Programme Specification

Course record information

Name and level of final award:	MSc The MSc in Advanced Software Engineering is an MSc that is Bologna FQ-EHEA first cycle degree or diploma compatible.
Name and level of intermediate awards:	Postgraduate Diploma Postgraduate Certificate
Awarding body/institution:	University of Westminster
Status of awarding body/institution:	Recognised Body
Location of delivery:	New Cavendish Street
Language of delivery and assessment:	English
Course/programme leader:	Dr Simon Courtenage
Course URL:	www.westminster.ac.uk/courses/subjects/business-information-systems/postgraduate-courses
Mode and length of study:	Full Time – 1 year Part-Time (Mixed) – 2 years minimum
University of Westminster course code:	W50
JACS code:	
UCAS code:	P023043
QAA subject benchmarking group:	Subject Benchmark Statement: Master's degrees in Computing 2011, available at http://www.qaa.ac.uk/en/Publications/Documents/SBS-Masters-degree-computing.pdf
Professional body accreditation:	TBC for BCS
Date of course validation/review:	2016
Date of programme specification:	May 2016

Admissions requirements

Consideration will be given to all applicants with a minimum lower second class (2.ii) honours degree from a British University or overseas equivalent in a subject related to computing or

software engineering. Suitable subjects might include Computing, Computer Science or Software Engineering, joint degrees including one of these subjects, or possibly other science or mathematically based subjects that include significant amounts of computer science material.

In admitting individual students to the course, however, an important consideration is the student's ability to benefit from the course and complete it satisfactorily. As a consequence, additional factors such as continuous professional development, training, and the current role of applicants or recent work experience may also be taken into account in individual cases.

All applicants are required to show competence in both written and spoken English; thus, overseas applicants whose first language is not English are normally required to have attained the equivalent of an IELTS score of at least 6.5 with 6.0 or above in each element prior to joining the course (more information on minimum scores for other language tests can be obtained from the admissions office).

All applicants are required to submit with their application, copies of their academic and/or professional qualifications and transcripts, two references (one of which should be academic, for applicants who have been in Higher Education in the 5 years prior to applying for the course), and a statement explaining the reasons they want to be admitted to the course, what they expect from the course, how they are going to achieve it, what they will bring to the course, what their career aspirations are and how they think the course can help them achieve those aspirations.

The admissions policy conforms to the Equal Opportunities Policy and the Admissions Policy of the University of Westminster. Each application is considered on its individual merits and decisions in admitting applicants to the course are made based on evidence that the applicant is likely to benefit from the course and to complete it satisfactorily.

Part-time students in employment are advised to seek the approval of their employers to attend the course, as they may be required to attend classes during work hours. As a number of modules are also delivered in the evening, students should also be aware of the need to make provision for evening attendance on campus.

Aims of the course

Software Engineers apply engineering discipline to the creation and maintenance of software applications and systems. In approaching their work, therefore, they use methods and methodologies that emphasise a systematic, quantifiable and disciplined approach. The aims of this course are to enable students to extend their knowledge of, and gain valuable experience in, software engineering as it applies to a number of new and important areas of IT and computing.

The rapid pace of technical change in software development is notorious and this has been accompanied and compounded by an increase in the complexity of the systems that are developed. Recently this has been most noticeable in the increase in mobile computing and the use of sophisticated hardware that require developer knowledge of new paradigms.

Many applications that run on these systems whether mobile or stationary are distributed in nature and will consume web services provided by service-oriented architectures and cloud-based platforms. There has also been an increase in the use of virtualisation techniques for providing flexible and maintainable systems. Businesses are now regularly using virtualised systems and techniques to lower cost and complexity and increase availability in computing environments.

Finally, the surge in cybersecurity issues and threats facing businesses and organisations that depend on IT systems has meant that software engineers need a thorough understanding of security when building and maintaining software applications and systems.

There is an acknowledged national shortage of IT and computing skills in the workforce. In the specific area of software development, a number of factors contribute to this. Most obviously, the rate of technological change means that an individual's specific knowledge frequently becomes out of date. Secondly, many significant technological developments originate in industry rather than academia, and are not yet firmly embedded in undergraduate curricula. Finally, many people enter the software industry without a specific educational background in computer science and acquire much vital knowledge in the workplace in relatively ad hoc ways.

In response to this, for many years the Department of Computer Science has been running courses that combine an emphasis on methodical approaches to the development of software applications and information systems with a determination to equip graduates with a portfolio of relevant research-oriented and practical skills and knowledge to compliment and expand their own knowledge.

