UWE BRISTOL

Faculty of
Computing, Engineering
and Mathematical Sciences

MSc Software Engineering by Distance Learning

Definitive Documentation – April 2006
Revised November 2007

Revised November 2007

Contents

Page Number

Revised November 2007

Part 1: Programme Specification

Programme Specification

Section 1: Basic Data

Awarding institution/body	University of the West of England
Teaching institution	University of the West of England
Faculty responsible for programme	Computing, Engineering and Mathematical Sciences
Programme accredited by	British Computer Society
Highest award title	MSc Software Engineering
Default award title	
Interim award title	PG Dip in Software Engineering PG Cert Software Engineering
Modular Scheme title (if different)	
UCAS code (or other coding system if relevant)	G70012
Relevant QAA subject benchmarking group(s)	Computing
On-going/valid until* (*delete as appropriate/insert end date)	
Valid from (insert date if appropriate)	1st September 2006

Authorised by… **Date:…**

Version Code 1

For coding purposes, a numerical sequence (1, 2, 3 etc.) should be used for successive programme specifications where 2 replaces 1, and where there are no concurrent specifications. A sequential decimal numbering (1.1; 1.2, 2.1; 2.2 etc) should be used where there are different and concurrent programme specifications

Section 2: Educational Aims of the Programme

The general educational aims of the faculty’s taught postgraduate programmes are:

- to provide an intellectual experience of advanced study, underpinned by staff expertise, research, and experience;
- to enable the student to further and deepen his/her knowledge, understanding and analytical abilities in a stimulating and challenging academic environment;
- to prepare the student for further professional development in his/her chosen field;
- to develop the student’s ability to conduct research in their chosen field;
- to offer postgraduate opportunities for part-time students in employment.

Specifically, after completion of this programme students will:

- Have developed advanced knowledge of all aspects of the software engineering process.
- Be able to apply software engineering principles in a professional manner.
- Understand the problems and constraints that routinely confront a Software Engineer and possess the underlying technical and cognitive skills to resolve them.
- Be able apply the research skills they have developed in the area of Software Engineering

Section 3: Learning Outcomes of the Programme

This programme is delivered within the common University framework of Blackboard. Details of Blackboard can be found on the university website at www.uwe.ac.uk

The programme is designed with an emphasis on a combination of resource-based and problem-based learning. Thus throughout, the learner is directed to a variety of resources and expected to undertake a series of activities in relation to them. In addition, the learner is also encouraged to undertake independent reading both to supplement and consolidate what is being taught/learnt and to broaden individual knowledge and understanding of the subject. For the most part, access to resources will be provided on-line. However, for some individual modules, it will be necessary to purchase a course text. Details of such necessary purchases are given in the individual module specifications.

To support the learning outcomes, all modules will provide on-line conferencing facilities within Blackboard. The conferences will provide learners with easy access to their fellow learners and provide a forum for the discussion of issues raised during the programme. Each forum will be monitored by the relevant module leader.

The award route provides opportunities for students to develop and demonstrate knowledge and understanding, intellectual skills, subject-specific skills and transferable skills as shown in the following pages.

A. Knowledge and Understanding

On successful completion of the programme students will have gained knowledge and understanding of:	***Teaching/Learning Methods and Strategies***	***Assessment***
1. Theoretical perspectives that underpin Software Engineering and distinguish it from other disciplines and which thus allow it to be used as the basis for organised software development. 2. Project planning, estimation and control methods and aids 3. The common activities associated with requirements elicitation, analysis, specification, validation, and management 4. The concerns associated with the analysis, design and implementation of object-oriented software 5. The problems associated with development within a particular application domain. 6. The nature of the research process 7. How to carry out research in order to synthesise a computer-based system that meets given requirements.	The general approach to teaching on this programme is to use the first semester to deliver theoretical concepts (SE Concepts/Project Management) and fundamental software engineering stages (Requirements Engineering/OO Design and Programming) and then to move to a more applied approach in semester 2. For example, in semester 2, the module “Groupwork Software Development” takes a holistic approach and allows the students the opportunity to pursue the whole software development lifecycle from requirements through to delivery. The module “Advanced Databases” provides the opportunity of looking at a particular application domain and considering how software is engineered in that area. In semester 2, students are also encouraged to think about the business of research and their dissertations in particular by the study of Research Methods. All of these outcomes are supported by the use of tutor input in the form of on-line narratives/presentations and activities, as described above. In addition specific outcomes are supported as follows: ▪ 1 by the use of on-line exercises and submissions to the group by individuals on selected areas of systems development as well by experiencing the use of SE principles in the group development of a software system. ▪ 2, 3 and 4 by the completion of on-line practical work, with samples solutions and a commentary from an academic ▪ 5 by the use of electronic conferencing and the completion of on-line exercises. ▪ 6 and 7 by reading, completing exercises and by individual supervision of the production of a dissertation. ▪	The learning outcomes are assessed in a variety of different ways including the assessment of contributions to audio/video conference collaborations; project reports, which may be submitted in stages; essays and exams. Research related learning outcomes are principally assessed by the production of a research proposal. Outcome 7 is assessed by a dissertation report and a software product.

