

Εργασία στο μάθημα "Τεχνολογία Λογισμικού"

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο

Χειμερινό εξάμηνο 2023-2024

Διδάσκων: Β. Βεσκούκης

Θεματικό πεδίο

Η διάθεση τηλεοπτικού και κινηματογραφικού υλικού όπως σειρές, ταινίες, ντοκιμαντέρ, κ.ά. μέσω δικτυακών μέσων (streaming) επιτρέπει το υλικό αυτό να συνοδεύεται από κάθε είδους μεταδεδομένα κ.ά. ψηφιακά στοιχεία (assets) όπως posters, trailers, βιογραφίες κλπ. Τα μεταδεδομένα αυτά αφορούν τους κάθε είδους συντελεστές, την ταξινόμηση σε είδη (genres), εικόνες, βιογραφικά πρωταγωνιστών, κ.ά. Με βάση τη δυνατότητα πρόσβασης σε τέτοια δεδομένα, το θεματικό πεδίο της εργασίας είναι η πλοήγηση και ανάλυση δεδομένων κινηματογραφικών ταινιών και σειρών, τα οποία διατίθενται ελεύθερα.

Εκτός από τα site των εταιριών παραγωγής, υπάρχουν διάφορα site που διαθέτουν δεδομένα ταινιών κλπ, όπως imdb.com, themoviedb.org, rottentomatoes.com, letterboxd.com, κ.ά. Τα περισσότερα από αυτά διαθέτουν ίσως και το σύνολο των δεδομένων τους, είτε για κατέβασμα, είτε μέσω API για ενσωμάτωση σε άλλες εφαρμογές. Τα διατιθέμενα δεδομένα που θα χρησιμοποιηθούν στην εργασία παρέχονται ελεύθερα από το δικτυακό τόπο imdb.com και αφορούν ταινίες και τηλεοπτικές σειρές για περισσότερα από 50 έτη. Έχοντας πρόσβαση σε τέτοια δεδομένα, μπορεί κανείς να πλοηγηθεί, να επιλέξει, αξιολογήσει, οργανώσει, αναλύσει στοιχεία σχετικά με κινηματογραφικό και τηλεοπτικό περιεχόμενο.

Διατύπωση ζητουμένου

Το αντικείμενο της εργασίας είναι η ανάπτυξη του "ntuaflix", μιας εφαρμογής λογισμικού για την πλοήγηση σε δεδομένα ταινιών και σειρών, την εξαγωγή στατιστικών στοιχείων με διάφορα κριτήρια, καθώς και τη δημιουργία και διαχείριση προσωπικών συλλογών, προτάσεις για παρακολούθηση, κ.ά. Η εργασία περιλαμβάνει την αναγνώριση και προδιαγραφή των απαιτήσεων, την αρχιτεκτονική και λεπτομερή σχεδίαση, την υλοποίηση επιλεγμένων λειτουργιών, τον έλεγχο και την τεκμηρίωση όλων αυτών. Η προδιαγραφή των απαιτήσεων και η σχεδίαση, η διαχείριση του κώδικα, το testing, καθώς και η διοίκηση του έργου, θα γίνουν με χρήση κατάλληλων εργαλείων. Το σύνολο των παραδοτέων εγγράφων θα παραχθεί αυτόματα, δηλαδή δεν θα υπάρξει συγγραφή κειμένων σε word κλπ.

Η αναγνώριση των λειτουργιών που θα περιλαμβάνει το λογισμικό, κατόπιν της συζήτησης που έγινε στο μάθημα, έχει οδηγήσει στη δημιουργία καταλόγου με τις δυνητικές λειτουργίες του ntuaflix, ο οποίος είναι διαθέσιμος στη σελίδα του μαθήματος στο helios.ntua.gr. Από τον κατάλογο αυτό οι ομάδες θα επιλέξουν τις λειτουργίες που θα υλοποιήσουν ενσωματώνοντάς τις σε περιπτώσεις χρήσης.

Τα τμήματα της εφαρμογής που θα κατασκευάσετε έχουν ως εξής:

- Ένα υποσύστημα back-end, το οποίο θα υποστηρίζει λειτουργίες πλοήγησης, επιλογής, ανάλυσης, αποθήκευσης ταινιών κλπ. Οι λειτουργίες αυτές θα διατίθενται μέσω ενός REST API, οι προδιαγραφές του οποίου θα ανακοινωθούν σε κατάλληλο χρόνο.

