

# Εργασία στο μάθημα «Τεχνολογία Λογισμικού»

Δρ. Μανώλης Πιτσικάλης  
manolispi@di.uoa.gr

Εαρινό εξάμηνο 2025

## 1 Περιγραφή

Τα συστήματα θαλάσσιας παρακολούθησης υποστηρίζουν την ασφαλή ναυσιπλοΐα διευκολύνοντας την ανίχνευση επικίνδυνων, ύποπτων και παράνομων δραστηριοτήτων πλοίων σε πραγματικό χρόνο. Τέτοια συστήματα χρησιμοποιούν συνήθως το Automatic Identification System (AIS) <sup>1</sup>, μια τεχνολογία εντοπισμού πλοίων στη θάλασσα μέσω ανταλλαγής δεδομένων. Το AIS χρησιμοποιεί έναν πομποδέκτη VHF, μια συσκευή εντοπισμού θέσης (π.χ. GPS) και άλλους ηλεκτρονικούς ναυτιλιακούς αισθητήρες, όπως γυροπυξίδα κ.α., παρέχοντας έτσι πολύτιμα δεδομένα σχετικά με το πλοίο και την τρέχουσα κατάστασή του. Η απόκτηση δεδομένων θέσης επιτυγχάνεται μέσω σταθμών βάσης AIS κατά μήκος των ακτών ή ακόμα και μέσω δορυφόρων όταν το πλοίο βρίσκεται εκτός εμβέλειας των επίγειων δικτύων.

Το ζητούμενο της εργασίας είναι να αναπτυχθεί μια διαδικτυακή πλατφόρμα που θα επιτρέπει την παρακολούθηση της ναυτιλιακής δραστηριότητας. Τα δεδομένα που θα πρέπει να χρησιμοποιήσετε για την εργασία σας είναι διαθέσιμα στο Zenodo<sup>2</sup>. Εκεί θα βρείτε μεταξύ άλλων ένα ιστορικό AIS dataset 6 μηνών (στον φάκελο P1) το οποίο και θα πρέπει να το αναπαράγετε (replay) σε πραγματική ταχύτητα για να προσομοιώσετε μια πραγματική ροή δεδομένων AIS. Η πλατφόρμα που θα υλοποιήσετε θα πρέπει να δείχνει ζωντανά τα πλοία στον χάρτη με βάση την θέση τους, την κατάσταση, τον τύπο και την πορεία τους (ένα παράδειγμα ενός τέτοιου χάρτη βρίσκεται εδώ<sup>3</sup>). Επιπλέον, αν κάποιος το επιλέξει, θα πρέπει να μπορεί να δει την προηγούμενη πορεία ενός πλοίου (π.χ. τις τελευταίες 12 ώρες). Οι εγγεγραμμένοι χρήστες θα πρέπει να μπορούν να αποθηκεύουν τον στόλο από πλοία τα οποία τούς ενδιαφέρουν, καθώς και να απεικονίζουν στον χάρτη μόνο πλοία με βάση κάποιο φίλτρο (π.χ., τα πλοία του στόλου τους, τα πλοία cargo κλπ). Τέλος, ο κάθε εγγεγραμμένος χρήστης θα πρέπει να μπορεί να ορίζει περιοχές ενδιαφέροντος στον χάρτη (zone of interest). Ο χρήστης θα πρέπει να μπορεί να ορίσει περιορισμούς, για τους οποίους θα εμφανίζεται κάποια ειδοποίηση κατά την παραβίασή τους. Ένας τέτοιος περιορισμός θα μπορούσε να είναι ότι τα πλοία μέσα στην περιοχή που έχει ορίσει ο χρήστης επιβάλλεται να έχουν ταχύτητα μικρότερη από 5 κόμβους. Για τη διευκόλυνσή σας, θα μπορεί να είναι ενεργή μόνο μία περιοχή ενδιαφέροντος. Οι χρήστες που έχουν προνόμια διαχειριστή θα πρέπει να μπορούν να τροποποιούν τα στατικά στοιχεία των πλοίων (τον τύπο τους). Για τα στατικά στοιχεία των πλοίων σας δίνεται το ακόλουθο αρχείο<sup>4</sup>.

## 2 Ομάδες εργασίας

Η εργασία θα γίνει από ομάδες των 5 ατόμων, οι οποίες θα υλοποιήσουν τον πλήρη κύκλο ανάπτυξης της πλατφόρμας (ανάλυση απαιτήσεων, σύνταξη προδιαγραφών, σχεδιασμός και αρχιτεκτονική, υλοποίηση και έλεγχοι αποδοχής, εγκατάσταση και λειτουργία). Η οριστικοποίηση των ομάδων εργασίας θα πρέπει να λάβει χώρα το αργότερο ως τις **23:59 την Δευτέρα 3/3**, μέσω της ενότητας «Ομάδες Χρηστών» του eclass.

## 3 Ελάχιστες κοινές προδιαγραφές

Οι κοινές προδιαγραφές για όλες τις ομάδες είναι:

1. Η πλατφόρμα που θα κατασκευάσετε θα αποτελείται τουλάχιστον από:
  - Ένα back-end υποσύστημα, το οποίο θα υποστηρίζει ένα ενιαίο RESTful API για τη εκτέλεση των σχετικών ενεργειών στο σύστημα.

