De Montfort University

Course Template

1. Basic information

- Course Name: Computer Games Programming
- Course Code: CC314A
- Level (UG, PG): Undergraduate
- Academic Period: 2014
- Faculty: Faculty of Technology
- Department: Games Programming & Intelligent Systems
- PMB COMP
- Offered at: DM - DMU Leicester
- Type (single, joint.): SI
- Highest Award : Bachelor of Science (Honours)
- All possible exit awards : Bachelor of Science; Certificate of Higher Education; Diploma of Higher Education; Institutional Undergraduate Credit
- Award notes :

Professional Body Recognition

- Accreditation by Professional/Statutory body:
  Yes
- Exemption by Professional/Statutory body:
  No
- Details
  BCS

- Modes of attendance: Main MOA: Full-Time
  Other MOA: Part-Time; Year Out/On Placement
- Mode Notes:

- Course leader: Ian Kenny

2. Entry Requirements and Profile

Normally 260 UCAS Points from at least two A Levels or equivalent, plus five GCSEs at grade C or above, including Maths and English

Typical A Level offers: At least one grade B and one grade C results with additional qualifications contributing towards the points score such as a third A Level or AS Levels

International Baccalaureate: 28+ points.

We welcome applications from mature students with non-standard qualifications and recognise all other equivalent and international qualifications.

If English is not your first language, an IELTS score of 6.0 or equivalent when you start the course is essential. English language tuition is available at DMU both before and during the course if required.

3. Course Description

Characteristics and Aims

Market research shows that specialist skills in games programming are in high demand. This course has been designed to deliver these skills. It does not produce graduates who have an all-round, but shallow, understanding of games development but produces skilled specialist graduates. These graduates are aware of the context of games development, but the emphasis is on the programming role, rooted in an understanding of computer science.

The course adopted guidelines which form the Games Programming category of the

International Game Developers Curriculum Framework. The course is informed by research and recommendations provided by Skillset which aims to build a bridge between the games industry and universities.

Games programming requires core skills across a range of areas, such as coding, programme design, artificial intelligence, handling multimedia, physics all set against a knowledge of game architecture and design. These core technical skills need to be supplemented by skills in team-working and team building, so the course develops these skills to enable graduates to be useful immediately.

The themes are

Programming/Games Programming:
Programming in C++ (compulsory)
Javascript and HTML (compulsory)
C++ for Games Programmers (compulsory)
Advanced Games Programming (compulsory)

Artificial Intelligence techniques:
AI & Modelling for Games (compulsory)
Fuzzy Logic and Knowledge-Based Systems
Mobile Robotics

Multimedia/Graphics/Sound:
Computer Graphics and Interactive 3D Modelling (compulsory)
Audio Post-production
Advanced Graphics

Game Design and Development:
Game Architecture, Design and Development (compulsory)
Systems Building: Methods and Management

On-Line and Mobile Games:
Mobile Games
Secure Web Application Development

Computer Science background:
Elements of Computing (compulsory)
Mathematics for Scientific Computing (compulsory)
Database design and implementation (compulsory)
Multi Service Networks

Teaching, Learning and Assessment Strategies

The modules offered on the course have a mixture of teaching and learning strategies to reflect the mix of theory and practice in the modules. Most modules will have formal contact in the form of:

§ Lectures. These are appropriate for presenting basic subject material, working through guided examples and demonstrating technical issues in software operation.

§ Tutorials. These provide students with the opportunity to practise and develop skills and techniques under guidance, and obtain feedback.

§ Laboratory classes. Computer Games programming has a strong practical element and students will only develop practical skills by extensive practise in the laboratory.

Student managed learning
All students are required to manage their own learning in the following ways:

§ Supported self-study. Students are issued with a module plan for each module. This gives a week by week plan of the module along with appropriate activities and reading.

§ Assessments. These serve to consolidate and develop a student's knowledge and

skills. Assessments can be formative as well as summative and could include piratical software development exercises, reports, essays, structured exercises, group and individual projects, tests and examinations. Projects offer the opportunity for students to develop their ability to conduct a sustained individual development exercise or investigation. Group projects also allow students to develop their ability to work in teams, to appraise group performance and deal with group problems.

Key skills including team-working, giving presentations, report writing, communication and undertaking research will be developed through the programme. The games development industry is particularly keen on students being well practised in group work since this is the usual mode of working when developing games in multi-disciplinary teams.

