­Έγγραφο απαιτήσεων λογισμικού (SRS)

ΠΡΟΣΑΡΜΟΓΗ ΤΟΥ ΑΝΤΙΣΤΟΙΧΟΥ ΕΓΓΡΑΦΟΥ ΤΟΥ ΠΡΟΤΥΠΟΥ ISO/IEC/IEEE 29148:2011

NTUAFLIX

# Εισαγωγή­

## 1.1 Εισαγωγή: σκοπός του λογισμικού

Το λογισμικό αυτό έχει ως στόχο την πρόσβαση των χρήστη σε τηλεοπτικό και κινηματογραφικό υλικό και την αλληλεπίδρασή τους με αυτό. Ειδικότερα, οι χρήστες αφού εγγραφούν στο σύστημα, έχουν την δυνατότητα να εκτελέσουν ποικίλες λειτουργίες.

Αρχικά, μπορούν να πλοηγηθούν στα στοιχεία των ταινιών δηλαδή στον τίτλο, στο είδος, στο trailer, στους ηθοποιούς, τους σκηνοθέτες και στο εξώφυλλό τους, είτε απευθείας από την αρχική σελίδα είτε κατόπιν αναζήτησης κατά τίτλο ή κατά είδος.

Επιπλέον, είναι δυνατή η δημιουργία προσωπικών συλλογών όπως «παρακολούθηση αργότερα» και «αγαπημένα».

Ο χρήστης θα μπορεί, επίσης, να αξιολογήσει τις ταινίες που παρακολούθησε βάζοντας «αστεράκια» και γράφοντας την κριτική του.

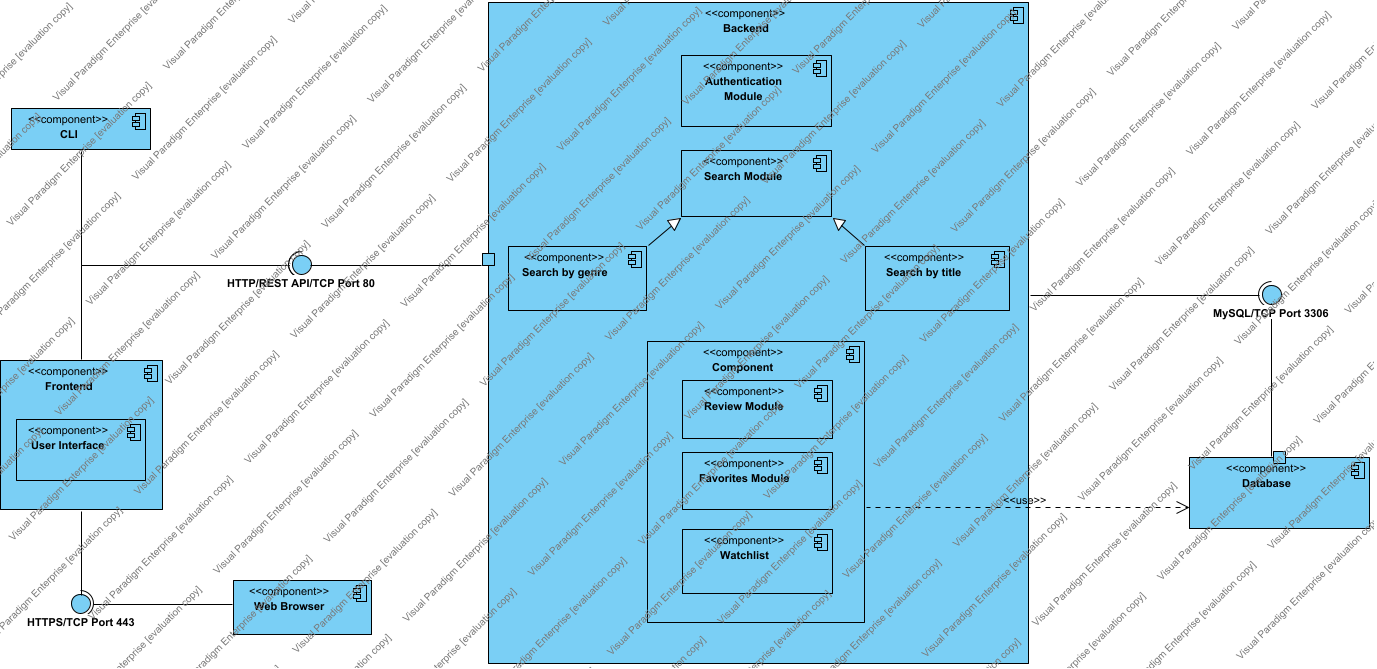
Πέρα από αυτά, θα υπάρχει και ένα CLI μέσω του οποίου οι χρήστες θα μπορούν να πραγματοποιούν όλες τις παραπάνω λειτουργίες, ενώ ο διαχειριστής θα έχει επιπλέον την δυνατότητα να προσθέτει χρήστες και ταινίες, να κάνει reset και health check για τον έλεγχο της υγείας του συστήματος.

