



**GALWAY-MAYO INSTITUTE OF
TECHNOLOGY**

Instiútúio Techneolaíochta na Sálúimhe-Maigh Eo

Master of Science in Computing

Programme Handbook 2010 / 2011

TABLE OF CONTENTS

	Page
PREFACE: Approved Programme Schedule	X
Timetable of Classes 2010 / 2011	XI
1. Overview of the MSc in Computing Programme	1
1.1 Aims and Objectives of the Masters Programme	1
1.2 Entry Requirements	1
1.3 Programme Duration	1
1.4 Programme Content	2
1.5 Programme Learning Outcomes	2
2. Module Information and Syllabus Content	5
2.1 Software Engineering	6
2.2 Object-Oriented Systems	10
2.3 Advanced Databases	15
2.4 Network and Forensic Computing	20
2.5 Research Seminar (Full-time candidates only)	25
2.6 Research Methods (Full-time candidates only)	28
3. Programme Calendar 2010 / 2011	30
APPENDIX I: Personal Notes Section	
APPENDIX II: Academic Staff – Contact Details	

GALWAY-MAYO IT Approved Programme Schedule

Program Code	GA_KCOMP_MGA	Version Number	5.0
Award Title	MASTER OF SCIENCE	First Approval	1997
Area of Specialisation	in Computing	Current Approval	Approved
		Next Review	

Stages 1 Intake First **2007**

Learning Modes FULL ACCS PART

Stage Award

Seq No	Course Code	Title of Examination Subject	Credits			Contact Hours	Component Marks				
			Status	NQAI Level	Number		Continuous Assessment	Project	Practical	Final Exam	Maximum
1	PROJ Z5007	Masters Project	E	9	20	1		100.00			100.00
2	RESE Z5002	Research Seminar	E	9	15	2	100.00				100.00
3	COAPZ5003	Computer Telephony Integration	E	9	10	4	20.00		30.00	50.00	100.00
4	RESE Z5003	Research Methods (Workshop)	E	9	5	2	100.00				100.00
5	COMP Z5030	Advanced Databases	E	9	10	4	40.00			60.00	100.00
6	COMP Z5037	Forensic Computing	E	9	10	4	40.00			60.00	100.00
7	COMP Z5031	Object Oriented Systems	E	9	10	4	50.00			50.00	100.00
8	COMP Z5032	Software Engineering	E	9	10	4	30.00			70.00	100.00
9	MGMT Z5008	Software Distribution & Mgt	E	9	10	3	30.00			70.00	100.00
20	COMP Z5035	Bioinformatics	E	9	10	4	40.00			60.00	100.00
		Credits Available			110						
		Credits Required			60						

M.Sc in Computing - Timetable (2010 / 2011)

	Mon	Tue	Wed	Thu	Fri
09:00 - 10:00					
10:00 - 11:00				PF18 Research Methods	PF18 Research Seminar
11:00 - 12:00		PF18 Forensic Computing	PF18 Software Engineering	PF18 Research Methods	PF18 Research Seminar
12:00 - 13:00		PF18 Forensic Computing	PF18 Software Engineering		
13:00 - 14:00				Object Oriented PF18 Systems	
14:00 - 15:00	CR4 Advanced Databases	CR4 Forensic Computing		Object Oriented PF18 Systems	
15:00 - 16:00	CR4 Advanced Databases	CR4 Forensic Computing	CR4 Software Engineering		
16:00 - 17:00	366 Advanced Databases		CR4 Software Engineering	Object Oriented CR4 Systems	
17:00 - 18:00	366 Advanced Databases			Object Oriented CR4 Systems	

1.1 Aims and Objectives of the Masters Programme

- To provide an opportunity to new graduates and established computing professionals for personal career development by undertaking formal post-graduate education on a full-time or part-time basis.
- To provide commercial and industrial companies with a locally-based means of upgrading the computing skills and knowledge of their staff.
- To increase the number of practitioners with advanced skills and to foster a spirit of co-operation and collaborative research between GMIT and other organisations; locally, nationally and internationally.

On completion of the M.Sc the participant will:

- Have developed core skills in key areas of advanced computing techniques.
- Be suitably informed of emerging developments in these areas.
- Be able to apply best practices across a range of software-related disciplines.

1.2 Entry Requirements

- Honours Degree in Computing or Honours Degree in Electronics/Computer Engineering with substantial software development experience.

1.3 Programme Duration

- The MSc in Computing can be completed as a one year full-time programme or alternatively as a part-time programme typically over two to three years. This twelve-month programme commences in early September and concludes in late August of the following year.
- The programme will commence in September 2010.
- Part-time students benefit from lectures and laboratories that are scheduled in block format, typically on one afternoon and one evening per week.
- To be eligible for the award of Master of Science (Computing), participants must successfully complete 4 taught Modules and a Masters Project. Course credits may be accumulated under the HETAC's ACCS system.

1.4 Programme Content

- Each taught module consists of 100 hours class contact, with participants undertaking a substantial workload on their own.

The taught modules for the 2010 - 2011 intake are:

- Software Engineering
- Object Oriented Systems
- Advanced Databases
- Forensic Computing

Full-time students must take all of the above modules; part-time students can elect to take one or more modules on offer from September 2010.

