



2023-03-05

Πανεπιστήμιο Πειραιώς
Προηγμένα Συστήματα Πληροφορικής – Ανάπτυξη Λογισμικού και
Τεχνητής Νοημοσύνης

Τελική Εργασία για το Μάθημα
ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

των

Αποστόλου Αθανάσιου
Μπιρμπάκου Γεώργιου
Ευαγγέλου Αλέξανδρου

Table of Contents

- Εισαγωγή.....3
- Περιπτώσεις Χρήσης.....4
- User Experience (UX).....7
- Υλοποίηση Εφαρμογής.....8
 - Backend Server.....8
 - Frontend Client.....9
 - Identity Provider.....9

Εισαγωγή

Στα πλαίσια της τελικής εργασίας του μαθήματος “Ανάπτυξη Λογισμικού για τον Παγκόσμιο Ιστό” αναπτύξαμε μια διαδικτυακή εφαρμογή με τις βασικές λειτουργίες της κεντρικής σελίδας ενός κινηματογράφου. Η εφαρμογή εκτελείται τοπικά από κάποιον υπολογιστή αφού δεν ζητείται στα πλαίσια του μαθήματος αυτού να ανέβει σε κάποιον server στο internet. Η εφαρμογή υποστηρίζει επισκέπτες καθώς και εγγραφή-σύνδεση χρηστών, οι οποίοι θα μπορούν να δουν βασικά στοιχεία όπως τις ταινίες, τα καταστήματα και τις προβολές και να αλληλεπιδράσουν με αυτά ανάλογα με τους ρόλους που διαθέτουν. Ονομάσαμε την εταιρία κινηματογράφων που προσομοιώνουμε “MovieScape”.

Περιπτώσεις Χρήσης

Περιγράφονται αναλυτικά οι περιπτώσεις χρήσης που υποστηρίζει η εφαρμογή μας. Η εφαρμογή μας ξεχωρίζει 3 ρόλους για τους χρήστες. Ο 1ος είναι όταν ο χρήστης δεν έχει συνδεθεί (anonymous), ο 2ος όταν ο χρήστης έχει απλά δικαιώματα (USER) και ο 3ος όταν έχει δικαιώματα διαχειριστή (ADMIN). Αν δεν αναφέρεται κάτι στην περίπτωση χρήσης τότε είναι προσβάσιμη από όλους τους χρήστες.

- **Περιήγηση στην Αρχική σελίδα.**

Κάποιος πελάτης εισέρχεται στην αρχική σελίδα της εφαρμογής. Εκεί φαίνονται η επικεφαλίδα (header) της σελίδας με τα links στις υπόλοιπες σελίδες. Στο κέντρο της σελίδας φαίνονται οι ταινίες που παίζονται τώρα (δηλαδή αυτές που έχουν κάποια ενεργή προβολή σε μελλοντική ημερομηνία). Στο τέλος της σελίδας φαίνεται το υποσέλιδο (footer) με κάποιες πληροφορίες για την εταιρία.

- **Εμφάνιση των ταινιών.**

Με περιήγηση στην σελίδα “ΤΑΙΝΙΕΣ” ο χρήστης βλέπει όλες τις ταινίες που παίζονται τώρα. Για κάθε ταινία μπορεί να πατήσει το κουμπί “ΠΛΗΡΟΦΟΡΙΕΣ” όπου θα τον μεταφέρει στην σελίδα με περισσότερες λεπτομέρειες για αυτή την ταινία είτε το κουμπί “ΠΡΟΒΟΛΕΣ” όπου θα τον μεταφέρει στην σελίδα των προβολών στην οποία θα εμφανίζονται μόνο αυτές για την συγκεκριμένη ταινία.

- **Εμφάνιση των λεπτομερειών μιας ταινίας.**

Πατώντας το κουμπί “ΠΛΗΡΟΦΟΡΙΕΣ” σε κάποια ταινία είτε από την αρχική σελίδα είτε από την σελίδα είτε από την σελίδα των ταινιών ο χρήστης μπορεί να δει τις λεπτομέρειες μια ταινίας. Αυτές περιλαμβάνουν στοιχεία όπως τους Σκηνοθέτες, το Σενάριο, την Διάρκεια, το Έτος, την Καταλληλότητα, την περίληψη και το trailer της ταινίας.

