Dublin Institute of Technology

BSc (Honours) Computing

BSc (Ordinary) Computing

Higher Certificate Computing

DT211

Programme Document (Part B)

March 2011

This document was prepared by the Programme Committee on behalf of the School of Computing

Table of Contents

Table of Tables

1. Introduction

1.1. Programme background and structure

This is the programme document for the programme in Computing offered by the School of Computing at the Dublin Institute of Technology. This programme leads to the awards of:

- BSc (Honours) in Computing
- BSc (Ordinary) in Computing
- Higher Certificate in Computing

The documentation is prepared in accordance with the DIT Quality Assurance Handbook to satisfy the quality assurance requirement for a Dublin Institute of Technology award.

1.1.1. School of Computing Programmes

The School of Computing is one of five schools in the College of Sciences and Health.

The School offers a range of programmes at undergraduate and postgraduate levels, as well as a variety of short course and minor award programmes. The programmes leading to major awards in the School of Computing are:

Undergraduate Full Time

1. BSc (Honours) Computing (exit option with BSc (Ordinary), Higher Certificate)
2. BSc (Honours) Computer Science (exit option with BSc (Ordinary), Higher Certificate)

Undergraduate Part Time

1. BSc (Honours) Information Systems and Information Technology (exit option with BSc (Ordinary), Higher Certificate)
2. Higher Certificate Computing (Recognition of Prior Learning Option)

Undergraduate International (Franchises)

1. BSc (Honours) Computer Science (exit option with BSc (Ordinary), Higher Certificate)

   Offered by partners in Harbin Institute of Technology, Harbin, China

2. BSc (Honours) Computer Science (exit option with BSc (Ordinary), Higher Certificate)

   Offered by partners in Institute of Finance Management, Dar es Salaam, Tanzania

Postgraduate Full Time

1. MSc Computing (Information Technology)
2. MSc Computing (Knowledge Management)
3. MSc Digital Games

Postgraduate Part Time

1. MSc Computing (Information Technology)
2. MSc Computing (Knowledge Management)
3. MSc Computing (Assistive Technology)
4. MSc Computing (Data Analytics)
5. Postgraduate Certificate in Software Development

1.1.2. School of Computing Staff

The School of Computing has 34 full time academic staff, one full time school administrator and two full time technical staff. All student registration, examination, administration and accounting functions are managed at the level of the campus, covering the five constituent schools of the College of Sciences and Health located in the Kevin St campus of the Institute, as well as a small selection of other schools from other faculties located in the Kevin St. campus. Other administrative functions are operated at the level of the Institute, including the admissions function. Additionally, the College provides a Head of Learning and Teaching to support the development and implementation of relevant activity in that area, and the implementation of quality assurance, and a Promotion and Recruitment Coordinator to support the marketing of the various programmes in the College of Sciences and Health.

The School has strongly developed its research profile in the past decade. Most staff members participate in research to some degree, with some staff members strongly involved in focussed research areas. This has been driven by two factors: recruitment of staff who are active in research; and facilitation of staff members to pursue research towards the achievement of doctoral qualifications.

There has been a very strong influence from staff research on the taught programmes in the school, in addition to the recruitment of postgraduate research students. Within the various programmes, staff research interests have led to the development of focussed elective modules, or focussed streams of modules, in advanced areas of computing, such as:

- Universal Design and Assistive ICT
- Artificial Intelligence
- Computer Games Development
- Business Systems Intelligence
- Security and Cryptography
- Mobile Robotics

The school has a strong track record in teaching and learning research. The school has been extremely successful over the past years in gaining funding under the Institute's Teaching and Learning Projects scheme and has successfully won a Teaching and Learning Fellowship as part of this scheme. All teaching and learning projects directly impact the student learning experience and there is an immediate transfer into learning and teaching practice in the school.

1.1.3. Engagement with Industry

The School has an Industry Liaison committee comprised of two members of staff, who are supported through assignment of a reduced teaching load. The committee is responsible for:

1. Placement of students for work experience.
2. Seeking and maintaining relationships with industrial partners.
3. Providing relevant information on school activity to industrial partners.
4. Identifying and organising research collaboration between the school and industry.
5. Promoting school programmes, in particular part-time programmes, for staff in industry.
6. Organising talks by industry for students on the school's programmes.
7. Liaising with the Institute's career advice service in providing support to students for preparation of CVs, as well as interview preparation and other relevant activity.
8. Recruiting industry practitioners for roles within the school, including final year project supervision, external examiner roles, membership of programme validation and review panels, guest lectureships and any other relevant casual roles that arise from time to time.
9. Organising conferences that involve academic and industrial partnerships.
10. Seeking support from industry for the provision of equipment.

The school currently has strong relationships with the following organisations:

1. Escher Group
2. LanComms
3. Horizon Open Systems
4. Sun Microsystems
5. Hibernia Group
6. HEANet
7. Musgrave

8. CISCO
9. Microsoft
10. SAS Institute

The school has a high proportion of staff with relevant industry experience and excellent contacts in industry. This serves as an input to the committee for the establishment of initial contact, as well as guiding and informing programme development.

The work placement option on this programme has been highly successful for the past year and will continue to be into the future, due to the close and strong links established with industry.

1.1.4. Professional and academic links

The BCS, the Chartered Institute for IT has accredited the B.Sc. in Computing (Hons.) as meeting the educational requirement for Chartered IT Professional (CITP) Further Learning Element registration and partially meeting the educational requirement for CEng registration for a period of five intakes from the 2009 intake up to and including the 2013 intake.

The Programme has strong industrial support from companies such as IBM, Hibernian Evros, Damovo and Fexco. These companies have employed our students as interns and on graduation over the past few years and have provided guest lecturers and advisors on expert panels.

The representative body for the Information and Communications Technology sector (ICT Ireland) and the Higher Education Authority (HEA) proposed and supported a pilot internship programme that is aimed at addressing the problem of the growing shortage of students graduating in ICT programmes.

2. Award

The programme is a four year honours degree programme in Computer Science, leading to the following award, after successful completion of all four years:

BSc (Honours) in Computing

This is a qualification at level 8 on the National Framework of Qualifications.

Students on the programme have the option of exiting (subject to having taken specific modules) the programme after successful completion of years 1, 2 and 3 of the programme and a Stage 3 Individual project, with the following award:

BSc (Ordinary) in Computing

This is a qualification at level 7 on the National Framework of Qualifications.

Students on the programme have the option of exiting the programme after successful completion of years 1 and 2 of the programme, with the following award:

Higher Certificate in Computing

This is a qualification at level 6 on the National Framework of Qualifications.

All modules on this programme are available as Continuing Professional Development short courses at their respective levels on the National Framework of Qualifications.

3. Programme aims and learning outcomes

3.1. Programme Aims

The main aim of the programme is to produce graduates with the necessary skills to take on appropriate professional positions in information technology upon graduation and grow into leadership positions or pursue research or graduate studies in the field; to function as IT developers and network and system support staff in a broad range of commercial and industrial environments. The focus of the course is to enable the graduate to integrate, manipulate and manage different technologies and integrate technologies to provide solutions in organizations and in the public arena. To this end, the student needs to have an in-depth practical knowledge of network theory, programming, data management, storage and manipulation and a solid understanding of software engineering and security.

The aim of the first two stages is to produce a student that has gained the knowledge and skills to program on a variety of platforms and can build and administer a computer and a network, to develop, implement and test a software system using a database and a web-based front end. The student should be aware of legal and ethical issues surrounding IT.

The aim of the third stage is to produce a student who has broadened their knowledge in networks, scripting, programming and databases and have the rudiments of security. They should know how to manage the acquisition of software, hardware and IT services and to develop extra software if appropriate. They should have undertaken and written up a body of work, in the form of report on a project or work placement in a professional environment.

The aim of the fourth stage of the degree is that a student should be able to apply technology to a problem to find a solution, which includes choosing the appropriate technology for the solution and critically analysing their and others' application of technology to a solution, with a view to improving the outcome. The fourth stage expands on their knowledge in two directions: a) It deepens their understanding of technologies, by giving them more and wider practical experience and b) it expands their theoretical understanding of the aims and objectives of the degree area and the philosophy behind the current best practices in the industry.

The programme places a large emphasis on practical skills. While a number of the modules in the programme cover material required for some external international certification bodies, these syllabi include extra theoretical content as would be required in an academic programme. Students graduating from the programme will be familiar with the material required to pass examinations leading to Cisco Certified Network Associate, and Oracle Certification in some Development and Administration areas.

In the third and fourth stages of the programme, some of the students may be accepted into an internship pathway, involving two days per week internship in industry.

3.2. Programme Learning Outcomes

The NQAI (National Qualification Authority of Ireland) guidelines provide a framework for defining the standards and learning outcomes to be achieved by students for a particular award-type. In line with DIT policy, The NQAI (National Qualification Authority of Ireland) guidelines have been used to support the identification and definition of learning outcomes for each of the awards that can be obtained within the DT211 programme: Higher Certificate in Computing (Level 6), B.Sc. (Ord.) in Computing (Level 7) and B.Sc. (Hons.) in Computing (Level 8).

The NQAI framework uses three strands for categorizing programming learning outcomes. For clarity and completeness, these categories have been used to group the learning outcomes for each award.

3.2.1. Higher Certificate in Computing (Level 6)

Knowledge:

On completion of Stages 1 and 2, the student will be able to demonstrate:

- A comprehensive working knowledge of the components and inner operation of a PC and a network and how they interact.
- An understanding of platform specific hardware and software installation and programming.
- An understanding of the diverse context in which IT is used and an appreciation of the importance of communications in IT.
- An understanding of the principles of discrete mathematics, program design and operating systems
- Specialised knowledge of network components, routing protocols and switching configurations.
- A fundamental understanding of how user requirements can be gathered and expressed in a variety of methodologies.
- A working knowledge of operating systems and systems administration.
- An understanding of the place and function of law and ethics in IT business organisations.

Know-how and skills

On successful completion of Stages 1 and 2, the student will be able to:

- Assemble and maintain a personal computer.
- install and use operating systems on different platforms.
- implement basic network routing infrastructures and configure multiple routing protocols to design and configure a simple LAN and internetwork.
- create the requirements section of an RFP and develop system models using a variety of methodologies.
- Design, write, document and test procedural programs, with persistent data.
- Demonstrate the wide applicability of discrete mathematics to computing
- Use both Structural and Object Oriented Programming concepts and implement these concepts in elementary programs.
- Design and implement a robust data model in a relational database and manipulate it using SQL.
- Develop a complete web site accessing data from a database server.

Competence

On successful completion of Stages 1 and 2, the student will be able to:

- Work in teams building computers and programming robots, applying theoretical knowledge gained and assuming responsibility for part of the work and collaborating with others to fulfil their goal.
- Manage and configure routing and switching software to implement networks.
- Evaluate and represent user requirements using a variety of methodologies.
- Interact with an operating system as an effective administrator
- Think logically, and express themselves clearly.
- Adapt published algorithms for use on given problems.
- Develop, and evaluate interactive software using a human-centred approach;

3.2.2. B.Sc (Ord) in Computing (Level 7)

Knowledge:

On completion of Stage 3, in addition to the learning outcomes of Stages 1 and 2, the student will be able to demonstrate:

- A specialised knowledge of the components and organisation of Wide Area Networks.
- A familiarity with security principles and tools for networks and operating systems.
- A good understanding of project management, software acquisition and testing.
- A working knowledge of scripting language, software libraries and deployment utilities.
- Have a working knowledge of the issues involved in designing web servers and of web technologies.
- A familiarity with end-user documentation requirements for deployment and version control.

Know-how and Skills

On completion of Stage 3, in addition to the learning outcomes of Stages 1 and 2, the student will be able to:

- Apply practical skills in the area of design, installation, configuration and security of Wide Area Networks.
- Design, implement and test web-based applications including related software, databases, interfaces and digital media.
- Use security tools for Operating System and Network security.
- Select and utilise the tools available for software development, maintenance, testing and deployment
- Demonstrate a proficiency in scripting, testing and version control on a variety of platforms.
- Identify issues in software, hardware and IT service acquisition and address project management and testing.
- Design and implement a moderately complex database, using advanced features, and design and develop layered procedural transactions to interact with it.
- Build graphic user interfaces using event driven programming in an object-oriented environment.

Competence

On completion of Stage 3, in addition to the learning outcomes of Stages 1 and 2, the student will be able to:

- Accept responsibility for deploying, testing, maintaining and documenting a secure installation using standard project management techniques.
- Investigate, determine and use software libraries, scripting languages and version control appropriate to the system being deployed.
- Accept responsibility for ensuring the security of an installation and for providing end-user documentation.
- Work successfully as part of a team, such as software development team working on medium to complex software programs/projects or a system support team.
- Work individually or in a team on a project or in a professional environment to deliver basic IT deliverables.

3.2.3. BSc (Hons) Computing (Level 8)

Knowledge

On completion of Stage 4, in addition to the learning outcomes of Stages 1, 2, and 3, the student will also be able to demonstrate:

- A deep understanding of the issues surrounding the management and administration of an Enterprise wide system infrastructure, including a distributed database.
- A detailed knowledge of the techniques involved in retrieving, manipulating and converting data from a variety of sources.
- A detailed knowledge of techniques of security provision and the requirement for them.
- A basic understanding of how to research and present research findings.

Know-how and Skills

On completion of Stage 4, in addition to the learning outcomes of Stages 1, 2 and 3 the student will be able to:

- Manage the installation and configuration of software systems to integrate diverse computer applications running on a range of physical and virtual platforms.
- Manipulate, interrogate and transport data using a variety of paradigms and choose the appropriate paradigm for the task.
- Provide security compliance to the enterprise system's infrastructure, using a variety of techniques, including steganography and cryptography.
- To formulate and implement I.T. strategies for an organisation.
- To conduct research into a range of assigned topics and into a specific project area.

Competence

On completion of Stage 4, in addition to the learning outcomes of Stages 1, 2 and 3, the student will be able to:

- Align enterprise systems infrastructure to organisational strategies, taking the purpose, design and context of the system into account and evaluating trade-offs between providing service levels and controlling costs.
- Evaluate available platforms, networks and tools, and identify an optimal configuration to support a set of computing needs.
- Provide and manage a secure software infrastructure up to and including an enterprise-wide secure distributed software infrastructure.
- Research topics relating to data management, enterprise system management and security.
- Take responsibility for a significant task from research, through design, development and deployment, in an area relevant to Information Technology, using appropriate platforms, technologies and methodologies.
- Write on an area of work, to an academic standard, explaining the background, process and outcome of their work, referencing appropriate material.

4. Access, transfer and progression

There are several methods of entry to the programme:

4.1.1. Regular application via the CAO

Minimum requirements for the programme are the Irish Leaving Certificate with a grade D3 or higher in six subjects at ordinary level including:

1. a grade C3 or higher in ordinary level Mathematics or a grade D3 or higher in Higher Level Mathematics,
2. English or Irish at either Ordinary or Higher Level,
3. a grade C3 or higher in at least 2 subjects at Higher Level.

4.1.2. Application with an equivalent qualification

Places may be offered to applicants with a qualification considered equivalent to the minimum requirements, as determined by the Institute.

4.1.3. A mature student application

Places may be offered to mature students who meet certain criteria with respect to suitability, analytical skills and professional experience. Students may be requested to attend for interview.

4.1.4. International Applications

Places may be offered to international applicants in line with the School, College and Institute policy on International applications.

4.2. Entry to later stages of the Programme

Entry to subsequent stages of the programme is normally by progressing from the previous stage. On occasion, it is also possible for students to transfer into the programme following the Institute's Advanced Entry procedures and where the Programme Committee is satisfied as to the equivalency of their qualifications.

5. Duration and General Structure

The full programme has four stages which can be completed in four years of full-time study. Two pathways are available on this programme as illustrated in Figure 1 below. One pathway (Full-Time Pathway) will require full-time attendance by students at DIT. The second pathway (Internship Pathway) will entail full-time attendance by students at DIT for the first two years and an internship programme involving three days per week attendance at DIT and two days per week internship in an IT role in a commercial company in Stages 3 and 4.

5.1. Overall Structure

There are three exit points from the programme:

- On successful completion of stages one and two, students may exit with a Higher Certificate in Computing (NQAI Level 6).
- On successful completion of stage three of the Full-Time Pathway, students may exit with a BSc (Ord) in Computing (NQAI Level 7), provided they have completed a Stage 3 individual project.
- On successful completion of stage four, students exit with a BSc. (Hons) in Computing (NQAI Level 8)

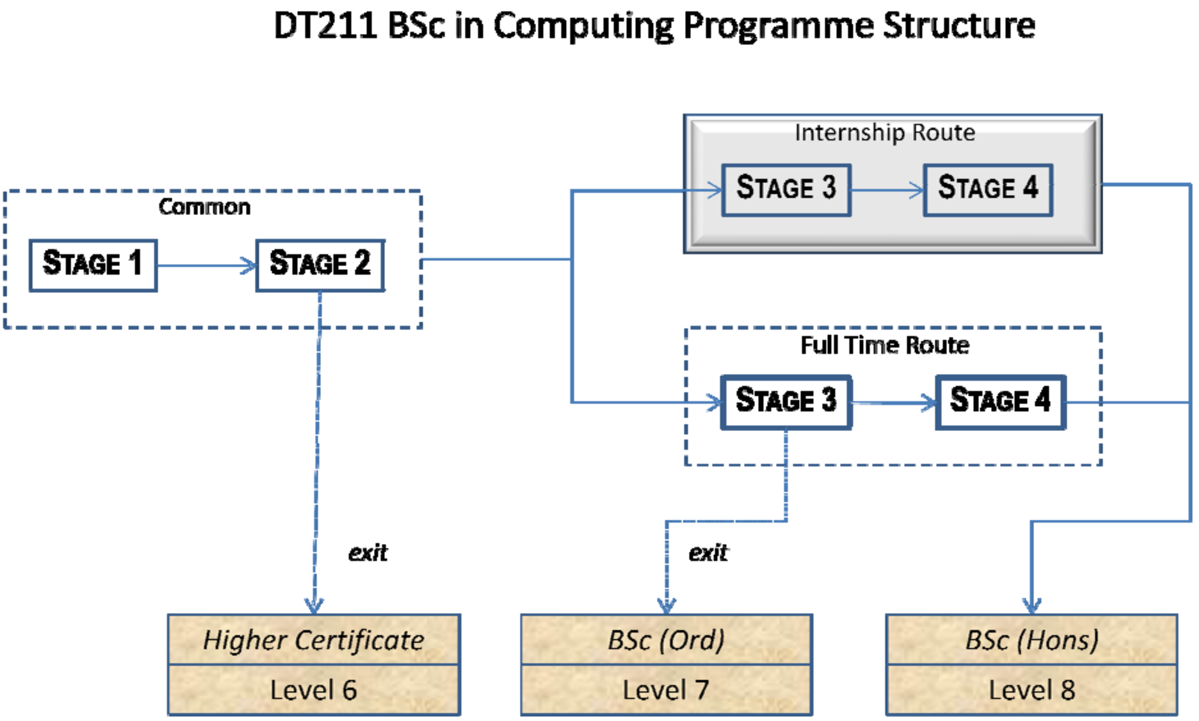

Figure 1 Programme Structure

Stages 1 and 2 are common across both pathways; in the third and fourth stages of both pathways the students study a core group of modules. In the fourth stage the core group of modules includes a substantial project which requires the students to research, design and implement and evaluate an IT project.

On successful completion of stage 2, students indicate a preference for the programme pathway for the third and fourth stages. Availability of places on each pathway will be determined by the programme committee and may vary from year to year.

In addition to the core modules, students following the Internship pathway will take part in an internship programme in industry for two days per week in the third and fourth stages. Students sign a two-year internship contract with an employer and are expected to complete the third and fourth stage of the programme. There is a contingency plan in place, should the employer become unable to honour the contract (see **Error! Reference source not found.**). Further details of Internship Procedures are found within the separately documented **School of Computing Industrial Placement Handbook.** In addition to the core modules, those students following the Full-Time pathway study a selection of optional modules where they can specialise in a particular area. As there is an exit point after stage three for students on the Full-Time pathway, these students undertake a project involving the research, design and implementation of a software system.

The European Credit Transfer System is used in awarding module credits. Each module is awarded the appropriate ECTS credits. The modules and the associated ECTS credits are detailed in the Curriculum section below.

6. Curriculum and assessment.

6.1. Overview

In this section, the contact hours and assessment strategy for each stage in each pathway is shown in a table.

Table 1 covers Stage 1 modules, all of which are core.

Table 2 covers Stage 2 modules, all of which are core.

Stage 3 is covered by three tables;

> Table 3 covers the core modules on the full-time stream (55 of 60 credits).
>
> Table 4 covers the options that are specified for the full-time stream. Students may take ***one*** of these options (5 credits).
>
> Table 5 covers the full suite of modules covered by students on the Internship stream.

Stage 4 is covered by three tables:

> Table 6 covers the core modules to be taken on the full-time stream (50 of 60 credits).
>
> Table 7 covers the pairs of options that are specified for the full-time stream. Students may take ***one pair*** of these options (5x2 credits).
>
> Table 8 covers the modules to be taken on the Internship stream (60 of 60 credits).

6.2. Stage 1 Modules

				Self	Total				
TITLE	**Lectures**	**Labs**	**Tutorials**	**Study**	**Hours**	**exam**	**CA**	**Credits**	
Building a PC	1	2		61	100		100	5	S1
Communications and Personal Development	3			61	100		100	5	S1
Information Technology Fundamentals	3			61	100	70	30	5	S1
Mathematics 1	2		1	61	100	70	30	5	S1
Program Design	2	1	1	48	100	60	40	5	S1
Programming	2	2	1	70	200	50	50	10	S1 & S2
Networking 1 - Fundamentals	2	2		48	100	50	50	5	S2
Introduction to Operating Systems	2	1		61	100	70	30	5	S2
Introduction to Algorithms	2	1	1	48	100	60	40	5	S2
Programming with Persistent Data	1	1	1	61	100	70	30	5	S2
Team Computing	1	3		48	100		100	5	S2
				628	1200			60	
Average hours per week	11.5	7.5	3						

Table 1 Modules in Stage 1 and their assessment

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Three-hour examinations for full-year subjects take place at the end of Semester 2. All Stage 1 modules are core.   A pass mark is 40%. General Assessment Regulations apply.

6.3. Stage 2 Modules

Title	Lectures	Labs	Tutorials	Self Study	Total Hours	% Exam	% CA	Credits	
Databases 1	1	2	1	48	100	50	50	5	S2
Human Computer Interaction	2	2		48	100	60	40	5	S2
Mathematics II	2		1	61	100	70	30	5	S1
Microprocessors	2	2		48	100	50	50	5	S1
Networking 2 - Routing	2	2		48	100	50	50	5	S1
Networking 3 - Switching	2	2		48	100	50	50	5	S2
Object-oriented Programming	2	2	1	70	200	60	40	10	S1 & S2
Operating Systems and System Admin	2	1		61	100	70	30	5	S1
Legal and Professional Issues	2		1	61	100		100	5	S2
Systems Infrastructure and Architecture 1	2	1		61	100	70	30	5	S1
Internet Applications & Web Development	2	2		61	100		100	5	S2
Total for Stage	20	15	4	628	1200			60	
Average hours per week	11	8.5	2.5						

Table 2 Modules for Stage 2 and their Assessment

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Three-hour examinations for full-year subjects take place at the end of Semester 2. All Stage 2 modules are core.   A pass mark is 40%. General Assessment Regulations apply.

6.4. Stage 3 Modules (Full-time stream)

TITLE	Lecture	Labs	Tutorials	Self	Total	% Exam	% C.A	Credits	
Databases 2	1	2	1	48	100	60	40	5	S1
Distributed Objects	1	2	1	48	100	40	60	5	S1
Graphical User Interface Programming	1	2	1	48	100	50	50	5	S1
Network Programming	2	1		61	100	70	30	5	S2
Networking 4 - WAN Technologies	2	2		48	100	50	50	5	S1
Security	2	1		61	100	50	50	5	S2
Software Installation and Maintenance	2	**2**		48	100	60	40	5	S2
Team Project*	1	2		161	200		100	10	S2
Systems Infrastructure and Architecture 2	2	1		61	100	70	30	5	S1
Web Development and Deployment	1	2	1	48	100		100	5	S2
Total for Stage	15	17	4	632	1100			55	
Average weekly hours, without option	7.5	8.5	2						

If an individual project is taken in lieu of this, it will normally operate over Semesters 1 AND 2. Individual projects are only taken by students who choose to exit with a BSc (Ord) in Computing. See syllabus for further details.

Table 3 Core modules for Stage 3 (Full-time pathway) and their assessment

The modules in Table 3 are all core to the full-time stream in Stage 3. In addition to the modules above, the student must take one module from the selection in Table 4. Depending on demand, some options may not run. If an option a student has chosen does not run, students will be informed of this in time to switch to an option that is running and are obliged to do so.

				Self	Total	%			
Module Title	**Lecture**	**Lab**	**Tutorials**	**Study**	**Hours**	**Exam**	**% CA**	**CREDITS**	
Universal Design and Assistive ICT	2	1		61	100	70	30	5	S1
Applied Intelligence	2	2		48	100	70	30	5	S1
Mobile Software Development	1	2	1	48	100	50	50	5	S1
Cloud Computing	2	2		48	100	70	30	5	S1
Computational Mathematics	2		1	61	100	50	50	5	S1
Mobile Robotics	2	2		48	100	70	30	5	S1

Table 4 Optional modules for Stage 3 (Full-time pathway)

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Project assessment will be documented separately.

A pass mark is 40%. General Assessment Regulations apply.

6.5. Stage 3 Modules (Internship stream)

				Self	Total	%	%		
TITLE	**Lecture**	**Labs**	**Tutorials**	**Study**	**Hours**	**Exam**	**C.A.**	**Credits**	
Internship*				100	400		100	20	S1 & S2
Databases 2	1	2	1	48	100	60	40	5	S1
Graphical User Interface Programming	1	2	1	48	100	50	50	5	S1
Network Programming	2	1		61	100	70	30	5	S2
Networking 4 - WAN Technologies	2	2		48	100	50	50	5	S1
Security	2	1		61	100	50	50	5	S2
Software Installation and Maintenance	2	**2**		48	100	60	40	5	S2
Systems Infrastructure and Architecture 2	2	1		61	100	70	30	5	S1
Web Development and Deployment	**1**	2	1	61	100		100	5	S2
Total for Stage	13	13	3	536	1200			60	
Average weekly hours, without option	6.5	6.5	1.5						

Internship students spend two days per week at an industrial placement.

Table 5 Modules for Stage 3 (Internship pathway) and their assessment

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Internship assessment is separately documented in the **School of Computing Industrial Placement Handbook**.

A pass mark is 40%. General Assessment Regulations apply.

6.6. Stage 4 Modules (Full-time stream)

TITLE	Lectures	Labs	Tutorials	Self Study	Total Hours	% Exams	% CA	Credits	
Fourth year project			1	399	400		100	20	S1 & S2
Advanced Data Management	1	2	1	96	100	50	50	5	S1
Systems Integration	2	2		96	100	60	40	5	S1
Advanced Security 1	2	1		97	100	60	40	5	S1
Advanced Security 2	2	1		97	100	50	50	5	S2
Enterprise Systems Infrastructure & Architecture	2		1	97	100	70	30	5	S2
System and Database Administration	1	2	1	96	100	50	50	5	S2
Total for Stage	10	8	4	978	1000			50	
Average weekly hours, without options	5	4	2						

Table 6 Core Modules for Stage 4 (Full-time pathway) and their assessment.

In addition to the modules listed in Table 6, full-time students must take a pair of related modules from the list in Table 7.

Paired options (1 pair to be taken)

Option	TITLE	Lectures	Labs	Self Study	Total Hours	% Exam	% C.A.	Credits	
1	Games Engines 1	2	2	48	100	50	50	5	S1
	Games Engines 2	2	2	48	100	50	50	5	S2
2	Rich Web Application Development	2	2	48	100	70	30	5	S1
	Enterprise Application Development	2	2	48	100	70	30	5	S2
3	Digital Audio	2	2	48	100	70	30	5	S1
	Music Technology	2	2	48	100	70	30	5	S2
4	Geographical Information Systems	2	1	61	100	50	50	5	S1
	Spatial Databases	2	1	61	100	50	50	5	S2
5	Computer Graphics	2	2	48	100	70	30	5	S1
	Image Processing	2	2	48	100	70	30	5	S2
	Average weekly hours for one pair of options	2	1.9						

Table 7 Option pairs for Stage 4 (Full-time pathway) and their assessment

An amendment sanctioned by the Programme Committee in December 2012 sanctioned the following:

- The addition of the module Bioinformatics as an optional module for full-time students.
- Students may choose any of the optional modules in Semester 2, provided they fulfil the pre-requisites for that module. While pairs show modules that complement each other, students are not restricted to choosing the module in Semester 2 that is paired with the module chosen in Semester 1.

