

# Εργασία στο Μάθημα “Τεχνολογίες Υπηρεσιών Λογισμικού”

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο

Εαρινό εξάμηνο 2024-2025

Διδάσκοντες: Β. Βεσκούκης, Π. Τσανάκας, Μ.Κόνιαρης

## Περιγραφή

Η εργασία αφορά την ανάπτυξη του δικτυακού λογισμικού **clearSKY**, το οποίο θα παρέχεται ως υπηρεσία (SaaS) που απευθύνεται σε εκπαιδευτικά ιδρύματα και σπουδαστές αυτών. Ο κύκλος της ανάπτυξης δεν περιλαμβάνει καθορισμό και προδιαγραφή απαιτήσεων, αλλά επιχειρεί να προσομοιώσει συνθήκες όπου στην πορεία της ανάπτυξης μιας εφαρμογής, απαιτήσεις που είναι σε ικανοποιητικό βαθμό καθορισμένες, εξειδικεύονται από την κάθε ομάδα. Οι απαιτήσεις αυτές δίνονται από τους διδάσκοντες σε μορφή ενός περιληπτικού εγγράφου SRS. Η εργασία εστιάζει στην αρχιτεκτονική, τις τεχνολογίες και τα σύγχρονα εργαλεία ανάπτυξης εφαρμογών που παρέχονται ως υπηρεσία, συμπεριλαμβανομένων επιλεγμένων εργαλείων AI, καθώς και στην εφαρμογή ευέλικτων μεθοδολογιών ανάπτυξης.

Το **clearSKY** είναι μια δικτυακή υπηρεσία που απευθύνεται σε διδάσκοντες όλων των βαθμίδων οι οποίοι ανακοινώνουν τις βαθμολογίες των μαθημάτων τους προτού τις οριστικοποιήσουν και επιτρέπουν έναν κύκλο αιτημάτων αναθεώρησης από τους σπουδαστές. Η διεξαγωγή της συγκεκριμένης διαδικασίας από κάθε διδάσκοντα ξεχωριστά, απαιτεί δαπάνη χρόνου η οποία μπορεί να αποφευχθεί με τη χρήση της **clearSKY**, με λογικές προσαρμογές από πλευράς των διδασκόντων οι οποίες εστιάζονται κυρίως στο μορφότυπο της βαθμολογίας. Για τους σκοπούς της εργασίας θα αρκεστούμε στο μορφότυπο excel που χρησιμοποιεί το νέο φοιτητολόγιο του ΕΜΠ. Το **clearSKY** θα τρέχει σε (υποθετικές) υποδομές νέφους με επαρκή για το σκοπό του υπολογιστική ισχύ. Οι χρήστες (διδάσκοντες, ή τα ιδρύματά τους) θα πληρώνουν ανάλογα με το πλήθος των μαθημάτων για τα οποία το χρησιμοποιούν. Η υπηρεσία θα είναι δωρεάν για τους διδασκόμενους.

Οι ελάχιστες υποστηριζόμενες λειτουργίες θα είναι οι ακόλουθες:

- Διαχείριση χρηστών
- Login με Google (\*)
- Εγγραφή ιδρύματος (\*)
- Αγορά credits από ιδρύματα (\*)
- Ανάρτηση αρχικής βαθμολογίας από διδάσκοντες
- Ανάγνωση βαθμολογίας από σπουδαστές
- Ανάγνωση στατιστικών τελικών βαθμολογιών για όλα τα μαθήματα από σπουδαστές
- Αίτημα αναθεώρησης βαθμολογίας από σπουδαστές
- Απάντηση διδάσκοντος σε αίτημα αναθεώρησης
- Ανάρτηση τελικής βαθμολογίας από διδάσκοντες

(\*) Ομάδες 5 - 6 ατόμων

Οι λειτουργίες αυτές προσφέρονται μέσω περιπτώσεων χρήσης οι οποίες περιέχονται σε έγγραφο SRS που δίνεται στο Helios. Επιπροσθέτως, στο Helios είναι διαθέσιμα παραδείγματα αρχείων excel με βαθμολογίες, καθώς και το αρχείο vpp.

