



Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής & Τηλεπικοινωνιών

Τεχνολογία Λογισμικού

8ο Εξάμηνο 2019-20

Διαχείριση έργων λογισμικού (1 / 2)

Δρ. Κώστας Σαΐδης (saiko@di.uoa.gr)

Διαχείριση έργων λογισμικού (1 / 2)

Περιεχόμενα

- Διαχείριση έργων (γενικά)
- Εισαγωγή στη διαχείριση έργων λογισμικού

A. Διαχείριση έργου (γενικά)

- Ορισμοί
- Βασικές έννοιες
- Βασικά εργαλεία

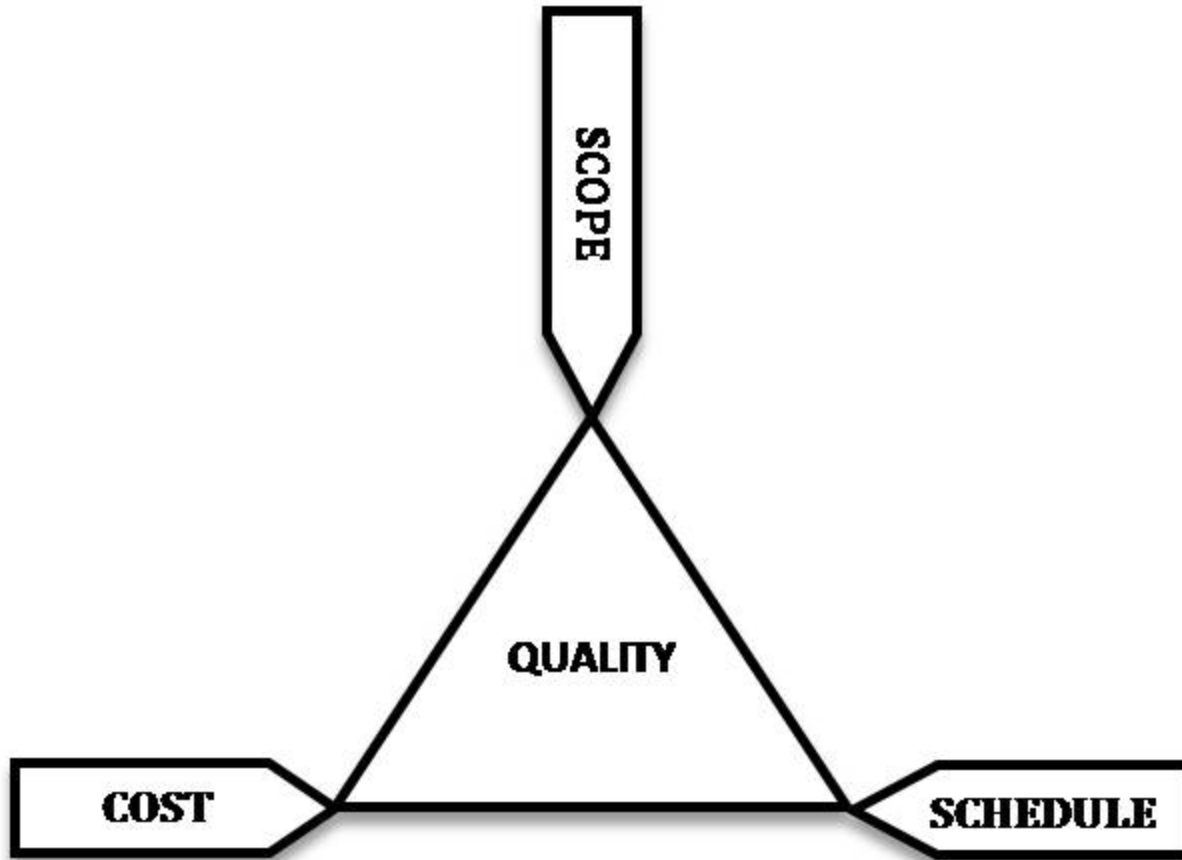
Έργο (project)

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)