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Project assessment will be documented separately.

A pass mark is 40%. General Assessment Regulations apply.

6.7. Stage 4 Modules (Internship stream)

				Self	Total	%	%		
TITLE	**Lectures**	**Labs**	**Tutorials**	**Study**	**Hours**	**Exam**	**C.A.**	**Credits**	
Internship				400	400		100	20	S1 & S2
Fourth year project			1	399	400		100	20	S1 & S2
Advanced Data Management	1	2	1	96	100	50	50	5	S1
Advanced Security 1	2	1		97	100	60	40	5	S1
Enterprise Systems Infrastructure & Architecture	2		1	97	100	70	30	5	S2
System and Database Administration	1	2	1	96	100	50	50	5	S2
Total for Stage	6	5	4	1185	1200			60	

Table 8 Stage 4 (Internship pathway) Modules and their assessment

Two-hour examinations for Semester 1 modules take place at the end of Semester 1. Two-hour examinations for Semester 2 modules take place at the end of Semester 2. Internship assessment is separately documented. A pass mark is 40%. General Assessment Regulations apply.

6.7.1. Project

In the fourth stage of the Programme, all students will be required to carry out a project and to submit a project report. This project report requires the students to use the skills they have developed over the programme to research, design and implement a substantial software system. The pass mark for the project is 40%. The project cannot be compensated against another subject. A student who is not successful in the project must repeat the project the following year and is normally permitted only one further attempt at the project unless decided by the Programme Committee.

Procedures and assessment criteria governing the Project are available from the Project Co-ordinator. These guidelines, procedures and assessment criteria are also available on the Web for easy access.

Students who wish to exit with a B.Sc (Ord) in Computing must complete a Stage 3 individual project. Students may request to do the individual project in Stage 3, instead of the Team Project. Students completing Stage 3 without completing an individual project must take an individual project in order to qualify for an exit with a B.Sc (Ord).

6.7.2. Internship

All students who are successful in all the stage 2 exams are eligible to participate in the Internship module.

Advanced entry students may be considered for the Internship pathway subject to approval by the Programme Committee. The Programme Committee may set a threshold on the numbers accepted for the Internship module. Students may, as one of the criteria, be selected based on previous performance.

The School of Computing has an Industry Liaison Committee, a member of which is the Work Placement and Internship Coordinator whose responsibility is to place students in companies to allow them to complete a 2 year internship. While significant effort will be made to place students, any students who are not placed before the start of the first semester must commence the fulltime stream.

A student who is not successful in the Internship module cannot repeat the Internship.

If the company can no longer sustain the employment of the student, the student may need to return to the full-time stream in Semester 2 of either stage. Internship will be assessed as normal, with ECTS credits dependent on the duration of the Internship already completed for the stage.

Students of Stage 3 can take up SENG3210 Team Software Development Project to make up the remaining 10 ECTS credits.

Students of Stage 4 can take up Semester 2 option modules from the full-time stream, provided they are deemed to have the equivalent of any pre-requisite for those modules. Students may also take Stage 2 Semester 2 option modules from the Degree in Computer Science, where they satisfy the pre-requisites.

Procedures, assessment criteria, and guidelines for the structure and content of all Internship documentation, reports and presentations are included in the School of Computing Industrial Placement Handbook. This document is available from the Industrial Placement Co-ordinator, and is also made available to all students on the Web at http://www.comp.dit.ie/industrialplacement/

6.8. Progression

Progression is normally permitted in accordance with the Institute's General Assessment Regulations. In order to progress from stage to stage, the student should pass, be exempt from, or compensate for all modules in the stage.

Compensation between modules is normally permitted between modules in a stage in accordance with the Institute's General Assessment Regulations (Modular Programmes) except for the following:

1. Students may not compensate for the Team Project module or the individual project module in Stage 3.
2. Students may not compensate for the Project module in Stage 4.
3. Students may not compensate for the Internship modules in Stages 3 and 4.

6.9. Exiting with an Ordinary Degree or a Higher Certificate

It is expected that all students will wish to progress from stage to stage. In order for a candidate to be granted a Higher Certificate or an Ordinary Degree, the candidate must apply in writing to the Examinations office. They may not apply for advanced entry into the following Stage until one academic year has elapsed since graduation.

Students who apply to exit with an Ordinary Degree must take a Stage 3 individual project.

6.10. Requirements to Pass a Module

Where a module is assessed using exam and non-exam assessment methods, in order for a student to pass that module, they must achieve a pass mark when the examination and non-examination marks are combined according to their respective weightings for that module.

Where a module is assessed without examination, the student must achieve a pass mark in the non-examination component of the module in order to pass the module.

6.11. Reassessment of a Module

If a student has not passed a module, they can take the next available assessable component (e.g. supplemental examination) even if this is a component they have already passed, if they would otherwise be denied an opportunity to attempt to progress within the same period of time (they may otherwise need to wait one further academic year to retake the non-examination component).

6.12. Exemptions

Exemptions can be offered to students based on prior learning in line with the School, College and Institute policies on the recognition of prior accredited and experiential learning.

Students who possess certificates from the Cisco Networking Academy Program of Cisco Systems Ltd. may be exempted from some elements of some networks modules on the programme in accordance with the table below. Students who wish to apply for exemption must apply in writing to the relevant stage mentor at the beginning of the academic year and furnish originals of the relevant certificates. Exemptions will be considered by the Exemptions Committee. The Exemptions Committee will be responsible for:

- evaluating the value of applicants' alternative qualifications and related industry experience for the purposes of gaining exemptions to the programme
- conducting applicant interviews where required
- assessing applications for exemptions from students for modules or elements of modules within the programme.

The Committee will consist of three members of staff from the School of Computing, two of whom will be members of the teaching staff for DT211 and the Chairperson of the DT211 programme. The membership of the Committee will be reviewed annually and will be nominated by Head of the School of Computing or his/her nominee. The Chairperson of the Admissions and Exemptions Committee is the Head of the School of Computing or his/her nominee.

Cisco CCNA Certificate for which Exemption may be Granted	Programme Module
CCNA 1: Networking Basics	TECH1401 Networking 1: Fundamentals
CCNA 2: Routers and Routing Basics	TECH2501 Networking 2: Routing
CCNA 3: Switching Basics and Intermediate Routing	TECH2502 Networking 3: Switching
CCNA 4: WAN Technologies	TECH3301 Networking 4: WAN Technologies

Table 9 Networks Modules for which exemptions may be granted

6.13. Awards

Since there are exit points from the programme after stage 2, stage 3 and stage 4, there is a need to calculate a final mark after each of these stages for the award of a certificate, ordinary degree or honours degree.

The final grade for the Higher Certificate will be determined from the averaged weighted marks of the Stage 2 modules.

The final grade for the BSc (Ord) will be determined from the average weighted marks of the Stage 3 modules. The Ordinary Degree project mark will be used in lieu of the Team Software Development project mark.

The final grade for the BSc (Hons) will be determined from the average weighted marks of the Stage 4 modules.

The average weighted mark will be calculated from the marks awarded to each module, weighted by the ECTS credits for that module. The formula used to calculate the final grades for each award is as follows:

$$\frac{\text{Sum (Mark achieved for a module * ECTS credits for the module)}}{\text{Total ECTS Credits for all modules included above}}$$

Any module for which an exemption has been given will not be included in the calculation.

The awards of Higher Certificate and BSc (Ord) may be made with the classifications of Distinction, Merit or Pass in accordance with the schedule set out in Table 18 below:

Average Mark	Classification
>=70%	Distinction
60%- 69%	Merit Upper Division
50%-59%	Merit Lower Division
40% - 49%	Pass

Table 10 Classification of Final Award – Higher Certificate and BSc (Ord)

The award of Degree (honours) may be made with the classifications of First Class Honours, Second Class Honours or Pass in accordance with the schedule set out in Table 19 below:

Average Mark	Classification
>=70%	First Class Honours
60%- 69%	Second Class Honours, Upper Division
50%-59%	Second Class Honours Lower Division
40% - 49%	Pass

Table 11 Classification of Final Award - BSc (Hons)

7. Programme Development Plan

Stage 1 of the revised programme, proposed in this review, will start for all students registering for the first time on the programme in September 2011

Stages 2, 3 and 4 of the revised programme will start for existing students of the programme registering for these stages in September 2011, with some minor adjustments to accommodate prior learning. The following exception will apply:

Students entering stage 3 in September 2011 will complete the IT Law and Professional Issues module at stage 3, as per the previous programme structure, instead of the Databases 2 module, as they will have completed the content of this module in stage 2.

8. Curriculum, examinations and syllabi

This chapter covers the curriculum in detail, giving the syllabus for each module. Each module has a module code that uniquely identifies it. It also has a module title.

In some cases, a student may only undertake a module if he / she has previously studied a pre-requisite module, or validated content similar to the pre-requisite module. Pre-requisite module codes are listed in the module heading.

Each module is worth a multiple of five ECTS credits.

The module is described under the headings:

- Module Author
- Module Description
- Module aim
- Learning Outcomes
- Learning and Teaching Methods
- Module Content
- Assessment Strategy
- Essential reading list
- Supplementary Reading List
- Web and other references
- Further Details
- Date of approval

The modules are presented in the Stage in which they occur. The content of the module descriptor is an indicative of what will be covered in the module.

8.1. Syllabi Stage 1

The following modules are taken in Stage 1:

Module Code	TITLE	Credits	
CMPU1002	Building a PC	5	S1
CMPU1004	Communications and Personal Development	5	S1
CMPU1012	Information Technology Fundamentals	5	S1
CMPU1018	Mathematics 1	5	S1
CMPU1024	Program Design	5	S1
CMPU1021	Networking 1 - Fundamentals	5	S2
CMPU1015	Introduction to Operating Systems	5	S2
CMPU1014	Introduction to Algorithms	5	S2
CMPU1025	Programming	10	S1 & S2
CMPU1028	Programming with Persistent Data	5	S2
CMPU1030	Team Computing	5	S2

Table 12 Modules in Stage 1

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1002*	*Building a PC*

8.1.1. Building a PC

Module author: Damon Berry

Module Description:

The core ingredient of this module is a progressive practical activity where students are given an opportunity to build a personal computer. As each component is added to the computer, the student learns about how that component functions and interacts with the rest of the computer system. Although it does not comprehensibly cover all of the constituent technologies, this module does provide a general and practical overview of the principles of operation of key components of a PC

Module aim

The aim of this module is to introduce students to practical aspects of PC architecture and assembly and to give students sufficient knowledge to help them to maintain and upgrade a personal computer

Learning Outcomes:

On completion of this module, the learner will be able to

- describe the components and peripherals of a functioning personal computer,
- provide an overview of the functioning of the various parts of a computer system
- demonstrate an understanding of interoperation of components and peripherals
- successfully build a computer system from constituent components in a safe and effective manner
- follow best practice in relation to PC construction
- demonstrate understanding of how information is passed between components of a computer system
- demonstrate understanding of operating systems fundamentals and operating system installation
- demonstrate understanding of external PC connections e.g. a simple standalone peer-to peer network

Learning and Teaching Methods:

Lectures, group work, assembly of a PC and installation and basic functionality of operating systems, self-directed learning, quizzes, summarising and vocal presentation of technical documentation.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1002***	***Building a PC***

Module content:

An introduction to Modern PC Architecture, including CPUs, RAM, ROM, CD ROM, DVD ROM, hard disk, PC buses, keyboard, monitor and mouse. The main issues associated with assembling a functioning computer system. Operating system fundamentals and installation, new personal computing technologies.

Module Assessment

Assessment is via 2-3 short assessments (totalling 15%) and one substantial end-of-module written assignment accompanied by an assembled computer with installed operating systems and assessment of students contribution to same (totalling 85%). All assignments will be on topics related to PC architecture, the task of assembling a PC and installing operating systems.

Essential Reading:

Bigelow, S.J., 2001, Troubleshooting, Maintaining and Repairing PCs Osborne McGraw-Hill 1500pp (Hardback), ISBN: 0072132728

Supplemental Reading:

Thompson, R.B, Thompson, B.F., 2003, PC Hardware in a Nutshell, O'Reilly, 874 pp, ISBN: 059600513X

Further Details:

To be delivered in one semester. Contact hours 3 per week; 1 lecture, 2 lab

Date of Academic Council approval …………………………..

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1004***	***Communications and Personal Development***

8.1.2. Communications and Personal Development

Module author: School of Computing

Module Description:

This module is designed to equip computing students with the necessary skills to function in a working environment. To do this, students must be good communicators and be able to clearly express their ideas to a range of stakeholders using a range of computer-based technologies. This part of the module covers both traditional media of communications (such as the writing of business documentation, particularly software specifications, graphical presentations and oral communications) and the use of network communication technologies (such as e-mail, threaded discussion boards and on-line chat), plus appropriate netiquette.

Additionally the module will cover the necessary study skills to cope with third-level content areas. To this end topics on this module will include time management, textbook studying, note-taking, library usage, using on-line resources, reducing test anxiety, improving concentration, learning memory strategies, and exam and assessment preparation.

Module aim

The aim of the module is to help students develop an appreciation of the importance of communications in the computing discipline. In most work-based computing environments, the ideas being dealt with are a combination of organisational and technical concepts all of which are often very abstract and complex, therefore the need for clear and concise communication is vital.

Learning Outcomes:

On completion of this module, the learner will be able to

- Apply effective note taking techniques in lectures
- Describe a range of learning theories and learner types
- Develop and apply an effective time management plan
- Apply effective test taking strategies to objective and essay tests
- Assess personal level of test anxiety and select appropriate strategies for dealing with stress
- Describe the importance of good communication and the problems with plagiarism
- Prepare various documentation to an accurate format
- Communicate effectively using oral, written and internet-based technologies
- Work effectively in teams and groups.
- Prepare and deliver a short presentation

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1004*	*Communications and Personal Development*

Learning and Teaching Methods:

The module will be delivered primarily through lectures and tutorials. The material will be developed in an informal way during lectures. It is envisaged that the students will assimilate much of the material through exercises. Much emphasis will be placed on worked examples and group discussion of exercises.

Because this in intended to be of practical use to the students, a large emphasis will be placed on allowing the students to tryout the approaches described in lectures. Each week, the students will be given a number of exercises that cover material relevant to the implementation of approaches discussed. These exercises get progressively more difficult and will incorporate material learned previously. Tutorials will be used to allow the students to get extra tuition in the more difficult areas. They will also be used to allow the students to ask for any extra help required. The tutorials will incorporate the delivery of additional exercises and examples and provide the student with the opportunity for one-to-one assistance from the supervisor.

Module content:

Introduction to DIT

Importance of Communications; Learning - individually and in groups and teams; Study Skills-individually and in groups and teams

Oral Communication and good presentation skills; Written Communication; Graphical communication; On-Line Communications

Module Assessment

Continuous Assessment - 100%

Three assignments, one in the first term and two scheduled in second term.

An example of the type of assignments is as follows:

Assignment based on company case study

Short research assignment

Oral presentation

Essential Reading: (author, date, title, publisher)

Cottrell, S., 2003, *The Study Skills Handbook*, 2nd ed., UK: Palgrave Macmillan.

Paradis, J.G., Zimmerman, M., 1997, The MIT Guide to Science and Engineering Communication, MIT Press

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1004***	***Communications and Personal Development***

Supplemental Reading: (author, date, title, publisher)

Cottrell, S., 2005, Critical Thinking Skills: Developing Effective Analysis and Argument, UK: Palgrave Macmillan

Tufte, E.R., 1997, Visual Explanations, Graphics Press

Web references, journals and other:

College Life at DIT: http://www.dit.ie/DIT/prospective/welcome/college-life.html

Study Tips: Manage Your Own Learning - http://www.ul.ie/~library/studyskills.html

What are Key Skills About? - http://www.schoolzone.co.uk/resources/articles/whatare_keyskills.asp

Key Skills Framework - http://www.action.ncca.ie/ga/key-skills/key-skills-framework

Further Details:

To be delivered in one semester or year- long.

This is a one semester course with three contact hours per week. Contact hours will consist of a combination of lectures and tutorials.

3 lectures/tutorial

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1012*	*Information Technology Fundamentals*

8.1.3. Information Technology Fundamentals

Module author: Art Sloan, Patricia O'Byrne

Module Description:

This module provides an overview of the discipline of Information Technology (IT) and describes how it relates to other computing disciplines.

Module aim

The aim of this module is to help students understand the diverse contexts in which IT is used and the challenges inherent in the diffusion of this type of technology.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Describe the role of the IT professional as the user advocate.
- Explain data quality and systems security
- Explain how the components of an IT system interrelate.
- Understand the issues of management of complexity in an information technology environment by applying best practices
- Illustrate the use of information and communication technologies to solve problems
- Outline the history of computing technology, the Internet, and the World-Wide Web
- Explain the relationship between IT and related and informing disciplines
- Explain how and to what extent IT has changed various application domains.

Learning and Teaching Methods:

Lectures, self-study, tutorials and any combination of discussion, case study, problem solving exercises, readings, seminars and computer-based learning.

Module content:

- User centeredness and advocacy
- Information assurance and security
- IT systems model
- Management of complexity
- Information and communication technologies: HCI, Networking
- History of computing technology
- Related disciplines: software Engineering, Mathematics and Statistics
- Application domains: science, business, legal issues

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1012***	***Information Technology Fundamentals***

Module Assessment

Methods of assessment to be used to measure the learning outcomes stated above are to be: a written examination and continuous assessment, including one or more of assignment, essay, problem-solving exercise and/or class or lab tests.

Examination:	70%
Continuous Assessment:	30%

Essential Reading:

Parsons, J. and Oja, D. New Perspectives on Computer Concepts 2011, Course Technologies - Cengage Learning, Inc., Kentucky, USA, 2010

Supplemental Reading:

Tajfar, A. Comprehensive Review of Information Technology Fundamentals, Virtualbookworm.com Publishing, Texas, USA

Web references, journals and other:

http://www.informaworld.com

Further Details:

One semester, three contact hours per week

Date of Academic Council approval **…………………………..**

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1024*	*Program Design*

8.1.4. Program Design

Module authors: Richard Lawlor and Shane Mulligan.

Module Description

This module is concerned with program design skills, with particular reference to using flowcharts, pseudocode and programming language constructs to model and design computer programs. Consideration is given as to how problem information might be represented in code or on paper and what program design steps may be performed to arrive at a solution. Abstraction, modularity and top-down design are central to this module.

Module Aims

- To introduce students to program design and the main techniques of program design. In particular, to introduce students to design strategies such as top-down and bottom-up and the techniques of stepwise refinement.
- To introduce the use of pseudocode and flowcharts in program design.
- To convey to students an understanding and appreciation of the power of abstraction whereby the essential information relating to a programming problem is abstracted and mapped onto programming constructs.
- To convey the importance of a well conceived design before rushing into code.

Learning Outcomes

On completion of this module, the learner will be able to:

- Abstract problem information and represent it on paper or an appropriate computing environment.
- Demonstrate a basic competence in the use of a program constructs to solve a problem
- Develop solutions to some elementary program design problems using top down design and stepwise refinement.
- Describe some simple program designs using pseudocode and flowcharts, and then implement the design.

Learning and Teaching Methods

The module will be delivered primarily through lectures, tutorials and laboratory work. The material will be developed in an informal way during lectures. It is envisaged that the students will assimilate much of the material through problem solving and exercises. Emphasis will be placed on worked examples and group discussion of exercises.

Practical work will consist of weekly laboratory sessions. This will also help the students understand how program design concepts can be mapped to a program language constructs.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1024***	***Program Design***

A large emphasis will be placed on allowing the students to tryout the problem descriptions and possible solutions in the laboratory. Exercises will be provided that cover material relevant to the implementation of problem solutions. These exercises get progressively more difficult and will incorporate material learned previously.

Tutorials will be used to allow the students to get help in the more difficult areas and provide students with the opportunity for individual assistance from the supervisor.

Module Content

- Problem Solving, Stages in Problem Solving.
- Data Types and Data Representation.
- Program Constructs.
- Abstraction, Problem Specification, Approaches to Problem Solving and Program Construction, Divide and Conquer, Stepwise Refinement, Top Down Design, Bottom Up Design.
- Recursion. Greatest Common Divisor, Factorial and Fibonacci.
- Pseudocode and Flowcharts in Program Design.
- Linear Data Structures - arrays and lists.

Module Assessment

Assessment will be based on a two hour end of semester written exam and continuous assessment during the semester.

Written exam	- 60%
Continuous Assessment	- 40%

Essential Reading

No specific textbook. Lecture notes and laboratory material as provided by the lecturer.

Supplemental Reading

Maureen Sprankle and Jim Hubbard - Problem Solving and Programming Concepts.

Web References

As specified by the lecturer.

Further Details

One semester (semester 1) module: 2 lectures per week, 1 laboratory hour, 1 tutorial hour

Date of Academic Council approval …………………………..

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1018*	*Mathematics 1*

8.1.5. Mathematics 1

Module author: Shane Mulligan.

Module Description:

This is an introductory module, to give the student a range of basic mathematical skills, and knowledge of how they are applied in various areas of computing and computer science. A number of common mathematical structures and methods will be presented, and their application to represent and solve simple problems. Their application to various areas in computing will be demonstrated.

Module aim

The aim of this module is to give the student the basic knowledge and competence to deal with mathematical concepts and problems that arise in computer science. It will give the student an understanding of discrete mathematics, and demonstrate the wide applicability of discrete mathematics to computing. It will present mathematics as an exact science, and train the student to think logically, and express themselves clearly.

Learning Outcomes:

On completion of this module, the student will be able to:

- Describe number theory concepts and how they can be used in computing.
- Perform the operations associated with sets, relations and functions, and relate practical examples to the appropriate set, function, or relation model.
- Describe symbolic logic and how it can be used to model real-life situations, e.g. represent sentences.
- Define sequences and series, and their definition using iteration.
- Explain and apply the rules for indices, and logs using base 2 and base 10.
- Describe matrices, and their operations, and apply simple matrix algebra.
- Describe and compute basic statistics and their application to data presentation and analysis.

Learning and Teaching Methods:

The learning and teaching methods will consist of lectures and tutorials. Exercises and continuous assessments will be given to ensure that the student understands and masters the material, and to give them practice at representing and solving simple problems. Computer based or online methods may also be used for assessment.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1018***	***Mathematics 1***

Module content:

Number Theory. Division Algorithm, Euclidean algorithm, Prime numbers, Fundamental Theorem of Arithmetic, Modular arithmetic, relevance to computing.

Number Systems and Boolean logic. Logic gates, diagrams, Introduction to Boolean algebra. Binary, decimal and hexadecimal numbers. Operations and conversion. Representing negative numbers, one's/two's complement.

Indices and logs. Arithmetic of, application and manipulation rules.

Set Theory: Definition. Algebra of sets, set operations, subsets, power set, Venn diagrams. Cartesian product, Computer representation of sets.

Relations: Definition, Binary relations. Equivalence relation properties, and application to databases.

Sequences and series. Definitions, sum of integers 1 to N, sum of squares.

Functions: Onto and one-to-one functions. Composition of functions, inverse functions. The floor function and the ceiling function. Linear and quadratic functions. Arithmetic operators and operator precedence. Application of the concept of a function to computer programming and to the computational complexity of algorithms.

Propositional Logic. Propositions, operators, representing English sentences with propositions.

Matrices: Definition, Matrix algebra (addition, multiplication, inverse). Application to representing systems of equations. Storing large data sets. Applications in computing, e.g. computer graphics, and computer representation of relations.

Statistics: Data collection and presentation in tables, stem-plots and histograms. Summarizing and describing numerical data. Measures of central tendency and spread of data, mean, mode, variance, standard deviation. Frequency distribution.

Module Assessment

The Module assessment will be by written Examination (70%), and Continuous Assessment (30%).

Essential Reading: (author, date, title, publisher)

Kenneth H. Rosen, 2003, Discrete Mathematics and its Applications, 5th Edition, McGraw-Hill.

Supplemental Reading:

Seymour Lipschutz, Marc Lipson, 1997, Schaum's outlines Discrete Mathematics 2nd Edition, McGraw-Hill.

Web references:

http://webcourses.dit.ie

Further Details:

3 contact hours per week. To be delivered in one semester; 2 hours lecture. 1 hour tutorial.

Date of Academic Council approval …………………………..

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***10***	***CMPU1025***	***Programming***

8.1.6. Programming

Module author: Programming and Algorithms Group

Module Description:

The module teaches the fundamental principles required to design, write, test and document structured procedural programs.

Module aim

The aim of this module is to

- Teach the fundamentals of procedural programming
- Teach the principles of good program design, implementation, documentation and testing.

Learning Outcomes:

On completion of this module, the learner will be able to

- Apply basic problem solving techniques to design a program using appropriate modules and data structures to specified requirements.
- Implement a program using previously a developed design
- Use an Integrated Development Environment (IDE) proficiently to develop programs
- To understand the compilation/link processes and interpret errors generated
- Design appropriate test data to ensure module and program correctness and robustness
- Debug a program using an IDE and by program tracing
- Write documentation for a program.

Learning and Teaching Methods:

This module will be taught using lectures, practical sessions in the laboratory and tutorials. All theoretical material will be taught in lecture class and this will include the rules and syntax of procedural programming. Each lecture will include many example programs to show the students how the material covered in the lecture is implemented.

Due to the practical nature of programming, a large emphasis will be placed on allowing the students to practice the development of programs in the laboratory. Each week, the students will be given a number of programming exercises that cover all material taught to them in their lecture class.

Tutorials will be used to allow the students to get extra tuition in the more difficult areas. They will also be used to allow the students to ask for any extra help required. The tutorial will incorporate

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***10***	***CMPU1025***	***Programming***

additional exercises and the ability for the supervisor to provide more one-to-one assistance with a student.

A Virtual Learning Environment (VLE) is used extensively in this module.

Module content:

- *Introduction:* What is a program? Source code. Machine code. Editing, Compiling Linking, Debugging. Use of an Integrated Development Environment (IDE). The command line.
- *Basic Data Types:* integer, floating-point and character data and variables.
- *Basic Input-Output:* Display data on a screen. Input data from the keyboard.
- *Programming Structures:* Conditional statements: Boolean values and expressions, logical and relational operators, if-statement, case-statement, compound conditional statements.
- Iterative constructs: while-statements, for-statements and nested control statements.
- *Introduction to Data Structures:* Strings, single-dimensional arrays, two-dimensional arrays, dynamically allocated arrays, user-defined structures, abstract data types, and enumerated data types. Command line arguments.
- *Structured Programming:* functions, parameter passing, returning values, global and local variables, nested functions, reusable code, library functions.
- *Implementing Basic Algorithms:* Summation, counting, numeric operations, swapping, maximum and minimum, simple string and array manipulation.
- *Testing and debugging:* Objectives and principles of testing. Choosing appropriate test data. Testing and debugging strategies. Debugging using an IDE. Debugging using a program trace.
- *Documentation:* Writing user and technical documentation. Style guidelines.

Module Assessment

Assessment of the module is a combination of the following:

Continuous Assessment (50%):

- Individual assignments ,
- Lab tests,
- On-line tests
- In-class written tests

Written examination(50%): One three hour, end of module examination.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***10***	***CMPU1025***	***Programming***

Essential Reading:

Depending on the procedural language used in this module, specific reading lists will be specified in advanced of the start of the module.

Supplemental Reading:

Dependent on the procedural language used.

Web references, journals and other:

Dependent on the procedural language used.

Further Details:

Class size is expected to be 80, broken into groups for tutorials and labs. Semesters: 2 Contact hours per week: Lectures: 2 hours Lab: 2 hours Tutorial: 1 hour

Date of Academic Council approval…………………………..

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1014*	*Introduction to Algorithms*

8.1.7. Introduction to Algorithms

Module authors: Shane Mulligan.

Module Description

This module is concerned with algorithms and data structures, and their implementation in a suitable language. A range of common data structures will be studied, and the algorithms that use these data structures. Students will gain a deeper understanding and competence in the algorithms studied, through their implementation, testing and comparison.

Module Aims

The aim of this module is:

- To introduce the student to a variety of data structures and how they may be implemented.
- To present the student with algorithms that make use of the data structures studied.
- To study the algorithms through the use of paper-based examples.
- To represent algorithms by the use of pseudo-code and/or flowcharts.
- To implement a selection of data structures and algorithms, test them and compare them.

Learning Outcomes

On completion of this module, the learner will be able to:

- Define a variety of data structures and their associated operations.
- Demonstrate a knowledge of standard algorithms by expressing them using pseudo-code.
- Analyse an algorithm in order to compute its complexity/operation count.
- Implement a range of data structures and algorithms, and test them to increase their knowledge and competence.