The rationale behind the MSc in Advanced Software Engineering is to draw on this experience to provide an education that will cover in-depth specific skills and best current practice in software development where there is currently a significant skills shortage, whilst at the same time instilling important research-based skills that will equip students for independent lifelong learning in fast-changing and technically challenging environment.

More specifically, the course provides a balanced study that aims to produce students capable of:

(CA1) thinking in a systematic and methodical way about the building and maintenance of software applications and systems;
(CA2) building and enhancing software applications and systems using current and novel technologies and platforms;
(CA3) understanding cybersecurity issues and threats, and using a systematic and methodical approach to security in the building of software applications and systems;
(CA4) critically evaluating alternative design and implementation strategies, and assessing the impact of new technologies and platforms
(CA5) independent in-depth analysis and development of a chosen topic relating to software applications and systems, making use of information and knowledge resources beyond the classroom and critically evaluating academic research and industrial practice;
(CA6) developing professional standards of work, both independently and collaboratively, as well as the interpersonal skills and attitudes required of a software engineer both in industry as well as in academia;
(CA7) being an autonomous and reflective learner who takes the initiative and responsibility for acquiring new knowledge and skills, employment and further study opportunities.

Employment and Further Study Opportunities

Today’s organisations need graduates with both good degrees and skills relevant to the workplace, i.e. employability skills. The University of Westminster is committed to developing employable graduates by ensuring that:

- Career development skills are embedded in all courses
- Opportunities for part-time work, placements and work-related learning activities are widely available to students

- Staff continue to widen and strengthen the University's links with employers in all sectors, involving them in curriculum design and encouraging their participation in other aspects of the University's career education and guidance provision
- Staff are provided with up-to-date data on labour market trends and employers' requirements, which will inform the service delivered to students.

Learning outcomes

Learning outcomes are statements on what successful students have achieved as the result of learning. These threshold statements of achievement and are linked to the knowledge, understanding and skills that a student will have gained on successfully completing a course.

Knowledge and understanding

On completion of the course, the student will be able to:

KU1: Demonstrate a systematic understanding of the knowledge base, and a critical awareness of current problems and/or new insights generated by both academic research and professional practice;

KU2: Have a comprehensive understanding of the methods, technologies and techniques applicable for the design, development, implementation and maintenance of software systems;

KU3: show originality and innovation in the application of knowledge, methodologies and techniques in the design and implementation of software systems;

KU4: show critical awareness of current research issues, problems and approaches;

KU5: understand and be able to participate within relevant professional, legal and ethical frameworks as a software engineering professional;

KU6: critically evaluate current and new technologies, methodologies and standards, and where appropriate propose new approaches.

Specific skills

The course is designed for those students who wish to further their career as software engineers, by providing them with specific skills appropriate to current industrial practice. These include:

SS1: ability to specify, design and implement software systems, using knowledge of software development processes, methodologies and tools;

SS2: ability to apply advanced object-oriented design and development methods and techniques to construct complex and innovative software solutions in a wide range of novel and different domains;

SS3: ability to recognize, appreciate and operate within relevant and appropriate legal, ethical and professional frameworks relating to software engineering within industry and academia;

SS4: ability to conduct independent in-depth analysis and/or study within an area related to software engineering, involving critical evaluation of advanced source material;

SS5: dependent on the choice of option modules, ability to critically evaluate, and apply knowledge and experience of, the following:

- Data management and data analytics
- Mobile and Web Application Development (native and managed)
- Security and authentication processes and techniques, as well as security risks and threats applied to the development and maintenance of software systems
- Service oriented architectures, cloud computing and implementation of web services

Key transferable skills

Upon completion of the course students will have developed a number of general rather than discipline-specific skills which any practitioner must have if s/he is to be successful. These Key Transferable Skills developed and assessed at postgraduate level are:

KTS1: Group working
Students will be able to (a) work effectively within a group both as group leaders and/or group members; (b) clarify tasks and make appropriate use of group members abilities; (c) negotiate and handle conflict with confidence; and (d) participate effectively in the peer review process;

KTS2: Learning resources
Students will be able to use a full range of learning resources to carry out literature reviews and engage in research activity;

KTS3: Self-evaluation
Students will be able to reflect on own and others functioning; participate effectively in the peer review process and analyse and identify ways to improve practice; continue to advance their knowledge and understanding, and recognise their development needs and to develop new skills to a high level;

KTS4: Management of information
Students will be able to competently undertake research tasks with minimum guidance; sieve through information clatter to identify relevance, to organise and present information effectively using different media;

KTS5: Autonomy
Students will be independent and self-critical learner, who can act autonomously in planning and implementing tasks and who will be able to guide the learning of others;

KTS6: Communication
Students can engage confidently in academic and professional communication with others, reporting on action clearly, autonomously and competently;

KTS7: Problem solving
Students have independent learning ability required for continuing professional study, making professional use of others where appropriate.

