



Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Η/Υ

Εθνικό Μετσόβιο Πολυτεχνείο

---

# Τεχνολογία Λογισμικού

7ο / 9ο Εξάμηνο 2018-19

---

Ν.Παπασπύρου, Αν.Καθ. ΣΗΜΜΥ, [nickie@softlab.ntua.gr](mailto:nickie@softlab.ntua.gr)

Β.Βεσκούκης, Αν.Καθ. ΣΑΤΜ, [v.vescoukis@cs.ntua.gr](mailto:v.vescoukis@cs.ntua.gr)

Κ.Σαΐδης, ΠΔ 407, [saiko@softlab.ntua.gr](mailto:saiko@softlab.ntua.gr)

# Διαχείριση έργων λογισμικού (1)

# Περιεχόμενα

- Διαχείριση έργων (γενικά)
- Διαχείριση έργων λογισμικού (εισαγωγή)

# Διαχείριση έργου (γενικά)

- Ορισμοί
- Βασικές έννοιες
- Βασικά εργαλεία

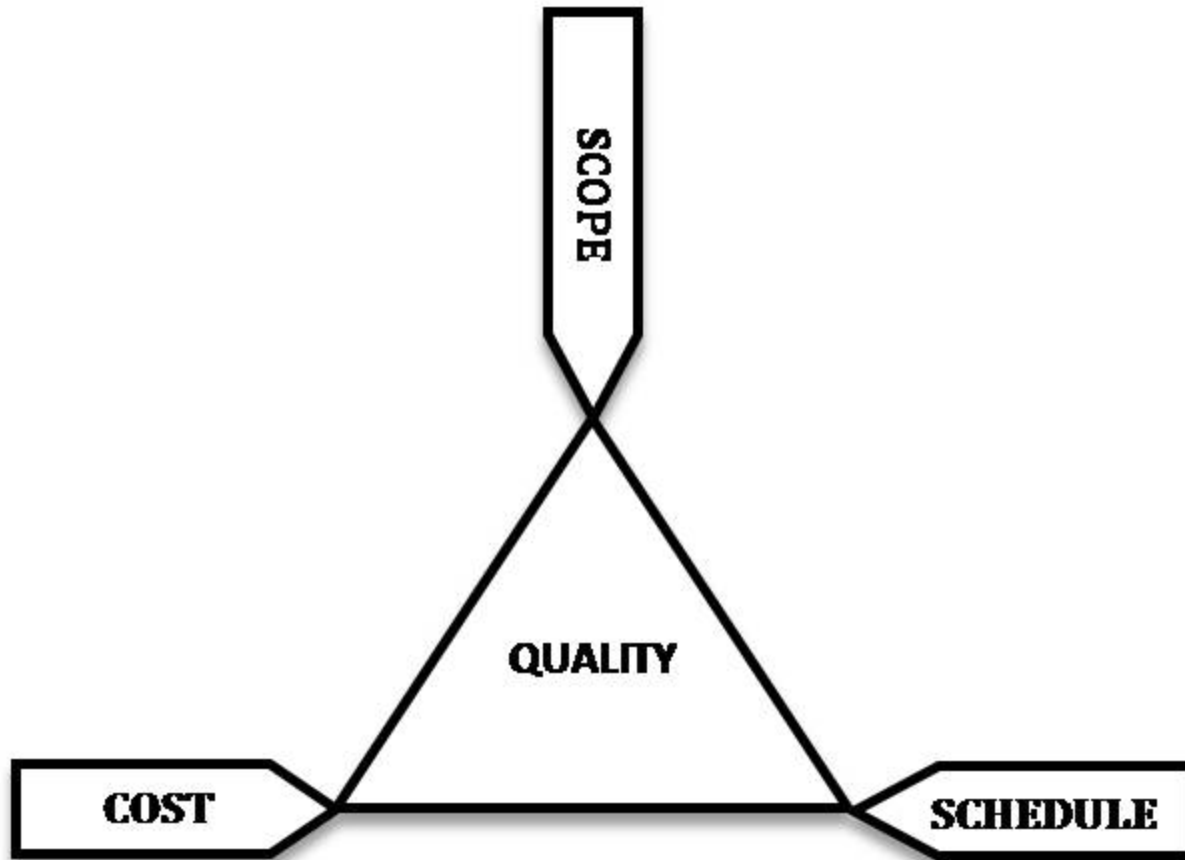
# Έργο (project)

Ένα σύνολο ξεχωριστών, σύνθετων και συνδεδεμένων ενεργειών που έχουν ένα στόχο και πρέπει να ολοκληρωθούν εντός συγκεκριμένου χρονικού διαστήματος και προϋπολογισμού με βάση συγκεκριμένες προδιαγραφές.