## 1.2 Διεπαφές (interfaces)

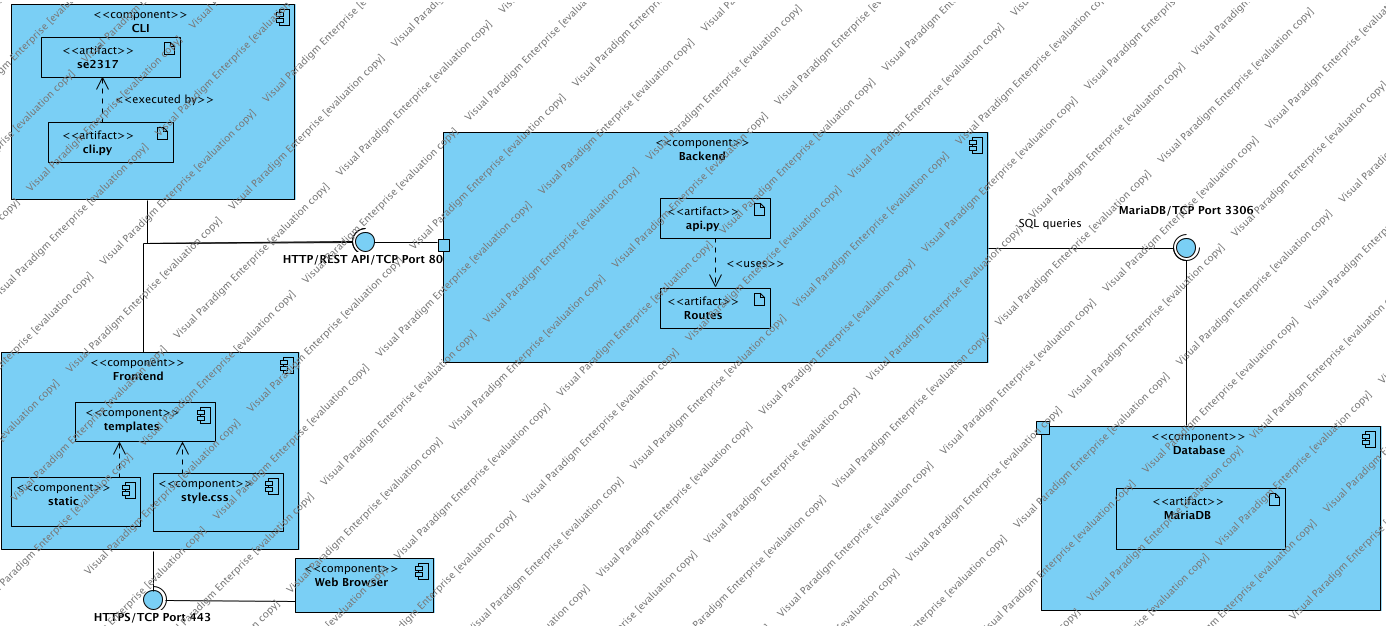
### 1.2.1 Διεπαφές με εξωτερικά συστήματα

Χρησιμοποιούμε το υποσύνολο δεδομένων που είναι ανεβασμένα στο helios, επομένως δεν γίνεται χρήση του IMDB API. Έτσι, το σύστημα μας δεν επικοινωνεί με εξωτερικά συστήματα.