- **Εμφάνιση των καταστημάτων.**

Με περιήγηση στην σελίδα “ΚΑΤΑΣΤΗΜΑΤΑ” ο χρήστης μπορεί να δει πληροφορίες για τα καταστήματα της εταιρίας. Οι πληροφορίες αυτές περιλαμβάνουν το τηλέφωνο, την περιγραφή, τις αίθουσες του καταστήματος με την περιγραφή τους, την διεύθυνση καθώς και ενσωματωμένο χάρτη Google Maps με τοποθετημένη πινέζα για αυτή την διεύθυνση. Πατώντας στο κουμπί “ΠΡΟΒΟΛΕΣ” ενός καταστήματος ο χρήστης μεταφέρεται στην σελίδα των προβολών στην οποία εμφανίζονται μόνο αυτές του συγκεκριμένου καταστήματος.

- **Εμφάνιση των προβολών.**

Με περιήγηση στην σελίδα “ΠΡΟΒΟΛΕΣ” ο χρήστης μπορεί να δει τις ενεργές προβολές (δηλαδή αυτές που πραγματοποιούνται σε μελλοντική ημερομηνία). Ο χρήστης μπορεί να επιλέξει τα εξής φίλτρα για να διευκολύνει την αναζήτησή του:

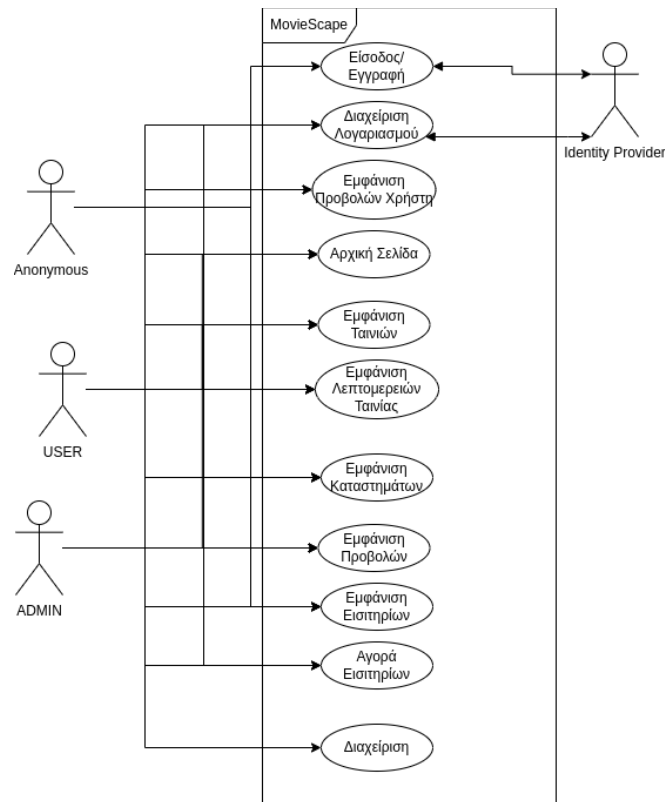
- Ταινία: όπου βλέπει μόνο τις προβολές για μια συγκεκριμένη ταινία
- Κατάστημα: όπου βλέπει τις προβολές μόνο για ένα συγκεκριμένο κατάστημα