# Περιορισμοί

Σε τέσσερις άξονες:

- Κόστος (cost)
- Χρόνος (schedule)
- Έκταση ζητούμενων (scope)
- Ποιότητα αποτελέσματος (quality)



By I, John Manuel Kennedy T., CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=4282986>

## Στην πράξη

- Συνήθως μπορούμε να ελέγξουμε και να διαχειριστούμε δύο από τους άξονες αυτούς.
- Πρέπει ο πελάτης να αποφασίσει σε ποιους άξονες υπάρχουν περιθώρια ευελιξίας.



## Για παράδειγμα

- Μικρότερος χρόνος + Υψηλότερη ποιότητα = Ακριβότερο
- Μικρότερος χρόνος + Φθηνότερο = Χαμηλότερη ποιότητα
- Υψηλότερη ποιότητα + Φθηνότερο = Μεγαλύτερος χρόνος

# Διαφορετικές προσεγγίσεις διαχείρισης

- Παραδοσιακή (waterfall)
- Αυξητική (incremental)
- Επαναληπτική (iterative)
- Προσαρμοστική (adaptive)
- κ.ά.

Αντίστοιχες μεθοδολογίες με τα έργα λογισμικού

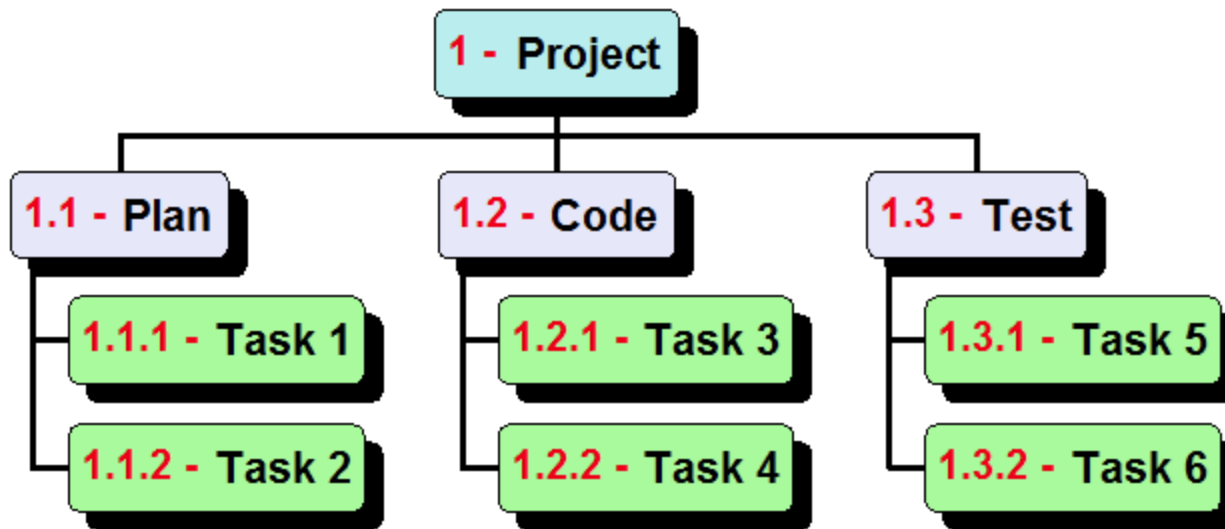
# Σχέδιο έργου (Project Plan)

- Διαχωρισμός του έργου σε υποέργα με ενδιάμεσα παραδοτέα (interim deliverables)
- Εκτίμηση προσπάθειας, κόστους και χρόνου
- Χρονοπρογραμματισμός και χρονικά ορόσημα (project milestones)

# Ιεραρχική κατανομή εργασιών (Work Breakdown Structure)

- Ιεραρχική περιγραφή όλης της "δουλειάς" που πρέπει να γίνει στο πλαίσιο του έργου για να ικανοποιηθούν οι απαιτήσεις του πελάτη.
  - Πακέτα εργασίας - υποέργα
  - Επιμέρους δράσεις και ενέργειες ανά πακέτο
  - Αλληλοεξαρτήσεις μεταξύ τους
- Δεν πρέπει να είναι ούτε πολύ λεπτομερής, ούτε πολύ περιληπτική (κάπου στη μέση).