# Component Diagram



# Deployment Diagram



### 1.2.2 Διεπαφές με το χρήστη

N/A

# 2. Αναφορές - πηγές πληροφοριών

Ν/Α

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

Λεπτομερής προδιαγραφή των λειτουργιών του λογισμικού σε επίπεδο περιπτώσεων χρήσης. Ο αριθμός των περιπτώσεων χρήσης ανάλογα με τον αριθμό των μελών της ομάδας σύμφωνα με την εκφώνηση.

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Διαπίστευση Χρηστών

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

* Χρήστες εφαρμογής («User»)
* Διαχειριστής («Admin»)

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

Σύνδεση στο διαδίκτυο και συμπλήρωση των απαιτούμενων στοιχείων.

#### 3.1.1.3 Περιβάλλον εκτέλεσης

* CLI/Web Server
* Backend Server
* Database

#### 3.1.1.4 Δεδομένα εισόδου

Δεδομένα εισόδου: Username, Password.

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

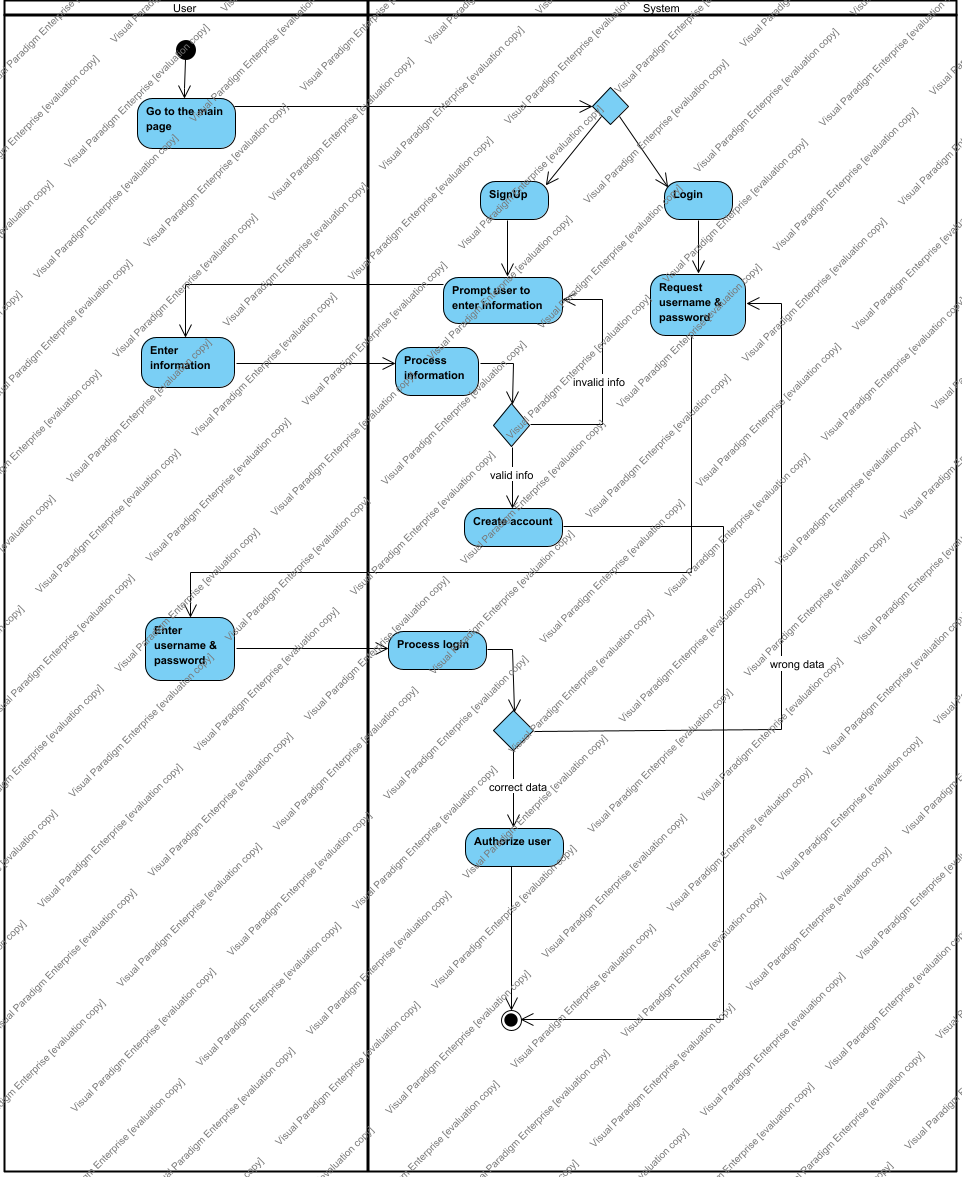
Sign Up:

* Συμπλήρωση στοιχείων εγγραφής
* Αποδοχή όρων χρήσης
* Έγκριση στοιχείων
* Αν εγκριθούν τα στοιχεία, σε παραπέμπει στη σελίδα του login
* Αν όχι, ξαναζητάει στοιχεία εγγραφής

Log In:

* Συμπλήρωση στοιχείων
* Επικύρωση στοιχείων
* Αν τα στοιχεία είναι έγκυρα, ο χρήστης παραπέμπεται στην αρχική σελίδα
* Αν όχι, ο χρήστης ενημερώνεται για την λανθασμένη συμπλήρωση των στοιχείων και του ξανά-ζητείται η συμπλήρωσή τους

# Activity Diagram



#### 3.1.1.7 Δεδομένα εξόδου

Έχουμε 2 περιπτώσεις:

* **Login**: Αν ο χρήστης συμπληρώσει τα σωστά στοιχεία, επαληθεύεται από το σύστημα και εισέρχεται στην εφαρμογή. Διαφορετικά, του ζητείται να τα επαναλάβει.
* **SignUp**: Το σύστημα παραπέμπει τον χρήστη να συμπληρώσει τα στοιχεία του. Αν είναι έγκυρα, τότε δημιουργείται νέο account για τον συγκεκριμένο user, αλλιώς ο χρήστης πρέπει να ξανασυμπληρώσει τα απαραίτητα πεδία.

#### 3.1.1.8 Παρατηρήσεις

Μέσω του CLI, πραγματοποιείται η εγγραφή των χρηστών από τον admin και οι χρήστες μπορούν να κάνουν login.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Αξιολόγηση Ταινιών

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

* Χρήστες εφαρμογής («User»)

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

* Επιτυχές Login
* Να είναι «User»

#### 3.1.2.3 Περιβάλλον εκτέλεσης

* Web Browser
* Back-End server
* DB server

#### 3.1.2.4 Δεδομένα εισόδου

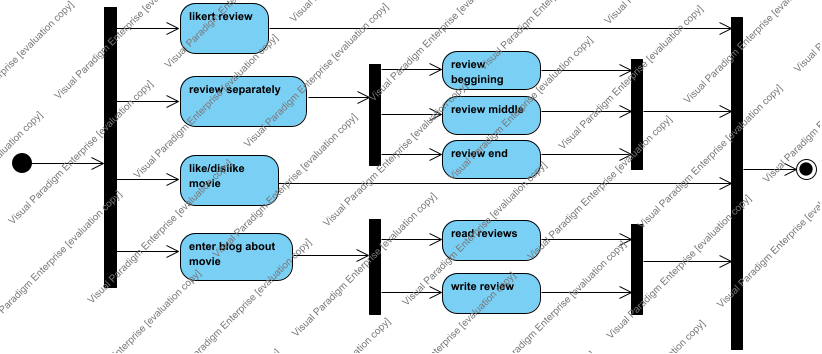
Απαιτείται να δοθούν από τον χρήστη στο σύστημα:

