

KEY PROGRAMME INFORMATION

Originating institution(s) Bournemouth University	Faculty responsible for the programme Faculty of Science and Technology
Final award(s), title(s) and credits BSc (Hons) Games Software Engineering – 120 (60 ECTS) Level 4 / 120 (60 ECTS) Level 5 / 120 (60 ECTS) Level 6 credits	
Intermediate award(s), title(s) and credits Dip HE Games Software Engineering – 120 (60 ECTS) Level 4 / 120 (60 ECTS) Level 5 credits Cert HE Games Software Engineering – 120 (60 ECTS) Level 4 credits	
UCAS Programme Code(s) (where applicable and if known) I610	HECoS (Higher Education Classification of Subjects) Code and balanced or major/minor load. 101267 Games (20%) 101020 Computer Games Programming (70%) 101019 Computer Games Graphics (10%)
External reference points <ul style="list-style-type: none"> • UK Quality Code for Higher Education; • Part A: Setting and maintaining academic standards; • Chapter A1: UK and European reference points for academic standards (October 2013) - incorporates the Frameworks for Higher Education Qualifications of UK Degree-Awarding Bodies (Qualification Frameworks); • QAA Honours Degree Subject Benchmark Statement: Computing (February 2016) • TIGA standards and guidelines. 	
Professional, Statutory and Regulatory Body (PSRB) links Awaiting accreditation from TIGA.	
Places of delivery Bournemouth University	
Mode(s) of delivery Full-time, Full-time sandwich	Language of delivery English
Typical duration 3 years full-time / 4 years full-time sandwich. Level 4: 1 year, Level 5: 1 year, Level 6: 1 year.	
Date of first intake September 2017	Expected start dates September
Maximum student numbers Not applicable	Placements Optional 30 week sandwich placement. Optional short placement (4 weeks minimum) with no coursework attached
Partner(s) Not applicable	Partnership model Not applicable
Date of this Programme Specification June 2021	
Version number V1.5-0922	
Approval, review or modification reference numbers E2017010 – approved 21/04/2017 NM161706 – 02/05/2017 P161701 – 06/07/2017 BU 1819 01 FST 1920 15, approved 04/03/2020 – Previously v1.2-0920 FST 2021 12, Approved 07/06/2021 – Previously v1.3-0921 FST2122 22, Approved 23/03/2022 – Previously v1.4-0922	
Author	

Programme Specification – Section 1

C Harvey

Programme Specification – Section 1

PROGRAMME STRUCTURE

Programme Award and Title: BSc (Hons) Games Software Engineering								
Year 1/Level 4								
Students are required to complete all 6 core units.								
Unit Name	Core/ Option	No of credits	Assessment Element Weightings			Expected contact hours per unit	Unit version no.	HECoS Subject Code
			Exam 1	Cwk 1	Cwk 2			
Games Design Principles	Core	20		30%	70%	40	v3.1	101268
Introduction to Programming	Core	20		30%	70%	40	v1.1	101020
Digital Technologies	Core	20		30%	70%	40	v2.2	101267
Mathematics for Computer Graphics	Core	20		30%	70%	40	v1.1	101019
Object Oriented Game Programming	Core	20		30%	70%	40	v1.1	101020
Game Development Pipeline	Core	20		30%	70%	40	v1.1	101267
Progression requirements: Requires 120 credits at Level 4.								
Exit qualification: Cert HE Games Software Engineering (requires 120 credits at Level 4)								