Full-time students must also take two research-oriented modules; *Research Methods* (workshop) and *Research Seminar* which collectively aim to develop core competencies in critical thinking, research techniques, research methodologies, technical writing & critique, and a process oriented approach to research work. Part-time candidates must complete a *Research Project*, which will be supervised and will be of approximately 450 - 500 hours duration.

1.5 Programme Learning Outcomes

Knowledge – (Breadth)

- The learner will have expert knowledge in object oriented design techniques, modern database configuration and performance tuning strategies, advanced software engineering methodologies and the application of complex forensic analytical tools to computer and network systems.
- The learner will be able to demonstrate the application of appropriate research methodologies and techniques within the domain of computing and software systems.

Knowledge – (Kind)

- The learner will be capable of demonstrating a critical understanding of established and emerging developments within the areas of object oriented development, enterprise database systems, software quality management and forensic computing techniques.
The learner will be capable of discussing current challenges and research foci in these subject areas.
- The learner will be capable of applying modern research methods appropriate to applied computing research problems / questions.

Know-How / Skill – (Range)

- The learner will be able to critically evaluate, at an advanced level, design, development and configuration aspects of complex, network-based, data centric, object-based software systems.
- The learner will be able to communicate to peers, both written and verbally, on core and advanced aspects of these foci of study, in an articulate, convincing and informed fashion.
- The learner will be capable of integrating theoretical knowledge and applied development skills to address complex, technology centric problems in the area of modular, open, data-dependent, security sensitive software systems.
- The learner will be capable of demonstrating an awareness of the present state of the art in the area of their chosen (or assigned) project / dissertation topic, including the ability to evaluate the established literature base in that subject area.
- The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to within the chosen (or assigned) applied research / dissertation topic.
- The learner will be capable of formally exhibiting his / her research capabilities within the chosen (or assigned) applied research / dissertation topic.

Know-How / Skill – (Selectivity)

- The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to the areas of object oriented systems, advanced database technologies, software engineering and forensic computing.
- The learner will be able to integrate disparate technologies and principles to successfully develop and deliver an appropriately integrated solution to a computer-based project.
- The learner will be able to apply research and critical thinking skills developed on the programme to a challenging computer-based problem.
- The learner will be able to evaluate, select and apply standard and customised research tools and methodologies of enquiry.

Competence – (Context)

- The learner will be able to identify sufficiently complex potential projects and research opportunities within the areas of object oriented systems, advanced database technologies, and forensic computing.
- The learner will be able to design and implement a computing solution that requires significant preliminary research.
- The learner will be able to analyse, appropriately justify, and professionally document designs proposed / implemented to solve assigned problems in the areas of object oriented systems, advanced database technologies, software engineering and forensic computing.
- The learner will be able to apply and critically evaluate best practices in object oriented design techniques, advanced data modelling, software engineering methodologies, and the application of forensic analysis techniques to network & computer systems.

- The learner will be able to demonstrate an appreciation of the professional standards relevant to the areas of object-oriented systems, advanced database technologies, software engineering and forensic computing.

Competence – (Role)

- The learner will be able to participate in peer collaboration and evaluation exercises.
- The learner will be able to work as a member of technology-focused project planning and development team.

Competence – (Learning to Learn)

- The learner will be able to critically evaluate his/her own work and research and reflect on the strength, weaknesses and future potential of such work.
- The learner will be able to demonstrate an understanding of the importance and need for continued personal development in the research and study disciplines of the programme.

Competence – (Insight)

- The learner will be able to demonstrate a critical appreciation of the design, development, implementation and on-going management of complex computing systems and the relevance of the contextual environment within which they reside.
- The learner will be able to maintain integrity and independence in professional judgement and conduct his / her work in a professionally ethical manner.

Core Module Information and Syllabus Content

SOFTWARE ENGINEERING

MODULE NAME	Software Engineering
MODULE CODE	COMP Z5032

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	3
	Tutorials	1
	Laboratory	
	Seminar	
	Other	

PRE-REQUISITE MODULES	Programme entry pre-requisites
CO-REQUISITE MODULES	

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE:

KNOWLEDGE – Breadth	The learner will have a thorough understanding of the critical issues in project management, both technical and motivational.
KNOWLEDGE – Kind	The learner will be capable of demonstrating a critical understanding of established and emerging developments within the area of software engineering and software quality management.

KNOW-HOW / Skill – (Range)	<p>The learner will be capable of integrating theoretical knowledge and applied development skills to address complex software engineering problems.</p> <p>The learner will be capable of demonstrating an awareness of the present state of the art in the management of software development and software quality.</p>
----------------------------	---

	The learner will understand the importance of metrics and be able to apply them in a wide range of software development environments.
KNOW-HOW / Skill – (Selectivity)	The learner will be able to apply research and critical thinking skills to a emerging fields within software engineering.

COMPETENCE – Context	The learner will be able to apply and critically evaluate best practices in software engineering methodologies.
COMPETENCE – Role	The learner will be able to participate in peer collaboration and evaluation exercises.
COMPETENCE – Learning to Learn	The learner will be able to demonstrate an understanding of the importance and need for continued personal development in the research, study and application of software engineering techniques.

SOFTWARE ENGINEERING – continued...

MODULE CONTENT...

