Εικόνα 41: Παράδειγμα χρήσης του search bar στο μενού

Στην Εικόνα 39, βλέπουμε το αποτέλεσμα της αναζήτησης που φαίνεται στην Εικόνα 38, αφού κάνουμε κλικ στο κουμπί “Search” . Γίνεται ταυτόχρονη αναζήτηση τριών κατηγοριών Χρηστών, Λιστών και Ταινιών.

Search results for "ex"

Users

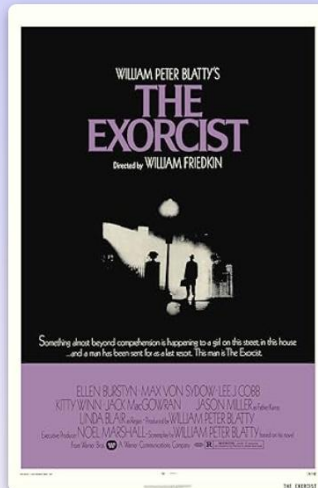


exatzipan53

Lists

No results found

Movies



Εικόνα 42: Σελίδα αποτελεσμάτων αναζήτησης μέσω του search bar στο μενού

6 Συμπεράσματα

Η ανάπτυξη της εφαρμογής FLIC αποτέλεσε μια δημιουργική αλλά και απαιτητική διαδικασία, η οποία κράτησε περίπου έναν χρόνο. Βασικός στόχος ήταν να δημιουργηθεί μια εύχρηστη, λειτουργική εφαρμογή που δίνει στους χρήστες της τη δυνατότητα να οργανώνουν και να μοιράζονται το περιεχόμενο που τους ενδιαφέρει με άλλους φίλους του κινηματογράφου. Κατά την υλοποίηση, υπήρξαν δυσκολίες, κυρίως επειδή η ανάπτυξη έπρεπε να συνδυαστεί με άλλες υποχρεώσεις και υπήρξε περιορισμός χρόνου. Παρ' όλα αυτά, η εμπειρία αποδείχτηκε εξαιρετικά χρήσιμη, καθώς μέσα από την πράξη αποκτήθηκε βαθύτερη κατανόηση τόσο στη σχεδίαση εφαρμογών όσο και στη διαχείριση βάσεων δεδομένων και στη σύνδεσή τους με πραγματικές ανάγκες χρηστών.

Παρόλο που η εφαρμογή ολοκληρώθηκε σε λειτουργικό επίπεδο, υπάρχουν πολλά σημεία που μπορούν να εξελιχθούν στο μέλλον. Ένα από τα στοιχεία που θα μπορούσαν να προστεθούν είναι η ενσωμάτωση API ταινιών, ώστε η ενημέρωση της βάσης δεδομένων να γίνεται αυτόματα, επιτρέποντας στην εφαρμογή να παραμένει πάντα ενημερωμένη με τις τελευταίες κυκλοφορίες. Επιπλέον, η ενσωμάτωση ενός συστήματος συστάσεων (Recommendation System) με τη χρήση τεχνητής νοημοσύνης θα βοηθούσε ώστε κάθε χρήστης να λαμβάνει προσωποποιημένες προτάσεις με βάση τις προτιμήσεις και το ιστορικό του. Τέλος, σημαντική προτεραιότητα αποτελεί και η συνεχής βελτίωση της ασφάλειας της εφαρμογής, εξασφαλίζοντας την προστασία των προσωπικών δεδομένων και την ασφαλή εμπειρία χρήσης για όλους τους χρήστες.

Με αυτές τις μελλοντικές επεκτάσεις, η FLIC έχει τη δυνατότητα να μετατραπεί σε μια ολοκληρωμένη εφαρμογή, που όχι μόνο θα καλύπτει τις βασικές ανάγκες των χρηστών, αλλά θα τους προσφέρει και μια πιο προσωπική, σύγχρονη και ευχάριστη εμπειρία.