Year 2/Level 5								
Students are required to complete all 6 core units.								
Unit Name	Core/ Option	No of credits	Assessment Element Weightings			Expected contact hours per unit	Unit version no.	HECoS Subject Code
			Exam 1	Cwk 1	Cwk 2			
3D Graphics Programming	Core	20		30%	70%	40	v1.1	101019
Physics for Games	Core	20		30%	70%	40	v2.1	101020
Programming for Interaction	Core	20		100%		40	v1.1	101020
Game Studio Project	Core	20		100%		40	v1.1	101267
AI Game Programming	Core	20		30%	70%	40	v2.1	101020
Commercialisation and Business Environment	Core	20		60%	40%	40	v3.1	101221
Progression requirements: Requires 120 credits at Level 5								
Exit qualification: Dip HE Games Software Engineering (requires 120 credits at Level 4 and 120 credits at Level 5)								
Year 3/Level P - Optional placement year in industry/business								
Optional sandwich placement is taken between levels 5 and 6.								
Progression requirements: Satisfactory completion of a minimum 30-week placement in industry. Students who do not choose to undertake the optional sandwich placement may progress directly from Level 5 to Level 6.								

Programme Specification – Section 1

Year 3/4/Level 6								
Students are required to complete all 4 core units.								
Unit Name	Core/ Option	No of credits	Assessment Element Weightings			Expected contact hours per unit	Unit version no.	HECoS Subject Code
			Exam 1	Cwk 1	Cwk 2			
Graphics and Computational Programming	Core	20		100%		40	v5.1	101019
Innovation, Enterprise and Business Development	Core	20		100%		40	v3.1	101221
Game Engine Programming	Core	20		100%		40	v1.1	101020
Individual Project	Core	60		100%		20	v3.2	101267

Exit qualification: BSc (Hons) Games Software Engineering
Sandwich UG award: Requires 120 credits at Level 4, 120 credits at Level 5, 120 credits at Level 6 and successful completion of a placement year
Full-time UG award: Requires 120 credits at Level 4, 120 credits at Level 5 and 120 credits at Level 6

AIMS OF THE DOCUMENT

The aims of this document are to:

- define the structure of the programme;
- specify the programme award titles;
- identify programme and level learning outcomes;
- articulate the regulations governing the awards defined within the document.

AIMS OF THE PROGRAMME

This programme aims to develop critically informed, agile and resourceful graduates, who:

- can produce computer games in a legally, ethically and critically-informed manner;
- have developed understanding of key issues in games development;
- combine the critical faculties that derive from a traditional academic degree with the professional skills and flexibility needed to get the best jobs in the games development industry;
- can manage their own personal development and lifelong learning;
- are prepared for a variety of output media;
- are equipped to be familiar with a variety of industry standard tools and agile enough to respond to new technologies and consumer trends.

BSc (Hons) Games Software Engineering is a course that focusses on the technical expertise required to develop computer games, including the core skills of software engineering. The core technical areas cover 2D and 3D graphics rendering, physics, artificial intelligence and human-computer interaction required for the development of high-end computer games platforms. The course will also involve working in multi-disciplinary (creative and technical) teams through the development of computer games.

ALIGNMENT WITH THE UNIVERSITY'S STRATEGIC PLAN

The BSc (Hons) Games Software Engineering programme is informed by and aligned with Bournemouth University's 2012-18 strategic plan and the fusion of excellent teaching, world-class research and professional practice that is at the heart of the institution's visions and values. Students are supported by academics with a wealth of industry experience, many of whom are actively engaged in the production of Computer Games Tools and Technologies for a wide range of commercial clients. Academics delivering the programme are actively engaged in cutting edge research, while students are encouraged to participate in a range of co-creation and co-publication projects. The programme's innovative pedagogic approach offers students the opportunity to learn by engaging in a series of practical, industry focused tasks, such as implementing artificial intelligence techniques for a practical range of purposes. These are aimed at equipping students with the full range of skills necessary to succeed in the software development environment, and are informed by the academic team's own industrial experience as well as by a network of industry contacts, who will also contribute directly to the programme by delivering guest lectures.

LEARNING HOURS AND ASSESSMENT

- Semesterised (CAS compliant)
- Expected contact time is 40hrs per 20cr Unit (if an exception needs to be made, we will provide a minimum of 40hrs)

Bournemouth University taught programmes are composed of units of study, which are assigned a credit value indicating the amount of learning undertaken. The minimum credit value of a unit is normally 20 credits, above which credit values normally increase at 20-point intervals. 20 credits is the equivalent of 200 study hours required of the student, including lectures, seminars, assessment and independent study. 20 University credits are equivalent to 10 European Credit Transfer System (ECTS) credits.

