

Module Title	Software Development 2
Is this an EXISTING MODULE?	No
If YES please give 'G Code'	
HESA Cost Centre	
Module Leader(s)	Dr Christopher Tubb
Staff Ref No	
CATS Tariff	20
CATS Level	5
HESA Subject Code	
Campus on which Module taught	Allt-yr-yn
Percentage of module not taught at UWCN	N/A
If course is taught externally please name other institution(s)	N/A
Franchise Centre(if applicable)	N/A
HEFCW ASC	6
Deleted Modules Titles	
Programme(s) that module is attached to: BSc Networked Information Systems BSc Internet & Multimedia Technology, BSc Internet & Network Technology	Compulsory (Y/N): Y
Professional Body Accreditation/Exemption	None
Contact Hours	60
Directed Studies Hours	70
Independent Studies Hours	70
Mode(s) of Attendance	FT/PT
Pre-requisites	Software Development 1; GUI Programming; Data Structures
Co-requisites	None
Semester(s) in which taught	3
Anticipated Number of Students	
Session of First Enrolment	2004/2005

Software Development 2

1. Rationale

This module will extend the skills gained by the student in the Software Development module of semester 1, introducing Object Oriented concepts and techniques and providing practical implementation experience through the use of a suitable Object Oriented programming language.

2. Aims

- To describe software object concepts and techniques.
- To relate software object concepts to both implementation languages and the “real world”.
- To provide practical experience of an object oriented language.
- To exploit an object oriented language for the creation of programs employing software object concepts.

3. Learning Outcomes

On successful completion of this module students will be able to:

- demonstrate a good understanding of object concepts such as Encapsulation, Abstraction, Inheritance and Polymorphism;
- discuss the properties of software object systems;
- create class definitions that model real world systems;
- create robust software which employs object concepts and techniques;
- use an object oriented programming language to achieve a stated task.

4. Indicative Content

Software Objects

Real world objects, Classes as object descriptions. Comparison with procedural methods. Encapsulation, Abstraction, Inheritance, Polymorphism. Scope issues. Persistence. Associations and relationships between objects. Messages and message passing. UML notation. Technical and commercial imperatives to object orientation.

Implementation issues

Using an object oriented programming language to implement class descriptions. Development of object oriented programmes using a contemporary development environment. Event and exception handling.

5. Teaching and Learning Strategy

As befits a module in which software implementation is an aim, there will be a large element of practical work for students to perform. Topics and concepts will be introduced using lecture, discussion and demonstrations, followed by exercises and examples for students to complete. Student will also be expected to achieve tutorial tasks which support the material covered in the lecture sessions.

6. Assessment Requirements

Assessment will comprise of two pieces of coursework, one written and one of a practical nature.

The written assignment will require that the student demonstrate informed understanding of the conceptual elements of the course.

The practical assignment will require that the student demonstrate the ability to apply these concepts through the development of software to achieve a stated task.

7. Indicative Reading

BUDD T. (2002) *An Introduction to Object Oriented Programming 3rd Edition*. Boston, Addison Wesley. 0201700026

LEDGARD, H. (1995) *The little book of object-oriented programming*. New Jersey: Prentice Hall. 013396342

MALIK D.S. and NAIR P.S. (2003) *Java Programming: From Problem analysis to Program Design*. Boston, Course Technology 0471376817

SHAMMAS, N. C. (1996) *Object-oriented programming for dummies*. Foster City IDG, 1568843321

WIENER R. (2000) *Fundamentals of OOP and data structures in Java*. Cambridge. Cambridge University Press. 0521662206

XIAOPING J. (2000) *Object-oriented software development using Java: principles, patterns, and frameworks*. Reading, Mass. Addison-Wesley. 020135084X

8. Professional Status

None

9. Student Evaluation

Student evaluation of the module will be achieved through the use of module questionnaires and direct student feedback.

10. Module Linkages

This module extends understanding and practical skills gained in *Software Development I*, *Data Structures*, and *GUI Programming*, to encompass object-oriented ideas.