Βιβλιογραφία

- [1] <https://letterboxd.com/about/faq/> - Σχετικά με την εφαρμογή Letterboxd
- [2] <https://www.imdb.com/> - Η εφαρμογή IMDb
- [3] <https://www.rottentomatoes.com/about>
Σχετικά με την εφαρμογή Rotten Tomatoes
- [4] https://www.w3schools.com/html/html_intro.asp - Σχετικά με την HTML
- [5] https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Styling_basics - Σχετικά με την CSS
- [6] <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
Σχετικά με την JavaScript
- [7] <https://getbootstrap.com/> - Το framework Bootstrap5
- [8] <https://www.php.net/docs.php> - Το Documentation της γλώσσας PHP
- [9] <https://www.omdbapi.com/> - Το Εξωτερικό API OMDb
- [10] <https://wampserver.aviatechno.net/> - Ιστοσελίδα του WAMP Server
- [11] <https://dev.mysql.com/doc/refman/8.0/en/introduction.html> -
- [12] <https://wall.alphacoders.com/> - Ιστοσελίδα Wallpaper Abyss
- [13] <https://www.freepik.com/> - Ιστοσελίδα FREEPIK
- [14] <https://bootstrapexamples.com/@anonymous/interactive-star-rating-2>
Κώδικας υποβολής βαθμολογίας με αστέρια

Παράρτημα

Παρακάτω μπορείτε να δείτε ενδεικτικά αποσπάσματα του κώδικα ανάπτυξης της εφαρμογής FLIC.

```
1 <?php
2
3 if ( !isset($_SESSION['username']) && isset($_POST['username'], $_POST['password']) ) {
4
5     require('db_conn.php');
6
7     $username = $_POST['username'];
8     $password = $_POST['password'];
9
10
11     $sql = "SELECT * FROM users WHERE username=:username";
12     $query = $db->prepare($sql);
13     $query->execute(array(':username'=>$username));
14     $result = $query->fetch();
15     $query->closeCursor();
16     $db = null;
17
18     $hash = password_hash($password, PASSWORD_DEFAULT);
19     //cryptographic salt for the password
20
21     $authorised=false;
22
23     if ($result == true && password_verify($_POST['password'], $hash)) {
24         //found in db
25         $authorised= true;
26         session_start(); // session for user starts
27         $_SESSION['username'] = $result['username'];
28         $_SESSION['user_id'] = $result['user_id'];
29         $_SESSION['fave_genre'] = $result['fave_genre'];
30         $_SESSION['avatar'] = $result['avatar'];
31     }
32
33     if ($authorised==true) {
34         header('Location: films_page.php'); //user page
35         exit();
36     }
37     else {
38         header("Location: login.php?msg=Αποτυχημένη διαπίστευση χρήστη!"); //error
39         exit();
40     }
41 } else {
42     session_start();
43     session_destroy();
44     header("Location: login.php?msg=Πρόβλημα - Δοκίμασε ξανά!");
45     exit();
46 }
```

Εικόνα 43:Λογική σύνδεσης (Log in) του χρήστη στην πλατφόρμα

```

1 <?php
2
3 /*ΣΥΝΔΕΣΗ ΣΤΗ ΒΑΣΗ*/
4 require('db_conn.php');
5
6 /*ΕΓΓΡΑΦΗ ΧΡΗΣΤΗ */
7
8 $username = $_POST['username'];
9 $password = $_POST['password'];
10 $birthday = $_POST['birthday'];
11 $email = $_POST['email'];
12 $email_conf = $_POST['email_conf'];
13 $fave_genre = $_POST['fave_genre'];
14
15 $usr_str = strlen($username);
16 $ps_str = strlen($password);
17
18 $ps_pattern = '/^(?=[a-z])(?=[A-Z])(?=[\W_]).{8,}$/' ;
19
20 $sql = "SELECT username FROM users WHERE username=:username";
21 $query = $db->prepare($sql);
22 $query->execute(array(':username'=>$username));
23 $result = $query->fetch();
24
25 if ($result == true){ //username was found in the db
26     header("Location: signup.php?msg=Υπάρχει χρήστης με αυτό το username!");
27     exit();
28 }
29
30 if ($usr_str < 8 || $usr_str > 15) {
31     header("Location: signup.php?msg=Το username δεν έχει 8-15 χαρακτήρες!");
32     exit();
33 }
34
35 if ($ps_str < 8) {
36     header("Location: signup.php?msg=0 κωδικός πρόσβασης είναι μικρός!");
37     exit();
38 } else if (!preg_match($ps_pattern, $password)) {
39     header(
40 "Location: signup.php?msg=0 κωδικός πρόσβασης δεν είναι κατάλληλος. Ξαναπροσπάθησε!");
41     exit();
42 }
43
44 if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
45     header("Location: signup.php?msg=Μη αποδεκτό email!");
46     exit();
47 } /*not sure if needed*/
48
49 $hash = password_hash($password, PASSWORD_DEFAULT);
50 //applying cryptographic salt to password
51
52 $data = [
53     ':username' => $username,
54     ':password' => $hash,
55     ':birthday' => $birthday,
56     ':email' => $email,
57     ':fave_genre' => $fave_genre,
58 ];
59
60 $sql = "INSERT INTO users(username, password, date_of_birth, email, fave_genre) VALUES
61 (:username,:password,:birthday,:email,:fave_genre)";
62
63 $query = $db->prepare($sql);
64 $query->execute($data);
65 $result = $query->fetch();
66 $query->closeCursor();
67 $db = null;
68
69 header("Location: login.php?msg=Επιτυχής εγγραφή!");
70
71 ?>