The assessment workload for a unit should consider the total time devoted to study, including the assessment workload (i.e. formative and summative assessment) and the taught elements and independent study workload (i.e. lectures, seminars, preparatory work, practical activities, reading, critical reflection).

Assessment per 20 credit unit should normally consist of 3,000 words or equivalent. Dissertations and Level 6 and 7 Final Projects are distinct from other assessment types. The word count for these assignments is 5,000 words per 20 credits, recognising that undertaking an in-depth piece of original research as the capstone to a degree is pedagogically sound.

Students who choose to undertake the sandwich placement after Level 5 will engage in 30 weeks of full-time work-based learning between Levels 5 and 6, students who do not undertake the sandwich placement will be eligible to progress directly on to Level 6.

STAFF DELIVERING THE PROGRAMME

Students will usually be taught by a combination of senior academic staff with others who have relevant expertise including – where appropriate per the content of the unit – academic staff, qualified professional practitioners, demonstrators/technicians and research students.

To ensure that the practical aspects of the programme are appropriately aligned with current industry practice, and to provide students with a broad range of high quality learning opportunities aimed at enhancing employability; some units may be jointly delivered by a mixture of both BU lecturers and appropriately qualified industrial professional practitioners.

INTENDED LEARNING OUTCOMES – AND HOW THE PROGRAMME ENABLES STUDENTS TO ACHIEVE AND DEMONSTRATE THE INTENDED LEARNING OUTCOMES

PROGRAMME AND LEVEL 6 INTENDED PROGRAMME OUTCOMES

<p>A: Subject knowledge and understanding</p> <p>This programme provides opportunities for students to develop and demonstrate knowledge and understanding of:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>A1 relevant theories, concepts and principles pertinent to games software engineering;</p> <p>A2 the tools, techniques and industry relevant software with which games developers operate;</p> <p>A3 appropriate research methodologies in carrying out independent research in computer games and produce a report demonstrating evidence of critical thinking;</p> <p>A4 the multi-disciplinary nature of games software engineering and the need to apply concepts from a range of scientific principles;</p> <p>A5 the full life-cycle of a computer games project;</p> <p>A6 an entrepreneurial understanding of the business and financial constraints in computer game development.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (A1- A6); • seminars (A1 – A6); • practical tutorial or lab sessions (A1-A6); • directed reading (A1 – A6); • use of the VLE (A6); • independent study time (A1-A6); • personal development (A1-A6); • independent research (for dissertation) (A3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (A1-A6); • examination (A1, A4); • dissertation (A3).
<p>B: Intellectual skills</p> <p>This programme provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme outcomes:</p>
<p>B1 critically evaluate theory and practice of programming principles;</p> <p>B2 analyse and synthesise information for computer-based systems;</p> <p>B3 integrate and synthesise evidence from a range of sources to support findings, proposed solutions and hypotheses;</p> <p>B4 effectively deploy appropriate methods and tools for the definition, construction and development of functioning computer games;</p> <p>B5 explain fundamental programming paradigms and contextual use cases, with knowledge of underpinning benefits and limitations.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (B1-B5); • seminars (B1-B5); • practical tutorial or lab sessions (B1-B5); • directed reading (B1-B5); • use of the VLE (B4); • independent study time (B1-B5); • personal development (B1-B5); • independent research (B3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (B1-B5); • examination (B1, B2, B5);