# WBS



[criticaltools.com](http://criticaltools.com)

# Χρονοδιάγραμμα (Timeline)

- Χρονο-προγραμματισμός των εργασιών
- Με βάση το WBS
- Είτε με απόλυτες ημερομηνίες
  - 1ο παραδοτέο στις HH/MM/EEEE
- Είτε με σχετικές ημερομηνίες
  - 1ο παραδοτέο στο δεύτερο μήνα (M2) από την έναρξη του έργου

# Χρήσιμες έννοιες

## Κρίσιμο μονοπάτι (critical path)

- Η ακολουθία των ενεργειών που αν αποτύχουν θα αποτύχει το έργο.

## Χαλαρό περιθώριο (slack, float)

- Το χρονικό διάστημα που μπορεί να καθυστερήσει μια ενέργεια χωρίς να καθυστερήσει κάποια επόμενη της ή το το έργο συνολικά.





# Εξαρτήσεις μεταξύ των επιμέρους εργασιών

- FS: Όταν ολοκληρωθεί η εργασία A, μπορεί να ξεκινήσει η B
- FF: Όταν ολοκληρωθεί η A, μπορεί να ολοκληρωθεί η B
- SS: Όταν ξεκινήσει η A, μπορεί να ξεκινήσει η B
- SF: Όταν ξεκινήσει η A, μπορεί να ολοκληρωθεί η B

# Χρόνος και Προσπάθεια

Κάθε εργασία στο πλαίσιο του έργου:

- έχει μια χρονική διάρκεια (ο αριθμός των ημερολογιακών ημερών που διαρκεί).
- απαιτεί μια προσπάθεια (ο αριθμός των εργατο-ωρών).

## Ανθρωπο-προσπάθεια

- Ανθρωπο-ώρα (AΩ)
- Ανθρωπο-ημέρα (AH) = 8 AΩ
- Ανθρωπο-μήνας (AM) = 20/21 AH
- Ανθρωπο-έτος (AE) = 11 AM

# Διαχείριση έργου λογισμικού

- Μοντέλα και μεθοδολογίες ανάπτυξης λογισμικού
- Διοίκηση ομάδας ανάπτυξης λογισμικού
- Τεχνικές και εργαλεία ελέγχου ανάπτυξης λογισμικού
- Διαχείριση συστατικών του λογισμικού

Μεγάλο μέρος του μαθήματος θα αφιερωθεί σε αυτό το θέμα!