Lifelong Learning

In addition to the above, by the end of the course, students will:

LL1: Be able to make decisions in complex and unpredictable situations and those where technical material must be quickly and reliably mastered;

LL2: Have acquired or reinforced appropriate habits of self-organisation, in terms of initiative and personal responsibility;

LL3: Be prepared for work towards a research degree or continued professional development.

Learning, teaching and assessment methods

Learning & Teaching

The teaching and learning strategy adopted on the course makes use of a variety of teaching methods to ensure that material is presented effectively to students and in a way that emphasizes the practical nature of the subject. The wide range of skills and knowledge required for the successful development of complex software systems means that a correspondingly wide range of strategies is adopted to develop these skills in students.

Formal lectures are used on the taught modules as a method for providing a firm grounding in the theory, methods and techniques relevant to the module topic. Tutorials are tutor-led sessions, and usually adopt a more problem-solving approach than is feasible in a formal lecture. Tutorials are usually held in computer laboratories as practical sessions in which students work at their own speed on set tasks. Other methods used are student presentations or student-led seminars, and a form of “surgery" where students can raise any problems related to the module.

The project is probably the most important aspect of the Master’s programme. It plays a unifying role in the course by providing, in effect, the equivalent of a programme of integrated assignments that draws directly on all of the taught modules of the course.

Students are expected to work on the project that is on a topic that each student has chosen, in the summer months after the end of the taught part of the course under the supervision of a member of academic staff.

Generally, there are three types of projects: (a) projects proposed by students themselves (typically such projects are based on idea(s) a student has come up with that were developed following a supervisor input to an appropriate for the level and standard project); (b) projects based on an idea suggested by teaching staff that a student has researched and developed to an appropriate for the level and standard project; and finally (c) work-based projects, the latter of which, in most cases, are undertaken by part-time students.

To help students build the required background for their project and develop further their research skills, students are required to take a project preparatory module as part of which they are introduced to various project areas; choose the topic/area of their project; are allocated a project supervisor who, in most cases, has research interests in the area of a student’s chosen project topic; research the area of their project; and devise a proposal detailed enough that will enable them to complete their project.

The supervisor acts in effect as someone who will guide students throughout the various phases of the project and who students will turn to in order to discuss their project work and receive feedback on the progress made and to have informed discussions on technical and research matters related to their project. Supervisors will also help students (a) decide on the scope of the project; (b) devise a project plan; (c) monitor their progress and adhere to target dates on provides; and (d) on how to tackle the writing up of the project report.

To support students in their studies and to allow access to module materials and course related information, significant use is made of web-based teaching materials. The modules’ pages on the

University's Virtual Learning Environment (Blackboard) and/or the faculty's intranet pages are used as repositories for lecture notes, presentation transparencies, course/assessment schedules, coursework (including feedback) and occasionally for assessment purposes. The course recognizes the importance of individuals being able to function equally well both as individuals and as members of team; thus, group activities are encouraged and promoted. To support and encourage student face to face interaction and collaborative work through exchange of emails, files, and online discussions, the facilities offered by the University's Virtual Learning Environment are commonly utilized.

To summarize, teaching and learning strategies involve the use of

- case studies, to improve students' analytical and problem solving skills;
- use of specialized software tools and packages, such as Development Environments and Computer Aided Software Engineering (CASE) tools, to build students hands on skills and understanding of such tools;
- presentations from outside speakers with industrial experience, to enable students see how the taught material is applied in industry;
- team/group work, to enable students develop further their teamwork skills to work effectively in a professional environment;
- research methods involving the use of library and online sources to develop students research and analysis skills.
- presentations and academic report writing as part of the assignments set, to develop further these important skills.

Assessment

Students are assessed in a number of different modes and according to a variety of criteria. In all cases, assessment is closely connected to both teaching methodology and learning outcomes. Learning outcomes in modules show specific achievement targets, part of whose purpose is to allow precise assessment and evaluation. However, it is important that assessment should be seen as a vital part of the learning process; for this, the formative element of the coursework-based assessment will be served through appropriate and timely feedback from tutors.