Programme Specification - Section 2

	<ul style="list-style-type: none"> dissertation (B3).
<p>C: Practical skills</p> <p>This programme provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>C1 demonstrate confidence and competence in the use of theory, practice and tools to specify, design and implement computer games;</p> <p>C2 conduct research into business and management issues;</p> <p>C3 use appropriate skills to communicate effectively in business situations;</p> <p>C4 work as part of a development team with an implicit understanding of appropriate and intrinsic methodologies;</p> <p>C5 Using core analytical techniques and design tools, write appropriate computer programs.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> lectures (C3); seminars (C1-C5); practical tutorial or lab sessions (C1-C5); independent study time (C2, C5); personal development (C1-C5); independent research (C2). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> coursework (C1-C5).
<p>D: Transferable skills</p> <p>This programme provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>D1 perform effectively when working in collaboration with others; deploy a range of interpersonal skills including effective listening, negotiating, persuasion and presentation;</p> <p>D2 undertake research and demonstrate literature review skills;</p> <p>D3 demonstrate openness and sensitivity to diversity in terms of other people, cultures and business and management issues;</p> <p>D4 manage their own motivation, tasks and behaviour in enterprising, innovative and professionally appropriate ways;</p> <p>D5 solve numerical problems and analyse information;</p> <p>D6 devise innovation to practical programming problems.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> lectures (D5, D6); seminars (D1-D6); practical tutorial or lab sessions (D1-D6); directed reading (D1-D6); use of the VLE (D5, D6); independent study time (D2, D4); personal development (D1, D4); independent research (D2). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> coursework (D1-D6); examination (D5, D6); dissertation (D2, D4).

LEVEL 5 / Dip HE INTENDED LEVEL OUTCOMES

<p>A: Subject knowledge and understanding</p> <p>This level provides opportunities for students to develop and demonstrate knowledge and understanding of:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>A1 relevant theories, concepts and principles pertinent to games software engineering;</p> <p>A2 the tools, techniques and industry relevant software with which games developers operate;</p> <p>A3 appropriate research methodologies to produce a report demonstrating evidence of implementation strategies and critical thinking;</p> <p>A4 the multi-disciplinary nature of games software engineering;</p> <p>A5 the integration of sub-systems into games and game-engines;</p> <p>A6 an entrepreneurial understanding of the business and financial constraints in computer game development.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (A1- A6); • seminars (A1 – A6); • practical tutorial or lab sessions (A1-A6); • directed reading (A1 – A6); • use of the VLE (A6); • independent study time (A1-A6); • personal development (A1-A6); • independent research (for dissertation) (A3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (A1-A6); • examination (A1, A4); • dissertation (A3).
<p>B: Intellectual skills</p> <p>This level provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme outcomes:</p>
<p>B1 feedback on theory and practice of programming principles;</p> <p>B2 analyse information on algorithms for a variety of principal game components;</p> <p>B3 give examples of evidence from a range of sources to support findings and hypotheses;</p> <p>B4 apply appropriate methods and tools for the definition, construction and development of functioning computer games;</p> <p>B5 explain intermediate programming paradigms and contextual use cases.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (B1-B5); • seminars (B1-B5); • practical tutorial or lab sessions (B1-B5); • directed reading (B1-B5); • use of the VLE (B4); • independent study time (B1-B5); • personal development (B1-B5); • independent research (B3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (B1-B5); • examination (B1, B2, B5); • in-class test (B1, B4, B5); • dissertation (B3).