Learning and Teaching Methods

The module will be delivered primarily through lectures, tutorials and laboratory work. The material will be developed during lectures. It is envisaged that the students will assimilate much of the material through examples and exercises. Emphasis will be placed on worked examples.

Practical work will consist of weekly laboratory sessions. This will also help the students understand how data structures and algorithms can be implemented in a programming language.

An emphasis will be placed on allowing the students to experiment with their implementations, e.g. by changing the input data.

Tutorials will be used to allow the students to get help in the more difficult areas and provide students with the opportunity for individual assistance from the supervisor.

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1014*	*Introduction to Algorithms*

Module Content

- Algorithms, Properties.
- Data Structures, Abstract Data Types.
- Common Data Structures, Linked-lists, Stacks, Queues, Trees.
- Standard algorithms, searching and sorting.
- Analysis of algorithms, computational complexity.
- Implementation and testing of algorithms.

Module Assessment

Assessment will be based on a two hour end of semester written exam and continuous assessment during the semester. The continuous assessment may consist of laboratory tests, practical assignments or a combination of these.

Written exam - 60%

Continuous Assessment - 40%

Essential Reading

No specific textbook. Lecture notes and laboratory material as provided by the lecturer.

Supplemental Reading

Deitel and Deitel – 2007, C How to program, 5th Edition, Prentice Hall.

Web References

As specified by the lecturer.

Further Details

One semester (semester 1) module and each week:

2 lectures per week

1 laboratory hour

1 tutorial hour

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1021*	*Networking 1 - Fundamentals*

8.1.8. Networking 1 - Fundamentals

Module author: Edina Hatunic-Webster

Module Description:

This course serves as an introduction to networking technologies. The course focuses on network terminology and protocols, Local-Area Networks (LANs), Wide-Area Networks (WANs), Open Systems Interconnection (OSI) models, cabling, Ethernet, Internet Protocol (IP) addressing and network standards.

Module aim

To provide the learner with a fundamental knowledge of networking technologies and of the relationships between the different parts of a computer network.

Learning Outcomes:

On completion of this module, the learner will be able to:

- describe the components and organisation of basic computer network installations.
- describe the Open System Interconnection (OSI) reference model layers.
- describe Transmission Control Protocol/Internet Protocol (TCP/IP) model layers.
- explain the purpose and operation of the main networking devices.
- Use a variety of networking utilities and tools

Learning and Teaching Methods:

The on-line course delivery involves a combination of lectures, self-paced study and weekly online continuous assessments.

Module content:

- Network Fundamentals
- OSI Reference Model Layers.
- Networking Media
- Planning and Cabling Networks
- Ethernet
- TCP/IP Protocol Suite and IP Addressing

Module Assessment

Assessment will be through a combination of continuous assessment and a written exam.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1021***	***Networking 1 - Fundamentals***

Marks will be allocated as follows:

Continuous Assessment (50%): The continuous assessment element of this course will focus on the practical aspects of the course.

Written Exam (50%): The written exam will be conducted under normal DIT Exam regulations and will be based on the theory covered during lectures.

Essential Reading:

COMER, Douglas E. "Computer Networks and Internets with Internet Applications", Prentice Hall.

TANNENBAUM, Andrew, S. "Computer Networks", Prentice Hall

LAMMLE, Todd. "CCNA Cisco Certified Network Associate Study Guide, Fifth Edition", Sybex.

Mark A. Dye, Antoon W. Rufi, "Network Fundamentals", CCNA Exploration Companion Guide, Cisco Press, 2007

Supplemental Reading:

Cisco Systems, 2003, "Cisco Network Academy Program: IT Essentials 1: PC Hardware and Software Companion Guide", Cisco Press.

Cisco Systems, 2005, "Cisco Network Academy Program: CCNA 1 and 2 Companion Guide, Third Edition", Cisco Press.

HALSALL, Fred. "Computer Networking and the Internet", Addison Wesley.

THOMAS, Tom., 2004,"Network Security First Step", Cisco Press

Web references, journals and other:

http://www.cisco.com

Further Details:

Duration: Single Semester Module; Contact Hours: 4 hours per week (2 hours of lecture, 2 hours of laboratory work, maximum of 20 students per laboratory.)

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1015*	*Introduction to Operating Systems*

8.1.9. Introduction to Operating Systems

Module author: Ken O'Brien

Module Description:

This module will serve as an introduction to Operating Systems. It provides an overview of the major components of a computer system and their interaction with the systems software. The module provides a fundamental understanding of the concepts of operating systems. Students will also learn how and why operating systems have evolved over years and the impact this has had on modern operating systems. The concepts will be reinforced with practical laboratory exercises in operating systems functionality, user interaction and management. This will be further backed up by a focus on command line interaction with various operating systems. Practical assignments will be given to develop practical operating systems skills. The module will, at a basic level introduce networked, client-server and distributed operating systems to the student. The module will provide the fundamentals for Advanced Operating Systems and the groundwork for other modules in computer science that assume a general understanding of operating systems principles and practice.

Module aims

The aim of this module is:

- to introduce the student to the principles of operating systems design
- to give the students a working knowledge of a modern operating system
- to provide the student with a sound knowledge of the various components and interactions of a modern operating system
- to develop a competency in practical interaction with an operating system

Learning Outcomes:

On completion of this module, the learner will be able to:

- Explain the benefits of an operating system in a computing environment
- List and describe the major components of an operating system and their basic functions
- Discuss the fundamental trade-offs involved in the design of operating systems
- Differentiate between the concept of processes and threads of control
- Classify scheduling policies with examples from different operating systems
- Appraise memory management techniques and virtual memory implementations
- Examine various file systems and illustrate their relationship with the IOCS
- Compare and contrast the strengths and weaknesses of different modern operating systems

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1015***	***Introduction to Operating Systems***

- Discuss networked, client-server and distributed operating systems and how they differ from single user operating systems
- Demonstrate proficiency in command line interaction with various operating systems

Learning and Teaching Methods:

In this module a number of teaching methods will be employed including lectures, practical sessions, tutorials and case studies. At least one industrial seminar may be arranged. Students may be introduced to the use of Virtual Operating System resources. Online student discussion groups, reflective blogs for use immediately after practical sessions and voluntary Q&A sections may also be included.

Module content:

- **Introduction:** Definition of an operating system, abstract views of an operating system, functions of an operating system, event-driven systems, efficiency & system performance goals, evolution of operating system designs, classes of operating systems and examples of operating systems.
- **Process and Threads:** Process and programs, programmers' view of processes, operating systems view of processes, concurrency, process states, thread of control, interacting processes.
- **Scheduling:** Non pre-emptive scheduling policies, pre-emptive scheduling policies, scheduling in practice, real-time scheduling, example scheduling in UNIX, Linux and Windows.
- **Memory Management:** Memory hierarchy, address spaces, static and dynamic memory, memory allocation to a process, continuous memory allocation, non-continuous memory allocation, swapping and relocation, paging, segmentation, paging with segmentation. Virtual memory basics, demand paging, page replacement policies, memory allocation to a process, page faults.
- **File System & IOCS:** Files and file operations, directories and directory operations, pathnames and filenames, multiple file systems, file types, file sharing, links and shortcuts, file locking, file attributes, disk structure, examples of UNIX, Linux and Windows file systems. Architecture of the IOCS**,** device drivers, types of devices, buffering, device driver structure.
- **Multiprocessor Systems**: Multiprocessor systems, multicomputer systems, clients and servers, distributed file systems, distributed processing, introduction to thin client computing.
- **Laboratory Work:** In addition to the lecture material studied in class, a weekly laboratory session focusing on Linux and UNIX-like operating systems will be scheduled. This session will be a hands-on approach to understanding and using the basics of Linux and UNIX-like operating systems. Topics covered include basic Linux commands, working with file systems, process management, the vi editor, working with shells, a brief introduction to shell scripting.

Module Assessment

This module should have a 70% weighting for the examination and a 30% weighting for the continuous assessment.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1015***	***Introduction to Operating Systems***

Essential Reading:

Flynn I.M. & McIver McHoes A. (2010), Understanding Operating Systems, 6th ed., Course Technology

John English, 2005, *Introduction to Operating Systems: Behind the Desktop*, Palgrave McMillian.

D.M. Dhamdhere, 2007, *Operating Systems: A Concept based Approach,* McGraw Hill.

Supplemental Reading:

William Stallings, 2009, Operating Systems: Internals and Design Principles, Prentice Hall.

Web references, journals and other:

http://williamstallings.com/OS/OS6e.html

Further Details:

This module will run over one semester, with three contact hours per week. Two hours for lectures and one hour for practical work.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1028***	***Programming with Persistent Data***

8.1.10. Programming with Persistent Data

Module author: Programming and Algorithms Group

Module Description:

This module introduces the fundamental programming techniques required for the efficient storage and retrieval of data.

Module aim

The aim of this module is to

- Teach the fundamentals of data storage methods and file formats
- Teach the procedural programming techniques to implement various storage methods and formats.

Learning Outcomes:

On successful completion of this module, the student will be able to:

- Distinguish between various data storage methods and formats,
- Choose the most appropriate data storage method for a specified requirement
- Design and write procedural programs to store and retrieve data in an efficient manner

Learning and Teaching Methods:

The module will be delivered primarily through lectures, tutorials, self-directed learning and practical laboratory exercises. A Virtual Learning Environment (VLE) is used extensively in this module.

Module content:

- *File Structure:* bytes, fields, records, attributes, rows, columns.
- *File Types:* ASCII, Binary, Relational
- *File Access:* Serial, Sequential, Random, Indexed, Index Sequential.
- *Data Manipulation*: Creating persistent data, Retrieving persistent data, Updating persistent data, Deleting persistent data.
- *File Design*: Efficient design of file structure and content.
- *Common File Techniques:* Multiple-file matching, sorting, merging, filtering.
- *File Security:* Reading and writing file permissions.
- *Programming with Common File Formats:* Reading and writing using common propriety software formats, e.g. spreadsheet, database, HTML

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1028***	***Programming with Persistent Data***

Module Assessment

Assessment of the module will be as follows:

Continuous Assessment (30%):

- Individual assignment
- Lab test

Written examination (70%): One two hour, end of module examination.

Essential Reading:

Depending on the procedural language used in this module, specific reading lists will be specified in advanced of the start of the module.

Supplemental Reading:

Born, Gunter,1995, *The File Formats Handbook,* Wadsworth Publishing Company,
ISBN 978-1850321170

Further Details:

class size is expected to be 80, broken into groups for labs and tutorials. Semesters: 1

Contact hours: one lecture, one laboratory hour and one tutorial hour.

Date of Academic Council approval …………………………

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
None	*None*	*5*	*CMPU1030*	*Team Computing*

8.1.11. Team Computing

Module authors: John Kelleher

Module Description:

This module introduces the student to the application of computer technology to mobile robotics from a software perspective.

Module aim:

The module will introduce core computer development concepts such as real-time hardware based design constraints, and team based software design and development. The module will use the case-study of developing software for a mobile robot platform to introduce these concepts. As part of the case study the module provides a broad introduction to the field of robotics and covers the basics of mobile robot control and surveys some common application areas. Students will work in teams that compete against one another in response to a challenge that that will test and develop their engineering design skills. The practical components of the course will give students experience in both individual and team based research and problem-solving.

Learning Outcomes:

- Define what a robot system is and discuss the applications of robot technology
- Explain and compare the operations, characteristics and applications of robot sensors and actuators
- Explain and compare different approaches to robot control.
- List the difficulties facing the development of hardware-based real-time software systems.
- Demonstrate the ability to implement software to run on a specific hardware base, for example a robot.
- Demonstrate an ability to work productively in a team.

Learning and Teaching Methods:

The module will be delivered through and mixture of lecture and lab sessions.

Module content:

- *Introduction:* Overview of robot history, classification, advantages and disadvantages and applications of robot technology.
- Introduction to the components of a robot system: power system, actuators, sensors, control system.
- *Actuators:* Review types such as electric, hydraulic, pneumatics, shape memory alloys and electric motors.
- Describe and compare the advantages and disadvantages of each type of actuator.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
None	***None***	***5***	***CMPU1030***	***Team Computing***

- *Sensors:* Contact Sensors, Internal Sensors, Sonar, Radar, Laser Range-Finding, Satellite Based Positioning, Data Fusion, Biological Sensing, Visual Sensors and Algorithms
- *Locomotion systems:* Compare and contrast the advantages and disadvantages of different types of locomotion system: legged versus wheeled system.
- *Robot Control System Architectures:* General Control Structure for Mobile Robots, Perception Action Models, Subsumption Hierarchies, Layered Models, Deliberative Agents

Module Assessment:

This module is assessed using both individual and competitive team-based continuous assessment. The continuous assessment element will be assessed either by means of an individual credit system where individuals gain credits for each laboratory session based on the level of quality and completion of the assigned practical work or via group projects or via a mixture of group projects and laboratory work. Additionally mini-theory tests may be carried out during the laboratory element of the module. The continuous assessment will carry a total mark of 100% for the module.

Essential Reading: (author, date, title, publisher)

Roland Siegwart and Illah R. Nourbakhsh, 2004, “Introduction to Autonomous Mobile Robots”, Bradford Books. ISBN 0-262-19502-X.

Supplemental Reading: (author, date, title, publisher)

- Saeed B. Niku, 2001, Introduction to Robotics, Analysis, Systems, Applications,Prentice Hall
- Newton C. Braga, 2002, Robotics, Mechatronics, and Artificial Intelligence, Newnes
- Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, and Sebastian Thrun, 2005, “Principles of Robot Motion : Theory, Algorithms, and Implementations”, The MIT Press. ISBN 0-262-03327-5
- Gregory Dudek and Michael Jenkin, 2000, “Computational Principles of Mobile Robotics”, Cambridge University Press. ISBN 0-521-56876-5
- John Iovine, 1998, Robots, Androids and Animatrons, McGraw- Hill

Web references, journals and other:

http://www.allaboutcircuits.com/ (last accessed December 2010)

http://people.howstuffworks.com/robot.htm (last accessed December 2010)

Further Details

Duration: 1 semester Contact Hours: 1 lecture hour and 3 lab hours per week

Date of Academic Council approval …………………………..

8.2. Syllabi Stage 2

The following modules are taken in Stage 2:

Module Code	Title	CREDITS	
CMPU2012	Mathematics II	5	S1
CMPU2013	Microprocessors	5	S1
CMPU2014	Networking 2 - Routing	5	S1
CMPU2018	Operating Systems and System Administration	5	S1
CMPU2021	Systems Infrastructure and Architecture	5	S1
CMPU2016	Object-oriented Programming	10	S1 & S2
CMPU2007	Databases 1	5	S2
CMPU2008	Human Computer Interaction	5	S2
CMPU2015	Networking 3 - Switching	5	S2
CMPU2011	Legal and Professional Issues	5	S2
CMPU2010	Internet Applications & Web Development	5	S2
	Total for Stage	60	

Table 13 Modules in Stage 2

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
Mathematics I		***5***	***CMPU2012***	***Mathematics II***

8.2.1. Mathematics II

Module author: Shane Mulligan.

Module Description:

This is a second module in mathematics, to give the student a broad range of basic mathematical skills, and a good knowledge of how they are applied in various areas of computing and computer science. Their application to various areas in computing such as data security, computer networks and artificial intelligence will be demonstrated.

Module aim

The aim of this module is to give the student the necessary knowledge and competence to deal with mathematical concepts and problems that arise in computer science.

An emphasis will be given to relating the mathematical concepts to their particular application areas.

It will give the student an understanding of discrete mathematics, and demonstrate the applicability of the various branches of discrete mathematics to computing. It will present mathematics as an exact science, and train the student to think logically, and express themselves clearly.

Learning Outcomes

On completion of this module, the student will be able to:

- Perform number theory calculations, and relate them to data security techniques.
- Demonstrate the application periodic functions in data communication.
- Use predicate logic to represent real-life situations, e.g. represent sentences, and draw logical conclusions.
- Define functions and demonstrate their applications in computing.
- Calculate probabilities of events, end expectations of random variables. Apply the tools of probability to solve problems in computing, e.g. analysis of algorithms.
- Describe queuing theory and its application to networks.
- Define graphs, their properties and their various applications in computing.

Learning and Teaching Methods:

The learning and teaching methods will consist of lectures and tutorials. Exercises and continuous assessments will be given to ensure that the student understands and masters the material, and to give them practice at representing and solving simple problems. Tutorials will give the student the opportunity to consider particular problems, or to clarify difficult concepts. Computer based methods may also be used for assessment.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
Mathematics I		***5***	***CMPU2012***	***Mathematics II***

Module content:

More Number Theory : Modular arithmetic, discrete logarithms, application to encryption.

Predicate logic. Predicates, quantifiers, use in data representation and deductive reasoning.

Functions: Hash functions, random numbers, iteration.

Periodic functions: maths for networks, waves, frequency domain, Fourier components.

Probability and statistics. Probability spaces, independence and dependence, repeated trials, random variables, expectation, probability distribution.

Queuing theory. Principles and application to networks.

Graph theory. Paths, cycles, special graphs, graph algorithms.

Module Assessment

The Module assessment will be by written Examination (70%), and Continuous Assessment (30%).

Essential Reading: (author, date, title, publisher).

Seymour Lipschutz, Marc Lipson, 1997, Schaum's outlines Discrete Mathematics 2nd Edition, McGraw-Hill.

Kenneth H. Rosen, 2003, Discrete Mathematics and its Applications, 5th Edition, McGraw-Hill.

Supplemental Reading

Murray R Spiegel, Larry J Stephens, 1999, Schaum's outlines Statistics 3rd Edition, McGraw-Hill.

Web references:

http://webcourses.dit.ie

and Lecturer's web page.

Further Details:

4 contact hours per week. To be delivered in one semester.

2 hours lecture and 1 hour tutorial.

Date of Academic Council approval ……………………….....

Pre-Requisite Modules code(s)	*Co-Requisite Modules code(s)*	*ECTS Credits*	*Module Code*	*Module Title*
		5	*CMPU2013*	*Microprocessors*

8.2.2. Microprocessors

Module authors: Frank Duignan, Richard Hayes

Module Description:

This module introduces the students to hardware and software aspects of microprocessor and microcontroller system development.

Module aim

The aim of this module is to provide students with an understanding of the inner operation of computers and the way in which they interact with external devices.

Learning Outcomes:

On completion of this module, the learner will be able to:

(1) Design, write and debug simple assembly-language programs for the 80x86 family of microprocessors.

(2) Design, write and debug simple C-language programs for the 80x86 family of microprocessors and a microcontroller.

(3) Perform arithmetic in the binary and hexadecimal number systems.

(4) Explain the role played by (and limitations of) the stack in a high level programming language.

(5) Explain the operation of simple circuits that interface microprocessors to external devices.

(6) Outline the operation of microprocessor system elements such as RAM, ROM, Timers and communications peripherals.

(7) Discuss the encoding of different types of computer data (e.g. ASCII characters, Unicode, floating point numbers).

(8) Discuss interrupt handling in microprocessor systems.

Learning and Teaching Methods:

Lectures, laboratory work, self-learning.

Module content

- Analogue systems vs. Digital systems
- The binary and hexadecimal number systems, signed and unsigned numbers.
- Microprocessor core elements: Registers, flags, calculation units, buses, RAM, ROM, Parallel I/O ports.
- Microcontroller programming : registers, memory map, I/O port mapping.
- Interfacing with actuators and sensors
- Analogue to digital and digital to analogue conversion
- Introduction to interrupts.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS Credits***	***Module Code***	***Module Title***
		5	***CMPU2013***	***Microprocessors***

- Introduction to serial communications
- Writing simple 80x86 assembler programs, addressing modes, decision making, looping.
- Calling subroutines and the behaviour of the stack.
- Relating simple C-programs to Assembler programs : Allocation of variables, assignment of values to variables, looping, decision making.
- System calls.
- The C-calling convention.
- Mixed language programming
- Code optimization

Laboratory programme

There are three sections in the laboratory programme:

- Microcontroller labs (4 weeks)
- x86 labs (4 weeks)
- Mini project (4 weeks). The mini project is typically a microcontroller application which includes input/output and structured programming.

Module Assessment

There are two main sections to the module mark:

Section 1:

- Two online/automated tests, one dealing with the 80x86 family of processors, the other with a microcontroller. Each accounts for 10% of the total module mark.
- Laboratory reports account for 20% of the module mark.
- Mini project work accounts for 20% of the module mark.

Section 2:

- The remaining 50% of the module marks are assigned to an end of module written exam which covers all of the course material. Students are expected to answer 3 questions out of a total of 4.
- Section 1 accounts for 50% of the total module mark.
- Similarly, Section 2 accounts for 50% of the total module mark.

Assessment of specific learning outcomes:

- (1),(2),(5),(8) Assessment based on performance in laboratory
- (3) to (8) Assessed using online tests and end of module test.

Pre-Requisite Modules code(s)	***Co-Requisite Modules code(s)***	***ECTS*** ***Credits***	***Module Code***	***Module Title***
		5	***CMPU2013***	***Microprocessors***

Essential Reading: On-line class notes.

Supplemental Reading:

Brey, Barry B., 1994 The Intel microprocessors : 8086/8088, 80186, 80286, 80386, and 80486 architecture, programming and interfacing. Prentice Hall.

Web references, journals and other:

Microcontroller datasheets and programming guides will be available on module website.

Students will also be expected to download, test and modify various sample programs from the module website.

Further Details:

Module to be delivered over one semester consisting of 2 hours class work, 2 hours laboratory work and self directed learning.

Self directed learning: Students will be expected to complete laboratory reports and prepare for other formal assessment components.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU1021		5	CMPU2014	Networking 2 - Routing

8.2.3. Networking 2 – Routing

Module author: Paul Doyle

Module Description:

This course serves as an introduction to networking routing protocols and concepts. The course focuses on network routing protocols both static and dynamic showing how the routing table is populated and interpreted. Static Routing and Dynamic Routing is covered with emphasis on a number of popular dynamic routing protocols. The module is structured to cover both fundamental theory and practical laboratory experience in building and diagnosing issues related to network routing.

Module aim

The goal is to develop a practical understanding of how a router learns about remote networks and determines the best path to those networks. This course includes both static routing and dynamic routing protocols.

Learning Outcomes:

On completion of this module, the learner will be able to

- Explain the theory behind CIDR and VLSM
- Demonstrate how to manage router software
- Configure multiple routing protocols.
- Implement basic network routing infrastructures

Learning and Teaching Methods:

The on-line course delivery involves a combination of lectures, self-paced study, weekly online continuous assessment and practical laboratory sessions with both simulators and physical routing equipment.

Module content:

- Static Routing
- Dynamic Routing
- Distance Vector Routing
- RIP V1 and V2
- VLSM and CIDR
- EIGRP
- Link-State Routing

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU1021		5	CMPU2014	Networking 2 - Routing

- OSPF

Module Assessment

Assessment will be through a combination of continuous assessment and a written exam.

Marks will be allocated as follows

- Continuous Assessment (50%): The continuous assessment element of this course utilises the CCNA online assessment facility.
- Written Exam (50%): The written exam will be conducted under normal DIT Exam regulations and will be based on the theory covered during lectures.

Essential Reading:

- Cisco Systems, 2009, "CCNA Exploration Course Booklet: Routing Protocols and Concepts, Version 4.0" Cisco Press
- COMER, Douglas E. 2001, "Computer Networks and Internets with Internet Applications", Prentice Hall.
- HALSALL, Fred. 2005, "Computer Networking and the Internet", Addison Wesley.
- TANNENBAUM, Andrew, S. 1996, "Computer Networks", Prentice Hall
- LAMMLE, Todd. 1998, "CCNA Cisco Certified Network Associate Study Guide, Illustrated Edition", Sybex

Supplemental Reading:

- Cisco Systems, 2004, "CCNA 1 and 2 Lab Companion, Revised (Cisco Networking Academy Program) (3rd Edition), Cisco Press
- HALSALL, Fred. "Computer Networking and the Internet", Addison Wesley.
- THOMAS, Tom., 2004,"Network Security First Step", Cisco Press

Web references, journals and other:

- http://www.cisco.com
- http://www.cisco.com/web/learning/netacad/index.html
- http://www.cisco.com/en/US/learning/netacad/course_catalog/CCNA.html

Further Details:

Duration: Single Semester Module; Contact Hours: 4 hours per week (2 hours of lecture, 2 hours of laboratory work.)Class Size: Labs should run with a maximum of 24 students per lab supervisor

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2018	Operating Systems and System Administration

8.2.4. Operating Systems and System Administration

Module author: **Ken O'Brien**

Module Description:

This module serves to extend the learner's knowledge of operating systems design and operation. It introduces students to the principles and practice of Systems Administration. The concepts are re-enforced with practical laboratory exercises in operating system management. Practical shell programming assignments are also given to develop practical operating systems skills.

Module aim

The aim of this module is to extend the students' knowledge of the principles of operating system design and to develop in them a working knowledge of operating systems and systems administration.

Learning Outcomes:

On completion of this module, the learner will be able to:

- describe the functions of the major components of an operating system
- describe the interactions between these components
- describe different types of operating system
- control the behaviour of an operating system through a command interface
- describe the fundamental tasks of a UNIX or a Windows systems administrator.
- perform basic system administration tasks on operating systems
- describe the user and group management mechanisms and tools on Windows and UNIX systems
- describe the system start-up and shutdown processes on a UNIX system.
- describe and modify the start-and shutdown order of services on a UNIX system.
- install configure and remove software systems on a UNIX platform
- automate simple system management functions by writing shell scripts.
- appreciate and evaluate of the impact of accessibility requirements on O/S functionality

Learning and Teaching Methods:

Lectures will be used to present the material and pace the learning process throughout the module. Laboratory exercises will be used to re-enforce the learning experience.

Practical assignments will be given throughout the module to allow students to gain experience of operating system programming and system administration.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2018	Operating Systems and System Administration

Module content:

Definition of an operating system, functions of an operating system, event-driven systems, protection mechanisms, virtual resources, evolution of operating system designs, examples of operating systems.

Using an Operating System: the programmer's view, event-driven programming, exceptions, the shell, graphical shells.

Filesystems: files and directories, pathnames and filenames, multiple filesystems, file types, filenames, file versions, links and shortcuts, file locking, file attributes, disk structure, backup systems, example filesystems.

processes and threads: threads, Interprocess communication, synchronisation, mutual exclusion, deadlock, livelock, starvation, priority inversion.

Memory Management: address spaces, virtual memory, demand paging, swapping, page tables, shared memory, page faults, copy-on-write, segmentation.

Hardware Support: computer operation, user and kernel modes, kernel memory, system calls, memory management hardware, caching, power management, hardware failure.

The kernel: kernel responsibilities, implementing threads, kernel objects, process management, scheduling,exception handling, memory management, kernel processes, daemons.

Device Drivers: The I/O subsystem, device drivers, types of device, buffering, device driver structure, Linux device drivers, Windows device drivers.

Multiprocessor Systems: Multiprocessor systems, multicomputer systems, clients and servers,

Role of a System Administrator, boot process, controlling processes, filesystem management, user management, system security, data security.

Laboratory Work: In addition to the lecture material studied in class, a weekly lab session focusing on the UNIX operating system will be scheduled. This strand will be a hands-on approach to understanding and using the basics of the UNIX operating system, including shell script programming. Operating Systems and Accessibility: Legal and Economic Mandate, Vendor Initiatives, API bottleneck, Clipboard issues.

Module Assessment:

Written examination 70%. Continuous assessment 30%.

Essential Reading:

English J. (2003), Introduction to Operating Systems, Palgrave Macmillan ISBN 0-333-99012-9

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2018	Operating Systems and System Administration

Supplemental Reading:

Flynn I.M. & A. McIver McHoes A. (2010), Understanding Operating Systems, 6th ed., Course Technology

Dalheimer M., Welsh M. (2006) Running Linux 5th ed. O'Reilly

Wirzenius L. Oja J. (2004) The Linux System Administrator's Guide, Linux Documentation Project

Stallings W. (2009), Operating Systems: Internals and Design Principles, 6th Ed. Prentice Hall

Further Details:

2 lectures and 1 laboratory per week, delivered over one semester

Date of Academic Council approval ……………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2021	System Infrastructure and Architecture I

8.2.5. System Infrastructure and Architecture I

Module author: Andrea Curley and Patricia O Byrne

Module Description:

One of the roles of the IT professional is to design and build systems and integrate them into an organization. This module develops the skills to elicit, document and validate requirements through a requirements engineering process. This module also aids students in the choice of a suitable systems development methodology for a specific system. This module is a precursor to *Systems Infrastructure & Architecture 2*.

