

# Τεχνολογία λογισμικού στην πράξη Εισαγωγή στο μάθημα

Διομήδης Σπινέλλης  
Τμήμα Διοικητικής Επιστήμης και Τεχνολογίας  
Οικονομικό Πανεπιστήμιο Αθηνών

dds@aueb.gr  
<http://www.dmst.aueb.gr/dds>  
@CoolSWEng

2021-03-25

## Τι είναι η τεχνολογία λογισμικού;

### Τεχνολογία λογισμικού

“The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.”

— ISO/IEC/IEEE Systems and Software Engineering Vocabulary (SEVOCAB)

### Στόχοι του μαθήματος

- Κάλυψη του σώματος γνώσης της τεχνολογίας λογισμικού
- Χρήση των κυρίων επαγγελματικών εργαλείων και πρακτικών
- Διαχείριση σχηματισμών
- Αυτοματοποίηση δόμησης
- Παρακολούθηση λαθών
- Υλοποίηση εργαλείων
- Συνεισφορά σε πραγματικό σύστημα
- Πρακτική σε
- εργασίες συντήρησης λογισμικού
- ομαδική εργασία
- συνεργασία σε παγκόσμιο επίπεδο
- χρήση δημοφιλών εργαλείων

### Δομή του μαθήματος

- Απαιτήσεις λογισμικού
- Σχεδιασμός λογισμικού
- Κατασκευή λογισμικού
- Έλεγχος λογισμικού
- Συντήρηση λογισμικού
- Διαχείριση σχηματισμών λογισμικού

- Διαχείριση τεχνολογίας λογισμικού
- Διεργασίες τεχνολογίας λογισμικού
- Μοντέλα και μέθοδοι τεχνολογίας λογισμικού
- Ποιότητα λογισμικού
- Επαγγελματικές πρακτικές τεχνολογίας λογισμικού
- Οικονομικά τεχνολογίας λογισμικού

## **Οργάνωση του μαθήματος και τρόπος διδασκαλίας**

- Ανεξάρτητη μελέτη
- Συμμετοχή
- Ερωτήσεις
- Εργασία
- Ομάδες εργασίας
- Εργαστήριο και εργαλεία
  - Τετάρτη 13:00 με 15:00 εργαστήριο 2

## **Οι σημειώσεις**

- Είναι οι παρουσιάσεις του μαθήματος
- Δε θα μοιράζονται
- Προσβάσιμες από την ιστοσελίδα <http://www.dmst.aueb.gr/dds/sweng-el>
- Καλύπτουν περιληπτικά τα θέματα του μαθήματος
- Ενημερώνονται συνέχεια
- Αν τις τυπώσετε θα εμφανιστούν σε μία σελίδα

## **Πρόγραμμα κάθε μαθήματος**

- Όρθια συνάντηση της κάθε ομάδας σε σχέση με την εργασία (3')
- Όρθια παρουσίαση από τους εκπροσώπους των ομάδων (5'–10')
- Συζήτηση πάνω στα θέματα που διαβάσατε πριν το μάθημα
- Σύντομη παρουσίαση των θεμάτων και της εργασίας του επόμενου μαθήματος

## **Βαθμολογία**

- Κατά τη διάρκεια του εξαμήνου
- Δεν θα υπάρξουν εξετάσεις
- Κατανομή
- Ομαδικές εργασίες (20%)
- Εργαστηριακές ασκήσεις (20%)
- Συμμετοχή στην τάξη (20%)
- Εργασία υλοποίησης (50%)

(Σύνολο 110%)

## Εργασία υλοποίησης

- Συνεισφορά σε έργο ΕΛ/ΛΑΚ
- Κατανομή βαθμών (110%)
- Κατανόηση και τεκμηρίωση του έργου (10%)
- Πλάτος των αλλαγών (15%)
- Ποιότητα υλοποίησης (15%)
- Ολοκλήρωση (10%)
- Έλεγχος (10%)
- Συνεργασία με την ομάδα ανάπτυξης (10%)
- Παρουσιάσεις (10%)
- Ποιότητα παραδοτέων (10%)
- Οργάνωση στο GitHub (wiki, issues, commits, branching) (10%)
- Κριτική στις καταχωρήσεις κώδικα — code reviews (10%)

## Επιλογή έργου ΕΛ/ΛΑΚ

- Σε γλώσσα που γνωρίζετε ή θέλετε να μάθετε
- Δομείται και εκτελείται σε πλατφόρμα που έχετε πρόσβαση / θα φτιάξετε
- Κάπως δημοφιλές (>5-10 αστέρια)
- Αναπτύσσεται από ομάδα (>2 μέλη)
- Δέχεται PRs
- Τεκμηριωμένο
- Ενεργό
- Εφαρμόζει διεργασίες (CI, επισκόπηση, θέματα)
- Αποφύγετε υλοποιήσεις αλγορίθμων, συλλογές, ελληνικά
- Διαλέξτε έργο που να χρησιμοποιείτε ή να σας ενδιαφέρει

## Συμμετοχή κατά τη διάρκεια του μαθήματος

- Καταχωρίζετε το χαρτί με τον κωδικό στον ιστότοπο [pixelbonus.com](http://pixelbonus.com)
- Μπορείτε να δείτε τα αποτελέσματα σε αυτή τη σελίδα

## Προετοιμασία για το μάθημα

- Κατεβάζετε το SWEBOOK v 3.0
- Τα κεφάλαια που θα διδαχθούν είναι διαθέσιμα και στα ελληνικά
- Αποκτάτε λογαριασμό στο GitHub
- Χωρίζετε σε ομάδες εργασίας 5-8 ατόμων
- Χωρίζετε σε ομάδες υλοποίησης 1-2 ατόμων
- Γράφεστε στο Moodle για το μάθημα
- Δηλώνετε τη συμμετοχή σας στο μάθημα μέσω αυτής της φόρμας
- Μην ξεχάσετε να δηλώσετε τα πρόσθετα καθήκοντα που σας ενδιαφέρουν