Programme Specification - Section 2

<p>C: Practical skills</p> <p>This level provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>C1 test a range of established techniques to implement a design solution;</p> <p>C2 conduct research into business and management issues;</p> <p>C3 use appropriate skills to communicate effectively in business situations;</p> <p>C4 work as part of a development team;</p> <p>C5 demonstrate an understanding of how graphics systems are integrated into games and game engines.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (C3); • seminars (C1-C5); • practical tutorial or lab sessions (C1-C5); • independent study time (C2, C5); • personal development (C1-C5); • independent research (C2). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (C1-C5).
<p>D: Transferable skills</p> <p>This level provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>D1 improve upon a range of interpersonal skills including effective listening, negotiating, persuasion and presentation;</p> <p>D2 undertake research and demonstrate literature review skills;</p> <p>D3 demonstrate openness and sensitivity to diversity in terms of other people, cultures and business and management issues;</p> <p>D4 manage their own motivation and time;</p> <p>D5 solve numerical problems for certain problems, for example artificial intelligence or matrix manipulation;</p> <p>D6 apply underpinning programming principles to real problems.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (D5, D6); • seminars (D1-D6); • practical tutorial or lab sessions (D1-D6); • directed reading (D1-D6); • use of the VLE (D5, D6); • independent study time (D2, D4, D5); • personal development (D1, D4); • independent research (D2). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (D1-D6); • examination (D5, D6); • dissertation (D2, D4).

LEVEL 4 / Cert HE INTENDED LEVEL OUTCOMES

<p>A: Subject knowledge and understanding</p> <p>This level provides opportunities for students to develop and demonstrate knowledge and understanding of:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:</p>
<p>A1 foundational theories, concepts and principles pertinent to games software engineering;</p> <p>A2 the tools, techniques and industry relevant software with which games developers operate;</p> <p>A3 describing low level concepts through report writing;</p> <p>A4 a range of scientific principles;</p> <p>A5 the life-cycle of a computer games project.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (A1- A5); • seminars (A1 – A5); • practical tutorial or lab sessions (A1-A5); • directed reading (A1 – A5); • independent study time (A1-A5); • personal development (A1-A5); • independent research (for dissertation) (A3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (A1-A6); • examination (A1, A4); • dissertation (A3).
<p>B: Intellectual skills</p> <p>This level provides opportunities for students to:</p>	<p>The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme outcomes:</p>
<p>B1 discuss theory and practice of programming principles;</p> <p>B2 analyse information for computer-based systems;</p> <p>B3 start to undertake evidence-based research;</p> <p>B4 gain experience with tools for the definition, construction and development of functioning computer games;</p> <p>B5 review programming paradigms.</p>	<p>Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • lectures (B1-B5); • seminars (B1-B5); • practical tutorial or lab sessions (B1-B5); • directed reading (B1-B5); • use of the VLE (B4); • independent study time (B1-B5); • personal development (B1-B5); • independent research (B3). <p>Assessment strategies and methods (referring to numbered Intended Learning Outcomes):</p> <ul style="list-style-type: none"> • coursework (B1-B5); • examination (B1, B2, B5); • dissertation (B3).

Programme Specification - Section 2

C: Practical skills This level provides opportunities for students to:	The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:
C1 identify industry-standard game production software and components; C2 review numbers, algebra, discrete mathematics, geometry and coordinate systems in the context of computer graphics; C3 identify the fundamental components and operations of computer systems; C4 recognise different development methodologies; C5 write appropriate computer programs.	Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes): <ul style="list-style-type: none"> • lectures (C3); • seminars (C1-C5); • practical tutorial or lab sessions (C1-C5); • independent study time (C2, C3, C5); • personal development (C1-C5).
	Assessment strategies and methods (referring to numbered Intended Learning Outcomes): <ul style="list-style-type: none"> • coursework (C1-C5).
D: Transferable skills This level provides opportunities for students to:	The following learning and teaching and assessment strategies and methods enable students to achieve and to demonstrate the programme learning outcomes:
D1 start to work in teams, gaining insight into tenets of programming practice; D2 undertake directed research; D3 manage their own time; D4 address numerical problems; D5 approach practical programming problems.	Learning and teaching strategies and methods (referring to numbered Intended Learning Outcomes): <ul style="list-style-type: none"> • lectures (D4, D5); • seminars (D1-D5); • practical tutorial or lab sessions (D1-D5); • directed reading (D1-D5); • use of the VLE (D4, D5); • independent study time (D2, D4); • personal development (D1, D4); • independent research (D2).
	Assessment strategies and methods (referring to numbered Intended Learning Outcomes): <ul style="list-style-type: none"> • coursework (D1-D5); • examination (D4, D5); • in-class test (D1-D5); • dissertation (D2, D4).