Objectives

On completion of this module the participant will:

- Incorporate best practice in software development
- Have a thorough understanding of the critical issues in project management, both technical and motivational
- Have a sound foundation for software re-engineering and reuse, encompassing both organisational and methodological aspects
- Understand the importance of metrics and be able to apply them in a wide range of software development environments
- Have a basis for the incorporation of formal methods in software development

Prerequisites

The participant is expected to be well-informed in the following areas:

- Software Development Process Models
- Software Requirements and Specification
- Requirements Management
- Analysis Methodologies
- Design Methodologies
- Appropriate texts: Software Engineering (R. Pressman)
 Software Engineering I (Sommerville)

Module Content

Review of traditional software development methodologies.

- Waterfall and Spiral Models, Iterative and Incremental Development, Risk Management.

Agile Software Development. Philosophy, Principles and Practices.

- Extreme Programming (XP) and the Scrum Software Development Process.
- Test-Driven Development (TDD)
- Automated Testing and Deployment
- Acceptance Testing
- Whitebox and Blackbox Testing Strategies.
- Software Antipatterns.
- Taxonomies, detection, refactoring and reuse.

Software Quality Management and Software Process Improvement.

- ISO 9001:2000, CMMI, ISO 15504 (SPICE) and ISO 12207
- Product Quality (including ISO 9126) and Software Usability (including ISO 9241)
- Software Configuration Management and Maintenance

Software Project Management

- Cost Estimation and Risk Management, Methodologies, Critical Systems Development, Software Evolution, Security Engineering.

Software Metrics and Information Quality.

- Estimation Techniques, Information Quality Frameworks, Information Quality Assessment.

Formal Methods. Z Specifications, Verification, Correctness, Proofs and Case Studies

Assessment

- Course work and Assignments (30%)
- Written Examination (70%)

Recommended Reading

- Software Quality, a Framework For Success - Sanders & Curran 1994
- Managing the Software Process - W. Humphrey 1989
- Introduction to the Personal Software Process - W. Humphrey 1997
- Introduction to the Team Software Process - W. Humphrey 2000
- Books on Testing by Beizer, Hetzel, Kaner and Myers
- A Discipline for Software Engineering - W. Humphrey 1995
- Software Engineering R. Pressman (5th Edit) 2000
- Software Engineering I. Sommerville (6th Edit) 2000
- Software Re-use - a Holistic Approach - E. Karlsson 1995
- Software Reuse and Re-engineering in Practice - Ed. P. Hall 1992
- The Three R's of Software Automation: Re-engineering, Repository and Re-usability - C. McClure 1992
- Software Inspection - Gilb and Graham 1993
- Software Quality Management and ISO 9000 - M. Jenner 1995
- ISO 9000 for Software Development - C. Schmauch 1995
- The SPIRE Handbook - Centre for Software Engineering 1998
- An Introduction to Formal Specification & Z - Potter, Sinclair & Till 1991

Recommended Journals

- IEEE Transactions on Software Engineering
- Association for Computing Machinery

Module Coordinator

Contact: John Healy (john.healy@gmit.ie)

Last Revision: June 2009.

OBJECT-ORIENTED SYSTEMS

MODULE NAME	Object-Oriented Systems
MODULE CODE	COMP Z5031

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	3
	Tutorials	
	Laboratory	1
	Seminar	
	Other	

PRE-REQUISITE MODULES	
CO-REQUISITE MODULES	

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE:

KNOWLEDGE – Breadth	Provide students with an expert knowledge of the object-oriented paradigm, its applications and limitations and develop an ability to critically analyse modern object-oriented approaches to systems design and development.
KNOWLEDGE – Kind	Understand the current challenges and limitations of object-oriented design & development and apply best-practice and emerging methodologies.
KNOW-HOW & SKILL – Range	Develop an expert knowledge of modern best practice in object-oriented systems design and development. This focus on best-practice will span architectural approaches to object-oriented development and fine-grained class design. Describe, analyse and communicate complex object-oriented designs using the Unified Modelling Language.
KNOW-HOW & SKILL – Selectivity	Apply object-oriented approaches to complex, semi-structured and unstructured problem domains. Develop the critical thinking skills required for undirected research, enquiry and evaluation.

COMPETENCE – Context	Evaluate and expand existing object-oriented best practices. Develop object-oriented models of complex problem domains and use research and best practice to solve tangible problems.
COMPETENCE – Role	Apply object-oriented best practices and professional standards to real-world problems. Undertake the appropriate research to apply object-oriented principles and practices to semi-structured and complex problems.

OBJECT ORIENTED SYSTEMS – continued...

MODULE CONTENT...

