

Liverpool John Moores University

Title: PRINCIPLES OF SOFTWARE ENGINEERING
Status: Definitive
Code: **5000ESE** (120623)
Version Start Date: 01-08-2016
Owning School/Faculty: Computer Science
Teaching School/Faculty: Computer Science

Team	Leader
Martin Randles	Y

Academic Level: FHEQ5 **Credit Value:** 20 **Total Delivered Hours:** 74
Total Learning Hours: 200 **Private Study:** 126

Delivery Options

Course typically offered: Standard Year Long

Component	Contact Hours
Lecture	24
Practical	24
Tutorial	24

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Portfolio	AS1	Coursework - Students will develop a portfolio of computer modelling exercises reflecting the different techniques covered during the module	50	
Exam	AS2	Examination	50	2

Aims

To introduce the student to project management techniques used in software engineering projects

To introduce the student to the models, tools and techniques of the development

process for software systems, including formal principles of modelling, enabling students to apply these techniques in the analysis and design of systems.

Learning Outcomes

After completing the module the student should be able to:

- 1 Apply the principles of best practice software engineering, within project planning, scheduling and management, to any practical example of software system development.
- 2 Evaluate the various models used for and within software engineering projects.
- 3 Use concepts from discrete mathematics to model aspects of computing systems.
- 4 Apply the established notations of sets, functions, relations, trees and graphs to computing examples.

Learning Outcomes of Assessments

The assessment item list is assessed via the learning outcomes listed:

Computer modelling	3	4
Examination	1	2

Outline Syllabus

Characteristics of large-scale software systems projects. Team membership & activities.

Software process models. CASE tool support.

Software project planning - scheduling tools, milestones, option and risk analysis.

Software project management techniques – cost modelling, milestone trend monitoring, activity networks, critical path analysis and Gantt charts.

Project monitoring and reviews, quality assurance.

Validation and verification techniques. Formal specification models.

Role of modelling in computing. Engineering principles.

Propositions and predicates, logical connectives, truth tables.

Concepts of set theory, set membership, union, intersection and difference.

Cartesian products; coordinate systems; vectors and matrices.

Functions and their properties; composition. Recursive definitions.

Functions of discrete and continuous variables.

Relations, inverse relations, composition.

Trees and graphs.

Learning Activities

Lectures, tutorial activities and computer lab practical sessions.

Notes

This module aims to introduce students to the fundamental principles of best practice in software engineering and system modelling, focusing on key topics that are central to all development processes and aspects concerned with the engineering of reliable distributed systems. The major issues of systems thinking and development are addressed utilizing software process models and relevant techniques of project management: The major activities and modelling techniques of modern software development are presented. The underlying formal concepts, based on discrete mathematical structures, are described and used in the modelling and programming of system functions.