The key elements in the assessment process are:

1. written assignments based on classroom learning and further reading and/or research;
2. practical assignments that involve applying ideas, concepts, methods, tools and techniques learnt or investigated to solve a problem or address the requirements of a brief;
3. practical and written classroom-based phase tests and final examinations under time constraints designed to test students' ability to apply and discuss concepts and techniques from the programme;
4. viva voce examinations and presentations.

A number of the taught modules in the programme are entirely assessed through coursework, but the diet of assessment for a significant number of modules involves both a coursework and an examination component, the latter of which normally takes the form of a two-hour examination at the end of the academic year. Where the assessment of a module involves both examination and coursework, the relative weightings of the examination and coursework components are normally 70/30 or 50/50 for option modules.

To pass a module, students must achieve an overall mark of 50% in the module. In addition, and depending on the module's assessment diet, students must achieve at least 35% (qualifying mark) in the coursework and/or in the examination. In particular, in order to pass a module whose assessment diet involves a combination of coursework and exam, students in addition to

achieving the pass mark overall, they must also achieve the qualifying mark in the exam and the coursework (on aggregate); in order to pass a coursework only assessed module, students are expected to achieve the pass mark overall and to also achieve the qualifying mark in each individual coursework component. Students, who fail to achieve the above, will be deemed as having failed the module and depending on their performance they may be offered a re-assessment opportunity.
At the discretion of the Assessment Board, a student may be re-assessed (re-sit) once only in any module other than the project module on each occasion that the student attempts the module. The following guidelines can affect potential re-assessments (in what follows the term assessment component should be understood as examination, coursework or grouping of assessment elements that the qualifying mark needs to be achieved, possibly on aggregate):

- If an overall mark of 50% or above is achieved and there is a particular component where a score of less than 35% is achieved, then the student will be deemed as not having passed the module and they may be offered a re-assessment in that component. The overall mark for a module successfully completed following a re-assessment will be capped at 50%.
- If an overall mark between 40% and 49% is achieved, then students may be offered reassessment in the components they have not achieved the passing mark. The overall mark for a module successfully completed following a reassessment will be capped at 50%. If an overall mark of less than 40% is achieved, then regardless of the score of individual components the student may have to retake the module the following year with attendance.
- If an overall mark of less than 40% is achieved, then regardless of the score of individual components the student may have to retake the module the following year with attendance.

The table below summarises the above guidelines:

		Assessment Component Mark	
		< 35%	≥ 35%
Overall Mark	50% or above	Reassess	Pass
	Between 40%-49%	Reassess	Reassess
	Less than 40%	Retake	Retake

Course structure

This section shows the core and option modules available as part of the course and their credit value. Full-time Postgraduate students study 180 credits per year.

Credit Level 7				
Module Code	**Module Title**	**Status**	**UK Credit**	**ECTS**
7SENG003W	Advanced Software Design	Core	20	10
7SENG007W	Concurrency and Parallelism	Core	20	10
7SENG001W	Enterprise Development	Core	20	10
7BDIN005W	Advanced Big Data Analytics	Option	20	10
7BDIN006W	Big Data Theory and Practice	Option	20	10
7BUIS027W	Cloud Computing Applications	Option	20	10
7CSEF002W	Cyber Security Threats and Countermeasures	Option	20	10
7BUIS008W	Data Mining and Machine Learning	Option	20	10
7COSC007W	Internet Security	Option	20	10
7IDCO004W	Mobile and Ubiquitous Computing	Option	20	10
7SENG002W	Mobile Application Development	Option	20	10
7MMCS006W	Usability and User Experience Design	Option	20	10

	Free choice module	Option	20	10
7BUIS019W	Research Methods and Professional Practice	Core	0	0
7SENG008W	Advanced Software Engineering Project	Core	60	30

Please note: Not all option modules will necessarily be offered in any one year. The availability of modules depends on resources and the number of students selecting a particular option module.

Full time students are expected to complete the course within a calendar year, whereas students doing the course in part-time mode are normally expected to complete it over a two-year period. The above means that full time students cover the taught part of the course over the two semesters of an academic year and that they work on their project during the summer months of the same year. Part time students cover the taught part of the course over four semesters (two years) and that they are expected to work on their project during the summer months their second (last) year of their studies.

Academic regulations

The MSc Advanced Software Engineering and its intermediate awards operate in accordance with the University's Academic Regulations and the Framework for Higher Education Qualifications in England, Wales and Northern Ireland published by the Quality Assurance Agency for Higher Education (QAA) in 2008.