<sup>1</sup><https://www.imo.org/en/OurWork/Safety/Pages/AIS.aspx>

<sup>2</sup><https://zenodo.org/records/1167595>

<sup>3</sup><https://www.marinetraffic.com>

<sup>4</sup><https://owncloud.skel.iit.demokritos.gr/index.php/s/k8eBG9Ze7B5TCjX>

- Ένα front-end υποσύστημα, το οποίο θα υλοποιηθεί ως διαδικτυακή εφαρμογή (Web app).
2. Για την προσομοίωση των ροών δεδομένων θα πρέπει να χρησιμοποιηθεί η πλατφόρμα Apache Kafka<sup>5</sup>.
  3. Τα front-end υποσυστήματα θα πρέπει να είναι τεχνολογίας SPA (Single Page Application). Στο πλαίσιο του μαθήματος θα γίνει φροντιστήριο / εργαστήριο για το React Framework.
  4. Θα πρέπει να υποστηρίζεται το πρωτόκολλο HTTPS για όλες τις χρηστικές και τις προγραμματιστικές διεπαφές της πλατφόρμας μέσω self-signed certificate.
  5. Η εργασία θα υλοποιηθεί με χρήση του συστήματος διαχείρισης εκδόσεων git. Ειδικότερα, θα πρέπει να γίνει χρήση κάποιας ηλεκτρονικής υπηρεσίας φιλοξενίας των git repositories της ομάδας (Github, Bitbucket, GitLab). Τα repositories θα πρέπει να είναι ιδιωτικά (private) και να έχουν πρόσβαση μόνο τα μέλη της ομάδας και ο διδάσκων.
  6. Η ομάδα θα πρέπει να κάνει χρήση ενός εργαλείου αυτοματισμού του «χτισίματος» του λογισμικού (build automation).
  7. Στο back-end υποσύστημα θα πρέπει να γίνει χρήση ενός εργαλείου για τον αυτοματισμό των ελέγχων του κώδικα (software test automation).
  8. Τα git repositories όλων των ομάδων θα έχουν ένα φάκελο docs που θα περιέχει τα έγγραφα τεκμηρίωσης της εργασίας.
  9. Η πλατφόρμα θα πρέπει να υποστηρίζει του εξής τρεις ρόλους: Ανώνυμος, Εγγεγραμμένος και Διαχειριστής.
  10. Η γλώσσα της πλατφόρμας θα πρέπει να είναι Ελληνικά ή Αγγλικά (ένα από τα δύο).

Κατά την διάρκεια του εξαμήνου, θα συζητήσουμε παραδείγματα τεχνολογιών που είναι σχετικές με την εργασία σας στα φροντιστήρια του μαθήματος. Αυτά τα παραδείγματα θα ανεβαίνουν σε σχετικό git repository.

## 4 Παραδοτέα και ημερομηνία εξέτασης

Η εργασία απαιτεί την κατάθεση τουλάχιστον τριών ενδιάμεσων παραδοτέων κατά τη διάρκεια του εξαμήνου. Η παράδοση—τόσο των ενδιάμεσων, όσο και του τελικού παραδοτέου—θα γίνεται μέσω git (λαμβάνοντας υπόψη την τελευταία ημερομηνία του commit που είναι εντός της εκάστοτε προθεσμίας). Στην παρούσα φάση, ανακοινώνεται το ζητούμενο του 1<sup>ου</sup> παραδοτέου.

### 4.1 1<sup>ο</sup> Παραδοτέο

Το 1<sup>ο</sup> παραδοτέο της εργασίας θα περιλαμβάνει:

- Την πρώτη έκδοση της τεκμηρίωσης του έργου (συνοπτικό έγγραφο SRS – Software Requirements Specification), το οποίο θα περιέχει μόνο την ανάλυση των απαιτήσεων (το «τι» θα υλοποιηθεί).
- Την πρώτη έκδοση της διεπαφής χρήσης των front—end εφαρμογών σε μορφή πρωτοτύπων (UI prototypes).

Στον φάκελο docs του git repository της κάθε ομάδας θα δημιουργηθούν τα αντίστοιχα αρχεία srs.md και ui.md, όπου θα ενσωματωθούν τα wireframes. Όποια ομάδα το επιθυμεί μπορεί να παραδώσει «ζωντανά» πρωτότυπα (HTML). Η ημερομηνία κατάθεσης του 1<sup>ου</sup> παραδοτέου είναι η **Δευτέρα 24/3, 23:59**.

## 5 Παράδοση εργασίας

Η εργασία θα πρέπει να έχει ολοκληρωθεί και παραδοθεί εντός της 1<sup>ης</sup> εβδομάδας μετά την ολοκλήρωση της εξεταστικής του Ιουνίου (σε ημέρα και ώρα που θα ανακοινωθεί). Η παράδοση της εργασίας γίνεται αυτόματα μέσω του git repository της ομάδας και θα ληφθεί υπόψη μόνο ο κώδικάς σας που δεν είναι μεταγενέστερος της ημερομηνίας εξέτασης, με βάση τη χρονοσήμανση των commits. Η εξέταση της εργασίας θα κρατήσει 1,5 – 2 ώρες και θα πρέπει συμμετέχουν όλα τα μέλη της ομάδας.

---

<sup>5</sup><https://kafka.apache.org/>

## 6 Χρήσιμες πληροφορίες/τεχνολογίες

Τεχνολογίες που θα είναι χρήσιμες στην εργασία σας (ενδεικτικά):

- Spring Boot
- REST API
- WebSockets
- Apache Kafka
- LeafLet.js
- React.js

### ★ ★ BONUS ★ ★

Το bonus της εργασίας είναι το ακόλουθο. Οι χρήστες θα μπορούν να ορίζουν περιοχές παρακολούθησης συγκρούσεων (collision monitoring zones) μέσα στις οποίες αν δύο πλοία βρίσκονται σε κίνδυνο σύγκρουσης θα εμφανίζεται κάποια ειδοποίηση. Μόνο μία τέτοια περιοχή θα μπορεί να είναι ενεργή κάθε φορά.