Liverpool John Moores University

Title:	SOFTWARE DEVELOPMENT
Status:	Definitive
Code:	**7005CCTV** (118648)
Version Start Date:	01-08-2011
Owning School/Faculty:	Computing and Mathematical Sciences
Teaching School/Faculty:	Computing and Mathematical Sciences

Team emplid	Leader
Denis Reilly	Y

Academic Level:	FHEQ7	**Credit Value:**	15.00	**Total Delivered Hours:**	30.00
Total Learning Hours:	150	**Private Study:**	120		

Delivery Options
Course typically offered: Runs Twice - S1 & S2

Component	Contact Hours
Lecture	12.000
Seminar	12.000
Tutorial	6.000

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Technology	AS1	Object-oriented Analysis, Design and Implementation (minimum 3,000 words).	50.0	
Technology	AS2	Class responsibility collaboration (CRC) based Software Development	50.0	

Aims

The course will develop the necessary skills for the development of object-orientated applications using the OO programming language. Students will work to gain the skills required to engineer OO-based software applications from initial specification, through to implementation, testing and documentation.

Learning Outcomes

After completing the module the student should be able to:

1 Critically analyse the use of object-oriented principles in the design of software applications.
2 Use object-oriented principles to produce UML specifications of software applications
3 Demonstrate mastery of UML specifications using the selected OO programming language.

Learning Outcomes of Assessments

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

Object-oriented Analysis	1	2
Software Development	3	

Outline Syllabus

Foundations of object-orientation.
Anatomy of OO classes – fields, constructors, methods.
Objects and classes – what is an object, object state, objects as parameters.
Object interaction – method invocation, objects calling objects.
Designing classes – responsibility-driven design.
Application structures – inheritance, subtyping, polymorphism.
Abstraction techniques – simulation, abstract classes, interfaces.
Handling errors – defensive programming, exceptions.
Designing applications – analysis and design, CRC cards, scenarios, class design, documentation, group cooperation.
Case Study – design of a chosen application.
Java-based user-interfaces – AWT and Swing APIs

Learning Activities

Online Lectures will be accompanied by practical online sessions. Students will be required to work in small groups to complete tasks, thereby encouraging communication and projects management skills

References

Course Material	Book

Author	Eckel, B.
Publishing Year	2004
Title	Thinking in Java
Subtitle	
Edition	3rd
Publisher	Prentice-Hall
ISBN	0-131-00287-2

Course Material	Book
Author	Arnold, K. Gosling, J.
Publishing Year	2000
Title	The Java Programming Language
Subtitle	
Edition	3rd
Publisher	Addison-Wesley
ISBN	0-201-70433-1

Course Material	Book
Author	Barnes, D.J., Kolling, M.
Publishing Year	2003
Title	Objects First with Java: A Practical Introduction using BLUEJ
Subtitle	
Edition	
Publisher	Prentice-Hall
ISBN	0-13-044929-6

Course Material	Book
Author	C. Thomas Wu
Publishing Year	2009
Title	An Introduction to OO Programming with Java
Subtitle	
Edition	
Publisher	McGraw Publisihing
ISBN	0073523305

Notes

The module lectures, tutorials and labs will use the BLUEJ development tool, which is a GUI-based development aid based on UML.