



## Πληροφοριακά Συστήματα (Εαρ. Εξ. 2019-2020) Απαλλακτική εργασία

Η εταιρεία στην οποία εργάζεστε σας έχει αναθέσει την υλοποίηση του Πληροφοριακού Συστήματος MovieFlix, ένα σύστημα το οποίο θα χρησιμοποιείται για την αναζήτηση, βαθμολόγηση και σχολιασμό ταινιών.

Στο MovieFlix θα υπάρχουν δύο κατηγορίες χρηστών: (i) οι διαχειριστές, και (ii) οι απλοί χρήστες. Πριν την εκτέλεση των λειτουργιών που περιγράφονται παρακάτω, οι χρήστες του MovieFlix θα πρέπει να κάνουν εγγραφή στο σύστημα. Η εγγραφή θα επιτρέπει την εισαγωγή νέων χρηστών στο σύστημα οι οποίοι θα κατατάσσονται στη κατηγορία «Απλός Χρήστης». Για την εγγραφή στο σύστημα απαιτούνται τα παρακάτω στοιχεία:

- e-mail: Το e-mail του χρήστη
- name: Το ονοματεπώνυμο του χρήστη
- password: Ο κωδικός που θα εισάγει ο χρήστης.

Ένας νέος χρήστης θα μπορεί να κάνει εγγραφή στο σύστημα αν και μόνο αν δεν υπάρχει ήδη άλλος με το ίδιο e-mail.

Ένας **απλός χρήστης** θα μπορεί να εκτελέσει τις παρακάτω λειτουργίες:

- **Είσοδος στο σύστημα:** Ο χρήστης θα εισάγει το e-mail του και το κωδικό του και αν τα στοιχεία είναι έγκυρα, θα γίνεται επιτυχής είσοδος στο MovieFlix. Διαφορετικά, θα εμφανίζεται ανάλογο μήνυμα που θα προτρέπει το χρήστη να εισάγει τα στοιχεία του και πάλι. Μόνο στη περίπτωση που ένας χρήστης έχει κάνει επιτυχημένα την είσοδό του στο σύστημα θα μπορεί να εκτελέσει τις παρακάτω λειτουργίες!
- **Αναζήτηση ταινίας:** Ένας χρήστης θα μπορεί να αναζητήσει ταινίες βάσει:
  - Τίτλου: Θα εμφανίζονται όλες οι ταινίες που έχουν τίτλο ίδιο με αυτόν που έχει δώσει ο χρήστης
  - Έτους κυκλοφορίας: Θα εμφανίζονται όλες οι ταινίες που έχουν έτος κυκλοφορίας, το έτος που έχει δώσει ο χρήστης
  - Πρωταγωνιστών: Θα εμφανίζονται όλες οι ταινίες στις οποίες θα έχει πρωταγωνιστήσει ο/η ηθοποιός που αναζήτησε ο χρήστης
- **Εμφάνιση πληροφοριών ταινίας:** Πιο συγκεκριμένα, για κάθε ταινία θα εμφανίζονται τα παρακάτω:



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

- Τίτλος
- Έτος κυκλοφορίας
- Πλοκή ταινίας
- Πρωταγωνιστές
- **Εμφάνιση σχολίων** που έχουν γίνει από άλλους χρήστες για αυτή τη ταινία. Πιο συγκεκριμένα, θα εμφανίζονται: το email του χρήστη που έχει κάνει το σχόλιο και το ίδιο το σχόλιο.
- **Βαθμολόγηση ταινίας**
- **Αφαίρεση βαθμολογίας** σε ταινία που έχει ο/η ίδιος/α βαθμολογήσει
- **Εισαγωγή ενός σχολίου** σε ταινία
- **Εμφάνιση όλων των σχολίων που έχει κάνει ο χρήστης**
- **Εμφάνιση όλων των βαθμολογιών που έχει δώσει ο χρήστης**
- **Διαγραφή σχολίου που έχει κάνει ο/η ίδιος/α** σε μία ταινία
- **Διαγραφή του λογαριασμού του** από το MovieFlix