I. Review of the Object Oriented Paradigm

- Abstraction & Encapsulation
- Generalisation & Polymorphism

II. Classes & Objects

- Class/ Instance Fields & Methods
- Creating & Initialising Objects
- Field Defaults & Initialisers
- Destroying & Finalising Objects
- Object hierarchies and Constructor Chaining
- Inner Classes
- Static & Dynamic Object Introspection. Reflection
- UML notation for classes and objects, UML Datatypes, Multiplicity, Attributes & Operations

III. Composition & Inheritance

- Aggregation, & Association. Uses, context and UML representation, Associations & Navigability, Reifying Associations, N-Ary Associations.
- Inheritance
 - Principle of Substitutability, Subclasses and Subtypes
 - Forms of Inheritance: Specialisation, Specification, Construction, Extension, Limitation, Combination
 - Inheritance in Java: Shadowing Superclass Fields, Overriding Superclass Methods, Modifiers and Inheritance. UML representations.
 - Interfaces & Abstract Classes: Implementing Interfaces, Immutability and Design by Contract, Abstract Classes v Interfaces. Describing Interface and Abstract Classes in UML.
 - Multiple Inheritance: The Deadly Diamond of Death, Using Mixin inheritance to simulate multiple inheritance.
 - Benefits & Costs of Inheritance, Inheritance & Encapsulation, Composition v Inheritance.

IV. Polymorphism

- Static and Dynamic object Binding. Polymorphism & Typing
- Ad Hoc v Universal Polymorphism
- Coercion and Overloading, Parametric Overloading
- Inclusion Polymorphism
- Overriding & Polymorphism, Abstract /Deferred Methods
- Polymorphism, Inheritance & Loose Coupling
- Case Study in Polymorphism: The Collections API
 - The Collection hierarchy and interfaces
 - Set, List, Map implementations.

V. Describing Object Interaction

- Interaction Diagrams: Collaborations, Classifier Roles, Association Roles
- Sequence Diagrams: Messages to Self, Object Creation & Destruction, Guards.

VI. Testing Object Oriented Applications

- Unit Testing and Test Driven Object-Oriented Development, Test Cases and Test Suits, Stubs and Self Shunting, JUnit and the JUnit API.

VII. Class and Package Design Principles

- Class Design
 - Single Responsibility & Open-Closed Principles
 - Liskov Substitution, Dependency Inversion & Interface Segregation
- Package Cohesion & Coupling
 - Reuse-Release Equivalence, Common Closure & Common Reuse Principles
 - Acyclic Dependencies, Stable Dependencies & Stable Abstractions
- Composite Design Patterns: Model-View-Controller, Architectural and Class Level applications of MVC, Producer-Consumer Model.

VIII. Creational Patterns

- Factory & Abstract Factory, Singleton & Concurrency Issues, Builder & Prototype Patterns.

IX. Structural Patterns

- Decorator: Case Study: The Decorator Pattern and the java.io Library
- Façade, Session and Message Façades. The Principle of Least Knowledge.
- Flyweight, Adapters, Bridge, Composite Patterns.
- Proxy: Dynamic, Remote, Smart & Synchronisation Proxies. Copy-on-write, Complexity Hiding.

X. Behavioural Patterns

- Chain of Responsibility, Invocation Handlers.
- Command: Invocation Encapsulation, Marcos, Queuing and Logging.
- Observer: Publisher and Subscribers, Applications of Producer-Consumer Model, Listeners and Push/Pull Models.

- State, Strategy and Template Patterns, The Hollywood Principle. Template v Strategy.
- Visitor: Dynamic Dispatch and Polymorphic Handshakes.
- Interpreter, Iterator, Mediator & Memento Patterns.

XI. Distributed Objects and Strategies

- Overview of architectural approaches. J2EE, CORBA and Web Services. N-Tier Applications and Object-Oriented Development.
- Data Externalisation and Marshalling. Naming Registries & Binding.
- Remote Method Invocation: Remote and Local Interfaces. Object stubs, skeletons. Interface Specification. Serialization & Object Parameters. Passing remote object parameters by reference, Distributed Garbage Collection, Callback Methods, Dynamic Class Loading, Gateway Objects, JRMP & IIOP.
- Distributed Polymorphism.
- Binding to remote objects using HTTP.
- Stateful and stateless remote objects
- Synchronous and asynchronous distributed object communication. Using message queues.
- Design Patterns for Distributed Architectures: Business Delegate, Business Interface, Home Factory & EJB Command Patterns.

XII. Object Binding with XML

- XML Schemas, Type Scope & Schema Reusability
- JAXB Binding Framework
 - Unmarshalling & Marshalling XML Documents, JAXB Customisation
- Object Persistence with DOM and PDOMs. Using Native XML Databases to persist object state.

XIII. Object-Oriented Databases

- Object-Relational Mapping and Strategies, Case Study: Hibernate.
- Migrating from Relational to Object-Oriented Databases.
- Features of Object-Oriented Databases: Complex Objects, Object Persistence, Transactions, Concurrency
- Object-Oriented DBMS Technologies: Pure OODBMs, Persistent Storage Managers, Object Wrappers.
- Extended RDBMs: Development with Object-Oriented Databases, e.g. O2, Ozone. Object-Oriented DBMS Applications.
- Data Transfer Object (DTO), DTO Factory, Domain Transfer Model, Generic Attribute Access.

