

## Liverpool John Moores University

Title: SOFTWARE SYSTEMS: DESIGN & EVOLUTION  
Status: Definitive  
Code: **6029COMP** (103019)  
Version Start Date: 01-08-2011

Owning School/Faculty: Computing and Mathematical Sciences  
Teaching School/Faculty: Computing and Mathematical Sciences

Team	Leader
Andrew Laws	Y

**Academic Level:** FHEQ6  
**Credit Value:** 12.00  
**Total Delivered Hours:** 50.00  
**Total Learning Hours:** 120  
**Private Study:** 70

### Delivery Options

Course typically offered: Semester 1

Component	Contact Hours
Lecture	24.000
Online	14.000
Tutorial	12.000

**Grading Basis:** 40 %

### Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Report	AS1	The production of an academic paper exploring a student-selected area of interest within the scope of the module.	100.0	

### Aims

*To deepen an understanding of the analysis, design and maintenance of software systems.*

*To explore the appropriateness, efficiency and effectiveness of various software systems design/development approaches in software system production e.g. Capability Maturity Model.*

*To appreciate the difficulties and effects of continued maintenance on software systems.*

*To explore the recent emergence of alternative approaches to software system production e.g. self-managing, autonomic systems.*

## **Learning Outcomes**

After completing the module the student should be able to:

- 1 Evaluate, compare and contrast systems design methods and their implementation.
- 2 Understand the role of software maintenance and its impact on software development and evolution.
- 3 Appreciate the approaches, challenges and rewards of developing self-managing software systems.

## **Learning Outcomes of Assessments**

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

Academic paper                      1    2    3

## **Outline Syllabus**

*Discussions on:*

*Problems facing the software industry.*

*Software systems design methods: data flow-orientated, object-orientated and data structure-orientated approaches*

*Software process improvement, Capability Maturity Model*

*Software maintenance, evolution and the Laws of Software Evolution*

*Autonomic Computing, challenges and approaches.*

## **Learning Activities**

Students will participate in interactive lectures/tutorials. They will be expected to read about the subject paying special attention to the indicative references.

## **References**

<b>Course Material</b>	Book
<b>Author</b>	Brooks, F.P.
<b>Publishing Year</b>	1995
<b>Title</b>	The Mythical Man-Month: Essays on Software Engineering
<b>Subtitle</b>	
<b>Edition</b>	2nd/Anniversary Edition

<b>Publisher</b>	Addison-Wesley
<b>ISBN</b>	

<b>Course Material</b>	Book
<b>Author</b>	Somerville, I.
<b>Publishing Year</b>	2007
<b>Title</b>	Software Engineering
<b>Subtitle</b>	
<b>Edition</b>	8th Edition
<b>Publisher</b>	Pearson Education Ltd
<b>ISBN</b>	139780321313799

<b>Course Material</b>	Book
<b>Author</b>	Lehman, Meir. M.
<b>Publishing Year</b>	0
<b>Title</b>	<a href="http://www.cs.mdx.ac.uk/staffpages/mml/">http://www.cs.mdx.ac.uk/staffpages/mml/</a>
<b>Subtitle</b>	- assorted papers
<b>Edition</b>	
<b>Publisher</b>	
<b>ISBN</b>	

<b>Course Material</b>	Book
<b>Author</b>	IBM Autonomic Computing Research
<b>Publishing Year</b>	0
<b>Title</b>	<a href="http://www.research.ibm.com/autonomic/">http://www.research.ibm.com/autonomic/</a>
<b>Subtitle</b>	
<b>Edition</b>	
<b>Publisher</b>	
<b>ISBN</b>	

---

## Notes

This module investigates the design and maintenance of software systems by comparing and contrasting the traditional engineering viewpoint used in developing such systems with the more recent emergence of a more organismic view of software development. Such a viewpoint leads inexorably to the development of self-managing systems.