ADMISSION REGULATIONS

The regulations for this programme are the University's Standard Undergraduate Admission Regulations with the following exceptions:

- applicants for whom English is not their first language must provide evidence of qualifications in written and spoken English. For BSc (Hons) Games Software Engineering students who do not have an appropriate UK English Language qualification, the English language entry requirement is IELTS (Academic) 6.0 with a minimum of 5.5 in each component, or equivalent. Equivalent English language qualifications as set out in document '[3H - Standards of English for International Students and English Language Qualifications](#)' will be accepted;

The University's standard Admission Regulations are available within section 3.1 of the *ARPP* on the BU website: <https://intranetsp.bournemouth.ac.uk/pandptest/3a-undergraduate-admissions-regulations.doc>.

Articulation:

Students who have successfully completed the Bournemouth University International College (Kaplan) Computing Pathway with a minimum of 50% course average and 60% in English will be automatically accepted for entry, without advanced standing, to Level 4 of BSc (Hons) Games Software Engineering.

PROGRESSION ROUTES

Recognition arrangements provide formally approved entry or progression routes through which students are eligible to apply for a place on a programme leading to a BU award. Recognition does not guarantee entry onto the BU receiving programme only eligibility to apply. In some cases, additional entry criteria such as a Merit classification from the feeder programme may also apply. Please see the Recognition Register (https://intranetsp.bournemouth.ac.uk/pandptest/7J_Recognition_Register_Public.xlsx) for a full list of approved Recognition arrangements and agreed entry criteria.

In order to take advantage of exciting new approaches to learning and teaching, as well as developments in industry, the current, approved Articulation/Recognition/Progression route(s) for this programme may be subject to change. Where this happens students will be informed and supported by the Faculty as early as possible.

ASSESSMENT REGULATIONS

The regulations for this programme are the University's Standard Undergraduate Assessment Regulations. The University's Standard Undergraduate Assessment Regulations are available from: <https://intranetsp.bournemouth.ac.uk/pandptest/6a-standard-assessment-regulations-undergraduate.pdf>

WORK BASED LEARNING (WBL) AND PLACEMENT ELEMENTS

Placements; this programme offers an optional placement year. This bears no credits. The duration of the placement is normally 30 weeks of supervised work experience and the aims of the placement year are to give the students experience of working within an appropriate professional environment which will contribute to their potential employability, mobility and global awareness. Completion of the four year degree, i.e. one with a 30-week placement included, will entitle students to a 'sandwich award'. Shorter (also optional) placements of 4 weeks with no coursework attached are also possible although the sandwich award is then no longer an option. Completion of the three-year full-time degree will, instead, entitle students to a 'full-time award'.

The placement is recognised at Bournemouth University as adding considerable value to graduate profiles and students are very strongly advised to follow the sandwich route. The non-sandwich route is designed for mature students who have experience of the world of work and who may need to

Programme Specification - Section 2

complete their course in three years for financial (or other) reasons. In some cases, on submission of relevant evidence, such students may be eligible for Recognition of Prior Learning (RPL). This will provide them with exemption from the placement year but will still entitle them to a sandwich degree.

The placement draws on some or all the units studied on the first two Levels (4 and 5) of this programme. Successful Level 4 and 5 completion is compulsory before proceeding to the 30-week or 4-week placement. It provides the opportunity for the student to develop their abilities and understanding of related subjects, as well as providing a platform for successful entry into the relevant profession (following graduation). It can also make a major contribution to the understanding of the final Level (6) units, further develops final projects or dissertation research by utilising the context of the work experience as appropriate and, finally, significantly enhances students' prospects of future employment.