**Αξιολόγηση:**

* Like/Dislike enumeration («LIKE», «DISLIKE»)
* Βαθμολογία Likert int
* Αξιολόγηση εξέλιξης ταινίας 2D int array
* Κριτική Ταινίας string

#### 3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

* Rate / Like movie
* Write / Read Review



#### 3.1.2.7 Δεδομένα εξόδου

Εμφανίζονται η συνολική αξιολόγηση της ταινίας από όλους τους χρήστες της εφαρμογής:

* Ο αριθμός όλων των likes και των dislikes
* Ο μέσος όρος του Likert rating της ταινίας
* Οι κριτικές των ταινιών από τους χρήστες εμφανίζονται σταδιακά κάνοντας scroll down
* Τα σχεδιαγράμματα των χρηστών για την αξιολόγηση της εξέλιξης της ταινίας

#### 3.1.2.8 Παρατηρήσεις

N/A

### 3.1.3 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 3: Αναζήτηση Ταινίας

#### 3.1.3.1 Χρήστες (ρόλοι) που εμπλέκονται

* Χρήστες εφαρμογής (“User”)

#### 3.1.3.2 Προϋποθέσεις εκτέλεσης

* Επιτυχές Login
* Να είναι “User”

#### 3.1.3.3 Περιβάλλον εκτέλεσης

* Web Browser

#### 3.1.3.4 Δεδομένα εισόδου

* Αναζήτηση με τίτλο string
* Αναζήτηση με ηθοποιό string
* Αναζήτηση με είδος string

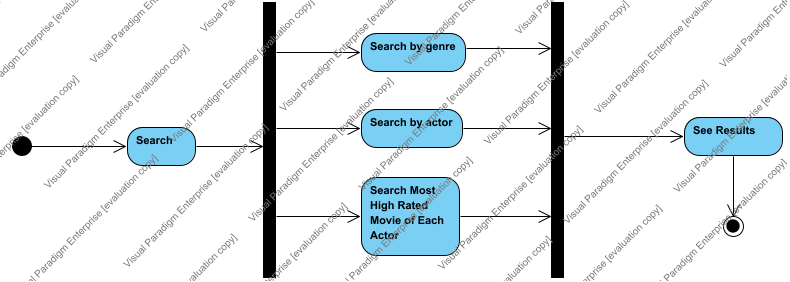
#### 3.1.3.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