All students should make sure that they access a copy of the current edition of the general University handbook called Essential Westminster, which is available at www.westminster.ac.uk/essential-westminster. The following regulations should be read in conjunction with Section 18: Modular Framework for Postgraduate Courses and relevant sections of the current Handbook of Academic Regulations, which is available at www.westminster.ac.uk/academic-regulations.

Award

To qualify for the award of MSc Advanced Software Engineering, a student must have:

- obtained a minimum of 180 credits at Level 7;
- attempted modules worth no more than 240 credits (a first attempt of any module will count as an attempt, and a re-attempt of any module that a student has failed will count as a further, separate attempt; re-assessment following referral at the first sit will not count as a further separate attempt); and
- satisfied the requirements contained within any course specific regulations for the relevant Course Scheme.

The University may award:

- a Masters Degree with Merit to a student whose marks average at least 60% across modules at Level 7.
- a Masters Degree with Distinction to a student whose marks average at least 70% across the modules at Level 7.

Intermediate Awards

These are awards that students are not normally registered for in the first instance. A student's registration may be changed to one of these exit awards, if a student has failed too many modules

and cannot be considered for the target award s/he is registered for or a student claims such an award because s/he is withdrawing the course.

Postgraduate Diploma in Advanced Software Engineering

In order to be awarded a Postgraduate Diploma (PgDip) in Advanced Software Engineering, a student must pass modules worth at least 120 credits and attempt modules worth no more than 240 credits. The modules a student needs to pass to be eligible for the award of the Postgraduate Diploma (PgDip) in Advanced Software Engineering qualification are all level 7 modules and include:

– all of the following core modules (60 credits):

Module Code	Module Title	Status	UK Credit	ECTS
7SENG003W	Advanced Software Design	Core	20	10
7SENG007W	Concurrency and Parallelism	Core	20	10
7SENG001W	Enterprise Development	Core	20	10

– and three of the following optional modules (80 credits):

Module Code	Module Title	Status	UK Credit	ECTS
7BDIN005W	Advanced Big Data Analytics	Option	20	10
7BDIN006W	Big Data Theory and Practice	Option	20	10
7BUIS027W	Cloud Computing Applications	Option	20	10
7CSEF002W	Cyber Security Threats and Countermeasures	Option	20	10
7BUIS008W	Data Mining and Machine Learning	Option	20	10
7COSC007W	Internet Security	Option	20	10
7IDCO004W	Mobile and Ubiquitous Computing	Option	20	10
7SENG002W	Mobile Application Development	Option	20	10
7MMCS006W	Usability and User Experience Design	Option	20	10

The University may award a Postgraduate Diploma with

- Merit to a student whose marks average at least 60% across the modules contributing to the award, or
- Distinction to a student whose marks average at least 70% across the modules contributing to the award.

Postgraduate Certificate (PgCert) in Advanced Software Engineering

In order to be awarded a Postgraduate Certificate in Advanced Software Engineering, a student must pass modules worth at least 60 credits and attempt modules worth no more than 240 credits.

The modules a student needs to pass to be eligible for the award of the Postgraduate Certificate in Advanced Software Engineering qualification are all level 7 modules and include:

– all of the following core modules (60 credits):

Module Code	Module Title	Status	UK Credit	ECTS
7SENG003W	Advanced Software Design	Core	20	10
7SENG007W	Concurrency and Parallelism	Core	20	10
7SENG001W	Enterprise Development	Core	20	10

The University may award a Postgraduate Certificate with

- Merit to a student whose marks average at least 60% across the modules contributing to the award, or

- Distinction to a student whose marks average at least 70% across the modules contributing to the award.

Support for students

Upon arrival, an induction programme will introduce students to the staff responsible for the course, the campus on which they will be studying, the Library and IT facilities and to the School Registry. Students will be provided with the Course Handbook, which provides detailed information about the course. Students are allocated a personal tutor who can provide advice and guidance on academic matters.

Learning support includes four libraries, each holding a collection of resources related to the subjects taught at their School. Students can search the entire library collection online through the Library Search service to find and reserve printed books, and access electronic resources (databases, e-journals, e-books).

Students can choose to study in the libraries, which have areas for silent and group study, desktop computers, laptops for loan, photocopying and printing services. They can also choose from several computer rooms at each campus where desktop computers are available with the general and specialist software that supports the courses taught at their School. Students can also securely connect their own laptops and mobile devices to the University wireless network.

The University uses a Virtual Learning Environment called Blackboard where students access their course materials, and can communicate and collaborate with staff and other students.