Module aim

The aim of this module is to:

- Identify different types of requirements through various requirements modelling techniques and convert these requirements into use-cases.
- Gain a detailed knowledge of use-case modelling and how it affects the rest of a project life cycle.
- Investigate numerous systems development methodologies.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Compare and contrast the various requirements modelling techniques for non-functional and functional requirements.
- Explain and give examples of use cases.
- Select appropriate user tasks for the application under consideration.
- Create the requirements section of an RFP.
- Develop system models as part of requirements engineering.
- Compare and contrast a number of software development methodologies.

Learning and Teaching Methods:

This module is taught through lectures and exercises in class combined with practical application through use of web and CASE tools.

Module Content:

Requirements: Stakeholders of a system and their needs; requirements modelling techniques; non-functional and functional requirements; roles played by external users of a system; requirements gathering in a system development lifecycle; system models; appropriate deliverables for this phase of the lifecycle; requirements section of an RFP.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2021	System Infrastructure and Architecture I

Use-Case Model: Use cases; types of event flows in a use case and under which conditions they occur; how use cases drive testing throughout the system lifecycle.

Methodologies: Overview and appraisal of methodologies such as Rational Unified Process, Information Engineering, Jackson Systems Development, Soft Systems Methodology, Agile Development.

Module Assessment:

Written Examination – 70%

Continuous Assessment – 30%

Essential Reading:

Sommerville, I. (2011), Software Engineering 9, Ninth Ed., Pearson.

Supplemental Reading:

Avison & Fitzgerald (2006), Information Systems Development, Fourth Ed. McGraw-Hill

Maciaszek, L.A. & Liong, B. L. (2005). Practical Software Engineering: A Case Study Approach. Pearson.

Skidmore, S. & Eva, M. (2004), Introducing Systems Development, Palgrave and MacMillan.

Pressman R.S. (2010), Software Engineering: A Practitioner's Approach, Seventh Ed., McGraw-Hill.

Further Details:

contact hours. To be delivered in one semester. 3 contact hours per week

Date of Academic Council approval ....

1. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU2016	Object Oriented Programming

8.2.6. Object-Oriented Programming

Module author: Programming and Algorithms Group

Module Description:

This module develops the students' programming and problem solving skills, progressing from procedural to object-oriented programming. Topics include the principles, practices, and applications of programming in object-oriented environment with applications to business and scientifically oriented problems. The techniques and language features of object-oriented design are implemented in programming projects. Emphasis is placed upon development of well-designed, efficient, maintainable object-oriented software.

Module aim:

The aim of this module is that the student become familiar with Object Oriented Programming concepts and implement these concepts in elementary object oriented programs.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Demonstrate an understanding of the underlying principles and concepts of Object-Oriented Programming
- Use UML in the design of OOP programs
- Write programs using Object-Oriented programming language
- Extensively testing using debugging tools to remove runtime errors from a program
- Document an Object-Oriented program
- Demonstrate an understanding of the advanced principles and concepts of Object Oriented Programming
- Design and implement object oriented programs using advanced Object Oriented constructs and design patterns
- Persistent objects: Formatted file input and output, direct file input and output
- Construct Program Libraries.

Learning and Teaching Methods:

- Lectures with demonstrations, Tutorials and / or Laboratory practicals based on lectures and tutorials
- A Virtual Learning Environment (VLE) is used extensively in this module.

1. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU2016	Object Oriented Programming

Module content:

- Comparison of Procedural Programming with OOP
- Review of procedural programming. Problems with procedural programming and the need for OOP
- Object-Oriented Principles and Concepts
- Introduction to class diagrams and OOP concepts.
- Relationships, Inheritance, Multiple Inheritance, Abstract Classes.
- Object Oriented Programming Language Constructs
- Abstract data types, classes, objects, messages, Instance variables, methods, encapsulation, private and public access, class variables, constructors, class interface, class implementation.
- Classes and objects, private and public class members, constructors, initialisation list, static data members, overloading, inline, separation of interface and implementation.
- Function overloading. Operator overloading.
- Destructors.
- Virtual functions and friend functions.
- Composition.
- Inheritance: Types of Inheritance, Construction, Destruction, Multiple Inheritance.
- Polymorphism.
- Abstract Classes.
- String class and character arrays. Pointers and dynamic memory.
- Generic Types, Static and Dynamic Binding, Polymorphism, Overloading.
- Exception handling.
- Streams and files.
- Templates: functions and classes. Genericity.
- Collections frameworks.
- Program Libraries.
- Testing techniques for object oriented programs.
- Laboratory Work:
- Design, code and test a series of object-oriented programs to re-enforce lecture content.

1. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU2016	Object Oriented Programming

Module Assessment:

Assessment of the module is a combination of the following:

Continuous Assessment (60%):

- Individual assignments
- Lab tests
- On-line tests
- In-class written tests

Written examination (40%): One three hour, end of module examination.

Essential Reading:

Depending on the language used in this module, reading lists will be specified in advance of the start of the module.

Supplemental Reading:

Dependent on the language used in this module.

Further Details:

Maximum class size is expected to be 80, broken into groups for labs and tutorials, Semesters: 2

Contact hours per semester: Lectures: 2 hours Lab / Tutorial : 2 hours

Date of Academic Council approval ....

2. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2007	Databases 1

8.2.7. Databases I

Module author: Patricia O'Byrne and Databases Group

Module Description:

The student will be introduced to the concepts and rationale behind the relational database. Data modelling and database design will be explored and the student will learn how to create and manipulate data in a relational database, ensuring data and relational data integrity. The student will apply the concepts to a case study.

Module aim

The aim of this module is to…

- Introduce the student to the concepts, justification of and rationale behind databases.
- Give the student the modelling skills to design and implement a robust data model.
- Equip the student with the ability to define and manipulate data in a relational database.

Learning Outcomes:

On completion of this module, the learner will be able to…………….

- Describe and justify the rationale behind the use of relational database management systems.
- Demonstrate an understanding of the desirable features of a database management system and how they are achieved.
- Identify and distinguish between data and meta-data, and the concepts of keys.
- Design a data model suited to a business application and implement it in a relational database.
- Define tables and views with appropriate constraints to ensure data integrity and relational integrity.
- Manipulate the data in a relational database using DDL and DML aspects of SQL.
- Apply knowledge of SQL to real-world query problems.

Learning and Teaching Methods:

This module will be delivered over four hours per week for one Semester. This will include two hours laboratory session. During the lecture, new material on the theories and practices in relational databases will be disseminated. Techniques will be applied in the practical laboratory sessions, including the use of a case study. Students will work individually or in teams, as appropriate.

2. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2007	Databases 1

Module content:

Concepts and rationale

- Discuss the content desirable features and rationale behind a database.
- Determine the data requirements for an application, judging whether given material is information, data or meta-data.

Data modelling

- Interpret, design and create Entity Relationship diagrams.
- Explain the relationship between functional dependencies and keys and give examples.
- Define entity integrity and referential integrity and give examples of user defined integrity constraints.
- Select appropriate business rules for a given scenario and apply them to the model.
- Describe the relationship between a logical model and a physical model and create both.
- Use a CASE tool to generate and reverse engineer logical and physical models.

SQL

- Create, Alter and drop single and related tables and other objects, using appropriate data types and constraints.
- Populate tables and modify and remove rows from tables, being cognizant of constraint issues.
- Formulate and test queries to return selection and projection, using functions to enhance returned data.
- Use relational operators and INNER and OUTER JOINs to query multiple tables.
- Use aggregation and sub-queries to return multi-dimensional data.
- Formulate and test simple transactions to update persistent data in the database, using COMMIT and ROLLBACK as appropriate.

Module Assessment

Module will be assessed by both non-exam and examination assessment. Students must pass the entire module. Practical work 50%, written examination 50%.

Essential Reading:

Elmasri & Navathe, 2010, 'Fundamentals of Database Systems' 6th Ed., Addison Wesley.

2. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2007	Databases 1

Supplemental Reading:

Fehily, C. 2008 ‘SQL: Visual Quick Start Guide’ 3rd Ed., Peachpit Press
Watson & Ramklass, 2008, ‘SQL Fundamentals 1 Exam Guide’, Oracle Press

Web references, journals and other:

Some references will vary, depending on the database being used.

www.oracle.com

www.sqlzoo.net

Further Details:

e.g. laboratory sessions limited to 20 – 25 by lab size. 4 contact hours per week. To be delivered in one semester.

Date of Academic Council approval …………………………

3. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2008	Human Computer Interaction (HCI)

8.2.8. Human Computer Interaction (HCI)

Module author: Andrea Curley and Dave Carroll

Module Description:

Human Computer Interaction (HCI) emphasises the significance of good interfaces and the relationship of interface design to successful human interaction with computer systems. This module provides the theoretical knowledge and practical experiences in the fundamental aspects of analysing, designing, prototyping, and evaluating user interfaces based on sound HCI principles.

Module aim

The aim of this module is to:

- Enable students to understand human behaviour with interactive objects;
- Provide students with the skills to develop, and evaluate interactive software using a human-centred approach;
- Provide students with an appreciation of human diversity and the need for accessibility and its implications for HCI design.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Discuss the background of HCI with its underpinnings from psychology and cognitive science;
- Select appropriate I/O devices and interaction styles to support users and their tasks;
- Employ a range of user centred design techniques which use established design principles and methodologies to solve HCI problems;
- Employ a range of low-fidelity and high-fidelity prototyping techniques;
- Employ a range of evaluation techniques;
- Discuss accessibility issues and their implications for user interface design.

Learning and Teaching Methods:

This module will be delivered using a combination of lectures and practical laboratory sessions. The main theoretical material will be delivered in lectures. The practical work will analyse design, prototype and evaluate a user interface will be carried out in the laboratory sessions.

Module Assessment

60% - written examination**;** 40% - continuous assessment

3. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2008	Human Computer Interaction (HCI)

Module content

- *Fundamentals of Human Computer Interaction* Scope and interdisciplinary nature of HCI; human capabilities; user groups.
- *Cognitive Foundations* Metaphors; mental models; perception; attention; memory; learning.
- *User interface technology* Input and output devices, interaction styles.
- *User-centred design process* HCI design process vs. software engineering design process; usability guidelines, principles and theories
- *User interface design and prototyping* User centered design; participatory design; screen design; low-fidelity, medium-fidelity and high-fidelity prototyping design principles and rules.
- *User interface evaluation* Usability specifications. Measurement criteria. Usability evaluation; cognitive walkthroughs, heuristic analysis, expert review, think aloud methods.
- *Usability issues and accessibility* Ease of use. Principles of universal design. Usability standards. HCI standards. Accessibility requirements. Implications for HCI.
  Legal imperative for accessibility.
- *Major Application Areas* Safe vs. hostile user environments: end-user interfaces, computer supported co-operative work, e-commerce, mobile computing, ubiquitous computing.

Essential Reading:

- Preece, J. et al (2002), *Interaction Design,* John Wiley & Sons.
- Norman, D. (2002,) *The Design of Everyday Things,* Basic Books.

Supplemental Reading:

- Benyon, D. (2010), Designing Interactive Systems, Pearson.
- Shneiderman, B. & Plaisant, C. (2010), Designing the User Interface, Pearson.
- Dix, A., Finlay,J., Abowd, G., & Beale, R. (2003), Human Computer Interaction, 3rd Ed, Prentice Hall.

Web references, journals and other:

- Jakob Nielsen's Website http://www.useit.com/
- Donald Norman's Website http://www.jnd.org/
- Ben Shneiderman's Website http://www.cs.umd.edu/~ben/
- ACM SIGCHI http://www.sigchi.org/
- BCS HCI http://www.bcs.org/server.php?show=nav.14296

3. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2008	Human Computer Interaction (HCI)

Further Details:

Four contact hours per week, two hours of lectures and two hours of labs. To be delivered in one semester.

Date of Academic Council approval ………………………….

4. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2014		5	CMPU2015	Networking 3 - Switching

8.2.9. Networking 3 – Switching

Module author: **Paul Doyle**

Module Description:

This course focuses on LAN switching and wireless LANs. The goal is to develop an understanding of how a switch communicates with other switches and routers in a small- or medium-sized business network to implement VLAN segmentation. This course focuses on Layer 2 switching protocols and concepts used to improve redundancy, propagate VLAN information, and secure the portion of the network where most users access network services. The module is structured to cover both fundamental theory and practical laboratory experience in building and diagnosing issues related to network routing.

Module aim

The goal is to develop an understanding of how switches are interconnected and configured to provide network access to LAN users and to enable students to integrate wireless devices into a LAN.

Learning Outcomes:

On completion of this module, the learner will be able to

- Explain Switch Concepts and configurations
- Demonstrate how to manage and configure VLANs
- Design and configure a simple LAN and internetwork.
- Compare and contract key characteristics of LAN environments

Learning and Teaching Methods:

The on-line course delivery involves a combination of lectures, self-paced study, weekly online continuous assessment and practical laboratory sessions with both simulators and physical routing equipment.

Module content:

- Switching Concepts
- LAN Design
- VLANs
- VTP & STP
- Inter-VLAN routing

4. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2014		5	CMPU2015	Networking 3 - Switching

- Basic Wireless Concepts and Configurations

Module Assessment

Assessment will be through a combination of continuous assessment and a written exam.

Marks will be allocated as follows

- Continuous Assessment (50%)
- Written Exam (50%)

Continuous Assessment (50%): The continuous assessment element of this course utilises the CCNA online assessment facility.

Written Exam (50%): The written exam will be conducted under normal DIT Exam regulations and will be based on the theory covered during lectures.

Essential Reading:

Cisco Systems, 2009, "CCNA Exploration Course Booklet: LAN Switching and Wireless, Version 4.0" Cisco Press

COMER, Douglas E. 2001, "Computer Networks and Internets with Internet Applications", Prentice Hall.

HALSALL, Fred. 2005, "Computer Networking and the Internet", Addison Wesley.

TANNENBAUM, Andrew, S. 1996, "Computer Networks", Prentice Hall

LAMMLE, Todd. 1998, "CCNA Cisco Certified Network Associate Study Guide, Illustrated Edition", Sybex

Supplemental Reading:

Cisco Systems, 2004, "CCNA 1 and 2 Lab Companion, Revised (Cisco Networking Academy Program) (3rd Edition), Cisco Press

HALSALL, Fred. "Computer Networking and the Internet", Addison Wesley.

THOMAS, Tom., 2004,"Network Security First Step", Cisco Press

Web references, journals and other:

http://www.cisco.com

http://www.cisco.com/web/learning/netacad/index.html

4. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2014		5	CMPU2015	Networking 3 - Switching

http://www.cisco.com/en/US/learning/netacad/course_catalog/CCNA.html

Further Details:

Duration:	Single Semester Module
Contact Hours:	4 hours per week (2 hours of lecture, 2 hours of laboratory work.)
Class Size:	Labs should run with a maximum of 24 students per lab supervisor

Date of Academic Council approval ………………………….

5. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
None	None	5	CMPU2011	Legal & Professional Issues

8.2.10. Legal & Professional Issues

Module author: Paul Bourke.

Module Description:

This module introduces students to the legal framework and business related areas of law, including contract, negligence, IT and employment law. The module provides a general overview of the law in the business area and is suitable for students who wish to gain some knowledge of basic legal principles. It will also introduce the students to the ethical issues that may arise in business and IT projects.

Module aim

The aim of this module is to:

- develop an understanding of the place and function of law in IT business organisations.
- to give the student knowledge of the legal institutions that affect businesses and the legal processes involved.
- to develop skills which enable the students to deal with legal problems in a business context and to develop analytical approach to such problems.
- to encourage an interest in legal issues through topical discussion.
- To explores ethical issues on the interface between information technology and society
- to develop a comprehensive and critical understanding of major concepts, theoretical models and management tools used to understand, explain and deal with business ethics problems.

Learning Outcomes:

On completion of this module, the learner will be able to:

- demonstrate a broad knowledge and integrated understanding of the structure, sources and substantive law governing contract, intellectual property, employment and information technology.
- demonstrate an understanding of the ongoing development of intellectual property and information technology law, as they respond to developments in science, the creative arts and business practices and identify deficiencies in current law where appropriate.
- apply the substantive law to a range of complex practical problems in business based on real case studies.
- analyse legal problems, classifying issues in terms of relevance and importance, and propose solutions.
- analyse and present a response to business ethics cases and problems.

5. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
None	None	5	CMPU2011	Legal & Professional Issues

- analyse the ethical issues that may arise with the implementation of an IT solution

Learning and Teaching Methods:

Lectures are used to provide a basic framework to the topic and key references. Tutorials make use of a mix of problem type questions, open-ended questions for discussion, and questions focusing on key issues in the topic. This approach enables students to think about how the law adapts to changing circumstances, making use of current legal and ethical issues that are reported in newspapers where appropriate to achieve the module learning outcomes.

Module content:

Sources of Irish Law: Bunreacht na hEireann - separation of powers, fundamental rights, organs of state. Legislation –superior and delegated. Common Law – doctrine of precedent. E.U. Law – Community law and national law, Directives, Regulation, actions in the Court of Justice.

Principles of Contract Law: Defining a contract, formation of a contract, written/oral. Essential elements – capacity, agreement, consent, consideration, intention, legality. Remedies for breach of contract – damages, recission, injunction. Sale of Goods Act 1982, EC Directive on Unfair Terms.

Law of Tort: Nature of a tort – liability issues. Defences to an action in tort – inevitable accident, assumption of risk, necessity, statutory authority, limitation of actions. Liability of employers, occupiers, possessor of skills. Remedies: damages, injunction.

Employment Law: Contracts of employment, legislation on dismissal, equality, working time and safety, E.E. Social Policy.

Information Technology Law: Ownership issues – data protection, copyright protection of computer programs and databases, patents. Data Protection Act 1988, Data protection Directive, the Europol Convention. Copyright Act 1963, 1993 Regulations, 1996 Pirated Goods Regulation, Database Directive 1996, Patents Act 1992 and European Patent Convention. Liability issues – contractual and tortious. Liability for defective products and services, breaches of security, breach of privacy, professional negligence.

Criminal Law Issues: computer misuse, hacking, viruses, fraud, Criminal Damage Act 1991.

Internet Law: Control of content, pornography, defamation, privacy and electronic commerce.

Disability: Legislation National and International, Definitions of Disability, International Classifications, Legislative Impact on IT industry.

Business ethics: covers the main ethical theories and perspectives that can be used to examine business ethics dilemmas, such as theories of justice and rights. By examining the complexity, controversy and globalised nature of many ethics issues such as global warming and labour conditions in developing countries, this theme also addresses the question of whether and how

5. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
None	None	5	CMPU2011	Legal & Professional Issues

business ethics problems can be successfully addressed in a way that satisfies a diverse range of stakeholders across the globe.

Business in Society: the role and responsibilities of business in society. Corporate social responsibility and stakeholder theory. The role of ethics in consumer purchase decisions and corporate reputation.

Ethics in the Workplace: Addresses the decision processes and organizational influences which cause people to make ethical and unethical decisions at work. Ethical issues pertaining to employees, such as discrimination, workplace surveillance and whistle blowing.

IT Ethics: Topics include ethical theory, privacy and security, spam, electronic commerce, the digital divide, open source software, medical informatics, bioinformatics and some neo-classical economics.

Code of conduct of Professional bodies:

Module Assessment

Continuous Assessments - 100%

These assessments are made up of a combination of in-class tests, group project, and presentation.

Essential Reading:

Brian Doolan, 2003, Principles of Irish Law 6th Ed., Gill and Macmillan

Kelleher & Murray, 2008, Information Technology Law in Ireland, Tottel

Supplemental Reading:

Fell et Al, 2007, IT Law, BCS.

Aine Keenan, 2002, Essentials of Irish Business Law, 3rd Ed., Gill and Macmillan

Ian Lloyd, 2004, Information Technology Law, OUP

Deborah G. Johnson, 1994, Computer Ethics, Prentice Hall.

Web references, journals and other:

Computers and Law Journal

National Disability Authority WWW.NDA.IE

Trace Center Wisconsin http://trace.wisc.edu

Further Details:

Duration: 1 semester. Contact hours: 3 hours (Lecture: 2; Lecture/Tutorial: 1) Max size: 30

Date of Academic Council approval ....

6. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2010	Internet Applications and Web Development

8.2.11. Internet Applications and Web Development

Module author: Cindy Liu

Module Description:

This module gives the learner an introduction to a range of World Wide Web client technologies and the skills needed to develop them. A combination of individual and group work is used to formulate an information architecture plan for a website and to create its style guide, assets and the XHTML code necessary to implement it.

In addition, introduce further develop skills needed to implement more advanced Web applications that use server-side technologies.

Module aim

The aim of this module is to:

- - Introduce the students to the full services available across the Internet.
- - Developing the skills needed to publish information on the Web.
- - Enhance communication skills
- - Access a database by connecting to a server

Learning Outcomes:

On completion of this module, the learner will be able to be able to:

- Identify and use a wide variety of Internet services.

- Develop XHTML and style sheets.

- Differentiate between and develop content using a variety of Web editors.

- Develop and integrate practical experience of a variety of client side technologies available for development of Web applications.

- Recognise and demonstrate the importance and principles of good Web design,

- Developed a complete Web site which provides information on a specific topic of choice and which is presented to their class.

- Create a connection to the database over the web to create dynamic content

- Evaluate a web site for accessibility.

6. Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU2010	Internet Applications and Web Development

Learning and Teaching Methods:

The module will be taught in a work based learning laboratory environment. Weekly problem-solving exercises will be presented, which will be supported by group based learning through a major group project. The learner will also be required to conduct some supplemental reading.

Module Content:

- Internet Services: Ftp, email, remote login, audio and video across the Internet,
- Web architecture: Client/server architecture.
- Client side technologies: HTML, style sheets, helper applications, scripting languages
- Web design: Hypertext design, information design, navigation design, web design guidelines. W3C/Web Access Initiative Guidelines. National Disability Guidelines. Accessibility Testing
- Recent Advances in Client-Side Technology: e.g. XML, JavaScript, VB Script and Cookies
- Database Internet Connectivity and Server side scripting

Module Assessment:

100% - continuous assessment.

Essential Reading:

Sebesta, R.W. (2011), *Programming the World Wide Web*. Sixth Ed., Pearson.

COMER, D. E. (2008) “Computer Networks and Internets with Internet Applications”, Prentice Hall.

Supplemental Reading:

David Flanagan (2006), *JavaScript: The Definitive Guide*, O'Reilly.

Chuck Musciano, Bill Kennedy (2006), *HTML and XHTML: The Definitive Guide*, O'Reilly.

Eric Meyer (2004), Cascading Style Sheets: The Definitive Guide, O'Reilly.

Web references, journals and other:

World-Wide-Web Consortium - http://www.w3schools.com/

JSP reference : http://www.jsptut.com/

The Apache Software Foundation http://apache.org/

Further Details:

The module will run over one semester, with four contact hours per week.

Date of Academic Council approval ………………………….

8.3. Core Syllabi Stage 3

MODULE CODE	TITLE	Credits	
CMPU3010	Databases 2	5	S1
CMPU3017	Graphical User Interface Programming	5	S1
CMPU3028	Networking 4 - WAN Technologies	5	S1
CMPU3044	Systems Infrastructure and Architecture I	5	S1
CMPU3027	Network Programming	5	S2
CMPU3034	Security	5	S2
CMPU3039	Software Installation and Maintenance	5	S2
CMPU3047	Web Development and Deployment	5	S2

Table 14 Core modules Stage 3

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2007, CMPU1025		5	CMPU3010	Databases 2

8.3.1. Databases 2

Module author: Patricia O'Byrne and Databases group.

Module Description:

The material covered will build on student's knowledge and skills to include design and transaction processing on a multi-user environment. The student will use a procedural language extension to allow for sequential, selective and iterative processing. Triggers, functions and procedures will enrich the student's ability to store and retrieve data securely. The student will apply the concepts to a case study.

Module aim

The aim of this module is to…

- Enhance design skills using Data Normalisation and Extended Entity Relational Modelling.
- Expand the student's ability to ensure data integrity using further database objects, such as triggers, functions and procedures.
- Furnish the student with the skills to undertake multi-layer transactions using a procedural language extension to the relational database, and a third-generation language.

Learning Outcomes:

On completion of this module, the learner will be able to……………

- Apply ER Modelling skills to a medium-sized database application area.
- Normalise data to third normal form.
- Demonstrate an understanding of ACID properties.
- Design and implement a multi-statement transaction using a procedural extension to the relational database.
- Write and use functions, procedures and triggers where appropriate.
- Configure user access and transactions to allow for multiple users with different needs to access the database.
- Practice multi-layer transaction processing.

Learning and Teaching Methods:

This module will be delivered over four hours per week for one Semester. This will consist of one hour lecture, one hour tutorial and one two-hour laboratory session. During the lecture, new material on modelling, transaction processing and programming with relational databases will be disseminated. Techniques will be applied in the practical laboratory sessions, including the use of a

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2007, CMPU1025		5	CMPU3010	Databases 2

case study. Where appropriate, students will work in teams. The tutorials will review theory and practice, to prepare them for practical and examination work, in accordance with the requirements of the students.

Module content:

- Techniques of Data Normalisation and Extended Entity Relationship Modelling.
- Architecture and layering of transactions in a relational database.
- Concepts of procedural programming as applied in the language extension.
- I/O to the client from the language extension and from a 3GL.
- Stored Functions and procedures.
- Trapping and propagating errors from the database to a client process.
- Allocating privileges to schema users, depending on the user requirements.
- ACID properties, concurrency and locking.
- Audit and constraint triggers.

Module Assessment

Module will be assessed by both non-exam and examination assessment. Students must pass the entire module. Continuous assessment 40%, examination 60%.

Essential Reading:

- Elmasri & Navathe, 2010, 'Fundamentals of Database Systems' 6th Ed., Addison Wesley.

Depending on database being used, examples are:

- Sunderraman, R., 2007, 'Oracle 10g Programming: A Primer, Addison Wesley
- Sack, J., 2008, SQL Server 2008 Transact-SQL Recipes: A Problem-Solution Approach, APress
- Geschwinde, E., 2001, PostgreSQL Developer's Handbook, 2nd Ed., Sams

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2007, CMPU1025		5	CMPU3010	Databases 2

Supplemental Reading:

Connolly & Begg, 2009, 'A practical approach to design, implementation and management' 5th ed. Addison Wesley

Web references, journals and other:

References to specific vendors will depend on the DBMS being used. Examples:

- www.oracle.com
- www.postgresql.org
- http://msdn.microsoft.com/en-us/library/bb510741(SQL.100).aspx

Further Details:

laboratory sessions limited to 20 – 25 by lab size. 4 contact hours per week. To be delivered in one semester.

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3017	Graphical User Interface Programming

8.3.2. Graphical User Interface Programming

Module author: Programming and Algorithms group

Module Description:

This module addresses the need among software developers to build graphic user interfaces using a variety of user interface components. It introduces the students to event driven programming and provides the skills necessary to design and develop user interface applications. In addition it extends the student's object oriented programming skills developed in the pre-requisite Object Oriented Programming modules.

Module aim

The aim of this module is:

- equip the student with knowledge and programming skills to build GUIs using user interface components and implement event programming techniques;
- expand the student's knowledge of more advanced programming concepts e.g. inheritance, interfaces, polymorphism;
- enable students to discriminate between alternative user interface programming approaches;
- demonstrate good GUI design principles.

Learning Outcomes:

On completion of this module, the learner will be able to:

- demonstrate thorough knowledge of and ability to apply event driven programming
- implement a user interface with user interface components and event handling
- incorporate basic graphics within an application manage persistent data in tandem with appropriate user interface components
- understand concurrency issues in event driven programming

Learning and Teaching Methods:

This is a practical module, largely driven by learning through doing. The lectures will be used to present concepts, discuss sample code and look at practical examples.

The key learning method is via the programming labs where students will be set programming lab exercises to implement the concepts covered in lectures. Tutorial sessions will be used to discuss problems that students had with the lab exercises, and to discuss any particular aspect of the module that the students are having difficulty with.   In addition, students will be asked to undertake practical assignment(s) where they will complete programming tasks in their own time as part of the continuous assessment.

Module content:

- If not previously studied, an introduction to the programming language and environment to be used;
- Overview of application programming interface for the relevant GUI environment;
- Creating a GUI, including components, windows, layouts ;
- Implementing event driven behaviour;
- Implementing the GUI in typical environments used, such as web;
- Use of persistent data (time allowing, linking to a database for query retrieval and data display);

Module Assessment

The assessment for this module will be 50% continuous assessment and 50% written exam. The continuous assessment will focus on the practical learning outcomes, with assessments such as practical programming assignments, use of the lab outputs as part of the assessment (at the discretion of the lecturer) and lab based exam(s).

