

# Πανεπιστήμιο Κύπρου

## Τμήμα Πληροφορικής

### ΘΕΜΑΤΑ ΤΕΧΝΟΛΟΓΙΑΣ ΛΟΓΙΣΜΙΚΟΥ: ΕΠΛ 603

Ακαδημαϊκό Έτος: 2009/2010 – Χειμερινό Εξάμηνο

#### Γενικές Πληροφορίες

Διδάσκων: Χρήστος Γκατζούλης, Επισκέπτης Λέκτορας

Γραφείο: ΘΕΕ 01 – B118

Τηλ: 2289 2746

Email: [xristos@ucy.ac.cy](mailto:xristos@ucy.ac.cy)

Ιστοσελίδα μαθήματος: <http://www.cs.ucy.ac.cy/courses/EPL603/>

#### A. Στόχοι

Στόχος του μαθήματος είναι να διδάξει θέματα τεχνολογίας λογισμικού με έμφαση σε δύο άξονες:

1. Αρχές σχεδιασμού και υλοποίησης κλιμακώσιμων συστημάτων λογισμικού, με βάση τις μοντέρνες μεθόδους ανάπτυξης τέτοιου είδους συστημάτων, όπως οι αρχιτεκτονικές λογισμικού και οι ενδιάμεσες πλατφόρμες.
2. Ειδικά θέματα ανάπτυξης συστημάτων λογισμικού όπως είναι η μηχανική απαιτήσεων, τα συστήματα πραγματικού χρόνου, design patterns, τα legacy συστήματα, ο έλεγχος του λογισμικού κλπ.

#### B. Προαπαιτούμενο Υπόβαθρο

Οι φοιτητές θεωρείται ότι κατέχουν βασικές γνώσεις σε αντικειμενοστρεφή προγραμματισμό και τεχνολογία λογισμικού. Αναφορικά με το προπτυχιακό πρόγραμμα του Τμήματος Πληροφορικής, οι φοιτητές θεωρείται ότι κατέχουν την ύλη των μαθημάτων ΕΠΛ 233 (Αντικειμενοστρεφής Προγραμματισμός), και ΕΠΛ 244 (Τεχνολογία Λογισμικού).

#### Γ. Τρόπος Διδασκαλίας και Αξιολόγησης Φοιτητών

Το μάθημα διδάσκεται σε 2 διαλέξεις και 1 εργαστήριο ανά βδομάδα. Η ύλη που θα διδαχθεί έχει χωρισθεί λογικά σε έναν αριθμό ενοτήτων και ακολουθεί σε μεγάλο βαθμό αυτή των βιβλίων [1, 2] που δίνονται στη βιβλιογραφία, τα οποία και είναι τα βασικά βιβλία του μαθήματος. Οι διαφάνειες από τις διαλέξεις θα είναι διαθέσιμες στην ιστοσελίδα με σκοπό να βοηθήσουν τον φοιτητή στη μελέτη της σχετικής ύλης.

Η αξιολόγηση θα γίνει με ημιεξέταση, τελική εξέταση και εργασίες.

Η αναλογία ως προς τον τελικό βαθμό έχει ως εξής:

**Εργασίες 25%**

**Ημιεξέταση 25%**

**Τελική Εξέταση 50%**

## Δ. Βιβλιογραφία

### Βασική βιβλιογραφία

[1] Clemens Szyperski, 'Component Software: Beyond Object-Oriented Programming', 2nd edition, Addison Wesley, 2002, ISBN 0-201-74572-0.

[2] Ian Sommerville, 'Software Engineering', 6th edition, Addison Wesley, 2001, ISBN 0-201-39815-X. (or 7th edition, 2004).

### Συμπληρωματική/βοηθητική βιβλιογραφία

[3] Roger S. Pressman, 'Software Engineering: A Practitioner's Approach', Mc Graw Hill, Fifth Edition, 2000, ISBN 0-07-709677-0.

[4] A. Shalloway, J.R. Trott, "Design Patterns Explained", Addison Wesley, 2<sup>nd</sup> Edition, 2005, ISBN 0-321-24714-0.

[5] Craig Larman, 'Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design', Prentice Hall, 1998, ISBN 0-13-748880-7.

[6] Len Bass, Paul Clements, Rick Kazman, 'Software Architecture in Practice', Addison Wesley, 1998, ISBN 0-201-19930-0.

## Ε. Ενδεικτική Ύλη

Τα θέματα περιλαμβάνουν:

- *Introduction to Software Engineering.*
- *Fundamentals.* Components-based systems. Components vs. Objects. The interaction between components. Patterns and software architectures. Basic principles of component software.
- *Middleware Platforms.* Definitions and use.
- *Design Patterns.* Example patterns (Singleton, Factory, Facade, Observer, Adapter) and case studies.
- *Requirements Engineering Process.* Analysis & elicitation, validation, management.
- *Legacy systems.* Structure and Design.
- *Real-time Software.* Design, monitoring and control systems.
- *Software Evolution.* Maintenance, evolution dynamics.
- *Software Testing.* Component testing, test design and automation.