1) Πάτημα Κουμπιού Search

2) Συμπλήρωση Κειμένου με βάση το κριτήριο που θες να αναζητήσεις

3) Προβολή Αποτελεσμάτων

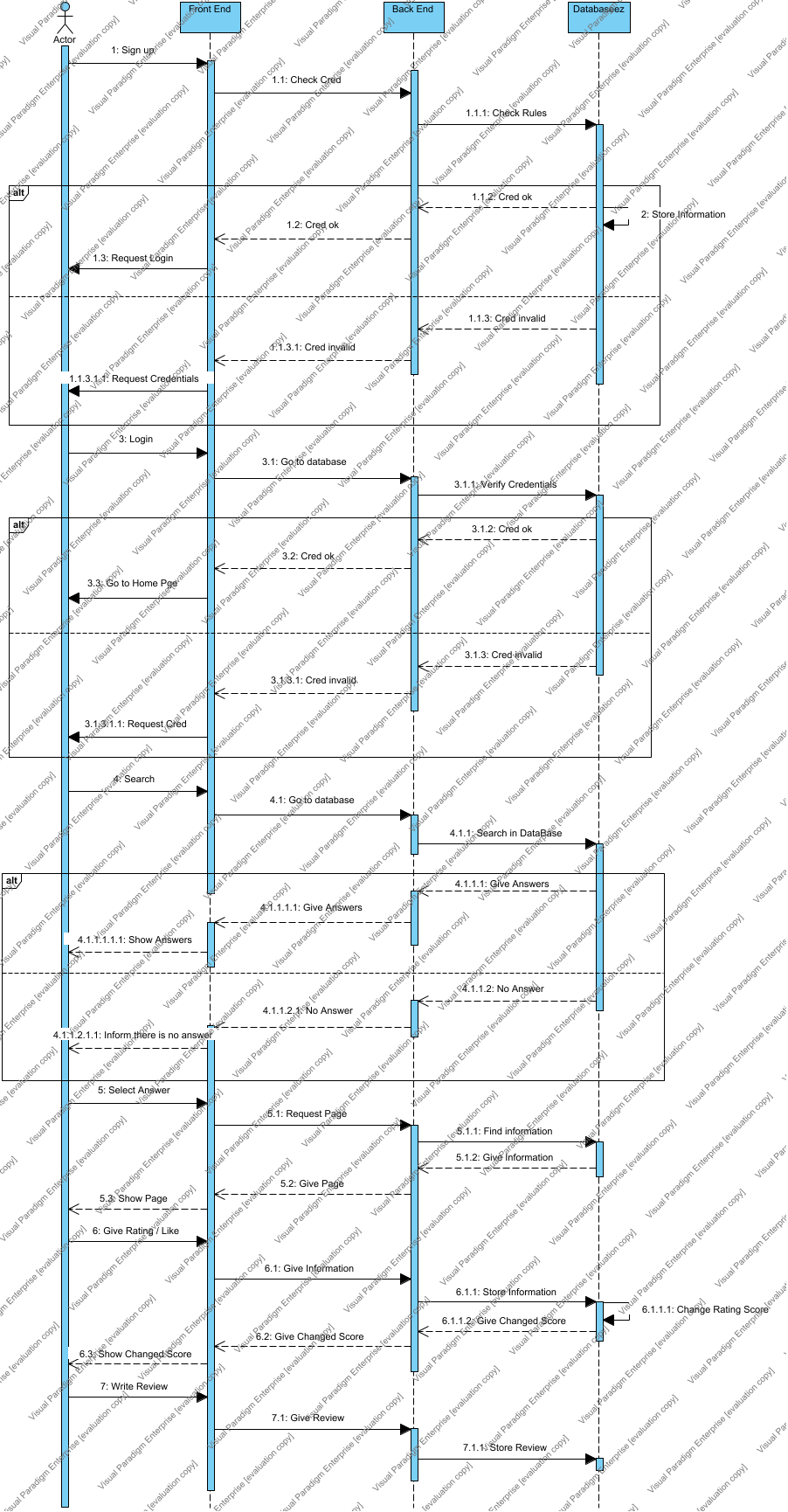
# Activity Diagram



#### 3.1.3.7 Δεδομένα εξόδου

Εμφανίζονται όλες οι ταινίες που πληρούν τα κριτήρια αναζήτησης. Δεν υπάρχει κάποια αλλαγή στα δεδομένα.