2. Μία εφαρμογή CLI (Command Line Interface) για τις παραπάνω λειτουργίες, η οποία θα λειτουργεί ως client του REST API που παρέχεται από το back-end υποσύστημα, προσφέροντας στο χρήστη της τη δυνατότητα να εκτελεί λειτουργίες μέσω ενός shell.
3. Μία frontend εφαρμογή (κατά προτίμηση δικτυακή), η οποία θα προσφέρει δυνατότητες πλοήγησης στις ταινίες - σειρές και υλοποίησης των επιλεγμένων λειτουργιών. Η εφαρμογή αυτή επίσης θα πρέπει να είναι client του REST API..

Ομάδες

Η εργασία θα γίνει σε **ομάδες των 3-6 ατόμων**, οι οποίες θα υλοποιήσουν τον πλήρη κύκλο ανάπτυξης του συστήματος (ανάλυση απαιτήσεων, σύνταξη προδιαγραφών, σχεδιασμός και αρχιτεκτονική, υλοποίηση και έλεγχοι αποδοχής, εγκατάσταση και λειτουργία). Το πλήθος των περιπτώσεων χρήσης που θα υλοποιηθούν συναρτάται με το πλήθος των ατόμων σε κάθε ομάδα. Από τη φύση του γνωστικού αντικειμένου της Τεχνολογίας Λογισμικού, ομάδες με λιγότερα από 4 άτομα είναι δύσκολο να έχουν αναλογικά λιγότερο φόρτο και δεν συνίστανται.

Για τη συνεργατική διαχείριση των εκδόσεων της τεκμηρίωσης και του πηγαίου κώδικα, είναι υποχρεωτική η χρήση του Github, καθώς και άλλων εργαλείων όπως περιγράφεται παρακάτω. Η συγκρότηση των ομάδων θα πρέπει να έχει ολοκληρωθεί το αργότερο μέχρι τις **22.10.2023**.

Ελάχιστες κοινές τεχνικές προδιαγραφές

Το σύστημα που θα αναπτύξετε θα πρέπει να υποστηρίζει τα εξής χαρακτηριστικά (ελάχιστες κοινές προδιαγραφές):

1. Οι λειτουργίες που θα υλοποιηθούν ομαδοποιημένες σε "περιπτώσεις χρήσης" (use cases) θα επιλεγούν υποχρεωτικά από τον κατάλογο που έχει ανακοινωθεί στο helios.
2. Το back-end θα παρέχει REST API, το οποίο απαιτείται να είναι συμβατό με το πρότυπο OpenAPI 3.0. Για τα υποχρεωτικά endpoints θα ανακοινωθούν προδιαγραφές και θα είναι κοινά για όλες τις ομάδες. Είναι αποδεκτή η δημιουργία και επιπλέον endpoints, ωστόσο αυτά που θα ανακοινωθούν πρέπει να υλοποιηθούν ακριβώς σύμφωνα με την προδιαγραφή ώστε να είναι δυνατός ο αυτόματος έλεγχος κατά την παρουσίαση της εργασίας.
3. Η γλώσσα των διεπαφών χρήστη στην εφαρμογή CLI θα είναι η αγγλική. Η γλώσσα των διεπαφών χρήστη θα είναι η ελληνική ή η αγγλική, με (προαιρετική) δυνατότητα επιλογής.
4. Κάθε ομάδα θα συντάξει σενάρια ελέγχου και θα ενσωματώσει την εκτέλεση των αντίστοιχων δοκιμών με αυτόματο τρόπο για τις λειτουργίες του back-end υποσυστήματος.
5. Για το frontend σε περιβάλλον web, πρέπει να υποστηρίζεται το πρωτόκολλο HTTPS για όλες τις διεπαφές, μέσω self-signed certificate.
6. Θα δοθεί μέσω helios υποσύνολο των δεδομένων του imdb που θα εισαχθούν (import) στο ntuaflix. Προαιρετικά, είναι δυνατή η λήψη δεδομένων μέσω του API του imdb. Σε περίπτωση μη χρήσης του API, η εισαγωγή του υποσυνόλου που θα δοθεί μέσω helios είναι υποχρεωτική.

Τεχνικές απαιτήσεις - εργαλεία

- **GitHub** ως περιβάλλον διαχείρισης κώδικα. Είναι υποχρεωτική η χρήση του username ntua-elXXXXX, με το email elXXXXX@mail.ntua.gr σύμφωνα με τις οδηγίες που έχουν δοθεί.