- Ημερομηνία από: όπου βλέπει μόνο τις προβολές που πραγματοποιούνται από αυτή την ημερομηνία και μετά
- Ημερομηνία μέχρι: όπου βλέπει μόνο τις προβολές που πραγματοποιούνται μέχρι εκείνη την ημερομηνία.
- **Εμφάνιση των εισιτηρίων μιας προβολής.**
Από την λίστα των προβολών πατώντας το κουμπί “ΕΙΣΙΤΗΡΙΑ” ο χρήστης μπορεί να δει περισσότερες λεπτομέρειες για μια προβολή καθώς και τα διαθέσιμα εισιτήρια σε μορφή τετραγωνικού χάρτη/πίνακα.
- **Αγορά εισιτηρίων.**
Η περίπτωση χρήσης είναι διαθέσιμη μόνο για ρόλους απλού χρήστη ή διαχειριστή. Από την σελίδα που εμφανίζονται τα εισιτήρια μια προβολής ο χρήστης μπορεί να επιλέξει 1 ή παραπάνω εισιτήρια (αν υπάρχουν διαθέσιμα) και να πατήσει το κουμπί “ΑΓΟΡΑ”. Τότε το σύστημα επικυρώνει το αίτημα και μετά από την επιτυχημένη αγορά εμφανίζει notification στον χρήστη και του στέλνει email με τις λεπτομέρειες της αγοράς.
- **Αγορά εισιτηρίων.**
Η περίπτωση χρήσης είναι διαθέσιμη μόνο για ρόλους απλού χρήστη ή διαχειριστή. Από την σελίδα που εμφανίζονται τα εισιτήρια μια προβολής ο χρήστης μπορεί να επιλέξει 1 ή παραπάνω εισιτήρια (αν υπάρχουν διαθέσιμα) και να πατήσει το κουμπί “ΑΓΟΡΑ”. Τότε το σύστημα επικυρώνει το αίτημα και μετά από την επιτυχημένη αγορά εμφανίζει notification στον χρήστη και του στέλνει email με τις λεπτομέρειες της αγοράς.
- **Σύνδεση-Εγγραφή χρήστη.**
Η περίπτωση χρήσης αφορά μόνο μη συνδεδεμένους χρήστες (anonymous). Πατώντας το κουμπί “Είσοδος/Σύνδεση” ο χρήστης μεταφέρεται σε εξωτερικό σύστημα παρόχου ταυτότητας (identity provider) όπου μπορεί να συνδεθεί με τον λογαριασμό του ή να δημιουργήσει καινούριο. Μπορεί επίσης να συνδεθεί με λογαριασμού Google Gmail ή Facebook. Μετά την επιτυχή σύνδεση/εγγραφή ο χρήστης ξανά μεταφέρεται στην εφαρμογή μας.
- **Διαχείριση λογαριασμού χρήστη.**
Η περίπτωση χρήσης αφορά μόνο συνδεδεμένους χρήστες (USER ή ADMIN). Πατώντας το κουμπί με το όνομα του χρήστη, τότε ο χρήστης μεταφέρεται στην σελίδα του λογαριασμού του. Εκεί μπορεί να επιλέξει “ΑΠΟΣΥΝΔΕΣΗ” για να αποσυνδεθεί. Μπορεί να επιλέξει επίσης “ΔΙΑΧΕΙΡΙΣΗ ΛΟΓΑΡΙΑΣΜΟΥ” για να μεταφερθεί στο εξωτερικό σύστημα παρόχου ταυτότητας και να αλλάξει τις λεπτομέρειες του λογαριασμού του (όπως το εμφανιζόμενο όνομα).
- **Εμφάνιση των προβολών του χρήστη.**
Η περίπτωση χρήσης αφορά μόνο συνδεδεμένους χρήστες (USER ή ADMIN). Πατώντας το κουμπί με το όνομα του χρήστη, τότε ο χρήστης μεταφέρεται στην σελίδα του λογαριασμού

του. Εκεί μπορεί να επιλέξει να δει τις προβολές του. Εμφανίζονται μόνο μελλοντικές προβολές με συγκεντρωμένα όλα τα εισιτήρια του χρήστη που έχουν αγοραστεί για αυτή την προβολή.

- **Διαχείριση εφαρμογής.**

Η περίπτωση χρήσης αφορά μόνο διαχειριστές (ADMIN). Με περιήγηση στην σελίδα “ΔΙΑΧΕΙΡΙΣΗ” ο χρήστης μπορεί να δει (read), να προσθέσει (add), να ανανεώσει (update) καθώς και να διαγράψει (delete) τις βασικές οντότητες της εφαρμογής οι οποίες είναι οι ταινίες, τα καταστήματα, οι αίθουσες των καταστημάτων και οι προβολές.



User Experience (UX)

Η εφαρμογή μας προσφέρει μια αρκετά καλή εμπειρία για τον χρήστη (user experience) καθώς οι βασικές λειτουργίες είναι εμφανείς και η περιήγηση και η ανάγνωση της πληροφορίας άνετη.

Οι σελίδα έχει σχεδιαστεί με βιβλιοθήκη που τηρεί τις προδιαγραφές του Material Design της Google. Αυτές προσφέρουν καθαρά και εύκολα προσβάσιμα γραφικά στοιχεία για αλληλεπίδραση με τον χρήστη. Το προκαθορισμένο μέγεθος των στοιχείων τα καθιστά εύκολα προσβάσιμα από κινητά και οθόνες αφής. Επίσης τα προκαθορισμένα εικονίδια ανταποκρίνονται στις σύγχρονες οδηγίες σχεδιασμού και είναι εύκολο να κατανοηθούν από τον χρήστη δίνοντάς του την ευκαιρία να συμπεράνει το είδος μιας πράξης προτού την εκτελέσει.