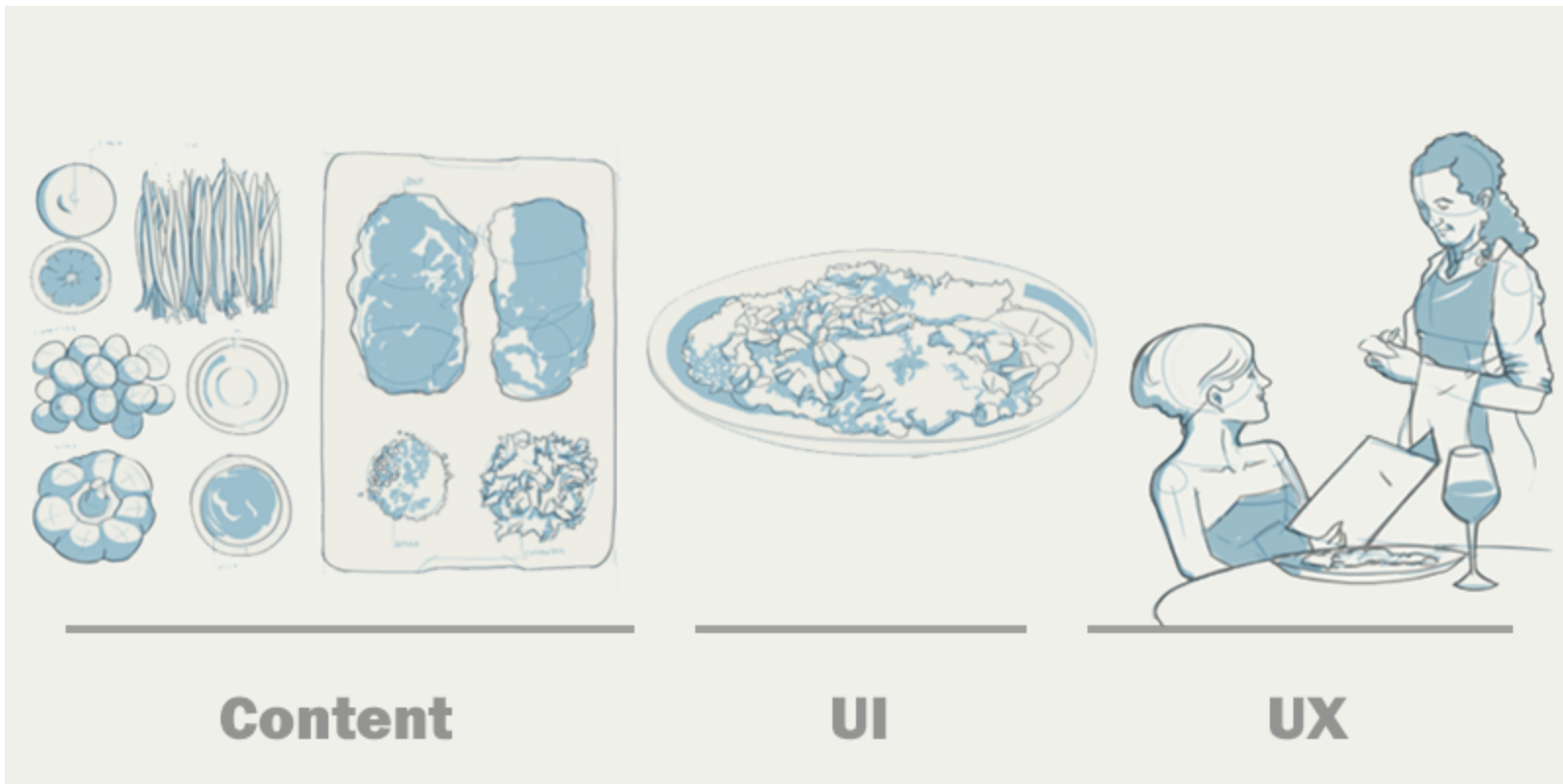
# Ρόλοι σε μια ομάδα ανάπτυξης λογισμικού

- Ανάλογα με την εφαρμογή/λογισμικό
- Χρησιμοποιούμε μια εφαρμογή διαδικτύου ως παράδειγμα

# Ενδεικτικοί ρόλοι

- Project/Product manager
- Architect
- Team lead
- Senior developer
- Junior developer
- Quality assurance engineer
- User interface designer
- User experience engineer

# UX vs UI



# Ενδεικτικοί ρόλοι

- Back-end engineer
- Front-end engineer
- Full-stack engineer
- Database developer
- Business analyst
- Data scientist
- Customer support
- DevOps engineer

# Συνήθως

- Σε μικρές ομάδες, ένα πρόσωπο έχει πολλούς ρόλους
- Σε μεγάλες ομάδες, ένα πρόσωπο έχει ένα ρόλο



# Διοίκηση της ομάδας

- Project/Product manager:
  - Οριοθέτηση στόχων ομάδας και χαρακτηριστικών λογισμικού
  - Διαχείριση χρόνου και προσπάθειας των μελών σε μακροσκοπικό επίπεδο

# Διοίκηση της ομάδας

- Architect
  - Λήψη των σημαντικών σχεδιαστικών αποφάσεων (major design decisions)
  - Θέσπιση των διεπαφών (interfaces), των συστατικών (components) και των εξαρτήσεων (dependencies)
  - Επιλογή εργαλείων (tooling)

# Διοίκηση της ομάδας

- Team Lead
  - Ανάθεση των επιμέρους εργασιών στα μέλη της ομάδας
  - Διαχείριση χρόνου και προσπάθειας των μελών σε καθημερινό επίπεδο