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

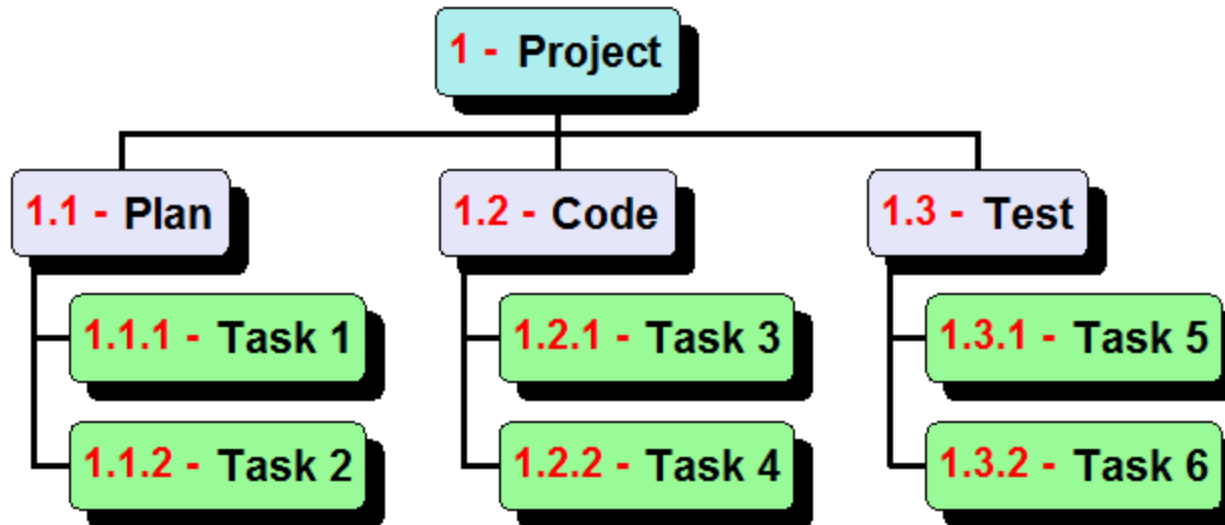
Σχέδιο έργου (Project Plan)

- 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



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

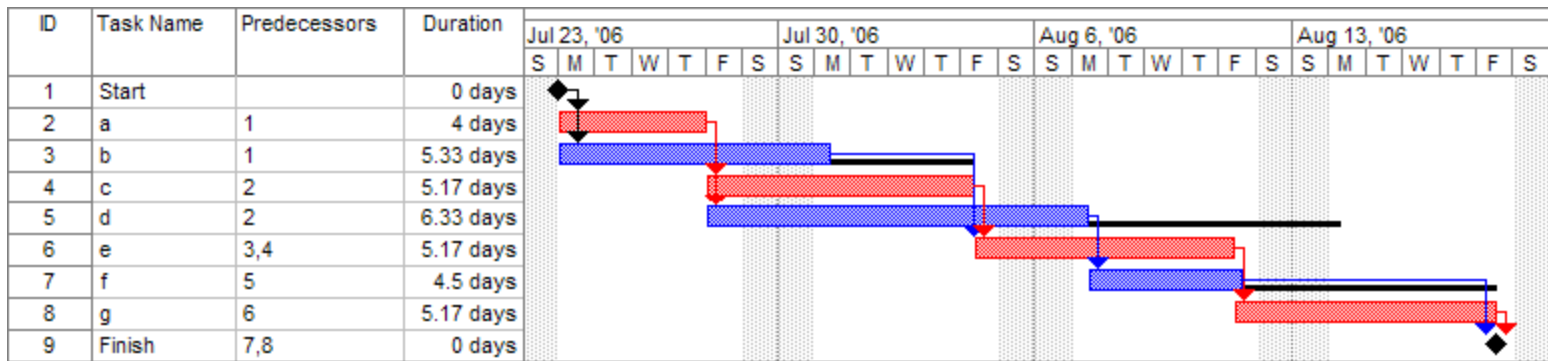
Κρίσιμο μονοπάτι (critical path)

- 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.