The written exam will be used to test the student's understanding of the concepts and their implementation, as learnt on the module. The exam will allow the students to demonstrate their knowledge by both descriptive explanations and via written coding examples.

Essential Reading:

This depends upon the programming environment used and will be specified by the lecturer at the start of the module.

Supplemental Reading: (author, date, title, publisher)

As for essential reading.

Web references, journals and other:

As for essential reading.

Further Details:

This is a one semester module, with two hours of practical labs, one hour lecture and one hour tutorial. The tutorial hour may be split into multiple sessions subject to class size. The nature of the tutorial requires that classes should not be excessively large.

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3028	Networking 4 – WAN Technologies

8.3.3. Networking 4 – WAN Technologies

Module author: K. O'Brien

Module Description:

In This course provides an introduction to Wide Area Networking (WAN) technologies. The course focuses on Advanced IP addressing techniques including Network Address Translation (NAT), Port Address Translation (PAT) and Dynamic Host Configuration Protocol (DHCP), WAN Technology and Terminology including Point to Point Protocol (PPP), Dial-On-Demand Routing (DDR) and Frame Relay, Network Management, Introduction to Optical Networking.

The module is structured to cover both theory and practical experience of Wide Area Networking technologies.

Module aim

The aim of this module is to provide the learner with a working knowledge of and practical experience of the installation and configuration of Wide Area Networks.

Learning Outcomes:

On completion of this module, the learner will be able to

- Describe the components and organisation of Wide Area Networks.
- Apply practical skills in the area of design, installation, configuration and security of Wide Area Networks.

Learning and Teaching Methods:

The on-line course delivery involves a combination of lectures, self-paced study and weekly online continuous assessments. Included in the on-line materials is a significant practical aspect to the course where learners will gain exposure to industry standard networking equipment, its configuration and management

Module content:

Overview of WAN Technologies, Point-to point protocol, Frame Relay, Network Security, Access Control Lists, Teleworker Services, IP addressing Services, Network Troubleshooting, emerging technologies

Module Assessment

Assessment will be through a combination of continuous assessment and a written exam.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3028	Networking 4 – WAN Technologies

Marks will be allocated as follows • Continuous Assessment (50%) • Written Exam (50%)

Essential Reading:

COMER, Douglas E. "Computer Networks and Internets with Internet Applications", Prentice Hall.

HALSALL, Fred. "Computer Networking and the Internet", Addison Wesley.

TANNENBAUM, Andrew, S. "Computer Networks", Prentice Hall

Vachon B., Graziani R. (2008) Accessing the WAN Exploration Companion Guide, First Edition", Cisco Press

LAMMLE, Todd. "CCNA Cisco Certified Network Associate Study Guide, Fifth Edition", Sybex.

Supplemental Reading:

Graziani R., Johnson A. (2007) ,Routing Protocols and Concepts, CCNA Exploration Companion Guide, Cisco Press.

Further Details:

Maximum of 20 students per laboratory.. To be delivered in one semester. Four hours per week, 2 hours of lecture, 2 hours of laboratory work.

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2021		5	CMPU3044	Systems Infrastructure and Architecture 2

8.3.4. Systems Infrastructure and Architecture 2

Module author: Andrea Curley and Patricia O Byrne

Module Description:

One of the roles of the IT professional is to design and build systems and integrate them into an organization. This module takes knowledge gathered in *System Infrastructure & Architecture 1* and uses it to source, evaluate and integrate components into a single system. It also covers the fundamentals of project management.

Module aim

The aim of this module is to:

- Identify the issues in the acquisition of software and hardware, and the sourcing of IT services.
- Address project management issues.
- Introduce the area of testing in an IT system.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Differentiate between build and buy in software and hardware acquisition.
- Differentiate between in-sourcing and out-sourcing for the acquisition of IT services, including support.
- Explain the importance of testing, evaluation and benchmarking in any IT sourcing decision.
- Explain the elements in a well-structured contract in an IT sourcing decision.
- Define integration in terms of components and interfaces providing examples of middleware platforms.
- Discuss the planning, budgeting, and scheduling issues in project management.
- Give examples of current testing standards and techniques.

Learning and Teaching Methods:

This module is taught through lectures and exercises in class combined with practical application through use of web and CASE tools.

Module content:

Acquisition and Sourcing: advantages and drawbacks of building and buying; in-sourcing and out-sourcing for the acquisition of IT services; testing, evaluation and benchmarking in any IT sourcing; primary components in an RFP; advantages and drawbacks of using RFPs in an IT sourcing decision;

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
CMPU2021		5	CMPU3044	Systems Infrastructure and Architecture 2

importance of a well-structured contract in any IT sourcing decision; given an RFP, recommend and justify one or more products that satisfy the criteria of the RFP.

Integration and Deployment: integration in terms of components and interfaces; examples of middleware platforms.

Project Management: key components of a project plan; importance of a cost/benefit analysis for a project plan; roles and responsibilities for key project personnel and stakeholders; project planning and tracking tools.

Testing and Quality Assurance: examples of current testing standards and techniques; evaluation and execution of acceptance tests; tools and techniques to create a testing environment.

Module Assessment

Written Examination – 70%

Continuous Assessment – 30%

Essential Reading:

Sommerville, I. (2011), Software Engineering 9, Ninth Ed., Pearson.

M. A. Parthasarathy (2007), Practical Software Estimation: Function Point Methods for Insourced and Outsourced Projects, O Reilly.

Supplemental Reading:

Tardugno, A.F, DiPasquale, Matthews, R.E (2000) IT Services: Costs, Metrics, Benchmarking, and Marketing, Prentice Hall.

Maciaszek, L.A. & Liong, B. L. (2005). Practical Software Engineering: A Case Study Approach. Pearson.

Skidmore, S. & Eva, M. (2004), Introducing Systems Development, Palgrave and MacMillan.

Pressman R.S. (2010), Software Engineering: A Practitioner's Approach, Seventh Ed., McGraw-Hill.

Further Details:

Contact hours. Two hours lecture, one hour lab per week. To be delivered in one semester.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3034	Security

8.3.5. Security

Module author: Dr. Fredrick Mtenzi.

Module Description:

Recent developments in the computing technology, mobile device technologies and the increasing broadband availability globally are shaping our lives in ways unimaginable a few years ago. While these developments have provided an unprecedented access to information and connectivity, they have also led to increased security concerns regarding the safety of our systems, data and wellbeing. The rush to market software which forms the trend of most of the modern technology has enhanced the release of insecure software.

This module provides an overview of the fundamental concepts of security. It will introduce students to the basic principles of securely using computers, wired and wireless networks and the internet. Emphasis will be placed on understanding issues which will lead to sensible security and an appreciation that security is the weakest link problem. The user security social responsibility which is becoming more important now will be covered in the legal and ethical aspects of security. Relevant concepts covered in the module will the subject of student experiments in the labs.

Module aim

The aim of this module is

- to introduce the students to the security principles,
- to give the students a thorough understanding of the network security issues,
- to provide them with sound practical knowledge of security tools, Operating Systems and Network security,
- to provide them with an in-depth practical security knowledge in real-life.

Learning Outcomes:

On completion of this module, the student will be able to:

- Define and describe network security,
- Discuss and relate the fundamental concepts of Security,
- Classify and analyse the nature and threat from malware,
- Design and deploy best practice techniques for securing Operating Systems and Networks,
- Compare and contrast different security tools and determine their appropriate use in a specific situation,
- Examine the trade-off of legal and ethical security issues in an organisation

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3034	Security

- Asses and rank different systems security approaches according to the protection they provide in real-life applications.
- Recognise and deploy the different hacking phases in systems.

Learning and Teaching Methods:

In this module a number of teaching methods may be employed including lectures, practical sessions, tutorials and case studies. At least one industrial seminar may be arranged. Focus should be placed on empowering the students to develop their skills independently of the presence of a tutor or lecturer.

Module content:

- Introduction to security
- Classical encryption techniques and stream ciphers
- User Authentication
- Access Control
- Physical and Infrastructure Security
- Operating Systems Security
- Network Security
- Security Tools
- Introduction to ethical hacking
- The role of people in security
- Legal and ethical aspects of security

Module Assessment

This module should have a 50% weighting for the examination and a 50% weighting for the continuous assessment. While it is important that the student can demonstrate their technical ability with coursework, it is equally important that they demonstrate an understanding of the theoretical aspects of security.

Essential Reading: (author, date, title, publisher)

William Stallings, 2005, Cryptography and Network Security: Principles and Practice, 5th ed, Prentice Hall

Supplemental Reading: (author, date, title, publisher)

Seymour Bosworth and M.E. Kabay, 2009, Computer Security Handbook, John Wiley & Sons. Inc.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3034	Security

William Stallings and Lawrie Brown, 2008, Computer Security: Principles and Practice, Prentice Hall

Markus Jakobsson, Zulfikar Ramzan, 2008, Crimeware: Understanding New Attacks and Defences, Symantec Press.

Charles P. Pfleeger and Shari Lawrence Pfleeger, 2007, Security in Computing, Prentice hall

Michael Gregg, 2008, Certified Ethical Hacker, Que Publishing

Vincent Nestler, Gregory White, Wm. Arthur Conklin and Corey Schou , 2010, Principles of Computer Security CompTIA Security+ and Beyond Lab Manual, McGraw-Hill Companies

Web references, journals and other:

IEEE security & privacy

ACM Transactions on Information and System Security (TISSEC)

http://www.sans.org/rr/

Further Details:

The module will run over one semester, with three contact hours per week. Two hours for lectures and one hour for the laboratory.

Date of Academic Council approval …………………………..

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3039	Software Installation and Maintenance

8.3.6. Software Installation and Maintenance

Module authors: Paul Kelly and Damian Bourke

Module Description:

Previously students will have demonstrated proficiency in Unix/Linux fundamentals including the File System Hierarchy and in the development of software using a variety of development tools, languages and development platforms. This module is intended to build on this proficiency and to take the student's development skills to a more practical level within the development process namely the roll-out and maintenance of software across the enterprise. This module introduces the student to the concepts behind software deployment. Essentially it builds upon the student's understanding of software development and operating systems. It introduces the techniques for installing and configuration of applications developed by the students across a number of platforms.

Module aim

The aim of the course is

- to provide students with the tools for the development, deployment and maintenance of a software system on stand-alone and networked environments.
- to provide the student with the necessary skills to write end-user documentation.

Learning Outcomes:

On completion of the course the learner will:

- be able to select and utilise the tools available for software development, deployment and maintenance.
- know how to create and maintain software libraries
- demonstrate a proficiency in a scripting language
- be able to deploy and configure application to run on a variety of platforms and environments
- be able to demonstrate appropriate version control
- be able to produce detailed documentation to assist the end-user in deploying software and implementing version control

Learning and Teaching Methods:

Teaching will be through a combination of lectures and labs. Lectures will introduce the concepts behind software development and deployment. Labs will be used to reinforce lectures through the use of weekly lab exercises and at least two assignments.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3039	Software Installation and Maintenance

Module content:

- Introduction
- Development Tools
- Static and Dynamic libraries
- Make utility
- Installation scripts
- Debugging Tools
- Version Control Tools
- Customising the shell: Shell variables
- Pipes, filters and Pattern matching
- Stream editing: sed
- Process Handling; Process Ids, foreground and background processes, Signal handling, PIDs
- Advanced Linux/Unix Shell scripting and PowerShell scripting within Windows:
- Macro processing: m4
- Programmable filters:
- Process management
- Package management tools and end-ser documentation

Laboratory exercises to include at a minimum:

- Development and deployment of an application on a variety of stand-alone platforms and on a networked environment
- Configuration of an application to run in a variety of environments

Module Assessment

Written Examination:	60%
Continuous Assessment:	40%

Essential Reading:

Edward C. Bailey, 1997, *Maximum RPM*, (Imprint: Sams) Macmillan Computer Publishing, Indiana. ISBN: 0-672-31105-4

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
			CMPU3039	Software Installation and Maintenance

Supplemental Reading:

Web references, journals and other:

Donnie Barnes, 1999, The RPM How-To Guide (http://www.rpm.org/RPM-HOWTO/), Red Hat Inc. Accessed 7th March 2006

Article: What You Need to Write Man Pages. By Peter Seebach http://www.linux.com/article.pl?sid=04/02/05/1651203 (Accessed: 12th March)

Further Details:

One semester comprising 2 hours of Lectures and 2 hours of Laboratory.

Date of Academic Council approval ………………

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3027	Network Programming

8.3.7. Network Programming

Module author: Ken O'Brien

Module Description:

This module introduces students to application programming interfaces using TCP/IP protocols and the Sockets programming interface.

Module aim

The aim of this module is to equip the student with a basic understanding of network programming using TCP/IP communication protocols.

Learning Outcomes:

On completion of this module, the student will be able to:

- create simple client and server programs using the Socket Interface
- list the major protocols in a TCP/IP protocol stack
- describe the functions of the different protocols

Learning and Teaching Methods:

Lectures will be used to present the material and pace the learning process throughout the module. Laboratory exercises will be used to reinforce the learning experience.

Practical assignments will be given throughout the module to allow students to gain experience practical network programming

Module content:

Use of TCP/IP, Designing Applications for a distributed environment, the client/server model and software design, concurrency in clients and servers, program interface to protocols, the socket interface, algorithms and issues in client software design, examples of client software, algorithms and issues in server software design , iterative connectionless servers, iterative connection-oriented servers, concurrent connection-oriented servers, remote procedure call mechanism, emerging networking technologies.

Module Assessment

Written examination 70%. Continuous assessment 30%

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3027	Network Programming

Essential Reading:

Comer D.E, Stevens D.L. (2003) Interworking with TCP/IP Volume 3 Client-Server Programming And Applications Linux/Posix Sockets Version, Prentice-Hall

Supplemental Reading:

Tanenbaum A. (2011) Computer Networks, Prentice Hall

Further Details:

Two hour lectures, one hour lab, To be delivered in one semester

Date of Academic Council approval ……………………….

Pre-Requisite Modules code(s)	**Co-Requisite Modules code(s)**	**ECTS Credits**	**Module Code**	**Module Title**
CMPU2010, CMPU1025	None	5	CMPU3047	Web Development and Deployment

8.3.8. Web Development and Deployment

Module author: Paddy Matthews

Module Description:

This course is a continuation and elaboration of the Web development concepts introduced in TECH2401, with students using server-side web technologies to deliver web-based solutions to standard problems.

Module aims:

- To enhance the web development skills developed in TECH2401 Internet Services & Web Development.
- To develop skills needed to implement more advanced Web applications that use server-side technologies.
- To introduce students to the concept of Model-View-Controller (MVC) architecture in the context of web development.
- To introduce students to framework-driven development.
- To familiarise students with XML (Extensible Markup Language) and its potential applications in the context of web services.
- To introduce students to the use of Ajax (Asynchronous JavaScript and XML) and its uses in the development of interactive web applications.

Learning Outcomes:

On completion of the module, students should be able to:

- Demonstrate a broad knowledge and have practical experience of a range of server side technologies.
- Develop dynamic Web applications that use server side processing.
- Develop Web applications that access, store and update information in databases.
- Demonstrate familiarity with the MVC architecture and framework-driven development through the creation of web applications using these concepts.
- Understand the purpose of XML and be capable of validating and manipulating XML documents.
- Develop simple web services using XML.
- Demonstrate familiarity with Ajax technologies and their uses in web development.
- Understand basic security requirements for web applications and how they might be facilitated.

Pre-Requisite Modules code(s)	**Co-Requisite Modules code(s)**	**ECTS Credits**	**Module Code**	**Module Title**
CMPU2010, CMPU1025	None	5	CMPU3047	Web Development and Deployment

Learning and Teaching Methods:

The learning and teaching methods will be the combination of lectures and laboratory work.

Module content:

- Server-side technologies: An overview of the various server side technologies, including Java Servlets, JSP, ASP, etc.
- Web Development:
- Model-View-Controller Architecture.
- Introduction to frameworks and framework-based implementation of MVC architecture.
- Introduction to dynamic programming using Ruby and JavaScript.
- Development of applications using a web framework (Ruby on Rails).
- Use of Ajax technologies in web development.
- Introduction to XML:
- XML syntax and document structure,
- notion of schemas and validation,
- document navigation using XPath and XQuery,
- transformations using XSLT,
- XML-based web services.
- Security threats and counter-measures – code injection, cross-side scripting, SSL.

Module Assessment:

Student learning will be evaluated via continuous assessment comprising two major assignments which will be demonstrated in class, and a number of minor assignments.

Essential Reading List:

Ruby, Thomas and Hansson, 2009, Agile Web Development with Rails, 3rd Edition, Pragmatic Bookshelf.

Thomas, 2009, Programming Ruby, 3rd Edition, Pragmatic Bookshelf.

Vohra and Vohra, 2006, Pro XML Development with Java Technology, Apress.

Pre-Requisite Modules code(s)	**Co-Requisite Modules code(s)**	**ECTS Credit s**	**Module Code**	**Module Title**
CMPU2010, CMPU1025	None	5	CMPU3047	Web Development and Deployment

Further Details:

Two hour lectures, one hour lab, one hour tutorial. To be delivered in one semester

Date of Academic Council approval ………………………….

8.4. Stage 3 Syllabi (Full-time stream)

MODULE CODE	MODULE TITLE	ECTS	SEMESTER
CMPU3045	Team Project[1]	10	S2
CMPU3011	Distributed Objects	5	S1
	Option	5	S1

Table 15 Core modules in Stage 3 (Full-time stream)

Ordinary Degree Students

CMPU3018	Individual Project	10

Table 16 Module for Ordinary Degree Exit

[1] Students who intend to exit with an Ordinary Degree should take the Ordinary Degree project instead of this. This module is NOT available to students intending to progress to Stage 4.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU3045	Team Project

8.4.1. Team Project

Module author: Ronan Bradley

Module Description:

This module offers the students the opportunity to use the knowledge and skills they have developed over the previous stages in the development of a complex software system. The project is a team project requiring the students to collaborate and organise themselves in a group to achieve the module objectives.

Module aim

This module aims to group students into small teams and familiarise them with a range of skills and knowledge necessary, to take a software application through its life cycle.

Learning Outcomes:

On completion of this module, the student will be able to:

- Demonstrate group interaction
- Demonstrate ability to deliver individual objectives within a team structure
- Demonstrate leadership and team collaboration skills
- Demonstrate an ability to follow development processes from initiation through design, implementation, test and delivery
- Implement a project design in an appropriate format from a requirements specification
- Demonstrate a choice of appropriate technology
- Implement role functionality
- Determine group integration.
- Develop presentation skills

Learning and Teaching Methods:

This module will be taught through a combination of lecture, tutorials and practical lab-based sessions.

The supervised practical sessions will be used to allow lecturers to review and assess the progress of the whole team and the individual team members. It will also be used for team demonstrations and presentations at least twice during the semester.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU3045	Team Project

Module content:

Students will be presented with a requirements specification, from which they must develop a software system in a group of between three and five students. These groups will be pre-assigned. Topics which will be covered as part of the module include

- Project planning and management (including documentation requirements)
- Test planning and test implementation
- Application of software engineering concepts to team development projects

Module Assessment

The module is assessed entirely by continuous assessment. The continuous assessment mark is determined through a combination of report submissions, submission of the student's project journal, presentations and interviews which take place throughout the semester. These are as follows:

- Group submission of project documentation artefacts (such as design documents, test plan, testing logs and project plans)
- Group presentation and defence of
  The design (prior to implementation)
  The final deliverable (including demonstration)
- Individual submission of a project journal
- Individual interview of each student within a group by the lecturer

A student's final mark will be composed of 40% of their group mark and 60% of their individual mark.

Essential Reading:

To be provided at the start of the module

Supplemental Reading:

To be provided at the start of the module

Web references, journals and other:

To be provided at the start of the module

Further Details:

Single Semester Module; 3 contact hour per week, 1 hour tutorial/lecture, 2 hours laboratory based review of progress and discussion with lecturer

Date of Academic Council approval ……………………………

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3011	Distributed Objects

8.4.2. Distributed Objects

Module author: June Barrett, Paul Kelly

Module Description:

This module builds upon the students procedural, object oriented and graphical user interface programming skills covered in other modules with emphasis.

Module aim

The aim of this module is to teach the students the use of persistent objects in an event driven environment and to teach advanced data structures and genericity, with emphasis on distributed objects.

Learning Outcomes:

On completion of this module, the learner will be able to :

- Create and manipulate persistent objects in relation to event driven environment.
- Persistent object manipulation
- Demonstrate the ability implement Multithreading.
- Select and implement appropriate distributed data structures
- Demonstrate object reuse through the use of genericity
- Perform network I/O, reusing core classes from an Object Oriented interface
- Build a front and back end object oriented event driven system

Learning and Teaching Methods:

This is a practical module, largely driven through doing. Lectures will used to present the theory, discuss sample code and examine practical examples.

The key learning method is via the programming labs where students will be set lab exercises to implement the concepts covered in lectures. In then labs the students will also have access to Relational Database .Tutorial sessions will be used to discuss problems that students had with the lab exercises, and to discuss any particular aspect of the module that the students are having difficulty with. In addition, students will be asked to undertake practical assignment(s) where they will complete tasks in their own time as part of the continuous assessment. During self study the students will use on-line third party tutorials.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3011	Distributed Objects

Module content:

- Overview of the Thread lifecycle
- Thread creation and manipulation using asynchronized and synchronized threads
- Input and output   persistent objects using streams.
- Creation and manipulation of distributed objects.
- Database connectivity, read, write and update persistent objects
- Advanced data structures , collections, sets and iterators
- Genericity
- I/O steam and database connectivity
- Read and write data to local and remote sites, remote Method Invocation

Module Assessment

Assessment of the module is a combination of the following:

Continuous Assessment (60%): (Individual assignments , Lab tests, On-line tests, In-class written tests).

Written examination (40%): One three hour, end of module examination.

Essential Reading:

This depends upon the environment used and will be specified by the lecturer at the start of the module.

Supplemental Reading:

This depends upon the environment used and will be specified by the lecturer at the start of the module.

Web references, journals and other:

This depends upon the environment used and will be specified by the lecturer at the start of the module.

Further Details:

Class size is expected to be a maximum of 60, broken into groups for tutorials and labs. Semesters: 1

Contact hours per semester: Lectures: 1 hour Lab: 2 hours Tutorial: 1 hour

Date of Academic Council approval …………………………

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU3018	Individual Project

8.4.3. Stage 3 Individual Project

Module author: **Damian Bourke.**

Module Description:

This module is required for all students intending to exit with a BSc (Ord) in Computing. It is NOT required for students who are progressing to Stage 4. This module will only be offered to students who are exiting with a BSc (Ord).

It requires the student to propose, design, implement, test, document and present a software project to demonstrate the level of knowledge gained over the course of their studies. The project must be an independent piece of work which is both coherent and well structured. The student is expected to be self-motivated so as to drive this work through to completion. They are expected to identify the key areas of the project and to make real decisions that will ultimately affect the end deliverable. The project deliverable will be an assessable, independent body of work that will demonstrate the student's ability to work on their own and their ability to communicate key aspects of the project.

Module aim

The aim of this module for the student is to complete a software system implementation. This includes all aspects of software development including, but not limited to; Analysis, Design, Development, Implementation Integration, and Documentation. In addition the student is required to produce an academic quality report outlining the key steps undertaken, the key decisions that were made, an evaluation of the outcomes, and the student's conclusions about the success or failure of the delivered system.

Learning Outcomes:

On completion of this module the student will be able to demonstrate the ability to:

- Define a problem area and write a project proposal
- Evaluate similar systems to their proposal identifying a set of clear user requirements
- Undertake research of the problem area to determine the boundaries and scope of the project.
- Undertake research of the proposed solution to identify appropriate technologies to use
- Select and implement a formal design methodology.
- Write a project plan and project manage the project to completion.
- Develop a fully operational software/hardware system.
- Produce a report (in English) of academic quality with appropriate referencing.
- Present and defend their findings.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU3018	Individual Project

Learning and Teaching Methods:

The project module involves a combination of seminars, self-paced work, research, and weekly meetings with their assigned supervisor. This is primarily a self-learning module with the supervisor providing guidance and feedback to the student throughout the project. The student is expected to state at the outset the objectives for the project and to meet these objectives throughout the year. The student is also given explicit verbal feedback at the Interim report stage which is a presentation by the student on the work completed so far and the work planned in the future.

In addition to the dedicated supervisory resources the School will also provide a second reader for the purposes of assessing the student's work.

Module content:

Students are provided with detailed guidelines on the project process. These guidelines detail the requirements of the project in terms of the project content and deliverables and they also provide guidance on the content and format of the Interim Report and Final Project Manual. The guidelines also explain the roles of the key stakeholders including the student, the supervisor, second reader and project co-ordinator.

Module Assessment

The module assessment will comprise 100% Continuous Assessment.

There are two key milestones within the project lifecycle at which the student is assessed. At the Interim Report stage the student is assessed on their research findings, system design, report writing and project management. The assessment is based on a key set of criteria set-out in the guidelines document.

At the Final Report stage the student is assessed on any additional research undertaken post-design, the system implementation phase, testing and evaluation phase, report writing, communication skills, project management, and the quality, complexity and usability of the delivered system. The assessment is based on a key set of criteria set-out in the guidelines document.

Essential Reading:

Damian Bourke, 2010. The Project Guidelines

Kathy Schwalbe, 2005, A Guide to the Project Management Body of Knowledge: Course Technology

Strunk, W. and White, E. B., 2000, *The Elements of Style* (4th ed.) Longman

Booth, W., Colomb, G.C., Williams, J.M., 2003, *The Craft of Research*, University of Chicago Press

Dawson, C., 2005, "A Practical Guide to Research Methods: A User-friendly Manual for Mastering Research Techniques and Projects", How To Books

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		10	CMPU3018	Individual Project

Supplemental Reading:

Evaluating Software Architectures Methods Studies

Web references, journals and other:

Further Details:

This module is to be delivered with dedicated supervisor contact hours equivalent to one hour per week for one semester, or half an hour per week for two semesters.

Under normal circumstances this project will be undertaken over two semesters for all full-time students.

In exceptional circumstances and at the discretion of the Head of School, students may apply to complete it in a single semester. Consideration will be given to their capacity to succeed in their project, based on their past performance and their workload.

Date of Academic Council approval ………………………….

8.5. Stage 3 Syllabi (Full-time options)

MODULE CODE	TITLE	Credits	
CMPU3004	Applied Intelligence	5	S1
CMPU3007	Cloud Computing	5	S1
CMPU3008	Computational Mathematics	5	S1
CMPU3026	Mobile Software Development	5	S1
CMPU3025	Mobile Robotics	5	S1
CMPU3046	Universal Design and Assistive ICT	5	S1

Table 17 Options available for Full-time Stage 3

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3004	Applied Intelligence

8.5.1. Applied Intelligence

Module Author: Brian Mac Namee & John Kelleher

Module Description:

This module introduces students to the power of artificial intelligence (AI) and machine learning (ML) by developing and exploring a series of ML and AI based applications primarily focused on using online data sources. While the module will cover AI and ML techniques these will not be covered in great depth, but rather the module will focus on the application of these techniques.

Module Aim:

The aims of this module are to:

- Introduce students to key techniques from artificial intelligence and machine learning
- Develop in students a deep understanding of how AI and ML techniques can be applied to real world problems.
- Engage students in the development of a range of real solutions using ML and AI techniques.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Demonstrate an understanding of a range of machine learning and artificial intelligence techniques
- Select appropriate ML and AI techniques to use to solve specific problems
- Compare the suitability of different AI and ML approaches to solving the same problem
- Develop a range of AI and ML solutions to real problems
- Critique the limitations of AI and ML approaches

Learning and Teaching Methods:

Lectures. tutorials, lab demonstrations and discussions. This module will be primarily practical and while material will be presented in lectures a large amount of time will be spent developing solutions to real problems based on the material covered in lectures. A virtual Learning Environment is extensively used in this module.

Module content:

The module content will be

- Introduction to Applied Intelligence
- What Is Artificial Intelligence (AI)?