- **Visual Paradigm** ως εργαλείο παραγωγής διαγραμμάτων UML. Μπορείτε να χρησιμοποιήσετε την trial έκδοση Visual Paradigm Enterprise ή το "Community Edition". Είναι υποχρεωτική η χρήση της έκδοσης Desktop - δεν γίνονται δεκτά διαγράμματα από την online έκδοση του εργαλείου.
- **GitHub** για διαχείριση έργων, με χρήση της σχετικής λειτουργικότητας που προσφέρεται εκεί.
- **Εργαλεία AI** της επιλογής σας για την υποστήριξη της ανάπτυξης προδιαγραφών, αρχιτεκτονικής, σχεδίου, κώδικα και ελέγχων.

Development stack που μπορούν να χρησιμοποιηθούν

- **Υλοποίηση κώδικα:** Python, javascript με nodejs/express. Επιτρέπονται και άλλες επιλογές (java, .net) με δήλωσή σας.
- **Διαχείριση δεδομένων:** ένα εκ των MySQL, MariaDB, PostgreSQL, Mongo, Elastic Search
- **Frontend:** η επιλογή frameworks και τεχνολογιών για frontend (π.χ. βιβλιοθήκες JS για διαγράμματα / γραφήματα στο web), είναι ελεύθερη.

Παραδοτέα

ΠΑΡΑΔΟΤΕΟ	3-4 άτομα	5-6 άτομα
Τεκμηρίωση		
Εγγραφο SRS - Software Requirements Specification	2 use cases	3 use cases
Διαγράμματα ER ή NoSQL class schemas	NAI	
Διαγράμματα UML Activity/State	αντίστοιχα με τα use cases	
Διαγράμματα UML Class/API	NAI	
Διαγράμματα UML Sequence	NAI	
Διαγράμματα UML Deployment	NAI	
Διαγράμματα UML Component	NAI	
Υλοποίηση		
Λειτουργίες backend	NAI	
Database dump (sql ή json)	NAI	
RESTful API	NAI	
API documentation - demo	NAI	
Command line interface (CLI)	NAI	NAI
Επικοινωνία με το χρήστη (frontend)	για 2 use cases	για 3 use cases
Εκτελέσιμη μορφή	NAI	
Testing		
API functional tests	NAI	NAI
CLI functional tests	NAI	NAI
CLI unit tests		NAI
AI tools usage logs		
Logs χρήσης εργαλείων AI σε προδιαγραφές, σχεδίαση, τεκμηρίωση, κωδικοποίηση, έλεγχο	NAI	

Μορφότυποι Παραδοτέων

Η απαιτούμενη μορφή των παραδοτέων φαίνεται στον παρακάτω πίνακα.

ΠΑΡΑΔΟΤΕΟ	Μορφότυπος	Όνομα αρχείου παραδοτέου	Σχόλιο
Τεκμηρίωση			
SRS - Software Req. Specification	docx με αυτόματη παραγωγή από Visual Paradigm	srs-softeng-XX.zip	Τα έγγραφα θα πρέπει να περιέχονται και στο ενιαίο αρχείο vpp
ERD ή NoSQL class schemas	Ενιαίο αρχείο Visual Paradigm (.vpp)	softeng22-XX.vpp	Προσοχή! Ένα (1) ενιαίο αρχείο vpp για όλα. Δεν γίνονται δεκτά διαγράμματα από σχεδιαστικά λογισμικά ή περισσότερα του ενός αρχεία vpp. Δεν γίνονται δεκτά διαγράμματα από το Visual Paradigm Online.
Διαγράμματα UML Activity / State			
Διαγράμματα UML Class/API			
Διαγράμματα UML Sequence			
Διαγράμματα UML Deployment			
Διαγράμματα UML Component			
Υλοποίηση			
Εισαγωγή, διαχείριση και πρόσβαση σε δεδομένα (backend)	Πηγαίος κώδικας	(σύμφωνα με το περιβάλλον ανάπτυξης)	Θα πρέπει να περιλάβετε ένα αρχείο README.md, παρέχοντας μια συνοπτική περιγραφή των συστατικών της εφαρμογής και των βημάτων που απαιτούνται για το στήσιμο του περιβάλλοντος ανάπτυξης (βιβλιοθήκες, κλπ).
Database dump (sql ή json)	SQL ή json		
RESTful API	Πηγαίος κώδικας στη γλώσσα υλοποίησης		
API documentation - demo	Postman scripts		
Command line interface (CLI)	Πηγαίος κώδικας στη γλώσσα υλοποίησης		
Επικοινωνία με το χρήστη (frontend)			
Εκτελέσιμη μορφή (build & deploy your code from source)			
Testing			
Back-end functional tests	test scripts	(σύμφωνα με το περιβάλλον ανάπτυξης)	
CLI unit tests			
CLI functional tests			
API functional tests			
AI tools usage logs			
Logs χρήσης εργαλείων AI σε προδιαγραφές, σχεδίαση, τεκμηρίωση, κωδικοποίηση, έλεγχο	json, txt αρχεία		Σύμφωνα με το εργαλείο AI usage logging στις διαφάνειες της σχετικής διάλεξης.