Όλες οι σελίδες της εφαρμογής μας ανταποκρίνονται και προσαρμόζονται στο μέγεθος της οθόνης (responsiveness) κάνοντας δυνατή και ευχάριστη την χρήση τους από διαφορετικές συσκευές όπως υπολογιστές, tablet ή κινητά.

Χρησιμοποιείται η, προτεινόμενη από την βιβλιοθήκη μας, γραμματοσειρά “Roboto” της οικογένειας “sans-serif” η οποία κάνει το κείμενο ευανάγνωστο σε όλες τις σελίδες.

Έχουμε επιλέξει λίγα χρώματα. Το κύριο (primary) χρώμα μας είναι το κόκκινο το οποίο έχει παρατηρηθεί ότι τραβάει έντονα την προσοχή των ανθρώπων και τους παροτρύνει να εκτελέσουν γρήγορα ενέργειες. Το προκαθορισμένο χρώμα από πίσω (background) είναι το λευκό αλλά ο χρήστης μπορεί να αλλάξει το θέμα είτε σε ανοιχτό (light) είτε σε σκοτεινό (dark).

Υλοποίηση Εφαρμογής

Η εφαρμογή μας αποτελείται από 3 βασικά μέρη. Έναν backend server, έναν frontend client και έναν identity provider.

Backend Server

Το backend μας είναι υλοποιημένο σε γλώσσα Java και χρησιμοποιεί το framework Quarkus.

Ο κύριος ρόλος του είναι η υλοποίηση ενός REST api το οποίο επιτελεί τις λειτουργίες της εφαρμογής.

Ο http server που εκκινεί το backend μας, πέρα από το REST api, μπορεί να σερβίρει και στατικές σελίδες. Την χρησιμοποιούμε αυτή την λειτουργία του για να σερβίρουμε και τον frontend client μέσω του backend server μας.

Το backend μας μιλάει μέσω OpenID Connect (oidc) πρωτοκόλλου με τον identity provider για να παίρνει πληροφορίες για τον συνδεδεμένο χρήστη και τους ρόλους του. Έτσι τα endpoints του api μας μπορούν να είναι είτε 1) προσβάσιμα από όλους, 2) προστατευμένα μόνο για τους authenticated users ή 3) προστατευμένα μόνο για συγκεκριμένους ρόλους του authenticated user.

Για την αποθήκευση χρησιμοποιούμε MySQL βάση δεδομένων στην οποία συνδέεται το backend μας. Η πρόσβαση γίνεται μέσω του γνωστού ORM Hibernate για java και χρησιμοποιώντας το extension “Quarkus Panache” για ευκολότερη πρόσβαση.

Όταν εκτελείται σε development mode σηκώνεται επίσης μια σελίδα “Swagger UI” στην διεύθυνση <http://localhost:8080/q/swagger-ui/> όπου δείχνει πληροφορίες για το REST api μας σε συμβατή μορφή με το OpenAPI specification.

Επιπλέον χρησιμοποιώντας έναν NGINX server με ρυθμισμένο reverse proxy μπορούμε να έχουμε πρόσβαση στην εφαρμογή μας μέσω https με ένα self signed ssl certificate στην διεύθυνση <https://aics-ticket.org>

Ο κώδικας μας αν και έχει χωρισμένες λειτουργίες βασισμένες σε αρχιτεκτονική MVC δεν είναι οργανωμένος σε φακέλους “controllers”, “services”, “entities/models” κτλπ. Είναι οργανωμένος σε μια layered architecture εμπνευσμένη από το Domain Driven Design.