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3004	Applied Intelligence

- What Is Machine Learning (ML)?
- Limits of ML and AI
- Applications of AI and ML
- Making Recommendations
- Collaborative Filtering
- Collecting Preferences
- Finding Similar Users
- Recommending Items
- User-Based or Item-Based Filtering?
- Applications of Recommender Systems: Online Retail Recommendation System, Social Network Friend Recommendation
- Discovering Groups
- Supervised versus Unsupervised Learning
- Hierarchical Clustering
- K-Means Clustering
- Clusters of Preferences
- Applications of Clustering: Customer Segmentation, Player Segmentation
- Searching and Ranking
- What's in a Search Engine?
- A Simple Crawler
- Building the Index
- Querying
- Content-Based Ranking
- Applications of Search: Building the next Google!
- Optimization
- Representing Solutions
- The Cost Function
- Random Searching
- Hill Climbing
- Simulated Annealing
- Genetic Algorithms

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3004	Applied Intelligence

- Optimizing for Preferences
- Network Visualization
- Applications of Optimization: Price Modelling, Parameter Selection
- Building Classification Models
- Documents and Words
- Modeling with Decision Trees
- Modeling with Naïve Bayes Classifier
- Modelling with k-Nearest Nieghbours Classifier
- Applications of Classification Models: Filtering Spam, Filtering Blog Feeds, Building Price Models, Social Network Friend Matching
- Evolving Intelligence
- What Is Genetic Programming?
- Genetic Programming Approaches
- Applications of Genetic Programming: Games, Opimisation, Search
- APIs and Languages
- Suitable APIs (e.g. Google API, Facebook API) an programming languages (e.g. Python) and techniques will be used to develop the sample applications described above.

Module Assessment

Continuous assessment - 100%

The module will be evaluated based entirely on continuous assessment. Students will be asked to develop a range of solutions to real problems using the approaches discussed in lectures. Students will also be tasked with presenting their opinions of the state of the art of AI and ML applications and he limitations of ML and AI based approaches in a series of in-class presentations.

Essential Reading:

- Toby Segaran, 2007, Programming Collective Intelligence: Building Smart Web 2.0 Applications, O'Reilly Media.

Supplementary Reading:

- Russell & Norvig, 2009, Artificial Intelligence – A Modern Approach (3rd ed), Prentice Hall
- Ethem Alpaydin, 2004, Introduction to Machine Learning, MIT Press
- Tom Mitchell, 1997, Machine Learning, McGraw Hill

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3004	Applied Intelligence

Web references, journals and other:

- Machine Learning, Springer Science + Business Media
- ACM Journal of Machine Learning Research, ACM
- IEEE Transactions on Pattern Analysis and Machine Intelligence
- IEEE Intelligent Systems

Further Details:

Duration: 1 semester

Contact Hours: 2 lecture hours and 2 lab hours per week

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3007	Cloud Computing

8.5.2. Cloud Computing

Module author: **Paul Doyle**

Module Description:

This module looks at the history and evolution behind cloud computing followed by a review of the latest technologies within it. This module is designed to provide the student with both a practical and theoretical understanding of existing cloud systems and their underlying technologies. The technologies focused on will include grid computing, virtualisation, distributed computing, cloud storage, security within the cloud in addition to reviewing a number of existing cloud environments.

Module aim

The aim of this module is to provide the learner with both a theoretical and practical understanding of the technologies used to enable Cloud Computing.

Learning Outcomes:

On completion of this module, the learner will be able to

a) Demonstrate an understanding of the fundamentals of Cloud Computing
b) Demonstrate an understanding of the evolution of Cloud Computing technologies
c) Demonstrate a practical understanding of cloud technologies within a laboratory environment.
d) Configure basic infrastructural components used within the cloud
e) Critically analyse different methods for implementing Cloud solutions

Learning and Teaching Methods:

The course delivery involves a combination of lectures and labs. Students will be expected to put into practice some of the module concepts by interacting with existing public cloud infrastructures.

Module content:

- What is cloud computing?
- Classifications of Clouds
- Evolution of Clouds
- Technologies within the cloud
- Virtualisation
- Storage and Security
- Hadoop
- Case Studies of Commercial Clouds (AZURE, Google APPS, Amazon EC2, OpenStack)

Module Assessment

Assessment will be through a combination of continuous assessment and a written exam.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3007	Cloud Computing

Marks will be allocated as follows

- Continuous Assessment (30%)
- Written Exam (70%)

Written Exam (70%): The written exam will be conducted under normal DIT Exam regulations and will be based on the theory covered during lectures.

Continuous Assessment (30%): The CA elements of the course will be based on the implementations of basic cloud technologies.

Essential Reading:

Velte/Elsenpeter 2009, Cloud Computing A Practical Approach, McGrawHill

Rosenberg/Mateos 2010, The Cloud at your service, Manning

Mather/Kumaraswamy/Latif Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice) 2009, O'Reilly Media

George Reese, 2009, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, O'Reilly

Supplemental Reading:

Armbrust, 2009, Above the clouds: A Berkeley view of Cloud Computing,UCB/EECS

Kesselman/Foster, 1998, The Grid: Blueprint for a new computing infrastructure, Morgan Kaufmann Publishers

Web references, journals and other:

http://aws.amazon.com/ec2/

http://www.ibm.com/ibm/cloud/

http://code.google.com/appengine/

http://www.microsoft.com/windowsazure/

http://www.grid.ie/

http://www.openstack.org/

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3007	Cloud Computing

Further Details:

This module will be taught over one semester, with four contact hours per week; 2 lectures, 2 laboratory practical hours.

The laboratory requirements for this module are such that students will require administration access to hardware systems in a controlled way such that they will not interfere with existing institute services.

Date of Academic Council approval …………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3008	Computational Mathematics

8.5.3. Computational Mathematics

Module author: John Gilligan, Dr Shane Mulligan

Module Description:

This is a an advanced module in Mathematics, which will give the student the necessary mathematical skills and tools, to deal with specialised areas in computing, such as Graphics, Image processing, Games development and Cryptography.

Module aim

The aim of this module is to give the student the necessary knowledge and competence to deal with specialised applications in computer science, such as graphics, image processing and cryptography. It will present the necessary mathematical tools and how they can be applied to various problem areas. It will present mathematics as an exact science, and a powerful tool to model and analyse problems.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Define and use linear algebra techniques in geometry.
- Explain the set-up and application of linear transformation, and matrix techniques in Graphics.
- Apply coordinate geometry algorithms in 2 and 3 dimensions.
- Develop and apply computational techniques which have application in topics such as image processing and intelligent analysis. These methods include, for example, numerical analysis and soft computing approaches and algorithms from number theory, and their application to cryptography.

Learning and Teaching Methods:

This module will consist of lectures and tutorials. Exercises and continuous assessments will be given to ensure that the student understands and masters the material, and to give them practice at representing and solving simple problems.

Module content:

Vectors and Matrices: Vectors, vector operations, cross product, normal vector. Matrices, multiplication, transpose, determinants.

Geometry. Co-ordinate systems, homogeneous coordinates. Line, circle, plane equations, parametric forms. Normal vector. Geometrical transformations and matrices, scaling, translation, rotations, and their composition. Projections and projection matrices.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3008	Computational Mathematics

Geometric algorithms: Euclidean distance, areas, intersection of line and plane, point-in-triangle test, area of a polygon.

Computational methods Numerical Algorithms, Methods from Number Theory, Soft Computing Methods, Applications in Computer Science

Module Assessment

Written Examination 70%

Practical 30%

Essential Reading:

Foley J. D., A. van Dam, S. K. Feiner, J. F. Hughes, 1990, Computer Graphics – Principles and Practice, 2nd Edition, Addison-Wesley.

Supplemental Reading: (author, date, title, publisher)

Bishop *Cryptography with Java Applets* Jones and Bartlett Comp Sci 2003

Karray and De Silva Soft Computing and Intelligent Systems Design Addison Wesley 2004

Web references, journals and other:

www.mathworks.com

Further Details:

Duration One semester: 3 hrs per Week.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3025	Mobile Robotics

8.5.4. Mobile Robotics

Module Author: Brian Mac Namee & John Kelleher

Module Description:

An introduction to mobile robotics including underlying theory and applications. The module will have a large practical component.

Module Aim :

This module provides a broad introduction to the field of mobile robotics, from a software perspective. It covers the basics of mobile robot control, software architectures, navigation and localization algorithms, and surveys common application areas. The practical aspects of the course will give students experience in developing real robotic systems.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Critique the appropriateness of using robotic solutions for various applications
- Describe the main components of a mobile robotic system, including sensors, actuators, control architectures and power options
- Differentiate between and compare different robotic sensor and actuator approaches
- Compare different approaches to robot locomotion
- Compare different approaches to robot localisation and mapping
- Develop mobile robotic solutions to sample problems

Learning and Teaching Methods:

The module will be delivered through a mixture of lectures and lab sessions with a stronger emphasis on lab work towards the latter half of the course. Lab work will take the form of a robot workshop in which students will be divided into teams to develop mobile robot solutions to appropriate problems. A virtual Learning Environment will be used in this module.

Module content:

The module content will be a combination of the following:

- Introduction
- Key questions for mobile robots
- Application areas
- Key components of mobile robotic systems

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3025	Mobile Robotics

- Sensors
- Basic concepts
- Contact sensors
- Internal sensors
- Range sensors
- Beacon based sensing
- Data fusion
- Biological sensing
- Visual sensors and algorithms
- Actuators
- Electronics basics
- Motors
- Artificial muscles
- Pneumatics
- New materials
- Mobile Robot Locomotion
- Wheeled robot topologies
- Legged robot topologies
- Legged robot gaits
- Basic mobile robot kinematics
- Representing robot positioning
- Forward kinematic model
- Mobile robot manoeuvrability
- Robot Control System Architectures
- General Control Structure for Mobile Robots
- Perception Action Models
- Subsumption Hierarchies
- Layered Models
- Deliberative Agents
- Mapping and Localisation
- Topologic maps

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3025	Mobile Robotics

- Metric maps
- Construction of occupancy grids
- Localisation approaches
- Introduction to SLAM (simultaneous localisation and mapping)
- Planning and Navigation
- Path-planning
- Collision Avoidance
- Advances in Robotics
- Implementing Robotic Systems
- An appropriate robotic API (e.g. ROS, Lejos)
- An appropriate mobile robot hardware platform (e.g. Lego Mindstorms, iRobot Roombas)

Module Assessment

Statement Written examination - 50%

Continuous assessment - 50%

Essential Reading:

Roland Siegwart and Illah R. Nourbakhsh, 2004, “Introduction to Autonomous Mobile Robots”, Bradford Books.

Supplementary Reading:

Howie Choset, Kevin M. Lynch, Seth Hutchinson, George Kantor, Wolfram Burgard, Lydia E. Kavraki, and Sebastian Thrun, 2005, “Principles of Robot Motion : Theory, Algorithms, and Implementations”, The MIT Press.

Gregory Dudek and Michael Jenkin, 2000, “Computational Principles of Mobile Robotics”, Cambridge

University Press.

Siciliano, Bruno; Khatib, Oussama (Eds.), 2008 “Springer Handbook of Robotics”, Springer.

Web references, journals and other:

Singularity Hub (www.singulaityhub.com)

NASA robotics (robotics.nasa.com)

Robots.net (www.robots.net)

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3025	Mobile Robotics

Further Details:

Duration: 1 semester

Contact Hours: 2 lecture hours and 2 lab hours per week

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3026	Mobile Software Development

8.5.5. Mobile Software Development

Module author: Programming and Algorithms Group

Module Description:

This modules covers the knowledge and practical skills to design and implement basic mobile phone applications, in a specific mobile development environment. In order to set mobile software development in context, it also covers background mobile communication technologies such as GSM, GPRS, EDGE, 3G, Bluetooth, and examines leading mobile development platforms and typical mobile data applications.

Module aim

The aim of this module is to enable the learner to develop mobile phone applications in a leading mobile phone development platform, whilst also providing an understanding of relevant mobile phone technologies in order to set the application development in context.

Learning Outcomes:

On completion of this module, the learner will be able to:

- - Develop mobile phone applications in a leading mobile development platform;
- - Understand and apply good design guidelines for mobile application development
- - Discuss and compare the leading mobile development platforms such as but not limited to: Android, Qt, Windows Mobile, and iOS (Apple);
- - Discuss and compare the mobile phone communication technologies such as, but not limited to: GSM, GPRS, EDGE, 3G, Bluetooth;

Learning and Teaching Methods:

Classroom based lectures will be used to explore and explain concepts, technologies, environment features, code examples and any other material relevant to the course that requires in class coverage.

A critical part of the module is the practical lab sessions where students will implement development concepts covered in lectures. Practical exercises will be completed within the lab , with practical assignments completed outside of class time, in the students own time. The students will use an Integrated Development Environment ni the lab to support their programming work if appropriate for the platform selected.

A tutorial session will also be included to provide a discussion forum where students can explain the problems that have had in lab sessions or with concepts covered in lectures. The tutorial session will also be used to introduce new material as needed to supplement material covered in lecture time.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3026	Mobile Software Development

Module content:

Technologies for mobile software development have and continue to develop and change rapidly. Therefore, no specific mobile platform is prescribed. The content is described generically, and will be covered for the particular platform used when the module is taught:

- Mobile phone development: typical applications, technologies, usage;
- Mobile communications technologies: local (e.g. Bluetooth) and network technologies (e.g. GSM, 3G);
- Use of the relevant IDE for application development;
- Mobile GUI components such as widgets, containers, menus;
- Mobile GUI Layouts;
- Mobile communications via the Internet;
- Using persistent data with mobile applications;
- Design consideration for mobile application development.

Module Assessment

This module will be based on 50% continuous assessment and 50% written examination. The continuous assessment will be based upon practical assignments that will reinforce and assess the practical learning outcome of application development. The written exam will test the student's understanding of the concepts covered of application development but will also assess the learning outcomes related to wider knowledge of mobile phone technologies.

Essential Reading: (author, date, title, publisher)

This will be prescribed by the lecturer at the beginning of the mobile as it will be specific to the mobile platform used.

Supplemental Reading: (author, date, title, publisher)

As per essential reading.

Web references, journals and other:

Relevant content references will be provided to students to support their learning activities

Further Details:

This module will run over one semester. There will be one weekly lecture, a two hour lab session and a one hour tutorial.

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3046	Universal Design and Assistive ICT

8.5.6. Universal Design and Assistive ICT

Module author:

John Gilligan, Dave Carroll, Damian Gordon, Ciaran O'Leary

Module Description:

This module embraces the paradigm of Assistive Technology. Its subject matter ranges from Disability Awareness, an overview of technology, the legal social and economic imperative of Accessible IT, Human Computer Interaction, Universal Design, Accessible Software and Accessible Web Design. It is intended to provide the learner with the necessary understanding and skills to develop Accessible User interfaces to Application Software and Accessible Web pages. The learner should understand the challenges of developing Inclusive IT solutions and be able to critique available solutions.

Module aim

The aim of this module is to:

- Describe the context of Disability and explore the issues involved in Assistive Technology.
- To investigate the specific challenges of an inclusive Information Technology society.
- To demonstrate the relevance of the learners own IT skills to facilitating an inclusive Information Technology society.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Describe the social and legal imperatives with regard to Assistive Technology.
- Appreciate the environmental constraints of people with disability.
- Describe the various technologies and specialised interfaces to assist people with disability.
- Describe principles of user interaction and identify various disabilities that impact the Human Computer Interface.
- Assess the accessibility of software within a universal design framework.
- Design and implement software interfaces with improved accessibility.
- Design and construct inclusive Web sites.

Learning and Teaching Methods:

This module will be delivered using a combination of Lectures, Practicals, Case Studies, Role Playing, Discussions, and Thinking Skills. The main theoretical material will be delivered in lectures. The

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU3046	Universal Design and Assistive ICT

practical work to assess the accessibility of software, and design and implement accessible software interfaces evaluate accessibility, and inclusivity of Web sites will be carried out in laboratory sessions.

Module content:

- *Disability Awareness:* Overview of different conditions. Analysis of requirements imposed by these conditions. Social, Legal and Economic Models of Disability. Service Delivery Models.
- *Technical Overview:* Relevant Technologies. Product Review. Software Support Systems. Hardware Overview. Specialized Interfaces. Augmentative Alternative Communication: Switch Access: Voice Recognition: Text to Speech Computer Vision: Environmental Control and Independent Living.
- *Specialized Interface design:* Analysis of Requirements. Technical Constraints. Design Issues. Universal Design, Challenging the HCI user Model. Enhanced models of HCI.
- *Inclusive IT* Universal Design Access issues, Matching Person and Technologies, Frameworks, Accessible Databases, Web Access. Usability, Technical Support.
- *The Education Paradigm* Connecting to Learn, Education and Technology, Learning Styles, Individual Education Plans.

Module Assessment

Written Examination 70%; Practical 30%

Essential Reading:

Cook and Hussey (2007), Assistive Technology Principles and Practice, (3rd Ed), Mosby.

Supplemental Reading:

- Scherer, Marcia J. (2003) Connecting To Learn, Educational and Assistive Technology for People With Disabilities, American Psychology Association
- Web references, journals and other:
  National Disability Authority WWW.NDA.IE
  Web Access Initiative www.w3.org/wai
  Higher Education Disability Site www.Ahead.ie
  Central Remedial Clinic www.crc.ie

Further Details:

Duration one semester: 2 Lectures 1 Practical per Week.

Date of Academic Council approval ....

8.6. Stage 3 Syllabus (Internship stream)

Students who choose the Internship stream forgo the right to exit on completion of Stage 3.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		20	CMPU3021	Internship (Third Year)

8.6.1. Stage 3 Internship

Module author: Paul Doyle

Module Description:

This module is linked with SPEC4901 and is the first half of a 2 year industrial placement which provides students with the opportunity to obtain professional industry experience for 2 days per week.

Module aim

The goal is to give students the opportunity to gain a structured learning experience within a professional / industrial environment so that they may consolidate and broaden the knowledge that they have gained in their academic studies to date.

Learning Outcomes:

On completion of this module, the learner will be able to

- Work in a professional environment
- Demonstrate an understanding of a particular project or area of work that they have been involved in
- Reflect on the learning experience and outcomes of their Internship
- Produce a professional report describing the details and experiences of their placement
- Describe the mission statement for the placement company
- Demonstrate an understanding of the company’s business model

Learning and Teaching Methods:

The students are brought through a six week, hour long induction course on writing a professional CV and preparing for interviews prior to obtaining an internship. For students who are offered an internship with a company, the learning experience is taken offsite as an industrial placement. During that time students may be working as part of a team or individually at the placement company, depending upon the individual requirements of the company. Students are assigned a DIT Internship Monitor for the duration of their internship.

Module content:

- Professional Development
- Structured reporting on work performed
- Teamwork within a professional organisation
- Reflective writing and reporting

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
		20	CMPU3021	Internship (Third Year)

- Presentation of work
- IT skills specific to the Internship

Module Assessment

Assessment will be based on continuous assessment. Marks will be allocated as follows

- Continuous Assessment (100%)

The continuous assessment element of this course is comprised of regularly submitted reports by students on internship, an end of year written report and an end of year presentation. The end of year presentation and report is jointly assessed by the Industrial placement monitor and the industrial placement coordinator.

Essential Reading:

Jennifer A. Moon, 2004 A Handbook of Reflective and Experiential Learning: Theory and Practice ,

Paul Doyle, DIT School of Computing Industrial Placement Handbook

Supplemental Reading:

David Boud, Reflection : 1985, Turning Experience into Learning, Routledge

Dannelle D. Stevens , Joanne E. Cooper, 2009, Journal Keeping: How to Use Reflective Writing for Learning, Teaching, Professional Insight and Positive Change , Stylus Publishing

Web references, journals and other:

http://www.comp.dit.ie/industrialplacement

http://www.dit.ie/careers/

Further Details:

Duration:	Placement covers year 3 and year 4. This descriptor covers Stage 3 and is linked to **SPEC4901**
Contact Hours:	Students will work in industry for 2 days per week

Date of Academic Council approval ……………………….

8.7. Stage 4 Syllabi (Core)

MODULE CODE	TITLE	CREDITS	SEMESTER
CMPU4002	Advanced Data Management	5	S1
CMPU4007	Advanced Security 1	5	S1
CMPU4049	System and Database Administration	5	S2
CMPU4026	Enterprise Systems Infrastructure & Architecture	5	S2
CMPU4029	Fourth year project	20	S1 & S2

Table 18 Core syllabi Stage 4

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
CMPU2007, CMPU3010		5	CMPU4002	Advanced Data Management

8.7.1. Advanced Data Management

Module author: Patricia O'Byrne and Databases Group

Module Description:

This module develops the student's ability to manipulate stored data and reformat data for storage or transport. The student learns to optimise queries in a relational database, use middleware to attach between client and server, implementing constraints and propagating errors where appropriate. The student learns to evaluate the correct storage and transport format for data, looking at a range of techniques such as relational data, object-oriented data and XML.

Module aim

The aim of this module is to…

- Convert data from one model paradigm to another.
- Store, manipulate and retrieve data from objects and XML in a relational database.
- Optimise the storage, retrieval and transport of data.

Learning Outcomes:

On completion of this module, the learner will be able to……………

- Demonstrate an understanding of Relational Algebra and calculus, applying them to complex queries to solve and optimise them.
- Evaluate and discriminate between the methods of constraint implementation and checking.
- Store and manipulate objects in an object relational database.
- Convert object models to relational models.
- Evaluate the appropriateness of storage structures to different application areas.
- Discuss and implement storage structures and models in XML, and how to query them using XPATH.
- Evaluate the application of SQL/XML and XQuery in converting data between relational and XML structures.

Learning and Teaching Methods:

This module will be delivered over four hours per week for one Semester. This will consist of one hour lecture, one hour tutorial and one two-hour laboratory session. During the lecture, new material will be disseminated. Techniques will be applied in the practical laboratory sessions. Where appropriate, students will work in teams. The tutorials will review theory and practice, to prepare them for practical and examination work, in accordance with the requirements of the students. A virtual learning environment and a central database may be used for this module.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
CMPU2007, CMPU3010		5	CMPU4002	Advanced Data Management

Module content:

Expressing queries in relational algebra and relational calculus. Using DeMorgan's theorem to rephrase queries. Advanced queries and optimising them. Verification and validation using client-side processing, triggers and table constraints. Types and Object representation in a large-scale hybrid relational database. Conversion of data models from object to relational and from relational to object and associated issues. XML, SQL/XML, XQuery, XPath. Criteria for choosing storage and transport mechanisms for data.

Module Assessment

Module will be assessed by both non-exam and examination assessment. Students must pass the entire module.

Essential Reading:

Elmasri & Navathe, 2007, 'Fundamentals of Database Systems' 5th Ed., Addison Wesley.

Depending on database being used:

- Sunderraman, R., 2007, 'Oracle 10g Programming: A Primer, Addison Wesley
- Sack, J., 2008, SQL Server 2008 Transact-SQL Recipes: A Problem-Solution Approach, APress
- Geschwinde, E., 2001, PostgreSQL Developer's Handbook, 2nd Ed., Sams

Supplemental Reading:

Connolly & Begg, 2009, 'A practical approach to design, implementation and management' 5th ed. Addison Wesley

Web references, journals and other:

www.oracle.com

www.postgresql.org

http://msdn.microsoft.com/en-us/library/bb510741(SQL.100).aspx

webcourses.dit.ie

Further Details:

e.g. laboratory sessions limited to 20 – 25 by lab size. 4 contact hours per week. To be delivered in one semester.

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4007	Advanced Security 1

8.7.2. Advanced Security 1

Module author: Dr. Fredrick Mtenzi.

Module Description:

Cryptography is the science of providing security for information through the reversible transformation of data. It can also be defined as the study and practice of scrambling information in a manner that is difficult to unscramble, and making scrambled information unintelligible. It is used as the basis of much computer security, in that it can be used to keep information confidential, and also preserve the integrity of data, particularly when being stored or being transmitted. The development of digital computing revolutionized the use cryptography, and has allowed its usage in most computing activities.

Apart from government, military and organisations, users need security provided in cryptography. For example, they will need this kind of security because they may be designing new software, discussing a marketing strategy, or planning a hostile business takeover. The aim of this course is to provide a detailed understanding of the issues involved in Steganography and Cryptography algorithms and techniques used in implementation. The course will introduce and make use of the relevant mathematical concepts such as number theory and finite fields. Practical applications of Steganography and Cryptography will be demonstrated and experimented by students as part of the laboratory exercises or assignments.

Module aim

The aim of this module is.

- to introduce the students to the principles of cryptography and steganography,
- to give the students a thorough understanding of cryptography and steganography algorithms,
- to provide them with an in-depth understanding of how cryptography and steganography provides security in real-life applications.

Learning Outcomes:

On completion of this module, the student will be able to:

- Describe the underlying principles of different cryptography and steganography algorithms.
- Design and implement simple cryptography algorithms using any high level programming language.
- Evaluate the effectiveness of cryptography algorithms according to well known security requirements.
- Apply relevant/appropriate mathematical results in the design and implementation of cryptography algorithms.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4007	Advanced Security 1

- Describe the principles and deployment of steganography techniques as applied in real-life security.
- Compare and contrast on the effectives and efficiency of cryptography and steganography algorithms.
- Recognise and justify the different scenarios of deploying cryptography and steganography algorithms.
- Select and combine different cryptography algorithms in order to achieve highly secure algorithms.
- Asses and rank cryptography and steganography algorithms according to the protection they provide in real-life applications.
- Explain the principles behind and usage rationale of various encryption techniques.

Learning and Teaching Methods:

In this module a number of teaching methods may be employed including lectures, practical sessions, tutorials and case studies. At least one industrial seminar may be arranged. Focus should be placed on empowering the students to develop their skills independently of the presence of a tutor or lecturer.

Module content:

- Introduction to cryptography
- Number theory, Discrete logarithms and Elliptic Curves
- Steganography
- Block Ciphers and Advanced Encryption Standard, Confidentiality Using Conventional Encryption.
- Public-Key Cryptography and RSA, Key management, Message Authentication and Hash Functions, Hash and Mac Algorithms, Digital Signatures and Authentication Protocols.

Module Assessment

This module should have a 60% weighting for the examination and a 40% weighting for the continuous assessment. While it is important that the student can demonstrate their technical ability with coursework, it is equally important that they demonstrate an understanding of the theoretical aspects of cryptography.

Essential Reading: (author, date, title, publisher)

William Stallings, 2005, Cryptography and Network Security: Principles and Practice, 5th ed, Prentice Hall

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4007	Advanced Security 1

Supplemental Reading: (author, date, title, publisher)

Niels Ferguson, Bruce Schneier, Tadayoshi Kohno, 2010, Cryptography Engineering: Design Principles and Practical Applications, John Wiley & Sons. Inc.

Behrouz A. Forouzan, 2008, Cryptography and Network Security, McGraw-Hill International Edition.

Wade Trappe, Lawrance C. Washington, 2006, Introduction to Cryptography with Coding Theory, Prentice Hall.

Web references, journals and other:

IEEE security & privacy

ACM Transactions on Information and System Security (TISSEC)

http://www.sans.org/rr/

Further Details:

The module will run over one semester, with three contact hours per week. Two hours for lectures and one hour for the laboratory.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4049	System and Database Administration

8.7.3. System and Database Administration

Module author: Patricia O'Byrne, Paul Kelly

Module Description:

This module teaches the student how to administrate a large database installation on an existing operating system, taking into account factors affecting security, scalability, availability, capability and performance costs.

Module aim

- The student will gain the necessary skills/knowledge to evaluate/assess/specify the security, performance, and availability and scalability requirements of an organisational database system.
- The student will plan, configure the Operating system, install appropriate database software, instantiate a server, partition, index and replicate database artefacts, provide secure user access, networking, archiving, backup and recovery that is appropriate to an organisation's needs.
- The student will learn to evaluate the complex trade-offs required between providing service levels and controlling costs with respect to database administration.

Learning Outcomes:

On completion of this module, the learner will be able to……………

- Evaluate organisational requirements for a data storage system.
- Choose the most appropriate configuration for integrating a large DBMS server onto an existing network, taking performance, security and cost into account.
- Partition, index, distribute and replicate database artefacts as appropriate.
- Configure system and database objects to maximise security, availability and cost.
- Configure and enable local and network user and group access, privileges and auditing, both to Operating System and database services.
- Plan and execute an archiving, backup and recovery strategy for all anticipated levels of failure, trading service level requirements with performance and space requirements.
- Develop an operational and maintenance plan, monitoring performance, security and cost with reference to Service Level agreements.

Learning and Teaching Methods:

During the lecture, new material on the theories and practices in systems and database administration will be disseminated. Techniques will be applied in the practical laboratory sessions. A virtual learning environment will be used for this module.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4049	System and Database Administration

Module content:

Operating System and Database System administration requirements. Interaction between the Operating System and the database. Roles of the System Administrator and Database Administrator. Organisational considerations when choosing a database installation. Configuration of the Operating System for the Database server. Choice and configuration of the database server for an organisation's requirements. Anatomy of a database. Startup, shutdown, monitoring and management of the database and operating system. Persistent storage strategies to optimise speed and cost. Database performance design, profiles and system security. Auditing techniques and implications. Networking techniques and implications. Archiving, backup and recovery with reference to service level agreements.

Module Assessment

Module will be assessed by both non-exam (50%) and examination (50%) assessment. Students must pass the entire module. Non-exam assessment will include practical work and a research component.