```

Εικόνα 44:Λογική εγγραφής (Sign up) του χρήστη στην πλατφόρμα

```

1 // ----- AJAX LOADING FOR OTHER MOVIES SECTION -----//
2 function initOtherMoviesAjaxLoader() {
3     const container = document.getElementById('other-movies-section');
4     if (!container) return; // Exit if the container doesn't exist
5
6     function loadMovies(page) {
7         // Save the current scroll position relative to the container
8         const containerTop = container.getBoundingClientRect().top + window.scrollY;
9         const currentScroll = window.scrollY - containerTop;
10
11         container.innerHTML = '<p>Loading movies...</p>';
12
13         fetch('other_movies_ajax.php?page=${page}')
14             .then(response => {
15                 if (!response.ok) throw new Error('Network response was not ok');
16                 return response.text();
17             })
18             .then(html => {
19                 container.innerHTML = html;
20
21                 // Restore scroll so the container stays in place
22                 window.scrollTo({
23                     top: containerTop + currentScroll,
24                     behavior: 'instant' // or 'auto'
25                 });
26             })
27             .catch(error => {
28                 console.error('Fetch error:', error);
29                 container.innerHTML =
30                     '<p style="color:red;">Failed to load movies. Please try again.</p>';
31             });
32
33         document.addEventListener('click', function(e) {
34             const clickedLink = e.target.closest('#other-movies-section a[data-page]');
35
36             if (clickedLink) {
37                 e.preventDefault();
38                 const pageNumber = clickedLink.dataset.page;
39                 if (pageNumber) loadMovies(pageNumber);
40             }
41         });
42
43         loadMovies(1);
44
45     };
46
47     // Initialize when DOM is ready
48     document.addEventListener('DOMContentLoaded', initOtherMoviesAjaxLoader);

```

Εικόνα 45:Υλοποίηση pagination στην σελίδα Films, με χρήση AJAX σε γλώσσα Javascript

```

1 // ----- AJAX LIVE SEARCH -----//
2 // This function is called when the user types in the search box (review modal)
3 function showResult(str) {
4     if (str.length == 0) {
5         document.getElementById("livesearch").innerHTML = "";
6         document.getElementById("livesearch").style.border = "0px";
7         return;
8     }
9     const xmlhttp = new XMLHttpRequest();
10    xmlhttp.onreadystatechange = function() {
11        if (this.readyState == 4 && this.status == 200) {
12            document.getElementById("livesearch").innerHTML = this.responseText;
13            //document.getElementById("livesearch").style.border = "1px solid #A5ACB2";
14        }
15    }
16    xmlhttp.open("GET", "livesearch.php?q=" + encodeURIComponent(str), true);
17    xmlhttp.send();
18 }
19
20 function selectMovie(id, title) {
21     document.querySelector('input[onkeyup]').value = title; // set main input
22     document.getElementById('movie_id').value = id; // set hidden id
23     document.getElementById("livesearch").innerHTML = "";
24     document.getElementById("livesearch").style.border = "0px";
25 }

```

Εικόνα 46:Υλοποίηση αναζήτησης ταινιών του modal “+ New Review”, με χρήση AJAX σε γλώσσα JavaScript

```

1 <?php
2 require('db_conn.php');
3
4 // Check if the search query is set
5 $q = isset($_GET['q']) ? trim($_GET['q']) : '';
6 if ($q === '') { exit; }
7
8 // Prepare and execute the SQL statement to search for movies
9 $sql = "SELECT movie_id, title FROM movies WHERE title LIKE :title LIMIT 10";
10 $stmt = $db->prepare($sql);
11 $stmt->execute([':title' => "%$q%"]);
12
13 // Check if any results were found
14 while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
15     $id = htmlspecialchars($row['movie_id']);
16     $title = htmlspecialchars($row['title']);
17     echo "<div class='list-group-item list-group-item-action'
18         style='cursor:pointer'
19         onclick='selectMovie(\"$id\", \" . json_encode($title) . \")'>
20         $title
21     </div>";
22 }
23