## Ομάδες, εργασία

Η εργασία θα γίνει από ομάδες των 3-6 ατόμων, οι οποίες καλούνται να υλοποιήσουν το σύνολο των λειτουργιών του **clearSKY** όπως αναφέρονται παραπάνω. Η υλοποίηση θα έχει ως αποτέλεσμα μια διαδικτυακή εφαρμογή η οποία θα πρέπει να τρέχει σε περιβάλλον Docker containers. Με την οριστικοποίηση των ομάδων θα διατεθούν από τους διδάσκοντες τα αντίστοιχα GitHub repositories.

Η αρχιτεκτονική που υποχρεωτικά θα υλοποιήσετε, θα είναι η αρχιτεκτονική **microservices**.

Τα εργαλεία που θα χρησιμοποιηθούν έχουν ως ακολούθως:

- **Σχεδίαση, αρχιτεκτονική:** Visual Paradigm, PlantUML με το [PlantUML-VP](#) plugin.
- **Υλοποίηση:** JS (node, express, κλπ), Python, React ή άλλο framework για frontend. DBMS επιλογής σας.
- **Διαχείριση** πηγαίου κώδικα: GitHub
- **Διαχείριση έργου:** GitHub
- **Deployment:** Docker containers
- **AI:** Συγκεκριμένα μοντέλα που θα ανακοινωθούν με το εργαλείο **ailog 1.5**.

## Παραδοτέα, βαρύτητες

Τα παραδοτέα της εργασίας έχουν ως εξής:

1. **Αρχείο vrr με την αρχιτεκτονική** της υλοποίησής σας σε μορφή διαγραμμάτων UML class, component, deployment και sequence, καθώς και σχεδίαση της υποδομής δεδομένων (ER για σχεσιακές ΒΔ, ή διάγραμμα κλάσεων για NoSQL). Το μοντέλο κλάσεων θα περιγράφει τους τύπους δεδομένων που ανταλλάσσονται (πχ json objects) και τα APIs που θα υλοποιηθούν. Σε καθένα από τα components που θα περιέχεται στα διαγράμματα, θα πρέπει να υπάρχει αναφορά στο αντίστοιχο αρχείο ή φάκελο πηγαίου κώδικα στο GitHub. Το μοντέλο UML θα πρέπει να είναι συνεπές, δηλαδή οι λειτουργίες των components και τα δεδομένα που ανταλλάσσονται να αναφέρονται στα διαγράμματα sequence, class (data structures) και class (APIs). Το deployment model πρέπει να περιέχει ports. (25%)
2. **Github repository με τον πηγαίο κώδικα**, συμπεριλαμβανομένων deployment scripts (20%)
3. **Η εφαρμογή σε λειτουργία** σε Docker containers. Προαιρετικά μπορείτε να ανεβάσετε την εφαρμογή σε υποδομή cloud της επιλογής σας. Θα γίνει προσπάθεια να σας διατεθεί εικονική μηχανή στην οποία να τρέξει η εφαρμογή σας (35%).
4. Ενημερωμένος δικτυακός τόπος στο **GitHub** με την πορεία διαχείρισης του έργου με 2-3 sprints (10%)
5. Τεκμηρίωση χρήσης εργαλείων AI (10%, binary grading με απαίτηση 60 εγγραφών ανά ομάδα)

## Χρονοδιάγραμμα

- **Τρίτη 14.4.2025** (Μ.Τρίτη): vrr με διαγράμματα κλάσεων (δομές δεδομένων), component (στοχευόμενη αρχιτεκτονική) και deployment (χρήση containers).
- **Τελική παράδοση** την ημέρα πριν την εξέταση του μαθήματος με υποβολή του τελικού vrr και του συνόλου του κώδικα σε λειτουργία.
- **Προφορική παρουσίαση** αμέσως μετά το τέλος της εξεταστικής.