Essential Reading:

Elmasri & Navathe, 2010, 'Fundamentals of Database Systems' 5th Ed., Addison Wesley. Or

Connolly & Begg, 2009, 'Database Systems: A Practical Approach to Design, Implementation and Management', 5th Ed., Addison Wesley.

Supplemental Reading:

Watson, J., 2008, OCA Oracle Database 11g Administration I Exam Guide (Exam 1Z0-052) , McGraw-Hill Osborne Media

Web references, journals and other:

www.oracle.com

webcourses.dit.ie

Further Details:

e.g. laboratory sessions limited to 20 – 25 by lab size. . To be delivered in one semester. 4 contact hours per week allocated as 2 hours lab, 1 hour lecture, 1 hour tutorial.

Date of Academic Council approval ……………………………

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4026	Enterprise systems infrastructure and architecture

8.7.4. Enterprise systems infrastructure and architecture

Module author: Enterprise systems infrastructure and architecture group.

Module Description:

This module covers the range of enterprise architectures and enterprise application systems that are used within organisations. It introduces the students to basic strategic formulation and implementation and how to evaluate an enterprise system's ability to formulate and implement different strategic objectives.

Module aim

The aim of this module is to provide students with knowledge of the purpose design and context of enterprise systems and how to align enterprise systems infrastructure to organisational strategies.

Learning Outcomes:

On completion of this module, the learner will be able to:

- Explain the concepts of enterprise wide IT systems and enterprise system architecture concepts.
- Describe the various processes within the organisation supported by Enterprise systems.
- To formulate and implement I.T. strategies for an organisation.
- Appraise how elements of enterprise systems can be used to produce information required to support organisational strategy.
- Evaluate the major considerations for enterprise systems infrastructure and architecture selection.
- Analyse and evaluate how to align enterprise systems infrastructure and architecture with organisational strategic formulation and implementation.

Learning and Teaching Methods:

This module will employ teaching methods and learning situations such as lectures, seminars and tutorials, as well as case studies.

Module content:

- *I.T. systems in the Organisation*: Describe common types and application areas: e.g. database systems, ERP systems, decision support systems, Internet/intranet/extranet systems and client server systems.
- Organisational enterprise system processes, including supply chain related processes, finance and human resources and IT infrastructure.
- *Strategic Management:* A description of how organisations formulate, implement strategies.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4026	Enterprise systems infrastructure and architecture

- Strategic information systems: Evaluate how data warehousing, data mining and OLAP are used to derive information critical to strategic formulation.
- *Strategic enterprise systems:* discussion on how I.T. infrastructure aid organisations to develop and choose strategic decisions.
- *Enterprise System architecture requirements:* a description of the major critical factors used to evaluate different enterprise system architectures.

Module Assessment:

Continuous assessment (30%): It will take a form a formal report discussing and analysing different enterprise systems architectures that are appropriate for organisation(s) with specific strategic goals

Written examination (70%): A two hour end of semester written exam.

Essential Reading:

David F.R. 2007 Strategic Management: concept and cases 11 edition, Pearson ISBN: 0-13-186949-3.

George M. Marakas (2005) Decision Support Systems in the 21st Century, 2nd Edition

Supplemental Reading:

Recent readings to be supplied before the commencement of the module

Web references, journals and other:

To be supplied during the running of the module.

Further Details:

2-hour lecture, and 1-hour tutorial. To be delivered in one semester.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4029	Fourth Year Project

Fourth Year Project

Module author: **Damian Bourke.**

Module Description:

This module is required for all students intending to exit with a stage four qualification. This module requires students to propose, design, implement, test, document and present a complex software project to demonstrate the level of knowledge gained over the course of their studies. The project must be an independent piece of work which is both coherent and well structured. The student is expected to be self-motivated so as to drive this work through to completion. They are expected to identify the key areas of the project and to make real decisions that will ultimately affect the end deliverable. The project deliverable will be an assessable, independent body of work that will demonstrate the student's ability to work on their own and their ability to communicate key aspects of the project.

Module aim

The aim of this module for the student is to either successfully completes a large-scale software system implementation or that they can critically research an area of Computing. Completing a large-scale software system implementation includes all aspects of software development including, but not limited to; Analysis, Design, Development, Implementation Integration, Documentation and Maintenance. Critically researching an area of Computing includes the correct use of the Scientific Method. This involves; using a well designed, defined and identified research method (qualitative, quantitative or a mixture of both); well-designed data collection and analysis (if applicable) and critical analysis of the research process. In either case the student undertakes a project which produces one of the above deliverables as well as producing an academic quality report outlining the key steps undertaken, the key decisions that were made, an evaluation of of the outcomes, and the student's conclusions about the success or failure of the delivered system/critical research.

Learning Outcomes:

On completion of this module the student will be able to demonstrate the ability to:

- Define a problem area and write a project proposal
- Evaluate similar systems to their proposal identifying a set of clear user requirements
- Undertake research of the problem area to determine the boundaries and scope of the project.
- Undertake research of the proposed solution to identify appropriate technologies to use
- Select and implement a formal design methodology.
- Write a project plan and project manage the project to completion.
- Develop a fully operational software/hardware system.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4029	Fourth Year Project

- Critically evaluate the delivered system against the proposed objectives and requirements.
- Produce a report (in English) of academic quality with appropriate referencing.
- Present and defend their findings.

Learning and Teaching Methods:

The project module involves a combination of seminars, self-paced work, research, and weekly meetings with their assigned supervisor. This is primarily a self-learning module with the supervisor providing guidance and feedback to the student throughout the project. The student is expected to state at the outset the objectives for the project and to meet these objectives throughout the year. The student is also given explicit verbal feedback at the Interim report stage which is a presentation by the student on the work completed so far and the work planned in the future.

In addition to the dedicated supervisory resources the School will also provide a second reader for the purposes of assessing the student's work.

Module content:

Students are provided with detailed guidelines on the project process. These guidelines detail the requirements of the project in terms of the project content and deliverables and they also provide guidance on the content and format of the Interim Report and Final Project Manual. The guidelines also explain the roles of the key stakeholders including the student, the supervisor, second reader and project co-ordinator.

Module Assessment

The module assessment will comprise 100% Continuous Assessment.

There are two key milestones within the project lifecycle at which the student is assessed. At the Interim Report stage the student is assessed on their research findings, system design, testing and evaluation criteria, report writing and project management. The assessment is based on a key set of criteria set-out in the guidelines document.

At the Final Report stage the student is assessed on any additional research undertaken post-design, the system implementation phase, testing and evaluation phase, report writing, communication skills, project management, and the quality, complexity and usability of the delivered system. The assessment is based on a key set of criteria set-out in the guidelines document.

Essential Reading:

Damian Bourke, 2010. The Project Guidelines

Kathy Schwalbe, 2005, A Guide to the Project Management Body of Knowledge: Course Technology

Strunk, W. and White, E. B., 2000, *The Elements of Style* (4th ed.) Longman

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4029	Fourth Year Project

Booth, W., Colomb, G.C., Williams, J.M., 2003, *The Craft of Research*, University of Chicago Press Dawson, C., 2005, "A Practical Guide to Research Methods: A User-friendly Manual for Mastering Research Techniques and Projects", How To Books

Supplemental Reading:

Evaluating Software Architectures Methods Studies

Web references, journals and other:

Further Details:

This module is to be delivered over two semesters with dedicated supervisor contact hours.

Date of Academic Council approval…………………………

8.8. Stage 4 Syllabi (Full-time Stream)

In addition to the core modules, all full-time students must study the following modules:

MODULE CODE	TITLE	ECTS	SEMESTER
CMPU4050	Systems Integration	5	S1
CMPU4008	Advanced Security 2	5	S2

Table 19 Stage 4 Full-time core modules

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4050	Systems Integration

8.8.1. Systems Integration

Module author: **Ronan Bradley.**

Module Description:

This module covers the issues relating to the integration of diverse computer systems in a modern networking environment and the technologies commonly used to address them. The module focuses on system and network services such as centralised authentication and authorisation, naming and file sharing

Module aim

The aim of this module is to provide the student with both a theoretical understanding of the design and operation of the system and network services required to integrate a heterogeneous network environment and hands-on experience of configuring these services to create an operational network supporting common services such as authentication, authorisation, naming and file sharing.

Learning Outcomes:

On completion of this module, the learner will be able to

- Describe and compare the purpose, configuration and operation of common system and network services such as DNS, ntp, LDAP, NIS, nfs and SaMBa
- Configure common network services to enable file sharing, naming and centralised authentication and authorisation
- Configure system authentication and authorisation mechanisms to use network services to create a single authentication/authorisation domain
- Analyse system performance to identify failures or performance issues relating to the system or the network services in use

Learning and Teaching Methods:

Course delivery involves a combination of lectures, self-paced study and relevant laboratory work. Included in the materials is a significant practical aspect to the course where learners will gain exposure to industry standard software systems with an opportunity to explore the issues surrounding systems integration in general.

Module content:

- Authentication and authorisation including
- User Management on UNIX and Windows Systems including adding and removing users and groups
- Configuration and integration of network authentication mechanisms on UNIX and Windows

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4050	Systems Integration

- Operation of common performance analysis tools to analyse the current performance characteristics of the system and network for the identification of actual and potential problems
- File sharing services for the integration of UNIX and mixed UNIX, Windows networks
- Configuring of file sharing services on UNIX such as SaMBa and nfs
- Criteria for selection of file sharing services in a given network environment
- Operation of the Domain Naming System and configuration using bind on a UNIX system
- Overview of other network services commonly used (such as NTP)

Module Assessment

Assessment by a combination of Continuous Assessment and Exam

Exam 60%, Continuous Assessment 40%

Essential Reading:

BELLOMO, M. M., James. (2000). Network and System Integration for Dummies, John Wiley & Sons.

HARVEL, L., WEBB, David., FLYNN, Steven., WHITEHURST, Todd. (2000). The UNIX and Windows 2000 Handbook: Planning, Integration and Administration, Prentice Hall PTR.

Supplemental Reading:

Marty Poniatowski, 2002, UNIX User's Handbook, 2nd ed., Prentice Hall PTR

ISBN: 0-13-065419-1

Steve Shah, Wale Soyinka. 2005, Linux Administration: A Beginner's Guide. Osborne/McGraw-Hill

ISBN: 0-07-226259-1

Web references, journals and other:

Relevant web references, journals and other will be indicated during the teaching of the module.

Further Details:

Single Semester Module

4 contact hour per week, 2 hours lecture, 2 hours lab

Date of Academic Council approval ……………………….....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4008	Advanced Security 2

8.8.2. Advanced Security 2

Module author: Dr. Fredrick Mtenzi.

Module Description:

In the last few decades, computers, mobile devices, networks and the Internet have become an integral part of our social fabric. As computer, networks and the Internet are used for communication and for varieties of online interactions and transactions, security has become the key issue in today's information technology world. Information security is required at all levels – the personal level, corporate level, and country level. There is a steady rise in the occurrence of security attacks. Security is a broad issue which is becoming increasingly important as computer networks become more widespread. It encompasses computer- and network-related crime, privacy issues, trust and confidence, and dependability of critical infrastructures. The evidence of the security threat is growing and new vulnerabilities are found each day. Those interested in exploiting these vulnerabilities are becoming a well-organized. Security threats are taking a variety of forms, including espionage, hacking, identity theft, crime, and terrorism. The level of sophistication and speed of development of the tools being used to create security breaches and attacks are growing exponentially.

With the increasing concern for safety and integrity of information against security attacks, it has become mandatory that organizations follow strict guidelines and security framework to assure the safety and protection of data and systems. To address these needs this module has will provide a detailed understanding of the issues involved in providing systems security. Further, the module will examine and critically analyze the security challenges faced by the society and the computing industry. It will elaborate and demonstrate the role of security in design, implementation and deployment of secure systems.

Module aim

The aim of this module is.

- to introduce the students to the security principles,
- to give the students a thorough understanding of the network security issues,
- to provide them with sound knowledge of provision of an enterprise secure systems, security audit and compliance,
- to provide them with an in-depth practical enterprise security.

Learning Outcomes:

On completion of this module, the student will be able to:

- Define and explain the components of network security,
- Discuss, relate and organise the fundamental concepts of Security,

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4008	Advanced Security 2

- Classify and analyse the nature and threat from malware,
- Design and develop best practice techniques of mitigating security threats,
- Critically analyze different aspects of security such as security testing, evaluation, auditing and policies,
- Assess security compliance of an organisation
- Compare and contrast international and regional security standards, legislations and laws.
- Asses and rank different systems security approaches according to the protection they provide in real-life applications.
- Recognise and justify the different scenarios of deploying systems security approaches.

Learning and Teaching Methods:

In this module a number of teaching methods may be employed including lectures, practical sessions, tutorials and case studies. At least one industrial seminar may be arranged. Focus should be placed on empowering the students to develop their skills independently of the presence of a tutor or lecturer.

Module content:

- Authentication Applications,
- Electronic Mail Security,
- Internet Protocol Security,
- Web security,
- Intruders, Crimeware, Firewalls
- Security Policies, Standards, Compliance
- Security Metrics and Auditing
- Penetration Testing
- Defences to security attacks
- The effect of Technological developments on Security

Module Assessment

This module should have a 50% weighting for the examination and a 50% weighting for the continuous assessment. While it is important that the student can demonstrate their technical ability with coursework, it is equally important that they demonstrate an understanding of the theoretical aspects of security.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4008	Advanced Security 2

Essential Reading: (author, date, title, publisher)

William Stallings, 2005, Cryptography and Network Security: Principles and Practice, 5th ed, Prentice Hall

Supplemental Reading: (author, date, title, publisher)

Seymour Bosworth and M.E. Kabay, 2009, Computer Security Handbook, John Wiley & Sons. Inc.

Andrew Lockhart, 2004, Network Security Hacks 100 Industrial-Strength Tips & Tools, O'Reilly

Markus Jakobsson, Zulfikar Ramzan, 2008, Crimeware: Understanding New Attacks and Defences, Symantec Press.

Ed Skoudis and Tom Liston, 2006, Counter Hack Reloaded: A step-by-step Guide to Computer Attacks and Effective Defences, Prentice hall

Bruce Schneier, 2004, Secrets and Lies: Digital Security in a Networked World, Hungry Minds Inc

Web references, journals and other:

IEEE security & privacy

ACM Transactions on Information and System Security (TISSEC)

http://www.sans.org/rr/

Further Details:

The module will run over one semester, with three contact hours per week. Two hours for lectures and one hour for the laboratory.

Date of Academic Council approval …………………………..

8.9. Stage 4 Syllabi (Full-time Options)

Paired options (1 pair to be taken)

MODULE CODE	TITLE	CREDITS	
CMPU4030	Games Engines 1	5	S1
CMPU4031	Games Engines 2	5	S2
CMPU4043	Rich Web Application Development	5	S1
CMPU4023	Enterprise Application Development	5	S2
CMPU4020	Digital Audio	5	S1
CMPU4042	Music Technology	5	S2
CMPU4032	Geographical Information Systems	5	S1
CMPU4046	Spatial Databases	5	S2
CMPU4017	Computer Graphics	5	S1
CMPU4034	Image Processing	5	S2

Table 20 Stage 4 Paired Options

An amendment sanctioned by the Programme Committee in December 2012 sanctioned the following:

- The addition of the module Bioinformatics as an optional module for full-time students.
- Students may choose any of the optional modules in Semester 2, provided they fulfil the pre-requisites for that module. While pairs show modules that complement each other, students are not restricted to choosing the module in Semester 2 that is paired with the module chosen in Semester 1.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4030	Games Engines 1

8.9.1. Games Engines 1

Module author: Dr Bryan Duggan

Module Description:

The purpose of this module is to introduce students to the core concepts required to program 3D game engines. This course combines a foundation in maths and physics programming for 3D games, 3D graphics and artificial intelligence for games. This course has a technical focus and gives students the opportunity to learn practical 3D games development from the ground up. This course explores analytical geometry, linear algebra, matrices, Newtonian physics and quaternions and applies these techniques to problems in game engine programming using standard API's such as DirectX and OpenGL.

Module aim

The aim of this module is for students to learn the fundamentals of 3D game engine programming.

Learning Outcomes:

- Demonstrate an understanding of trends and advances in computer game technology.
- Solve common problems in computer games such as entity position, orientation and movement by using a variety of mathematical tools.
- Program a scene in 3D using standard API's.
- Represent and program a flexible camera suitable for a variety of game types (FPS and RTS).
- Make use of a physics engine to perform physics integration
- Make use of an audio engine to perform functionality such as 3D positional audio and occlusion
- Apply software design patterns to computer games engines
- Use UML and object oriented design to model computer games engines.
- Critically analyse how the techniques learned on the course are used in commercial games.

Learning and Teaching Methods:

Class time is split into a series of interactive "studio classroom" based lectures and practical problem solving in labs. In lectures, students have access to a PC, with appropriate software and development kits, so that material and examples can be examined in a live environment. In addition, students will be expected to proactively and independently seek out resources on the internet and from the library to supplement their own learning.

A Virtual Learning Environment (VLE) will be employed to distribute all teaching materials and to support student interaction with both other students and academic staff. Where new material is not presented in lectures, such material will be made available through the VLE and students are expected to proactively use this resource.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4030	Games Engines 1

Module content:

Mathematics for games

Vectors, trigonometry, matrices, translations, rotations, scaling, linear algebra, equations of lines, rays and planes, intersections of lines rays and planes

3D graphics

Introduction to the game loop, mathematics programming, the rendering pipeline, vertex buffers, texturing, meshes, particle systems, camera models.

Physics Simulation

Newtonian physics (force, acceleration, velocity, position, mass). Friction. Hamiltonian mechanics (quaternions, torque, angular velocity and acceleration, equations of motion and rotation.

Game Engines

Asset management, design patterns, programming a flexible camera, UML, object oriented game design.

Middleware

Physics - rigid bodies, joints, integration, collision detection. Audio – 3D positional audio, occlusion

Module Assessment

This module has a 50% weighting for the examination and a 50% weighting for the continuous assessment. Continuous assessment will consist of a single significant assignment, which may be individual or team based. Students will be expected to document and present their assignment work in the form of an ePortfolio or a blog. Students will be expected to enter their assignment into national and international competitions such as the XNA Ireland Challenge and the Imagine Cup. While it is important that students can demonstrate their technical ability through continuous assessment, it is equally important that they demonstrate both an understanding of the mathematical concepts and an appreciation for how the techniques learned on the course are used in commercial games. This will be assessed in the examination. Students will not be required to write code in the examination, but will be required to solve problems using the mathematics and physics concepts learned and to critically analyse examples of the techniques learned on the course in commercial games.

The suggested method of assessing the individual learning outcomes for the module is as follows:

Essential Reading:

Introduction to 3D Game Programming with DirectX 10, Frank D Luna, Jones & Bartlett Publishers; 1 edition (October 25, 2008)

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4030	Games Engines 1

Game Physics Engine Development, Second Edition: How to Build a Robust Commercial-Grade Physics Engine for your Game, Ian Millington, Morgan Kaufmann; 2 edition (August 6, 2010)

Supplemental Reading:

Advanced Animation with DirectX (Focus on Game Development), Jim Adams , Course Technology PTR; 1 edition (May 22, 2003)

Web references, journals and other:

http://gamedevelopers.ie

http://gamesfleadh.ie

http://imaginecup.com

http://dreamspark.com

http://seriousgames.ie

Further Details:

4 hours per week for one semester

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
CMPU4030		5	CMPU4031	Games Engines 2

8.9.2. Games Engines 2

Module author: Dr Bryan Duggan

Module Description:

The purpose of this module is to build on the skills learned in Game Engines 1 so that students can learn how *higher order* behaviours are implemented in computer games. This course begins by exploring perception and path finding and continues with a comparison of techniques used to implement higher order NPC (Non Player Character) decision making. This course has a strong practical and technical focus and gives students the opportunity to learn practical games development at a low level.

Module aim

The aim of this course is to build on the knowledge learned in Game Engines 1 so that students can learn how higher order behaviours are implemented in computer games.

Learning Outcomes:

- Compare approaches to implementing perception and propose optimised solutions to perception problems.
- Critically analyse how perception is implemented in commercial computer games.
- Program simulations using steering behaviours
- Implement a variety of path following and path finding algorithms.
- Demonstrate an understanding of the issues involved in practical, optimised path finding.
- Use UML and object orientated design to model both technical architectures and behaviours in computer games.
- Propose architectures for NPC higher order decision making.
- Critically analyse how the techniques learned on the course are used in commercial games.

Learning and Teaching Methods:

Class time is split into a series of interactive “studio classroom” based lectures and practical problem solving in labs. In lectures, students have access to a PC, with appropriate software and development kits, so that material and examples can be examined in a live environment. In addition, students will be expected to proactively and independently seek out resources on the internet and from the library to supplement their own learning.

A Virtual Learning Environment (VLE) will be employed to distribute all teaching materials and to support student interaction with both other students and academic staff. Where new material is not presented in lectures, such material will be made available through the VLE and students are expected to proactively use this resource.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
CMPU4030		5	CMPU4031	Games Engines 2

Module content:

Perception

Audio perception models, visual perception models, algorithms for visual perception, algorithms for intersection of lines, rays and planes.

Steering behaviours

Unit steering behaviours (seek, pursuit, arrive path following, obstacle avoidance, interpose, hide). Group behaviours (separation, alignment, cohesion, flocking). Combining steering behaviours (weighted sum, prioritised dithering, spatial partitioning).

Path following and path finding

Graphs, breath first search, depth first search, Djikstra's shortest path, the A* algorithm, data structures, optimisations, path smoothing.

Higher order NPC decision making

Finite state machines, concepts and examples, state transition diagrams, the state machine design pattern, goal driven agent architectures, composite goals, sub goals, goal arbitration.

Module Assessment

This module has a 50% weighting for the examination and a 50% weighting for the continuous assessment. Continuous assessment will consist of a single significant assignment, which may be individual or team based. Students will be expected to document and present their assignment work in the form of an ePortfolio or a blog. Students will be expected to enter their assignments into national and international competitions such as the XNA Ireland Challenge and the Imagine Cup. While it is important that students can demonstrate their technical ability through continuous assessment, it is equally important that they demonstrate both an understanding of the mathematical concepts and an appreciation for how the techniques learned on the course are used in commercial games. This will be assessed in the examination. Students will not be required to write code in the examination, but will be required to solve problems using the mathematics and physics concepts learned and to critically analyse examples of the techniques learned on the course in commercial games.

The suggested method of assessing the individual learning outcomes for the module is as follows:

Essential Reading:

- Mat Buckland (2005), Programming Game AI by Example, Worldware Publishing Inc.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
CMPU4030		5	CMPU4031	Games Engines 2

Supplemental Reading:

- AI Game Development: Synthetic Creatures with Learning and Reactive Behaviors, Alex J. Champandard, New Riders Games (December 6, 2003)
- Pierre Rautenbach (2008) 3D Games Programming: Using DirectX 10 and Open GL, Cengage Learning Business Press

Web references, journals and other:

- http://gamedevelopers.ie
- http://gamesfleadh.ie
- http://imaginecup.com
- http://dreamspark.com
- http://seriousgames.ie

Further Details:

- 4 hours per week for one semester

Date of Academic Council approval ……………………….....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4043	Rich Web Application Development

8.9.3. Rich Web Application Development

Module author: Dr. Michael Collins

Module Description:

This module teaches how to develop rich web applications with a particular emphasis on client-side development. The module shows how key design characteristics will affect the decision on selecting specific technologies to develop a web application. Advances in underlying Internet frameworks will also be examined and how these can aid in developing sophisticated applications.

Module aim:

The aim of this module is to:

- Teach the student how to develop rich web applications.
- Teach the methods and techniques in selecting appropriate technologies according to specific design characteristics in developing sophisticated web applications.
- Provide the student with sufficient knowledge in updating existing web applications using more appropriate technologies that are specifically designed for certain environments.

Learning Outcomes:

On completion of this module, the student will be able to:

a) Develop a sophisticated web application using predominantly client-side technologies.
b) Make appropriate decisions in selecting suitable technologies to implement a complex web application.
c) Distinguish the importance of correct design patterns for web applications that involve different components.
d) Analyse existing web applications and be able to identify areas that may be improved using more suitable mechanisms.

Learning and Teaching Methods:

This module will teach the student how to implement a rich web application correctly. It will incorporate how to correctly design a web application and identify its core entities. Exercises will be given to the student to design and architect different web applications and these will supplement their theoretical knowledge. The content of this module contains a strong practical element that will involve initial theoretical material to be delivered in lectures and subsequent exercises to be completed in the laboratory. These exercises form the primary source of learning for the student so it will be expected that students complete these each week. If an exercise is not completed during scheduled laboratory time, it will be assumed that the student will complete it during their self-study time.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4043	Rich Web Application Development

The student will be given many references to books and online material that supplement content delivered during their lectures. This will form part of their self-study to be done outside of normal contact hours.

Module content:

The content of the module will include:

- Ruby – Using Ruby to develop rich web applications
- AJAX – Introduction to AJAX, AJAX and JavaScript, developing applications with AJAX, advantages of AJAX against using other client-side technologies.
- Python – Introduction to Python, using Python in rich web application development
- Adobe Flex – Developing rich and sophisticated interfaces to web applications, comparison to other similar GUI technologies, XML

Module Assessment:

- Assessment of the module will be: Written examination: 70%; Continuous assessment: 30%

Essential Reading: (author, date, title, publisher)

- Dave Thomas, Chad Fowler, Andy Hunt, 2004, Programming Ruby, 2nd Edition, O'Reilly
- Paul Barry, 2010, Head First Python, 1st Edition, O'Reilly
- Anthony T. Holdener, 2008, AJAX: The Definitive Guide, 1st Edition, O’Reilly
- Yakov Fain, Victor Rasputnis, Anatole Tartakovsky, 2010, Enterprise Development with Flex, 1st Edition, O'Reilly

Supplemental Reading: (author, date, title, publisher)

- Ian F. Darwin, Jason Brittain, 2003, Tomcat: The Definitive Guide, 1st Edition, O’Reilly and Associates
- Alaric Cole, Elijah Robison, 2010, Learning Flex 4, 1st Edition, O'Reilly
- Simson Garfinkel & Gene Spafford, 2001, Web Security, Privacy & Commerce, 2nd Edition, O’Reilly and Associates

Web references, journals and other:

Further Details:

The module will be delivered over one semester. Contact hours are:

4 contact hours per week: Lectures: (2 hours); Practical (Lab) work: (2 hours)

Date of Academic Council approval …………………………

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4023	Enterprise Application Development

8.9.4. Enterprise Application Development

Module author: Dr. Michael Collins

Module Description:

This module shows how to architect enterprise applications and how to use enterprise-level Java technologies to develop very sophisticated applications. Core design fundamentals such as concurrency, scalability and redundancy will be taught to students and how to incorporate these and other important factors in a multi-tiered web application.

Module aim:

The aim of this module is to:

- Teach the student how to design and develop sophisticated enterprise applications using several Java related technologies.
- Teach correct enterprise application design principles and how to incorporate these in developing a web application.
- To provide the student with sufficient knowledge in recognising common problems in web applications.
- Teach the student how to incorporate persistent data in a multi-tiered enterprise application model.

Learning Outcomes:

On completion of this module, the student will be able to:

a) Design and develop sophisticated enterprise applications using a variety of Java related technologies.
b) Incorporate efficient design principles in the development of a complex web application.
c) Distinguish the need for specific components in order to implement a design.
d) Implement a sophisticated Graphical User Interface (GUI) of a web application using enterprise-level Java components that are independent of the server-side technologies.
e) Maintain state in a sophisticated application.
f) Incorporate authentication in a multi-tiered application.
g) Implement session tracking using a variety of techniques and be able to select the most appropriate method based on the design.

Learning and Teaching Methods:

This module will teach how to design and develop sophisticated enterprise applications using a mixture of theory and practical work. Lectures will be used to deliver the theoretical material to the student and will include sample code to show the student how the material is implemented. References, documentation and other resources will also be provided to supplement all material delivered. It will be expected that the student conducts self-study outside of normal contact hours to complement this.

Due to the strong practical nature of this module, a large emphasis will be placed on the development of web applications in the laboratory. Each week, the student will be given a practical exercise based on material delivered in their lecture. This will compliment their overall understanding in the respective area.

Module content:

The content of the module will include:

- **Java Servlets:** Review of Servlet APIs, processing HTTP requests, HTTP Authentication with Servlets, Session management, using cookies with Servlets, using the Session API, Database management with Servlets and JDBC, comparing Servlets with other server-side technologies.
- **JavaServer Faces:** Introduction to JSF, What is JSF, JSF Features, How JSF fits for web applications, the JSF Architecture and design pattern, JSF Components, JSF Tags, JSF validation process, JSF lifecycle.
- **Enterprise Architectures:** The Model-View-Controller (MVC) design pattern, the role of Enterprise JavaBeans (EJB) in web application development. Java Persistence API in web application development. Setting up an Application Server, establishing a connection to a database.