**Sequence Diagram**



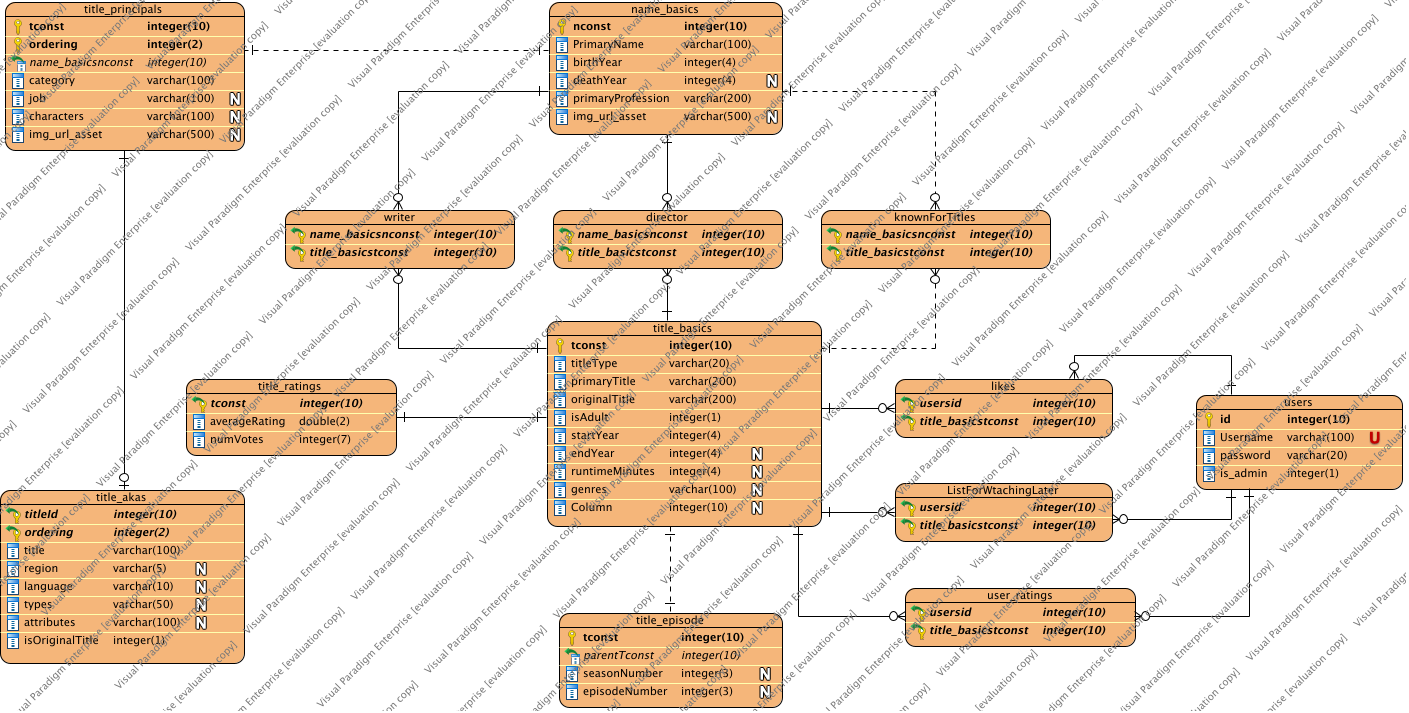
­­

## 3.2 Απαιτήσεις επιδόσεων

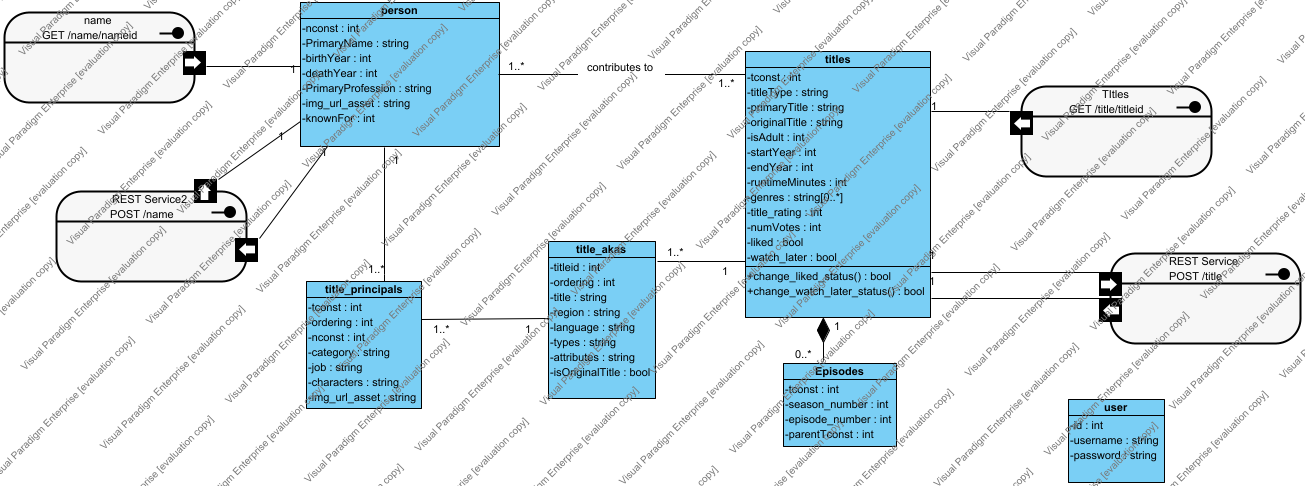
Ποσοτική τεκμηρίωση μέτρων και κριτηρίων επιθυμητών επιδόσεων με αναφορά στα ποσοτικά χαρακτηριστικά εισόδων και φορτίου του λογισμικού.