Assessment

- Continuous Assessment(s) – 50%
- Final Examination (written) – 50%

Required Reading

- *Object-Oriented Software Engineering* - I. Jacobson 1993
- *The Java Programming Language*, 3rd Ed, Arnold & Gosling, Addison-Wesley, 2000, ISBN: 0201704331
- *Learning UML*, Sinan Si Alhir, O'Reilly, 2003, ISBN: 0-596-00344-7
- *Design Patterns - Elements of Reusable Object Oriented Software*, Erich Gamma et al, Addison Wesley, 1994, ISBN 0-201-63361-2

Recommended Reading

- *The Object-Oriented Thought Process*, Matt Weisfeld, SAMS, 2000, ISBN: 0672318539
- *Java Programming with CORBA*, Brose, Vogel & Duddy, Wiley, 2001, ISBN: B000050BU5
- *Java & XML Data Binding*, McLaughlin, O'Reilly, 2002, ISBN:0-596-00278-5
- Booch, G., Jacobsen, I., and Rumbaugh, J. (1997). The UML specification documents. Rational Software Corp., www.rational.com
- Booch, G., Rumbaugh, J. and Jacobson, I. (1999). The Unified Modeling Language user guide. Addison Wesley Longman, Inc. Reading, MA.USA.
- Fowler, M., and Scott, K. (2000): UML Distilled. Addison-Wesley. Reading, MA. USA.
- IEEE Transactions on Software Engineering
- Association for Computing Machinery
- Journal of Object-Oriented Programming

Module Coordinator

Contact: John Healy (john.healy@gmit.ie)

Last Revision: June 2009.

ADVANCED DATABASES

MODULE NAME	ADVANCED DATABASES
MODULE CODE	COMP Z5030

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	3
	Tutorials	
	Laboratory	1
	Seminar	
	Other	

PRE-REQUISITE MODULES	Programme entry pre-requisites (Students are expected to be versed in database fundamentals, SQL and Object-oriented concepts).
CO-REQUISITE MODULES	

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE.

KNOWLEDGE – Breadth	<p>The learner will have expert knowledge of database configuration and performance tuning strategies.</p> <p>The learner will be able to demonstrate the application of appropriate research methodologies and techniques within the domain of computing and software systems.</p>
KNOWLEDGE – Kind	<p>The learner will be capable of demonstrating a critical understanding of established and emerging developments within the area of enterprise database systems.</p> <p>The learner will be capable of discussing current challenges and research focus for database systems</p> <p>The learner will be capable of applying modern research methods appropriate to applied computing research problems / questions.</p>

KNOW-HOW / Skill – (Range)	<p>The learner will be able to critically evaluate, at an advanced level, design, development and configuration aspects of complex, data centric software systems.</p> <p>The learner will be capable of integrating theoretical knowledge and applied development skills to address complex, technology centric problems in the area of modular, open, data-dependent software systems.</p>
KNOW-HOW / Skill – (Selectivity)	<p>The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to the areas of advanced database technologies.</p> <p>The learner will be able to integrate disparate technologies with database management systems and principles to successfully develop and deliver an appropriately integrated solution to a computer-based project.</p> <p>The learner will be able to apply research and critical thinking skills developed in the module to a challenging data centric problem.</p>
COMPETENCE – Context	<p>The learner will be able to identify sufficiently complex potential projects and research opportunities within the advanced databases subject area.</p> <p>The learner will be able to design and implement a computing solution that requires significant preliminary research.</p> <p>The learner will be able to analyse, appropriately justify and professionally document designs proposed / implemented to solve assigned problems in the area of advanced database technologies.</p> <p>The learner will be able to apply and critically evaluate best practices in advanced data modelling,</p> <p>The learner will be able to demonstrate an appreciation of the professional standards relevant to the areas of advanced databases.</p>

COMPETENCE – Role	<p>The learner will be able to participate in peer collaboration and evaluation exercises.</p> <p>The learner will be able to work as a member of technology-focus project planning and development team.</p>
COMPETENCE – Learning to Learn	<p>The learner will be able critically evaluate his/her own work and research and reflect on the strengths, weaknesses and future potential of such work.</p> <p>The learner will be able to demonstrate an understanding of the importance and need for continued personal development in the research and study disciplines of the programme.</p>
COMPETENCE – Insight	<p>The learner will be able to demonstrate a critical appreciation of the design, development, implementation and on-going management of complex computing systems and the relevance of the contextual environment within which they reside.</p> <p>The learner will be able to maintain integrity and independence in professional judgement and conduct his / her work in a professionally ethical manner.</p>

Module Objectives

The aim of this module is to give students a deeper and broader view of the topic, through consideration of the elements of constructing database system software, and by examining the variety of systems and applications currently in use and being researched.

Having successfully completed the module, the learner will be able to demonstrate knowledge and understanding of:

- The issues involved in developing database management software.
- The reasons for the variety of database types now available.

ADVANCED DATABASES – continued...

I. Relational Databases

A Review of Relational Theory
Relational Algebra, Normalisation, SQL Queries
Query Optimisation
Function Dependencies
Inference Axioms (Armstrong & B-Axioms)
Entity-Relationship Modelling
Enterprise Data Modelling
Zachman Charts
Data Dictionary
Security and Integrity issues
Use of CASE tools.
Design of large database systems.
Fourth Generation Languages
Database Middleware
Gateways to Legacy Databases

II. Database Machines

Parallel Database Processors.
Possible Architectures
Example systems
Performance
Data Warehousing

III. Multimedia and Object Orientation

Concepts and Techniques for Multimedia Data Systems
Features of Object Oriented and Semantic Databases
Motivation - OO Concepts Applied to Databases
Object Oriented Database Systems and Applications

IV Databases and the World Wide Web

Database Architecture in the Web Environment
Enterprise DBMS Deployment
Freeware Data Services

V. Distributed Databases

Distributed Database Principles
Advantages and Associated Problems
Performance Issues, Data placement and Allocation
Query optimisation
Concurrency Control and Recovery
Integrity and Security Issues.