Διάγραμμα Gantt



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

B. 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!

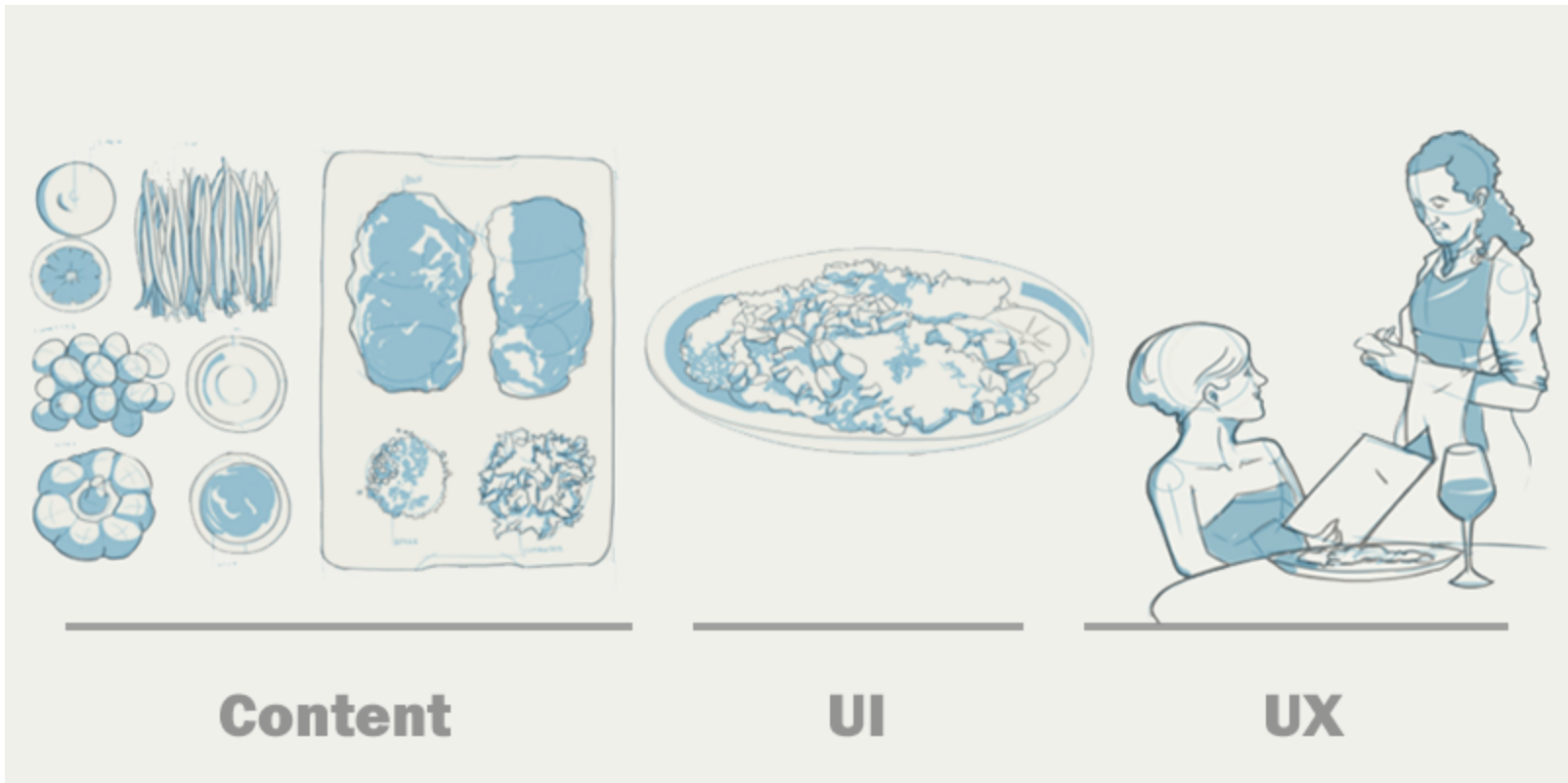
Ρόλοι σε μια ομάδα ανάπτυξης λογισμικού

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

Ενδεικτικοί ρόλοι

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

UX vs UI



Ενδεικτικοί ρόλοι (συνέχεια)

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

Διοίκηση της ομάδας

Επικεφαλής μηχανικός

- Ανάλογα με το έργο, μπορεί να συνδυάζει και τις τρεις ιδιότητες
- Στο μάθημα δίνουμε έμφαση σε αυτόν το "ρόλο"

Το ζητούμενο

Πώς ο επικεφαλής μηχανικός

- Ο οποίος συμμετέχει σε μια ομάδα ανάπτυξης
- Διαχειρίζεται τεχνικά τη διαδικασία ανάπτυξης
- Διασφαλίζοντας την ποιότητα του τελικού αποτελέσματος

Ανεξάρτητα από

- Τους ρόλους των ανθρώπων στην ομάδα
- Τη μεθοδολογία ανάπτυξης
- Την αρχιτεκτονική του υπό ανάπτυξη λογισμικού