## 3.3 Απαιτήσεις οργάνωσης δεδ­­ομένων

# Entity Relationship Diagram



# Class API Diagram



### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Έχουμε δύο τύπους χρηστών “User” και “Admin”. Οι “User” αποκτούν πρόσβαση σε πληροφορίες των ταινίων μέσω των use cases αλληλεπιδρώντας με το front end. Ο «Αdmin» έχει πλήρη εποπτεία της βάσης και έχει πρόσβαση στα προσωπικά δεδομένα όλων των χρηστών όπως username, password. Επίσης λογαριασμοί χρηστών θα δημιουργούνται από τον “Admin” του συστήματος μέσω του CLI.

### 3.4 Περιορισμοί σχεδίασης

Front-End

* Για τη μορφοποίηση, τη διαδραστικότητα και τη γενική διαμόρφωση της ιστοσελίδας θα γίνει η χρήση **Javascript**, η οποία συνεπάγεται κάποιους περιορισμούς στην ταχύτητα, ανάλογα με τη χρήση της, καθώς και τη μεταβλητότητα που μπορεί να παρουσιάσει σε διαφορετικούς browsers.
* Μέσα στα αρχεία της Javascript ενσωματώνουμε κώδικες **HTML** για την κατασκευή του σκελετού της εφαρομογής και **CSS** για την μορφοποίησή της

Back-End

* Χρήση της γλώσσας **Python** για την υλοποίηση του back-end της εφαρμογής.
* Για τη συγγραφή δοκιμαστικού (test) κώδικα και τη διενέργεια ελέγχων στα endpoints του back-end χρησιμοποιήθηκε το Postman

CLI

* Το Command Line Interface γράφτηκε σε Python.
* Η διενέργεια ελέγχων έγινε με το εργαλείο **pytest** (full-featured Python testing tool)

Version Control

* Χρήση του Git (και πιο συγκεκριμένα της πλατφόρμας Github) για την διαδικασία version control του συστήματος.

## 3.5 Λοιπές απαιτήσεις

### 3.5.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Το λογισμικό πρέπει να είναι διαθέσιμο 24/7 και να μπορεί να χρησιμοποιείται από πολλαπλούς χρήστες ταυτόχρονα.

### 3.5.2 Απαιτήσεις ασφάλειας

Πρόσβαση στα δεδομένα των ερευνών να είναι επιτρεπτή ΜΟΝΟ στον διαχειριστή/δημιουργό της.

* Χρήση HTTPS και SSL certificate.
* Πρόσβαση στα δεδομένα των χρηστών να είναι επιτρεπτή ΜΟΝΟ στον διαχειριστή/δημιουργό της, καθώς και η δημιουργία αυτών.

### 3.5.3 Απαιτήσεις συντήρησης

* Λόγω της στατικής φύσης των δεδομένων, αρκεί να μπορεί το σύστημα να τροφοδοτηθεί από κάποιο backup αποθηκευτικό χώρο.
* Συντήρηση του συστήματος θα εκτελείται κάθε 3 μήνες και σε έκτακτες περιστάσεις. Η συχνότητα αυτή υπόκειται σε αλλαγές ανάλογα με τον αριθμό χρηστών και την απόδοση του συστήματος.