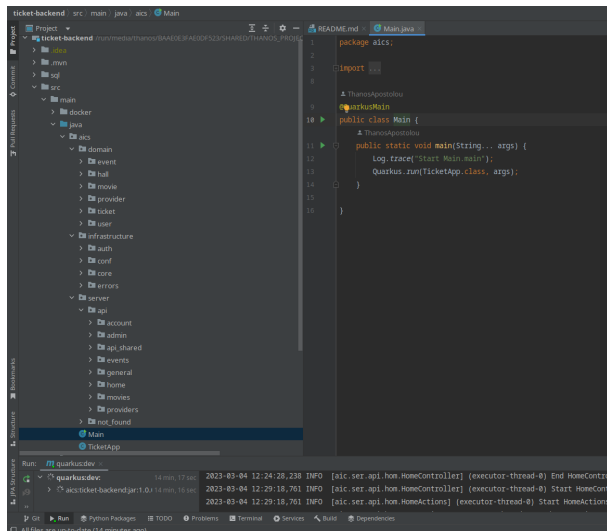


Figure 1: Backend code screenshot

Layered Architecture

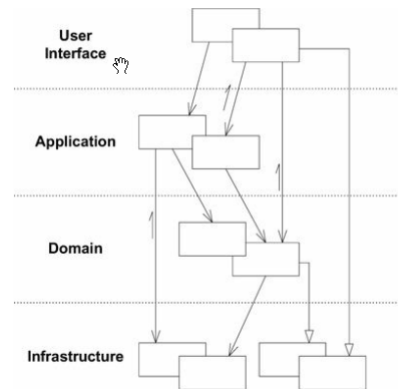


Figure 2: Layered Architecture from Domain-Driven Design: Tackling Complexity in the Heart of Software, by Eric Evans

Frontend Client

Το frontend μας είναι υλοποιημένο σε γλώσσα Typescript και χρησιμοποιεί το framework React.

Χρησιμοποιείται επίσης η βιβλιοθήκη “Material UI (mui)” για την πρόσβαση σε έτοιμα components που ακολουθούν το material design.

Υποστηρίζεται client side routing μέσω της βιβλιοθήκης “react-router”. Όλες οι σελίδες και οι υποσελίδες τους χωρίζονται σε κομμάτια (chunks) και φορτώνονται lazily (lazy loading). Έτσι όταν ένας χρήστης περιηγείται σε μια σελίδα φορτώνονται μόνο τα αρχεία (html, css, javascript) για εκείνη την σελίδα και όχι για όλη η εφαρμογή.

Το frontend μας μιλάει μέσω OpenID Connect (oidc) πρωτοκόλλου με τον identity provider για να παίρνει πληροφορίες για τον συνδεδεμένο χρήστη και τους ρόλους του. Το access_token που λαμβάνει έπειτα από επιτυχές login στέλνεται κάθε φορά στο backend μας ως “Bearer token” στο Authorization header. Για ευκολότερη χρήση των tokens καθώς και την ανανέωση τους (refresh) όταν αυτά λήγουν χρησιμοποιείται η βιβλιοθήκη “keycloak-js”.

Στο τέλος η εφαρμογή χτίζεται (build) ως στατικά αρχεία και τοποθετείται μέσα στα resources της backend εφαρμογής μας ώστε να μπορεί να σερβιριστεί από εκεί.

Identity Provider

Για identity provider χρησιμοποιούμε το open source solution Keycloak.

Έχουμε ρυθμίσει σε αυτόν 2 clients. Έναν public web client στον οποίο συνδέεται η frontend εφαρμογή μας μέσω oidc “Standard Flow” που δηλαδή σε ορολογία OAuth 2.0 αντιστοιχεί στο “Authorization Code Flow” (εννοείται πλέον με PKCE). Έναν δεύτερο client στον οποίο συνδέεται το

backend μας με την χρήση ενός “secret” μέσω του oidc “Service accounts roles” flow που δηλαδή σε ορολογία OAuth 2.0 αντιστοιχεί στο “Client Credentials Grant” flow.

Έχουμε ρυθμίσει τον identity provider με την σειρά του να συνδέεται με τους εξωτερικούς identity providers “Facebook” και “Google” μέσω OAuth 2.0 ώστε να μπορεί ο χρήστης να συνδεθεί με τον λογαριασμό του σε Facebook ή Google/Gmail.

Έχουμε προσθέσει τους ρόλους για την εφαρμογή μας “TICKET_USER” και “TICKET_ADMIN”. Κάθε νέος χρήστης που δημιουργείται λαμβάνει τον ρόλο “TICKET_USER” εξ ορισμού. Ο ρόλος TICKET_ADMIN μπορεί να δοθεί σε κάποιον χρήστη μόνο από έναν admin με πρόσβαση στο keycloak.