Προθεσμίες

- Παράδοση 1** Περιλαμβάνει την μέχρι τότε έκδοση του συνόλου της τεκμηρίωσης, όπως αναφέρεται στον αντίστοιχο πίνακα, μαζί με το μέχρι τότε τμήμα της υλοποίησης (εφόσον υπάρχει). Θα υποβληθεί ένα (και μόνο) αρχείο vrr. Επίσης, θα καταγραφεί ένα στιγμιότυπο της εικόνας του repo στο GitHub. Η υποβολή θα πρέπει να έχει γίνει στο σύστημα helios το αργότερο **έως τα μεσάνυχτα της 10ης Δεκεμβρίου 2023**.
- Παράδοση 2** Περιλαμβάνει το σύνολο της εργασίας. Η προθεσμία είναι **έως τα μεσάνυχτα της ημέρας πριν την εξέταση**.
- Παρουσίαση** το ΠΣΚ μετά την ολοκλήρωση της εξεταστικής, σύμφωνα με πρόγραμμα που θα ανακοινωθεί.

Βαθμολογία - βαρύτητες

Η βαθμολογία του μαθήματος προκύπτει κατά 50% από την εργασία και κατά 50% από το γραπτό. Επιπλέον, η εργασία παρουσιάζεται και προφορικά από τα μέλη της ομάδας, μετά τη λήξη της εξεταστικής. **Για τη λήψη προβιβάσιμου βαθμού στο μάθημα απαιτείται προβιβάσιμος βαθμός σε αμφότερες την εργασία και την τελική εξέταση.**

Η βαθμολογία της εργασίας προκύπτει με τις ακόλουθες βαρύτητες:

- Τεκμηρίωση 30%
- Υλοποίηση 30%
- Testing 15%
- Χρήση εργαλείων σχεδίασης και διαχείρισης έργων 15%
- Χρήση εργαλείων AI σε όλο τον κύκλο ζωής 10%

Σημείωση 1: Η βαθμολογία 10% της χρήσης εργαλείων AI είναι binary, δηλαδή είτε αποδίδεται ολόκληρη (10/10), είτε καθόλου (0/10). Το 10 αποδίδεται όταν ισχύουν **όλα** τα ακόλουθα: τουλάχιστον 2 καταγραφές στη φάση των προδιαγραφών, από 4 στις φάσεις της αρχιτεκτονικής και της σχεδίασης και από 10 στις φάσεις της συγγραφής κώδικα και ελέγχου. Συνολικά απαιτούνται **τουλάχιστον 30 καταγραφές ανά ομάδα**, με υποβολή του πλήρους διαλόγου με το εργαλείο AI και με συμμετοχή όλων των μελών της ομάδας.

Σημείωση 2: Κανονικοποίηση, εφόσον απαιτείται, γίνεται στη γραπτή εξέταση σε επίπεδο θέματος και όχι για το σύνολο του γραπτού. **Η βαθμολογία της εργασίας των μελών μιας ομάδας ενδέχεται να διαφοροποιηθεί ανάλογα με τα δεδομένα που θα καταγραφούν στο helios και στο GitHub, καθώς και στο εργαλείο καταγραφής της χρήσης εργαλείων AI.**

Σημείωση 3: Η τελική φόρμουλα βαθμολόγησης γραπτού και εργασίας είναι η ακόλουθη:

$$\text{FinalGrade} = \text{ceiling}(\text{sumproduct}(\text{PartialGrades}; \text{weights}); 1)$$

όπου $\text{ceiling}(7,1;1) = \text{ceiling}(7,9;1) = 8$

Συμπληρώσεις κλπ τροποποιήσεις της εκφώνησης θα ανακοινώνονται στο helios