At University level, Services for Students provide advice and guidance on accommodation, financial and legal matters, personal counselling, health and disability issues, careers and the chaplaincy providing multi-faith guidance. The International Office provides particular support for international students. The University of Westminster Students' Union also provides a range of facilities to support all students during their time at the University.

School and Course Support
Student Handbook and Module Guides: there is a general University postgraduate student handbook as well as a course specific handbook. All module leaders provide module guides that include: a module description, teaching schedule, assessment schedule and lecture notes.

Personal Tutor System: the course leader normally acts as personal tutor for all students, until a student is allocated a project supervisor, who then acts as the personal tutor.

*Computer equipmen*t: the two main computer systems used on the course, these are Microsoft Windows and Mac OSX computers running in various laboratories. Students also have access to mobile devices and embedded hardware where necessary.

Students have access to 18 computer laboratories.

Course Committees: are the forum for students and staff to present their views on the operation and development of the course. The Course Committee meets once per semester and reports on all aspects of the course. There is also a campus level student committee that provides feedback on general issues, related to the campus.

Reference points for the course

Internally:
- The University’s Handbook of Academic Regulations

- The University's Mission Statement and 2020 Strategy
- The University's Quality Assurance and Enhancement Handbook
- Research activity in Mobile Computing, Cloud Computing, and Software Development.

Externally:
QAA Subject Benchmark statements. University and SEEC (credit consortium) level descriptors.

Professional body accreditation:

BCS, Guidelines on Course Accreditation – Information for Universities and Colleges

Quality management and enhancement

Course management

The management structure supporting the course is as follows:

- Dr Simon Courtenage: Course leader, responsible for day-to-day running and overall management of the course and development of the curriculum.
- Dr Aleka Psarrou: Head of Department, holds academic responsibility for the course, and for the other courses within the Department of Computer Science within the Faculty of Science and Technology.
- Prof. Jane Lewis, Dean of School, holds overall responsibility for the course and for other courses run by the Faculty of Science and Technology.

The Course Leader is responsible for the academic management and organisation of the course. The Course Leader, who is also the Admissions Tutor, Personal Tutor and Project Co-Ordinator for the course, is assisted by an Examinations Officer. The Course Team comprises the Course Leader and all the members of staff who teach on the course. Typically, each module is delivered by a module team. Each module has a Module Leader, who is responsible for co-ordinating the module team and for the delivery, resourcing and smooth running of the module.

Course approval, monitoring and review

The course was initially approved as the MSc Computer Science by a University Validation Panel in 2010. The panel included internal peers from the University and external subject specialists from academia and industry to ensure the comparability of the course to those offered in other universities and the relevance to employers. Periodic course review helps to ensure that the curriculum is up-to-date and that the skills gained on the course continue to be relevant to employers.

The course is monitored each year by the School to ensure it is running effectively and that issues which might affect the student experience have been appropriately addressed. Staff will consider evidence about the course, including the outcomes from each Course Committee, evidence of student progression and achievement and the reports from external examiners, to evaluate the effectiveness of the course. The Annual Monitoring Sub-Committee considers the School action plans resulting from this process and the outcomes are reported to the Academic Council, which has overall responsibility for the maintenance of quality and standards in the University.

Student involvement in Quality Assurance and Enhancement

Student feedback is important to the University and student views are taken seriously. Student feedback is gathered in a variety of ways. The most formal mechanism for feedback on the course is the Course Committee. Student representatives will be elected to sit on the Committee to represent the views of their peer group in various discussions. The University and the Students' Union work together to provide a full induction to the role of the Course Committee.

All students are invited to complete a Module Feedback Questionnaire before the end of each module. The feedback from this will inform the module leader on the effectiveness of the module

and highlight areas that could be enhanced. The University also has an annual Student Experience Survey, which elicits feedback from students about their course and University experience.

Students meet with review panels when the periodic review of the course is conducted to provide oral feedback on their experience on the course. Student feedback from course committees is part of the Schools' quality assurance evidence base.

For more information about this course:

Admissions Tutor: Dr Simon Courtenage
Dept of Computer Science
Faculty of Science and Technology
Tel: +44 (0) 20 7911 5000 ext. 64555
Fax: +44 (0) 20 7911 5089
Email: courtes@westminster.ac.uk

Course Enquiries:
Tel: +44 (0) 20 7915 5511
Email: admissions@westminster.ac.uk
Web: http://www.westminster.ac.uk/schools/computing/postgraduates