Assessment

- Course work and Assignments (40%)
- Final Examination (60%)

Recommended Reading

- Fundamentals of Database Systems, 5th Edition
Elmasri & Navathe (2007)
- An Introduction to Database Systems
C.J Date (2003)
- Database Systems, 3rd Edition
T Connolly, C Begg & A Strachan (2002)

Recommended Journals

- IEEE Transactions on Software Engineering
- Association for Computing Machinery

Module Coordinator

Contact: Owen Foley (owen.foley@gmit.ie)

Last Revision: June 2009.

FORENSIC COMPUTING

MODULE NAME	Forensic Computing
MODULE CODE	COMP Z5037

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	2
	Tutorials	
	Laboratory	2
	Seminar	

PRE-REQUISITE MODULES	Programme entry pre-requisites	
CO-REQUISITE MODULES		

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE:

KNOWLEDGE – Breadth	The learner will have expert knowledge in the application of complex forensic analytical tools to computer and network systems.
KNOWLEDGE – Kind	The learner will be capable of demonstrating a critical understanding of established and emerging developments within the area of forensic computing techniques.

KNOW-HOW / Skill – (Range)	<p>The learner will be able to critically evaluate to an advanced level; design, development and configuration aspects of complex local and wide area networks.</p> <p>The learner will be able to communicate to peers, both written and verbally on such core development and configuration aspects in an articulate convincing and informed fashion.</p>
KNOW-HOW / Skill – (Selectivity)	The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to network & forensic computing.

COMPETENCE – Context	<p>The learner will be able to analyse, appropriately justify and professionally document advanced solutions to complex network security problems.</p> <p>The learner will be able to apply and critically evaluate best practices in network security and forensic analysis of computer systems.</p> <p>The learner will be able to demonstrate an appreciation of the professional standards relevant to practicing as a forensic computing analyst.</p>
COMPETENCE – Role	<p>The learner will be able to participate in peer collaboration and evaluation exercises specific to network configuration and computer / network forensic analysis.</p> <p>The learner will be able to work as a member of a network design and analysis team.</p>
COMPETENCE – Learning to Learn	<p>The learner will be able to critically reflect on the strength, weaknesses and future potential of emerging technologies in the area of forensic computing</p>
COMPETENCE – Insight	<p>The learner will be able to demonstrate a critical appreciation of the design, development, implementation and on-going secure management of complex network systems and the relevance of the contextual environment within which they reside.</p>

FORENSIC COMPUTING – continued...

Objectives

- To develop expert knowledge in the application of forensic analytical tools to computer and network systems.
- To be capable of demonstrating a critical understanding of established and emerging developments within the area of forensic computing techniques.
- To be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to forensic computing.
- To be able to demonstrate an appreciation of the professional standards relevant to practicing as a forensic computing analyst.

Technical Prerequisites

Degree level knowledge of operating systems, data communications and network technologies.

MODULE CONTENT...

I. Computer Forensic Analysis

Context of Computer Forensic Analysis
Legislative Framework
Collecting Evidence
The Forensic Workstation
Live Response
Acquiring a Forensic Duplication
Forensic Analysis of Windows Operating Systems
File System Forensics
Root Kits
Data Hiding Techniques
Steganographic Techniques
Device Forensics (USB & Other Memory Devices)
Encrypted Files
Password Recovery
Emerging Forensic Analysis Tools & Practices

II. Forensics and Internet Core Protocols

Protocol Design: IP, TCP, UDP, ICMP, BOOTP, DHCP, DNS
Multicasting Protocols
Private Network Interconnection (NAT and VPN)
Classless and Subnet Address Extensions (CIDR)
Ipv6
Emerging Standards
Major Network Services and APIs
Transport Level Interface Programming

Threads
Input and Output Streams
Sockets for Clients
Sockets for Servers

III. Network & Application Security

Cryptographic Theory
Secret-Key Algorithms
Public-Key Algorithms
Digital Signatures
Authentication and Key Distribution
Design of Secure Sockets Layer
Implementing Secure Socket Applications
Web Security using Secure HTTP
Emerging Internet Security Protocols
File / Application Security

IV. Forensic Analysis of Network Traffic

Network Intrusions and Attacks
Direct vs Distributed Attacks
Buffer Overflows
Accidental Attacks
Address Spoofing
Packet (Header) Spoofing
ARP Spoofing (Poison ARP)
DNS Spoofing
Forensic Tools for Network Investigations
(TCPDump, Ethereal and Ethereal)
Online Forensics

V. Dissemination of Forensic Results

Quality of Evidence
Process and Tool Validation
The Forensic Report
Report Generation Tools
Expert Witness Testimony
Precedent / Case Study Analysis

Assessment

- Course work and Assignments (40%)
- Written Examination (60%)