Ένας **διαχειριστής** θα μπορεί να εκτελέσει τις λειτουργίες ενός απλού χρήστη και τις παρακάτω:

- **Εισαγωγή νέας ταινίας στο MovieFlix:** Θα εισάγεται μία ταινία στο σύστημα. Για να εισαχθεί μια ταινία, απαραίτητη είναι η εισαγωγή του τίτλου της και ενός τουλάχιστον πρωταγωνιστή
- **Διαγραφή ταινίας από το MovieFlix:** Η διαγραφή θα γίνεται βάσει τίτλου ταινίας. Αν υπάρχουν παραπάνω από μία ταινίες με τον ίδιο τίτλο, να διαγράφεται η παλαιότερη (αυτή δηλαδή με το παλαιότερο έτος κυκλοφορίας)
- **Ενημέρωση υπάρχουσας ταινίας:** Θα ενημερώνονται ένα ή περισσότερα πεδία από τα παρακάτω:
  - Τίτλος: Αλλαγή του τίτλου της ταινίας
  - Έτος κυκλοφορίας: Αλλαγή του έτους κυκλοφορίας
  - Πλοκή: Αλλαγή της πλοκής
  - Πρωταγωνιστές: Εισαγωγή νέων πρωταγωνιστών ή/και διαγραφή ήδη υπάρχοντων
- **Διαγραφή σχολίων άλλων χρηστών από μία ταινία** (Ένας διαχειριστής δηλαδή, θα μπορεί να διαγράψει σχόλια και άλλων χρηστών πέρα από τα δικά του)



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

- **Αλλαγή κατηγορίας ενός χρήστη από απλό σε διαχειριστή** (ένας διαχειριστής θα μπορεί να αναβαθμίσει έναν χρήστη κατηγορίας «Απλού» σε «Διαχειριστή»)
- **Διαγραφή οποιουδήποτε απλού χρήστη** (δεν θα επιτρέπεται δηλαδή η διαγραφή ενός διαχειριστή)

Να υλοποιηθεί web service με χρήση Python και Flask το οποίο θα παρέχει τα απαραίτητα endpoint στους χρήστες του, ώστε να μπορούν να εκτελεστούν οι παραπάνω λειτουργίες. Το web service που θα υλοποιηθεί, να συνδέεται με ένα container της MongoDB. Σε αυτό θα υπάρχει η βάση δεδομένων **MovieFlix** η οποία θα έχει τα παρακάτω collection:

- **Users:** Το collection αυτό θα περιέχει τους χρήστες του MovieFlix. Για κάθε χρήστη θα πρέπει να αποθηκεύονται τα παρακάτω:
  - name: Ονοματεπώνυμο του χρήστη
  - e-mail: Το e-mail του χρήστη
  - password: Το password του χρήστη
  - comments []: Τα σχόλια που έχει κάνει ο κάθε χρήστης σε μία ταινία
  - category: Κατηγορία χρήστη (απλός ή διαχειριστής)
- **Movies:** Το collection αυτό θα περιέχει τις ταινίες που υπάρχουν στο MovieFlix. Για κάθε ταινία θα πρέπει να αποθηκεύονται τα παρακάτω:
  - title: Τίτλος ταινίας
  - year: Έτος κυκλοφορίας
  - description: Περιγραφή της πλοκής της ταινίας
  - actors []: Πίνακας που θα περιέχει τους πρωταγωνιστές της ταινίας
  - rating: Βαθμολογία ταινίας στο MovieFlix
  - comments []: Πίνακας που θα περιέχει τα σχόλια που έχουν κάνει οι χρήστες του MovieFlix για αυτή τη ταινία.