Further information on the Department's placements policy and procedure can be sourced in the Creative Technology Placements Handbook and also here (for Bournemouth University policy on the same subject):

<http://intranetsp.bournemouth.ac.uk/pandptest/4k-placements-policy-and-procedure.DOCX>

Programme Specification - Section 2

Programme Skills Matrix

Units		Programme Intended Learning Outcomes																					
		A 1	A 2	A 3	A 4	A 5	A 6	B 1	B 2	B 3	B 4	B 5	C 1	C 2	C 3	C 4	C 5	D 1	D 2	D 3	D 4	D 5	D 6
L E V E L 6	Graphics and Computational Programming	x		x	x			x	x	x	x	x				x	x		x		x	x	x
	Innovation, Enterprise and Business Development						x			x	x		x	x	x	x		x	x	x		x	
	Game Engine Programming	x		x	x			x	x		x	x	x				x		x		x		x
	Individual Project		x			x		x		x				x	x		x	x	x		x		
L E V E L 5	3D Graphics Programming	x	x	x	x			x	x	x	x	x					x		x		x	x	x
	Physics for Games	x			x			x			x	x					x				x	x	x
	Programming for Interaction	x			x			x			x		x				x						x
	Game Studio Project			x		x		x		x	x		x	x	x	x	x	x	x	x			x
	AI Game Programming	x		x	x			x		x	x	x					x				x		x
	Commercialisation and Business Environment						x			x	x		x	x	x	x		x	x	x			
L E V E L 4	Games Design Principles	x			x						x		x						x	x			
	Introduction to Programming	x						x			x	x					x				x		x
	Digital Technologies	x	x		x				x		x		x										
	Mathematics for Computer Graphics	x		x	x			x		x	x	x					x		x		x	x	
	Object Oriented Game Programming	x	x					x			x	x				x	x	x		x	x		x
	Game Development Pipeline	x	x			x			x		x		x				x				x		x

Programme Specification - Section 2

<p>A – Subject Knowledge and Understanding This programme provides opportunities for students to develop and demonstrate knowledge and understanding of:</p> <ol style="list-style-type: none"> 1. relevant theories, concepts and principles pertinent to games software engineering; 2. the tools, techniques and industry relevant software with which games developers operate; 3. appropriate research methodologies in carrying out independent research in computer games and produce a report demonstrating evidence of critical thinking; 4. the multi-disciplinary nature of games software engineering and the need to apply concepts from a range of scientific principles; 5. the full life-cycle of a computer games project; 6. an entrepreneurial understanding of the business and financial constraints in computer game development. 	<p>C – Subject-specific/Practical Skills This programme provides opportunities for students to:</p> <ol style="list-style-type: none"> 1. demonstrate confidence and competence in the use of theory, practice and tools to specify, design and implement computer games; 2. conduct research into business and management issues; 3. use appropriate skills to communicate effectively in business situations; 4. work as part of a development team with an implicit understanding of appropriate and intrinsic methodologies; 5. Using core analytical techniques and design tools, write appropriate computer programs.
<p>B – Intellectual Skills This programme provides opportunities for students to:</p> <ol style="list-style-type: none"> 1. critically evaluate theory and practice of programming principles; 2. analyse and synthesise information for computer-based systems; 3. integrate and synthesise evidence from a range of sources to support findings, proposed solutions and hypotheses; 4. effectively deploy appropriate methods and tools for the definition, construction and development of functioning computer games; 5. explain fundamental programming paradigms and contextual use cases, with knowledge of underpinning benefits and limitations. 	<p>D – Transferable Skills This programme provides opportunities for students to:</p> <ol style="list-style-type: none"> 1. perform effectively when working in collaboration with others; deploy a range of interpersonal skills including effective listening, negotiating, persuasion and presentation; 2. undertake research and demonstrate literature review skills; 3. demonstrate openness and sensitivity to diversity in terms of other people, cultures and business and management issues; 4. manage their own motivation, tasks and behaviour in enterprising, innovative and professionally appropriate ways; 5. solve numerical problems and analyse information; 6. devise innovation to practical programming problems.

APPENDICES

BSc Games Software Engineering