4. Outcomes

Generic outcome headings	**What a student should know and be able to do upon completion of the course**
• **Knowledge & understanding**	Students will be able to: Illustrate programming principles and best practice, using object-oriented and procedural programming environments, and use software libraries. Describe and illustrate the architecture of a computer system, how data is stored, manipulated and exchanged between components. Describe, evaluate and use a range of artificial intelligence techniques when designing a solution for a software system; Describe how multimedia objects are stored, processed and integrated into software system applications; Illustrate the basic principles behind systems software as used in operating systems and computer networks.
• **Cognitive skills**	Students will have developed and be able to demonstrate the following cognitive skills: Solving problems; Critical reasoning; Analysing/evaluating; Modelling objects/environments; Awareness of professional/ethical considerations
• **Subject specific skills**	Students will be able to: Integrate a range of games design features into an overall design for a games system; Apply software design techniques in software development. Construct usable client web pages for on-line systems using suitable tools; Adapt software development techniques to

	suit the particular requirements of games programming; Use games development systems, games engines and other relevant production tools in constructing usable computer games or computer game components.
• **Key Skills**	Students will have developed, and be able to demonstrate, the following key skills: Communication - group discussion, handling information from various complex documents. Application of number - handling the dynamics in games systems. Information technology - across a range of activities. Improving own learning and performance - all modules require self-study and preparation for module delivery. Problem solving - identifying programming solutions, modelling issues. Working with others - learning to work effectively in groups and improve group performance over time. Applying methodologies/concepts. Using methods/tools in specification and design.

5. Structure and Regulations

Relationship Details

Module	Credits	Level	Take/Pass		Semester	Locations
CTEC1801	**30.00**	**1**	**Must Take**		**Y**	**DM**
IMAT1205	**15.00**	**1**	**Must Take**		**Y**	**DM**
IMAT1212	**30.00**	**1**	**Both**	**Y**	**DM**	
IMAT1213	**15.00**	**1**	**Must Take**		**Y**	**DM**
IMAT1606	**30.00**	**1**	**Must Take**		**Y**	**DM**
IMAT2427	**30.00**	**2**	**Must Take**		**Y**	**DM**
IMAT2605	**30.00**	**2**	**Must Take**		**Y**	**DM**
IMAT2800	**30.00**	**2**	**Must Take**		**Y**	**DM**
IMAT2902	**30.00**	**2**	**Must Take**		**Y**	**DM**
SAND2802	**0.00**	**2**	**Neither**		**1, 2, X, Y**	**DM**
CTEC3110	**15.00**	**3**	**Neither**		**Y**	**DM**
CTEC3604	**30.00**	**3**	**Neither**		**Y**	**DM**
IMAT3111	**30.00**	**3**	**Neither**		**Y**	**DM**
IMAT3404	**15.00**	**3**	**Neither**		**Y**	**DM**
IMAT3406	**15.00**	**3**	**Neither**		**Y**	**DM**
IMAT3425	**30.00**	**3**	**Neither**		**Y**	**DM**
IMAT3451	**30.00**	**3**	**Must Take**		**Y**	**DM**
IMAT3606	**30.00**	**3**	**Must Take**		**Y**	**DM**

IMAT3608	**30.00**	**3**	**Neither**	**Y**	**DM**
TECH3024	**30.00**	**3**	**Neither**	**Y**	**DM**

Structure

Structure notes

1 The course includes an optional work placement year.

Students successfully completing the 360 credit taught programme and the work placement year will be awarded a sandwich degree.

The requirements to progress into the sandwich are determined by Faculty Policy which requires that normally a student must have passed a minimum of 60 credits at level 5.

Students successfully completing the 360 credit taught programme but opting not to take the work placement year, will be awarded a full-time degree.

Course Specific Differences or Regulations

1 The requirements to progress into the sandwich are determined by Faculty Policy which requires that normally student must have passed a minimum of 60 credits at level 2.

Numbers at sites, including partner institutions

1

Relevant QAA Subject Benchmarking statement(s)

1 Computing

6. Quality Assurance Information

QA of Workbased Learning

Liaison with Collaborative Partners

Procedures for Maintaining Standards

The Programme is managed by a programme leader together with a programme team. They are guided by the prevailing academic regulations and modular scheme handbooks produced by Registry.

An external examiner is attached to the programme who acts as a critical friend. He/She attends the assessment board and scrutinises student work and marking to ensure that standards have been maintained at an apposite level.

Each year the programme leader completes a Programme Enhancement Plan which is approved by the Programme Board/Subject Authority Board and Faculty Academic Committee.

The student voice is heard via student representatives on the Programme Board and the Staff Student Consultative Committee. Feedback from students is gathered by end of module questionnaires and programme questionnaires.

The programme is subject to a periodic review in line with University requirements.

Course Handbook Descriptor