```

Εικόνα 47: Αναζήτηση στη βάση δεδομένων - livesearch.php

```

1  <?php
2      ob_start();
3      session_start();
4
5      require('db_conn.php');
6
7      $user_id = $_SESSION['user_id'];
8      $new_avatar = $_POST['selectedAvatar'];
9
10     $sql = "UPDATE users
11     SET avatar = :new_avatar
12     WHERE user_id = :user_id";
13     $query = $db->prepare($sql);
14     $query->execute(array(':new_avatar'=>$new_avatar, ':user_id'=>$user_id));
15     $result = $query->rowCount();
16
17     // Update session variable - to be able to see the change immediately
18     $_SESSION['avatar'] = $new_avatar;
19
20     $query->closeCursor();
21
22     if ($result > 0) {
23         header("Location: myprofile.php?msg=Επιτυχής αλλαγή εικόνας προφίλ!");
24         exit();
25     } else {
26         header("Location: myprofile.php?msg=Αποτυχής αλλαγή εικόνας προφίλ!");
27         exit();
28     }
29     $db = null;
30     ob_end_flush();
31  ?>

```

Εικόνα 48: Λογική αλλαγής εικόνας προφίλ χρήστη

```

<!--LIKE BUTTON-->
<?php $sql3 = "SELECT count(*) AS exist FROM list_likes WHERE user_id = :user_id AND list_id = :list_id";
$query3 = $db->prepare($sql3);
$query3->execute(array(':user_id'=>$user_id, ':list_id'=>$list_id));
$res = $query3->fetch();
if ($res['exist'] == 0) { ?>
    <a class="btn d-flex align-items-center gap-2" id="like" type="button" href="like_list?list_id=<?php echo $list_id; ?>">
        <svg xmlns="http://www.w3.org/2000/svg" width="25" height="25" fill="currentColor" class="bi bi-heart" viewBox="0 0 16 16">
            <path d="m8 2.748-.717-.737C5.6.281 2.514.878 1.4 3.053c-.523 1.023-.641 2.5314 4.385.92 1.815 2.834 3.989 6.286 6.357 3.4
        </svg>
        <span>Like</span>
    </a>
<?php } else if ($res['exist'] == 1) { ?>
    <a class="btn d-flex align-items-center gap-2" id="unlike" type="button" href="like_list?list_id=<?php echo $list_id; ?>">
        <svg xmlns="http://www.w3.org/2000/svg" width="25" height="25" fill="currentColor" class="bi bi-heart-fill" viewBox="0 0 16 16">
            <path fill-rule="evenodd" d="M8 1.314C12.438-3.248 23.534 4.735 15-7.534 4.736 3.562-3.248 8 1.314"/>
        </svg>
        <span>Unlike</span>
    </a>
<?php }
$query3->closeCursor(); ?>

```

Εικόνα 49: Υλοποίηση κουμπιού "Like" - Open List

```

1  <?php
2      session_start();
3      require('db_conn.php');
4
5      $user_id = $_SESSION['user_id'];
6      $list_id = $_GET['list_id'];
7
8      $sql = "DELETE FROM list_likes WHERE user_id = :user_id AND list_id = :list_id";
9      $query = $db->prepare($sql);
10     $query->execute(array(':user_id'=>$user_id, ':list_id'=>$list_id));
11     $res = $query->rowCount();
12
13     if ($res == 0) {
14         $sql = "INSERT INTO list_likes VALUES (:user_id, :list_id)";
15         $query = $db->prepare($sql);
16         $query->execute(array(':user_id'=>$user_id, ':list_id'=>$list_id));
17         $result = $query->rowCount();
18     }
19
20     header('Location: open_list.php?list_id='.$list_id);
21     //exit();
22
23  ?>