Να γίνει containerize το web service που θα υλοποιηθεί περιγράφοντας ακριβώς τα βήματα που πρέπει να ακολουθηθούν από το Docker στο Dockerfile για τη δημιουργία του image. Στη συνέχεια, να δημιουργηθεί το αρχείο docker-compose.yml το οποίο θα συνδέει τα δύο container (δηλαδή το web service και η MongoDB) ώστε να τρέχουν μαζί.

Το container της βάσης δεδομένων, να έχει volume σε ένα φάκελο του host που θα ονομάζεται data, ώστε στη περίπτωση που το container διαγραφεί, να αποφευχθεί η απώλεια των δεδομένων.



### Τρόπος παράδοσης:

Θα πρέπει να ολοκληρωθούν **όλα τα παρακάτω** ώστε να εξεταστεί και βαθμολογηθεί η εργασία:

1. Να δημιουργηθεί ένα private repository στο GitHub με όνομα MovieFlix2020\_AM\_Erwnymo\_Onoma, στο οποίο να δώσετε πρόσβαση στα GitHub accounts: csymnoul και jdtotow.

Το repository που θα δημιουργήσετε θα περιέχει πέραν του **κώδικα** του web service που υλοποιήσατε, ένα **README.md** αρχείο το οποίο θα περιγράφει επακριβώς τις λειτουργίες του συστήματος, καθώς και το τρόπο με τον οποίο μπορεί κάποιος να το εκτελέσει στον υπολογιστή του.

2. Στον Εύδοξο, στην ενότητα Εργασίες έχει δημιουργηθεί μία νέα εργασία με τίτλο «Απαλλακτική Εργασία». Εκεί θα πρέπει να ανεβάσετε ένα zip/rar αρχείο που θα περιέχει τα πακάτω:

- Ένα **txt** αρχείο που θα περιέχει το url για το repository που έχει δημιουργηθεί στο GitHub. Το αρχείο αυτό να ονομαστεί **AM\_Erwnymo\_Onoma.txt**.
- Ένα **doc** αρχείο με όνομα **AM\_Erwnymo\_Onoma.doc** που θα περιέχει τα εξής:
  - i. Διάγραμμα ροής δεδομένων του πληροφοριακού συστήματος
  - ii. Πίνακας ρίσκων για την υλοποίηση του πληροφοριακού συστήματος
  - iii. Προτεινόμενο διάγραμμα Gantt για τη διαχείριση της υλοποίησης του πληροφοριακού συστήματος

### Ημερομηνίες παράδοσης / εξέτασης:

- Οι εργασίες θα πρέπει να κατατεθούν (GitHub και ΕΥΔΟΞΟΣ) **πέντε (5) ημέρες** πριν ημερομηνία εξέτασης του μαθήματος βάσει του Προγράμματος της Εξεταστικής Περιόδου Ιουνίου-Ιουλίου 2020.
- Η εξέταση των εργασιών θα πραγματοποιηθεί διαδικτυακά σε ομαδική κλήση κατά την ημερομηνία της εξέτασης του μαθήματος όπως θα ανακοινωθεί στο πρόγραμμα εξετάσεων από το Τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς.



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

**ΠΡΟΣΟΧΗ:**

1. Εργασίες με commit στο GitHub με ημερομηνία μεταγενέστερη της ημέρας παράδοσης της εργασίας θα μηδενιστούν!
2. Η εργασία είναι αυστηρώς ατομική! Ο κώδικας των εργασιών θα εξεταστεί για αντιγραφή και σε περίπτωση που βρεθούν διπλότυπα θα μηδενίζονται οι προσπάθειες όλων των εμπλεκομένων!

Για απορίες σχετικά με την εργασία να στείλετε email: [simvoul@unipi.gr](mailto:simvoul@unipi.gr), [totow@unipi.gr](mailto:totow@unipi.gr), [dimos@unipi.gr](mailto:dimos@unipi.gr)