Module Assessment:

Assessment of the module will be as follows:

- Written examination: 70%
- Continuous assessment: 30%

Essential Reading: (author, date, title, publisher)

Hunter and Crawford, 2001, Java Servlet Programming, 2nd Edition, O'Reilly

Larry Brown, Marty Hall, 2006, Core Servlets and JavaServer Pages, 2nd Edition/Volume II, Prentice Hall

Hans Bergsten, 2006, JavaServer Faces, 3rd Edition, O'Reilly

Ian F. Darwin, Jason Brittain, 2003, Tomcat: The Definitive Guide, 1st Edition, O'Reilly and Associates

Vlada Matena and Beth Stearns, 2000, Applying Enterprise JavaBeans, 1st Edition, Prentice Hall.

Supplemental Reading: (author, date, title, publisher)

Bruce Perry, 2004, Java Servlet and JSP Cookbook, 1st Edition, O'Reilly and Associates

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4023	Enterprise Application Development

Web references, journals and other:

World-Wide-Web Consortium http://www.w3.org/

Oracle – Java http://www.oracle.com/technetwork/java/index.html

Apache Group http://jakarta.apache.org

Further Details:

The module will be delivered over one semester. Contact hours are:

4 contact hours per week:

Lectures: (2 hours)

Practical (Lab) work: (2 hours)

Date of Academic Council approval ……………………….…

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4012	Bioinformatics

8.9.5. Bioinformatics

Module author: Bioinformatics group

Module Description:

This course introduces the student Bioinformatics, which uses computer databases to store, retrieve and assist in understanding biological information. Genome-scale sequencing projects have led to an explosion of genetic sequences available for automated analysis. These gene sequences are the codes, which direct the production of proteins that in turn regulate all life processes such as normal organism development and many pathological conditions. The student will be shown how gene expressions lead to the creation of proteins the building blocks of cells which in turn are the foundation of all organisms. Analysis these sequences by micro-array data lead to a much fuller understanding of many biological processes. Students will be introduced to the basic concepts behind Bioinformatics and Computational Biology tools.

Module aim:

The aim is to introduce computer science students a background and better understanding of the process of how genes through the process of regulation result after a number of intermediate stages in the formation of an amino acid sequence which has a direct relationship to the underlying gene base pair sequence. To show how this relationship leads to the inheritance of characteristics and the evolutionary process in general. Emphasise the significance amino acid sequences in the formation of proteins and specifically the regions involved in protein functionality. To develop the theory of how these basic building blocks lead to cell functionality and different biological processes e.g. organism development. To familiarise the student with how data can be generated and obtained, which can then used to develop a further understanding of the aforementioned biological processes.

Learning Outcomes:

On completion of this module, the student will be able to:

- Discuss the process of genes, hereditary, gene regulation and evolution of the genome.
- Discuss the importance of gene sequences in the genomic process and how it relates to the formation of amino acid sequences.
- Analyse an amino acid sequence or gene sequence and discuss how it can result in the formation of the primary, secondary and tertiary structures of proteins the basic building blocks of all organisms.
- Discuss the fundamental principles of cell biology
- Describe and analyse the principles of micro-arrays and their importance to the field of bioinformatics.
- Discuss how to obtain and use the freely available micro-array data
- Develop an understanding of the underlying computational techniques used in bioinformatics.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4012	Bioinformatics

Learning and Teaching Methods:

This module will employ teaching methods and learning situations such as lectures, seminars and tutorials, as well as case studies

Module content:

- What is meant by the genome, what are genes and how is their expression regulated, what are the main types of genomes and their properties and how the genes establish different physical traits and how mutations result in the evolution of organisms.
- How gene sequence produce basic amino acid sequences : the transcription of DNA and the translation of RNA.
- How are the basic amino acid sequences converted into function proteins ..
- What are cells and their different components. What are the functions of the different components of a cell. How to these components interact to produce cell functionality.
- Evaluate the principles of micro-array analysis and discuss how such techniques allow for the observation of the expression of thousands at the same time.
- What techniques can be applied to micro-array data to remove background noise and thus ensure the micro-array data is more accurate for analysis.
- Discuss probability theory and its role in Statistical Inference.

Module Assessment:

Continuous assessment (30%); A report discussing how biological data can be utilised to evaluate biological systems.

Written examination (70%); A two hour end of semester written exam.

Essential Reading:

- Alberts Bruce and Martin Raff, 2003 Essential Cell Biology: An introduction to the molecular biology of the cell. Garland Science Textbooks.
- Clark, David P. 2005. Molecular Biology, Elsevier Academic Press
- Microarray Gene Expression Data Analysis: A Beginner's Guide, Helen C. Causton, John Quackenbush, Alvis Brazma, 2003, Blackwell Publishers
- **Markley Scott and Leon Darryll Sequence Analysis in a Nutshell: A Guide to Tools** A Guide to Common Tools and Databases **2003**
- Bower, James M and Bolouri Hamid, 2001, *Computational modelling of genetic and biochemical networks* Cambridge, Massachusetts; London, England: The MIT Press.

Supplemental Reading:

To be supplied before commencement of the module

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4012	Bioinformatics

Web references, journals and other:

- Genetics Virtual Library
- Microbiology Resources
- Biology related search engine
- Gene Expression and Micro-array analysis resources

Further Details

Further Details: 2-hour lecture, and 1-hour tutorial. To be delivered in one semester.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4020	Digital Audio

8.9.6. Digital Audio

Module author: Bryan Duggan & Dan Barry

Module Description:

Computers have been used for decades to generate process and analyse musical sounds. Their ability to perform these tasks has grown so great that a single computer with appropriate hardware and software can provide the functionality of a traditional recording studio. Digital audio and computer music are now ubiquitous, whether part of a multi-media application, a computer game, the latest singles in the charts or streaming audio over the Internet. The aim of this module is to introduce students to digital audio techniques. This course has a strong practical and technical focus.

Module aim:

The aim of this module is for students to learn how audio is digitised and processed.

Learning Outcomes:

On completion of this module, the student will be able to:

- Demonstrate an understanding of trends and advances in digital audio and sampling technology.
- Demonstrate an understanding of how issues such as sample rate, resolution, the Nyquist frequency and so on affect the quality of sampled audio.
- Be able to program effects such as vibrato, tremolo and ADSR envelopes using additive synthesis.
- Be able to analyse sampled audio using a range of tools such as the Fourier Transform.
- Be able to use a patch diagram to program audio.
- Demonstrate a basic understanding of music theory.
- Be able to program simulations of sounds and instruments.

Learning and Teaching Methods:

Class time is split into a series of interactive "studio classroom" based lectures and practical problem solving in labs. In lectures, students have access to a PC, with appropriate software and development kits, so that material and examples can be examined in a live environment. In addition, students will be expected to proactively and independently seek out resources on the internet and from the library to supplement their own learning.

A Virtual Learning Environment (VLE) will be employed to distribute all teaching materials and to support student interaction with both other students and academic staff. Where new material is not presented in lectures, such material will be made available through the VLE and students are expected to proactively use this resource.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4020	Digital Audio

Module content:

- **Digital Audio** Concepts :Phasers, basics of sound signals, sampling, resolution, frequency, the Nyquist frequency, compression artefacts.
- **Introduction to Psychoacoustics in Computer_Music_**Perception of intensity, the human ear, perception of temporal features, perception of frequency, perception of noise, fusion and the perception of timbre.
- **Sound synthesis:** Soundcards and sound production, digital oscillators, FM & AM Synthesis, wavetables, soundcard architectures.
- **Programming effects:** Tremolo, vibrato, ADSR envelopes, additive synthesis, understanding patch diagrams, analysing signals, Fourier analysis.
- **Music Theory:** The well tempered scale, major and minor keys, sharps, flats and naturals, the relationship between notes and frequency, time signatures.
- **Digital audio formats:** Compression Principles, Sub-band Coding, Transform Coding, Compression Formats: MPEG Layer I/II/III/2 ACC. MP3, PCM, AU, WAVE, AIFF. Streaming Media.

Module Assessment:

This module has a 70% weighting for the examination and a 30% weighting for the continuous assessment. Continuous assessment will consist of assignments and in class examinations. While it is important that students can demonstrate their technical ability through continuous assessment, it is equally important that they demonstrate an understanding of the concepts. This will be assessed in the examination.

The suggested method of assessing the individual learning outcomes for the module is as follows:

Essential Reading: (author, date, title, publisher)

Curtis Roads, (2000), The Computer Music Tutorial, MIT Press,

Supplemental Reading: (author, date, title, publisher)

Eduardo Reck Miranda, (2001), Composing Music with Computers, Focal Press.

Richard Boulanger (2000) The Csound Book: Perspectives in Software Synthesis, Sound Design, Signal Processing,and Programming (Paperback)

Web references, journals and other:

http://webcourses.dit.ie

http://csounds.com

Journal of Computer Music

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS credits	Module Code	**Module Title**
		5	CMPU4020	Digital Audio

Further Details

The module will run over one semester, with four contact hours per week.

Date of Academic Council approval ……………………….

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU4042	Music Technology

8.9.7. Music Technology

Module author: Bryan Duggan & Dan Barry

Module Description:

Computers have been used for decades to generate process and analyse musical sounds. Their ability to perform these tasks has grown so great that a single computer with appropriate hardware and software can provide the functionality of a traditional recording studio. Digital audio and computer music are now ubiquitous, whether part of a multi-media application, a computer game, the latest singles in the charts or streaming audio over the Internet. The aim of this module is to introduce students to techniques and applications of digital audio in music technology. The course explores applications such as sequencing and spatial audio using popular toolkits, hardware and API's. This course has a strong practical and technical focus.

Module aim:

The aim of this module is to introduce students to applications and techniques of digital audio in music technology.

Learning Outcomes:

On completion of this module, the student will be able to:

- Demonstrate an understanding of trends and advances in music technology.
- Produce multi-track music using popular music and digital audio software packages.
- Demonstrate an understanding of the underlying principles and techniques used by these packages.
- Apply effects such as panning and reverberation and demonstrate an understanding of how these effects are produced.
- Develop music and sound applications included MIDI based applications.
- Demonstrate an understating of the techniques and algorithms employed by 3D positional audio toolkits.
- Program 3D positional audio using standard API's.
- Demonstrate a basic understanding of music theory.

Learning and Teaching Methods:

Class time is split into a series of interactive "studio classroom" based lectures and practical problem solving in labs. In lectures, students have access to a PC, with appropriate software and development kits, so that material and examples can be examined in a live environment. In addition, students will be expected to proactively and independently seek out resources on the internet and from the library to supplement their own learning. Students will have access to cutting edge audio hardware and will gain hands on experience using the hardware in practical class sessions.

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU4042	Music Technology

A Virtual Learning Environment (VLE) will be employed to distribute all teaching materials and to support student interaction with both other students and academic staff. Where new material is not presented in lectures, such material will be made available through the VLE and students are expected to proactively use this resource.

Module content:

Sequencers Sampled audio, MIDI, quantisation, mixing, plug-in architecture, latency.

Programming Sound Applications and MIDI Programming music, representing rhythm and pitch, wavetables vs FM synthesis. Introduction to MIDI, MIDI Format: tracks, events, timing, channels, instruments, wire protocol, Programming MIDI, MIDI standards.

Audio Effects Theory and Design Understanding Equalisation, Reverberation, Dynamic Range Processors and various other audio FX. The DSP required to implement such effects.

3D Positional Audio Speaker configurations, vectors, virtual 3D space, the listener, the Doppler effect, sound cones, DirectXSound, OpenAL, hardware acceleration, EAX.

Digital Audio and Music Technology Application Areas Impact of technology on music production. Digital music in multi-media, the Internet and streaming audio, computer games.

Module Assessment:

This module has a 70% weighting for the examination and a 30% weighting for the continuous assessment. Continuous assessment will consist of assignments and in class examinations. While it is important that students can demonstrate their technical ability through continuous assessment, it is equally important that they demonstrate an understanding of the concepts. This will be assessed in the examination.

The suggested method of assessing the individual learning outcomes for the module is as follows:

Essential Reading: (author, date, title, publisher)

Curtis Roads, (2000), The Computer Music Tutorial, MIT Press,

Supplemental Reading: (author, date, title, publisher)

Eduardo Reck Miranda, (2001), Composing Music with Computers, Focal Press.

Richard Boulanger (2000) The Csound Book: Perspectives in Software Synthesis, Sound Design, Signal Processing,and Programming (Paperback)

Web references, journals and other:

http://webcourses.dit.ie

http://csounds.com

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
		5	CMPU4042	Music Technology

Journal of Computer Music

Further Details

The module will run over one semester, with four contact hours per week.

Date of Academic Council approval …………………………

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4032	Geographical Information Systems

8.9.8. Geographical Information Systems

Module Author(s): Mark Foley

Module Description

A Geographical Information System (GIS) is a computer system designed to facilitate the collection, management, and analysis of large volumes of geographical knowledge.

This course will focus on the fundamental principles of GIS, the practical techniques of implementing a GIS and the creation and use of a GIS for spatial analysis.

Module Aim

The aim of this module is to take students from no prior knowledge of GIS to a position where they (i) understand the role and current state of the art in Geographical Information Systems, (ii) can analyse a problem in GIS and build an appropriate solution and (iii) are familiar with the basic techniques of spatial analysis and modelling.

Learning Outcomes

On successful completion of the module the students will be able to

- Understand the role of GIS and its application in solving practical problems.
- Understand how geographic data is represented in a computer.
- Understand the unique character of geographic data and how this is mapped to the real world.
- Be able to confidently use the market-leading commercial GIS package.
- Understand the rudiments of spatial databases and when and where to use database technology.
- Understand how GI data is created and acquired and be aware of possible sources of data.
- Understand the art and science of cartography and map design.
- Solve problems in spatial analysis especially in the areas of visualization, query/measurement and design/modelling.

Learning and Teaching Methods

Lectures, self-study, labs, tutorials, and any combination of discussion, case study, problem-solving exercises, readings, seminars, and computer-based learning.

Module Content

The module content will include the following topics. Material may be added to or deleted from this list over the lifetime of the module to reflect the changing nature of the relevant technologies.

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4032	Geographical Information Systems

Principles

- Introduction to GIS and its applications.
- Representing geography and the nature of geographic data.
- Georeferencing.
- The notion of uncertainty in GI data.

Techniques

- GIS software.
- GIS data collection.
- Creating and managing GI data.
- Introduction to distributed and web GIS.

Analysis

- Cartography and map production.
- Visualization of GI data.
- Query and measurement
- Data summary and inference
- Introduction to spatial modelling

There will be also be hands-on work with the leading GIS software packages and students will apply the course material to practical problems and case studies.

Module Assessment

The methods of assessment to be used to measure the learning objectives stated above are written examination and continuous assessment including one or more of assignment, essay, problem-solving exercise, oral presentation, and class or lab tests.

Continuous Assessment 50%

Examination 50%

Essential Reading List

The material listed here represents a sample of possible reading material for the module. The actual reading material will be determined by the lecturer, taking account of availability of new material and changing technologies which could, in turn, determine module content.

Geographic Information Systems and Science, 3rd ed.; Paul A. Longley, Michael F. Goodchild, David J. Maguire, David W. Rhind; Wiley, 2010; ISBN: 978-0-470-72144-5.

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4032	Geographical Information Systems

Supplementary Reading list

Introduction to Geographic Information Systems, 5th ed.; Kang-tsung Chang; McGraw Hill, 2009; ISBN: 978-007-126758-8.

Geospatial Analysis - a comprehensive guide, 2nd ed.; Michael de Smith, Michael F. Goodchild and Paul A. Longley; Winchelsea, 2007; ISBN: 978-1906221-980.

Software

Practical hands-on experience using GIS will be provided through the use of a current market-leading GIS software package.

Further Details

Contact Hours

Three hours per week; 1 hours lecture / tutorial, 1 hour lab.

Delivery

One semester

Date of Academic Council approval ....

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4046	Spatial Databases

8.9.9. Spatial Databases

Module author: Patrick Browne

Module Description:

This module focuses on the use of database management systems (DBMS) to manage spatial information. A Spatial Database is a specialised DBMS that is designed to facilitate the collection, querying, management, and analysis of large volumes of geographical knowledge. A spatially enabled DBMS is a central component of a Geographical Information System (GIS). Spatial databases have a major role to play in managing the national physical and informational infrastructure. The understanding of advanced spatially enabled DBMS is vital in implementing any information system where geographic data is required. This module focuses on the role of the DBMS in geographical applications.

Module aim

The aim of this module is to familiarise students with the features of a spatially enabled database systems and to develop applications that use the spatial database extensions.

Learning Outcomes:

On completion of this module, the student will be able to:

- use a database to store and query spatial data
- develop applications that use a spatially enabled DBMS, e.g. spatial data mining, navigation.
- distinguish and use appropriate database models
- apply the various query languages appropriate to spatial querying to given situations
- understand the extensions required by application programs to handle spatial data

Learning and Teaching Methods:

Lectures, self-study, labs, tutorials, and any combination of discussion, case study, problem-solving exercises, readings, seminars, and computer-based learning.

Module content:

Foundations Fundamental geographic concepts for GIS The world in spatial terms, how natural and man made features can be stored in a DBMS. Qualitative and quantitative location e.g. coordinate systems and geo-referencing. Maps as representation of the world and of information. Geometric and thematic information.

Algorithms and Data Structures for GIS: Abstract Data Type (ADT), spatial ADTs and their operations e.g. intersection, overlay, network traversal, auto-correlation, statistical operations, searching.

Pre-requisite Module codes	Co-Requisite Module code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4046	Spatial Databases

Spatial representations: ADTs for representation, raster, vector, TIN, quadtrees, R-trees, scan orders, polygon coverage, discrete objects, networks, time, connections and topology, networks, distance and direction, flow and diffusion, spatial hierarchies, boundaries, spatial patterns, attributes, and representations of relationships.

Applications of spatial databases: Spatial data mining, spatial data on the World Wide Web, transportation networks, natural resources, soil data, oceanography, land cover, geology, climate, terrain, land records, administrative boundary data, demographic studies, and health data.

Spatial databases Spatial data: definitions, formats, models, queries the relational model, advanced SQL, data modelling techniques, implementing a simple database, post relational database models, object-relational and object-oriented models, spatial data structures, spatial indexing e.g. R-Tree, networking, database issues in GIS.

The course will involve practical work on a range of appropriate software e.g. spatially extended DBMS ,map servers, statisital packages, markup languages, client and server side programming..

Module Assessment

The methods of assessment to be used to measure the learning objectives stated above are written examination and continuous assessment including one or more of assignment, essay, problem-solving exercise, oral presentation, and class or lab tests.

Continuous Assessment 50%

Examination 50%

Essential Reading:

Shekhar and Chawla; Spatial databases – A tour, 2003, Prentice Hall, 2003

Supplemental Reading:

(1) Obe, Hsu; PostGIS in Action, 2010 Manning.

(2) Rigaux, Scholl, Voisard; Spatial databases, 2002, Morgan Kaufmann

Web references:

The Open Geospatial Consortium, Inc. (OGC) - http://www.opengeospatial.org/

Further Details:

This module will be delivered in one semester. 2 hours lecture / tutorial and 1 hour lab per week.

Date of Academic Council approval ....

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4017	Computer Graphics

8.9.10. Computer Graphics

Module author: Paul Kelly Brian Mac Namee

Module Description:

This is an introductory module in computer graphics. This module introduces the learner to the branch of computer science which deals with the theory and technology for computerised image synthesis. The module examines the mathematical and algorithmic basis of graphics concepts and their implementation in a representative interactive graphics system. A major emphasis is placed on the structure and use of a device-independent application programmer's interface (API) to create graphical applications.

Module aim

The aim of this module is to teach the fundamentals of computer graphics. Its theoretical spectrum is to cover the complete graphics pipeline and include major aspects of computer graphics such as geometrical transformations, projections, surface modelling, illumination and shading, hardware considerations and issues in visual realism. Students should also be made aware of the diverse application areas for graphics technologies. The theoretical content of the course is to be supported with a strong practical element to include both programming and the use of suitable supplementary packages.

Learning Outcomes:

On successful completion of this module, the learner will

- Know the fundamental techniques and algorithms used in computer graphics
- Apply the mathematics and algorithms behind computer graphics
- Write software that uses a graphics API
- Implement some of the techniques taught in the course using appropriate packages
- Critically analyse the impact of technological developments on the area of visualisation
- Display an awareness the application areas of computer graphics technologies
- Select appropriate graphics techniques to meet application requirements (e.g. performance)
-

Learning and Teaching Methods:

Lectures, tutorials, lab demonstrations and discussions. Students will be expected to put into practice some of the module concepts by developing small sample applications. A Virtual Learning Environment is used extensively in this module.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4017	Computer Graphics

Module content:

The module content will include, but not be limited to (image processing is a fast moving discipline and so new topics will be introduced to the module over time), the following:

- Graphics I/O and Storage Devices. Scan Conversion. Plotting points, lines, shapes. Filling. Boundary block transfer.
- Mathematics of Transformations, Geometric, Coordination, Composite, Instance Transformations, Windows and viewports. Viewing transformations, clipping and shielding. The synthetic camera. Coordinate systems. Hierarchy models. Fractals.
- Perspectives and Parallel Projections. 2D and 3D viewing pipelines. Hidden Surfaces. Depth comparisons. Depth buffer algorithm. Scan line algorithms. Painters algorithm. Subdivision algorithms. Hidden line elimination.
- Animation. 3-D dynamics, occlusion, solid objects and interference, scene generation. Surface Treatment. Ray tracing, radiosity, rendering. Colour theory.
- Visual realism: Illumination. Textures. Shadows. Fog. Reflections. Gourad and Phong shading models. Illumination and reflectance models.
- Image File Formats.
- Splines, B-splines, Cubic splines.
- New and current topics in computer graphics.

Module Assessment

Written examination - 70%

Continuous assessment - 30%

Essential Reading:

Francis S. Hill and Stephen M. Kelley,2007, Computer Graphics Using openGL (3rd edition), Prentice-Hall, 2006.

or

Donald Hearn, M. Pauline Baker and Warren Carithers, 2010, Computer Graphics with Open GL 4/e, Prentice Hall.

Supplementary Reading:

Mason Woo, Jackie Neider, Tom Davis, OpenGL Architecture Review Board , 2008,

OpenGL Programming Guide: The Official Guide To Learning Opengl, Addison Wesley

James Foley, 2005, Computer Graphics: Principles and Practice, Addison Wesley

Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
	None	5	CMPU4017	Computer Graphics

I. Angel, 2005, Interactive Computer Graphics: A Top-Down Approach Using OpenGL (5th Edition)

Web references, journals and other:

OpenGL Home: http://www.opengl.org

OpenGL Programming Guide (The Red Book) :http://www.glprogramming.com/red/

OpenGL Reference :http://www.talisman.org/opengl-1.1/Reference.html

GLUT Reference:http://krow.net/dict/spec3/spec3.html

Nate Robbins' OpenGL tutorials:http://www.xmission.com/%7Enate/tutors.html

NEHE Tutorials: http://nehe.gamedev.net

Further Details:

Duration: 1 semester

Contact Hours: 2 lecture hours and 2 lab hours per week

Date of Academic Council approval ………………………….

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
	None	**5**	CMPU4034	**Image Processing**

8.9.11. Image Processing

Module author: Brian Mac Namee & Paul Kelly.

Module Description:

This is a first course in Image Processing. The course begins with low level image processing and progresses its way up to the beginnings of image interpretation. The learner will be required to apply their understanding of the concepts involved through building applications that manipulate images through the use of suitable packages (e.g. Matlab, Scilab or OpenCV).

Module aim

The aim of this module is to teach the fundamentals of image processing. This will include image representations, fundamental image processing operations, segmentation and the beginnings of image analysis. Its theoretical content is to be supported with a strong practical element that encompasses both programming and the use of suitable packages.

Learning Outcomes:

On completion of this module, the learner will be able to:

Demonstrate an understanding of the fundamentals of image processing

- Implement image processing techniques to solve image processing problems
- Compare different approaches to solving the same image processing tasks
- Select appropriate image processing techniques to solve image processing problems.
- Critically analyse how image processing feeds into the processes of image analysis and understanding
- Describe the state of the art of image processing in particular application areas

Learning and Teaching Methods:

Lectures. tutorials, lab demonstrations and discussions. Students will be expected to put into practice some of the module concepts by developing sample applications. A Virtual Learning Environment is extensively used in this module.

Module content:

The module content will include, but not be limited to (image processing is a fast moving discipline and so new topics will be introduced to the module over time, the following:

- Introduction to image processing
- Image processing hardware considerations
- Image acquisition and representation

Dublin Institute of Technology

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	**Module Title**
	None	**5**	CMPU4034	**Image Processing**

- Fundamentals of image processing
- Point operations
- Neighbourhood operations
- Geometric operations
- Mathematical morphology
- Frequency domain processing
- Segmentation
- Thresholding
- Edge detection
- Boundary detection
- Image analysis and understanding
- Template matching
- Hough transform
- Shape descriptions

Module Assessment

Written examination - 70%

Continuous assessment - 30%

Essential Reading:

Rafael C. Gonzalez & Richard E. Woods, 2002, Digital Image Processing 2/e, Prentice Hall

Supplementary Reading:

David Vernon, 1991, Machine Vision: Automated Visual Inspection and Robot Vision, Prentice Hall. (Available online)

Web references, journals and other:

OpenCV wiki pages: http://opencv.willowgarage.com/wiki/

Further Details:

Duration: 1 semester

Contact Hours: 2 lecture hours and 2 lab hours per week

Date of Academic Council approval ....

8.10. Stage 4 Syllabi (Internship)

Internship students take the Internship module as well as the core modules in Stage 4.

Pre-Requisite Modules code(s)	Co-Requisite Modules code(s)	ECTS Credits	Module Code	Module Title
CMPU3022	**None**	**5**	CMPU4039	**Internship (Fourth year)**

8.10.1. Stage 4 Internship

Module author: **Paul Doyle**

Module Description:

This module is linked with SPEC3102 and is the second half of a 2 year industrial placement which provides students with the opportunity to obtain professional industry experience for 2 days per week.

Module aim

The goal is to give students the opportunity to gain a structured learning experience within a professional / industrial environment so that they may consolidate and broaden the knowledge that they have gained in their academic studies to date. Students should gain a deep understanding of the business within which they are placed.

Learning Outcomes

On completion of this module, the learner will be able to

- Work in a professional environment
- Demonstrate extensive understanding of a particular project or area of work that they have been involved in
- Reflect on the learning experience and outcomes of their Internship
- Produce a professional report describing the details and experiences of their placement
- Evaluate their contribution to the company
- Critically assess the relationship between their academic knowledge and industrial experience
- Demonstrate an understanding of the company's business model

Learning and Teaching Methods:

During the time in industry students may be working as part of a team or individually. Students are assigned a DIT Internship Monitor for the duration of their internship and deliver reflective documents demonstrating their learning as part of continuous assessment.

Module content:

- Professional Development
- Structured reporting on work performed
- Teamwork within a professional organisation
- Reflective writing and reporting
- Presentation of work

Pre-Requisite Modules code(s)	**Co-Requisite Modules code(s)**	**ECTS Credits**	**Module Code**	**Module Title**
CMPU3022	**None**	**5**	CMPU4039	**Internship (Fourth year)**

- IT skills specific to the Internship

Module Assessment

Assessment will be based on 100% continuous assessment.

The continuous assessment element of this course is comprised of:

- regularly submitted reports by students on internship,
- an end of year written report and
- an end of year presentation.

The end of year presentation and report is jointly assessed by the Industrial placement monitor and the industrial placement coordinator.

Essential Reading:

Jennifer A. Moon, 2004 A Handbook of Reflective and Experiential Learning: Theory and Practice ,

Lydia E. Anderson, 2007, Professionalism: Real Skills for Workplace Success, Prentice Hall

Paul Doyle, DIT School of Computing Industrial Placement Handbook

Supplemental Reading:

David Boud, Reflection : 1985, Turning Experience into Learning, Routledge

Dannelle D. Stevens , Joanne E. Cooper, 2009, Journal Keeping: How to Use Reflective Writing for Learning, Teaching, Professional Insight and Positive Change , Stylus Publishing

Web references, journals and other:

http://www.comp.dit.ie/industrialplacement

http://www.dit.ie/careers/

Further Details:

Duration:	Placement covers year 3 and year 4 and is linked to **SPEC3102**
Contact Hours:	Students will work in industry for 2 days per week

Date of Academic Council approval …………………………