```

Εικόνα 50: Λειτουργία του κουμπιού "Like" - Open List

```

<ul class="navbar-nav">
  <?php if (!isset($_SESSION['username'])) { ?> <!-- if not logged in -->
    <li class="nav-item"><a class="nav-link" href="user_page.php">Films</a></li>
    <li class="nav-item"><a class="nav-link" href="user_page.php">Lists</a></li>
    <li class="nav-item"><a class="nav-link" href="user_page.php">Reviews</a></li>
  <?php } else { ?> <!-- if logged in -->
    <li class="nav-item"><a class="nav-link" href="films_page.php">Films</a></li>
    <li class="nav-item"><a class="nav-link" href="lists_page.php">Lists</a></li>
    <li class="nav-item"><a class="nav-link" href="reviews_page.php">Reviews</a></li>
    <li class="nav-item"><a class="nav-link" href="myprofile.php">My Profile</a></li>
    <li class="nav-item"><a class="nav-link" href="logout.php">
      <svg xmlns="http://www.w3.org/2000/svg" width="25" height="20" fill="currentColor" class="bi">
        <path fill-rule="evenodd" d="M10 12.5a.5.5 0 0 1-.5.5h-8a.5.5 0 0 1-.5-.5v-9a.5.5 0 0
        <path fill-rule="evenodd" d="M15.854 8.354a.5.5 0 0 0-.708 1-3a.5.5 0 0 0-.708.708
      </svg>
    </a></li>
  <?php } ?>
</ul>

```

Εικόνα 51: Navigation Menu της πλατφόρμας

```

<?php if (isset($_SESSION['username'])) { ?>
  <ul class="navbar-nav ms-auto">
    <form action="search_page.php" class="d-flex" role="search" method="get">
      <input class="form-control me-2" id="search_value" name="search_value" value="<?php echo isset($_GET['search_value'])
      ? $_GET['search_value'] : null; ?>" type="search" placeholder="Search here" aria-label="Search">
      <button class="btn btn-outline-light" type="submit">Search</button>
    </form>
  </ul>
<?php } ?>

```

Εικόνα 52: Πεδίο αναζήτησης στο μενού (Συνδεδεμένος χρήστης)

```

<!-- STAR RATING -->
<div class="star-rating mb-3">
  <input type="radio" id="star1" name="rating" value="5">
  <label for="star1">
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-star-fill" view
    <path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73l.173 6.765c-.329-.314-.158-.888.283-.9514.898
    </svg>
  </label>
  <input type="radio" id="star2" name="rating" value="4">
  <label for="star2">
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-star-fill" view
    <path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73l.173 6.765c-.329-.314-.158-.888.283-.9514.898
    </svg>
  </label>
  <input type="radio" id="star3" name="rating" value="3">
  <label for="star3">
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-star-fill" view
    <path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73l.173 6.765c-.329-.314-.158-.888.283-.9514.898
    </svg>
  </label>
  <input type="radio" id="star4" name="rating" value="2">
  <label for="star4">
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-star-fill" view
    <path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73l.173 6.765c-.329-.314-.158-.888.283-.9514.898
    </svg>
  </label>
  <input type="radio" id="star5" name="rating" value="1">
  <label for="star5">
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-star-fill" view
    <path d="M3.612 15.443c-.386.198-.824-.149-.746-.592l.83-4.73l.173 6.765c-.329-.314-.158-.888.283-.9514.898
    </svg>
  </label>
</div>

```

Εικόνα 53:Υλοποίηση βαθμολογίας με αστέρια - Reviews

```

1 //----- STAR RATING -----//
2
3 // Add event listener to star ratings for animation effect
4 document.addEventListener('DOMContentLoaded', () => {
5   document.querySelectorAll('.star-rating:not(.readonly) label').forEach(star => {
6     star.addEventListener('click', function() {
7       this.style.transform = 'scale(1.2)';
8       setTimeout(() => {
9         this.style.transform = 'scale(1)';
10      }, 200);
11    });
12  });
13 });
14
15 // Initialize star ratings for readonly containers
16 document.addEventListener('DOMContentLoaded', () => {
17   document.querySelectorAll('.star-rating-readonly').forEach(container => {
18     const rating = container.getAttribute('data-rating'); // also returns "3"
19     const stars = container.querySelectorAll('svg');
20     stars.forEach((star, index) => {
21       if (index < rating) star.classList.add('filled');
22     });
23   });
24 });

```

Εικόνα 54:Λειτουργία βαθμολογίας με αστέρια, με χρήση JavaScript