Recommended Reading

- Internetworking with TCP/IP (Vol. 1 Principles, Protocols & Architecture), 5th Edition. *D. Comer* (2006).
- Computer Networks (4th Edition) , *A. Tanenbaum* (2003).
- Internetworking with TCP/IP (Vol. 2 Design, Implementation & Internals), 5th Edition. *D. Comer* (2006).
- Real Digital Forensics. *K Jones, R Bejtlich, C Rose* (2008)
- File System Forensic Analysis. *B Carrier* (2007)
- Computer Forensics and Investigations. *B. Nelson, A. Phillips, F. Enfinger, C. Steuart* (2005).
- Building Cisco Multilayer Switched Networks . *R Froom* (2006).
- Building Scalable Cisco Internetworks. *Paquet / Teare* (2006).
- Security in Computing, 4th Edition. *C.P Pfleeger & S.L Pfleeger* (2007)
- Computer Forensics: Principles and Practices. *L Volonino, R Anzaldua & J Godwin* (2007)

Recommended Journals

- Digital Investigation (The International Journal of Digital Forensics & Incident Response) ISSN: 1742-2876
- Communications of The ACM
- IEEE Transactions on Communications
- IEEE Network Magazine
- IEEE Communications Magazine
- Computer Communications Review
- Various Internet RFCs (Requests for Comments)

Module Coordinator

Contact: Dr. Sean Duignan (sean.duignan@gmit.ie)

Last Revision: June 2009.

RESEARCH SEMINAR

MODULE NAME	Research Seminar
MODULE CODE	RESE Z5002

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	
	Tutorials	
	Laboratory	
	Seminar	2
	Other	

PRE-REQUISITE MODULES	Programme entry pre-requisites
CO-REQUISITE MODULES	

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE:

KNOWLEDGE – Breadth	<p>Demonstrate the application of appropriate research methodologies and techniques within the domain of computing and software systems.</p> <p>Demonstrate an awareness of the present state of the art in a specialist area of computing including the ability to evaluate the established literature base in that subject area.</p>
KNOWLEDGE – Kind	Apply modern research methods appropriate to applied computing research problems / questions.
KNOW-HOW / Skill – (Range)	Independently acquire and assess relevant knowledge that is contextually appropriate and specific to an applied area of computing research.

KNOW-HOW / Skill – (Selectivity)	<p>Apply research and critical thinking skills to a challenging computer-based problem.</p> <p>Design and implement a computing solution that requires significant preliminary research.</p> <p>The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to an applied area of computing research.</p> <p>The learner will be capable of formally exhibiting his / her research capabilities within an area of applied computing.</p>
COMPETENCE – Context	<p>The learner will be able to design and implement a computing solution that requires significant preliminary research.</p>
COMPETENCE – Learning to Learn	<p>The learner will be able critically evaluate his/her own work and research and reflect on the strength, weaknesses and future potential of such work.</p> <p>The learner will be able to demonstrate an understanding of the importance and need for continued personal development in the research and study disciplines of the programme.</p>
COMPETENCE – Insight	<p>The learner will be able to maintain integrity and independence in professional judgement and conduct his / her work in a professionally ethical manner.</p>

Full-time candidates will be required to attend and participate in a weekly seminar that will be supported by an electronic element (*e.g.* Dedicated discussion fora, with the Institute's Learning Management System).

The 'Research Seminar' serves as a showcase / dissemination mechanism for basic and advanced research activity undertaken by learners over the duration of the programme. This two-hour weekly seminar will be jointly led by the (current) three-person programme teaching board thus allowing for a diverse range of topics to be presented, discussed, critiqued, evaluated and synthesised. Increased (diverse) staff participation will also encourage sociality and collaboration; desirable skills / abilities which are emphasised in the programme learning outcomes. A significant emphasis is placed on attendance and participation by all as this seminar accounts for 15 credits and serves as the principal mechanism for confirming ongoing learner engagement in scholarly research.

In the early weeks of the programme, each participant will work on the development of an initial research project proposal in a topic or area of interest to them; the scope of which will be determined through seminar discussion and evaluation. It is envisaged that this proposal will be appropriately refined and further scoped to become a formal project / definition of research study in time. The weekly seminar (and online support tools) will be the principal mechanisms facilitating this scoping and refinement exercise.

Participants will be scheduled during seminar time to report on progress-to-date at regular intervals, and advised by the seminar board on required actions, as appropriate. Additionally, all candidates will make formal presentations to the group at defined intervals as follows:

- Literature & Technology Review
- Progress Updates (Iterative)
- "Mock" Oral Defence
- Research Project Prototype Presentation
- Final Oral Examination

It is envisaged that each project will require 450 - 500 hours of effort and must be well documented and verified on completion. The oral examination will be a principal verification mechanism.

Projects will be assessed by the Research Seminar Course Board and by external examiners. The following assessment criteria will normally apply:

- Ability to define the objectives of the project
- Demonstration of the participant's understanding of the subject area
- Technical difficulty of project
- Quality of the research of relevant literature
- Contribution to existing subject knowledge
- Innovation and/or potential commercial impact
- Documentation and delivery of project
- Overall structure and impact of project
- The degree to which the objectives have been met.

Most projects will be computer based and should normally involve a substantial amount of software development. This is in keeping with the overall goals of the course to reinforce participant's existing knowledge and to provide relevant and useful new skills in the area of advanced computing techniques and software development.