## **Ενδεικτικός χρονοπρογραμματισμός**

- Ορόσημο 1: Συμμετοχή και ομάδες (14/3)
- Ορόσημο 2: Επιλογή έργου (26/3)
- Ορόσημο 3: Παρουσίαση έργου (29/3)
- Ορόσημο 4 (50%): Παρουσίαση σχεδιασμού της συνεισφοράς (26/4)
- Ορόσημο 5 (τελικό): Παρουσίαση της υλοποίησης

## **Ο κώδικας στην ανάπτυξη λογισμικού**

- Ο κώδικας ως όχημα επιστημονικής επικοινωνίας
- Τι κερδίζουμε διαβάζοντας ποιοτικό κώδικα
- Πώς καταλαβαίνουμε κακής ποιότητας κώδικα
- Απαιτήσεις που οδηγούν σε περίεργο κώδικα
- Πώς ερχόμαστε σε επαφή με έτοιμο κώδικα
- Λόγοι χρήσης κώδικα
- Το τοπίο του ΕΛ/ΛΑΚ

## **Τι κερδίζουμε διαβάζοντας ποιοτικό κώδικα**

- Αρχιτεκτονικές
- Δομές δεδομένων
- Μεθόδους προγραμματισμού
- Αλγορίθμους
- Μορφοποίηση
- Τεκμηρίωση
- Διεπαφές (API)
- Επαναχρησιμοποίηση
- Γλώσσες προγραμματισμού

## **Πώς καταλαβαίνουμε κακής ποιότητας κώδικα**

- Ασυνεπής μορφοποίηση
- Αχρείαστη περιπλοκότητα
- Προφανή λογικά λάθη και ελλείψεις
- Προβλήματα μεταφερσιμότητας
- Απουσία συντήρησης
- Προβλήματα ασφάλειας και αποδοτικότητας
- Τεχνικό χρέος

## **Απαιτήσεις που οδηγούν σε περίεργο κώδικα**

- Μεταφερσιμότητα
- Αποδοτικότητα
- Έλλειψη πόρων (π.χ. μνήμης)
- Παραδείγματα έναντι κώδικα παραγωγής

- Προστασία διανοητικής ιδιοκτησίας

### **Πώς ερχόμαστε σε επαφή με έτοιμο κώδικα**

- Διατηρούμε την ψυχραιμία μας (Don't Panic!)
- Δομούμε
- Τρέχουμε τις δοκιμές
- Διαβάζουμε
- Βελτιώνουμε
- Δομούμε και ελέγχουμε ξανά
- Συνεισφέρουμε

### **Πώς χρησιμοποιούμε κώδικα για ...**

- ... παράδειγμα
- ... συντήρηση
- ... ανάπτυξη
- ... μεταφορά
- ... αναμόρφωση
- ... επαναχρησιμοποίηση
- ... επιθεωρήσεις

### **Το τοπίο του ΕΛΛΑΚ**

- Ιστορία
- Άδειες: GPL, LGPL, Affero, BSD, MIT, Apache, ...
- Αποθετήρια: GitHub, GNU Savannah, Apache, Eclipse, GitLab...
- Διεργασία ανάπτυξης

### **Παραδείγματα ΕΛΛΑΚ**

- ΛΣ: Linux, FreeBSD, NetBSD, OpenBSD
- Βάσεις δεδομένων: PostgreSQL, MariaDB, SQLite
- Διορθωτές: Vim, GNU Emacs, Atom
- Μεταγλωττιστές: GCC, LLVM
- Ανάπτυξη εφαρμογών: OpenJDK, Mono, OpenCL
- Γραφικά: Inkscape, GIMP, Blender, Ghostscript
- Ολοκληρωμένα εργαλεία ανάπτυξης: Eclipse, NetBeans, KDevelop
- Επεξεργασία κειμένου: LaTeX, Pandoc, LibreOffice
- Ιστοκεντρικές εφαρμογές: React, Angular, Django, Ruby on Rails, PHP

### **Προετοιμασία για το επόμενο μάθημα**

- Διαβάστε το κεφάλαιο 1 του SWEBOOK v 3.0
- Αποκτήστε (αν δεν έχετε) λογαριασμό στο GitHub.

- Διαβάστε τον οδηγό στην ιστοσελίδα How to Contribute to Open Source
- Επισκεφτείτε το αποθετήριο How to choose (and contribute to) your first open source project
- Άσκηση:
- Εξαγωγή απαιτήσεων για ένα σημαντικό-δημοφιλές έργο ανοιχτού λογισμικού.
- Συγκεκριμένοι στόχοι:
  - Αναγνώριση ενδιαφερόμενων μερών
  - Αναγνώριση λειτουργικών και μη λειτουργικών απαιτήσεων
  - Μελέτη συνεπειών αλλαγής απαιτήσεων
- Βίντεο: <http://www.youtube.com/watch?v=VDjoiRkt06M>
- Ετοιμάστε την κάρτα συμμετοχής με βάση αυτό το πρότυπο.
- Να φέρετε μαζί σας μια μονή διαφανή θήκη CD/DVD κανονικού (όχι λεπτού) πάχους

### **Άδεια διανομής**

Εκτός αν αναφέρεται κάτι διαφορετικό, όλο το πρωτότυπο υλικό της σελίδας αυτής του οποίου δημιουργός είναι ο Διομήδης Σπινέλλης παρέχεται σύμφωνα με τους όρους της άδειας Creative Commons Αναφορά-Παρόμοια διανομή 3.0 Ελλάδα.