RESEARCH METHODS (Workshop)

MODULE NAME	Research Methods (Workshop)
MODULE CODE	RESE Z5003

ECTS CREDITS:	WEEKLY HOURS DURATION	
	Lectures	
	Tutorials	2
	Laboratory	
	Seminar	
	Other	

PRE-REQUISITE MODULES	
CO-REQUISITE MODULES	

INTENDED ASSESSABLE LEARNING OUTCOMES ON SUCCESSFUL COMPLETION OF THIS MODULE:

KNOWLEDGE – Breadth	The learner will be able to demonstrate the application of appropriate research methodologies and techniques within the domain of computing and software systems.
KNOWLEDGE – Kind	The learner will be capable of applying modern research methods appropriate to applied computing research problems / questions. The learner will be capable of discussing current challenges and research foci in selected areas.
KNOW-HOW / Skill – (Range)	The learner will be capable of independently acquiring and assessing relevant knowledge that is contextually appropriate and specific to an applied area of computing research. The learner will be capable of formally exhibiting his / her research capabilities within an area of applied computing.

KNOW-HOW / Skill – (Selectivity)	<p>The learner will be able to apply research and critical thinking skills to a challenging computer-based problem.</p> <p>The learner will be able to evaluate, select and apply standard and customised research tools and methodologies of enquiry.</p>
-------------------------------------	--

COMPETENCE – Context	<p>The learner will be able to demonstrate an appreciation of the professional standards relevant to the process of research in computer science.</p>
COMPETENCE – Role	<p>The learner will be able to participate in peer collaboration and evaluation exercises.</p>
COMPETENCE – Learning to Learn	<p>The learner will be able to critically evaluate his/her own work and research and reflect on the strength, weaknesses and future potential of such work.</p> <p>The learner will be able to demonstrate an understanding of the importance and need for continued personal development in the application of research methods.</p>
COMPETENCE – Insight	<p>The learner will be able to maintain integrity and independence in professional judgement and conduct his / her work in a professionally ethical manner.</p>

RESEARCH METHODS (Workshop) – continued...

4.1 Aims and Objectives

This is a short course on research methods specific to research in Computer Science and Informatics. Attendance is compulsory for all full-time research students on the M.Sc in Computing.

This module is designed to provide students with an introduction to the methods used to carry out a postgraduate research project in computing and related disciplines. It is designed for students from a wide variety of backgrounds and aims to help them to develop critical thinking and to learn research techniques. In particular, it will provide them with the skills that they will need to undertake their own research on the MSc programme.

4.2 Module Outline

- The Philosophy and Nature of Inquiry.
- The Nature and Objectives of Research.
- Research Design and Planning.
- Research Methodologies for Computing and Informatics.
- Choosing and Planning a Research Project
- Writing a Research Proposal.
- Reviewing the Literature.
- Tools of Research.
- Data Collection and Analysis.
- Managing / Achieving Objectives.
- Writing the Research Report.
- Presenting Data, Drawing Conclusions, Identifying Future Work.
- Presenting Research Findings.

Assessment

- Participation, In Class Discussion Sessions + Online Forum
- Literature Review
- Project Plan
- Research Presentation

Module Coordinator

Contact: Dr. Sean Duignan (sean.duignan@gmit.ie)

Last Revision: June 2009.

M.Sc in Computing – Programme Calendar (2010 / 2011)

September 2010

Sep 14th: Term I Commences
Sep 24th: Programme Board Meeting

October 2010

Oct 8th: Research Focus / Proposals Due
Oct 25th: Public Holiday (Monday)

December 2010

Dec 10th: End of Term I
Dec 23rd: College closes (half day) for Christmas

January 2011

Jan 17th: Term II commences Wednesday
Jan 21st: Programme Board Meeting

March 2011

Mar 17th: St Patrick's Day (Thursday)

April 2011

April 15th: Term II Ends (Easter Break)

May 2011

May 2nd: Public Holiday (Monday)
May 9 – 26: Summer Examinations

June 2011

June 13 – 17: Student Consultation Days

August 2011

August 24th: Final Research Papers Due

September 2011

Sept 1 – 2: Final Oral Examinations

Notes

Notes

Notes

M.Sc in Computing (2009 – 2010)

Academic Staff – Contact Details

Name	Role / Responsibility
<p>Dr. Seán Duignan Lecturer Department of Maths & Computing School of Science Galway-Mayo Institute of Technology Dublin Road Galway Ireland Tel: ++353 (91) 742143 email: sean.duignan@gmit.ie</p>	<p>Module Co-ordinator: (Forensic Computing)</p> <p>Module Co-ordinator: (Research Methods)</p> <p>Research Supervisor</p>
<p>Owen Foley Lecturer Department of Maths & Computing School of Science Galway-Mayo Institute of Technology Dublin Road Galway Ireland Tel: ++353 (91) 753161 email: owen.foley@gmit.ie</p>	<p>Module Co-ordinator: (Advanced Databases)</p> <p>Research Supervisor</p>
<p>John Healy Lecturer Department of Maths & Computing School of Science Galway-Mayo Institute of Technology Dublin Road Galway Ireland Tel: ++353 (91) 742604 email: john.healy@gmit.ie</p>	<p>Module Co-ordinator: (Software Engineering)</p> <p>Module Co-ordinator: (Object Oriented Systems)</p> <p>Research Supervisor</p>