B. Intellectual Skills

On successful completion of the programme students will have the intellectual skills necessary to:	***Teaching/Learning Methods and Strategies***	***Assessment***
1. To comprehend new concepts both for use during the course and, later, professionally 2. Understand human roles and how various functions interact to successfully produce an artefact. 3. Recognize the role of the engineering approach in the production of an artefact – in particular a software system. 4. Formulate, analyse, visualise, synthesize and communicate plans and designs to solve problems in software applications 5. To explore, evaluate and use software development technology 6. Evaluate alternative approaches to problem solving within an application domain. 7. Critically analyse theoretical perspectives relevant to the research process 8. Evaluate research methodologies, tools and techniques, and the process of research	In general, the skills listed on the left are developed by asking the learner to read some (usually) on-line background material. Students then follow activities that encourage reflection, analysis and/or critique of their reading. Individual skills are also supported as follows. Learning outcomes are addressed in various modules, the principle ones are described here for each LO: ▪ Outcome 1,Software Engineering concept module which introduces different models of SE and encourages students to think more deeply and critically about these by requiring individual presentations to be posted to the group of students enrolled on the module. The presentations explore topics introduced in presentations produced by the tutor. ▪ Both Software Groupwork Development and Project Management address 2 via asynchronous conference collaborations and individual and group exercises. In particular, the role of management is considered together with the risks that might impede attainment of a successful product is examined in Project Management. ▪ 3 is considered in a number of modules (Software Engineering Concepts, Groupwork Software Development and Requirements Engineering). In particular, Requirements Engineering uses case-studies as a means of exploring the engineering of software requirements. ▪ 4 is principally addressed in the context of OO Design and Programming where students undertake a range of programming exercises. Exercises and, subsequently, model solutions are presented on-line. During published time-slots, students will be able to access ‘real-time’ support through the use of computer-mediated telephone support and instant messaging. ▪ 5 is considered in both Requirements Engineering and OO Design and Programming, in which students are encouraged to consider the use of CASE tools. ▪ Learning outcome 6 is considered throughout the programme by the use of examples and case-studies. ▪ Research Methods covers the tools and techniques of 7, and their evaluation. The dissertation is the specific medium through which the appropriateness of these tools is considered in relation to a particular problem (LO8)	Assessment is by a variety of means including examinations, software development exercises, . essay-type questions and group projects 7 , 8 are assessed by the research proposal and the production of a supervised dissertation.

C. Subject, Professional and Practical Skills

On successful completion of the programme students will have gained Subject/Professional/Practical Skills that enable them to:	***Teaching/Learning Methods and Strategies***	***Assessment***
1. Apply SE concepts throughout the lifecycle of a non-trivial software project (SE Concepts, Group Software Dev., Dissertation) 2. Use established methods to assess project risk (Project Management) 3. Use requirements engineering methods to develop software requirements specifications for software and computer-based systems (Requirements Engineering) 4. Use a range of software tools (throughout) 5. In depth knowledge of an OO programming language. (OO Design and Programming, Component based Software Development) 6. Development data-based using advanced technologies (Advanced databases) Prepare a research proposal in the subject area (Research Methods)	For the most part, subject specific skills reside in individual modules. The list given is the minimum that a graduate of MSc Software Engineering could be expected to possess. Additional specific practical skills are likely be gained through undertaking the dissertation, which requires the production and demonstration of a software artefact. These will vary with the dissertation topic. ▪ 1 through on-line conferencing and presentations, individual and group exercises and supervised research ▪ 2 through the staged solving of a demanding planning exercise ▪ 3 through directed reading and a practical exercise ▪ 4 through exercises that require the use of a range of tools e.g. CASE tools and IDEs ▪ 5 through programming experience. ▪ 6 through conferencing, reading and practical exercises. ▪ 7 through conferencing and investigative reading.	Outcomes are assessed by a variety of means including examinations, practical exercises, conference contributions and the research proposal.

D. Transferable Skills and Other Attributes

It is assumed that students enter this programme having already developed the transferable skills listed. The programme offers opportunities to develop these skills further, as described in the table below:

Transferable Skills and Other Attributes	***Teaching/Learning Methods and Strategies***	***Assessment***
1. Communication skills**:** to communicate principally in writing, including for instance, the results of technical investigations, to peers and/or to “problem owners”.	The emphasis on group work in some modules particularly ensures that 1 and 7 are met.	Written and practical assignments assess skills 1 – 4. The dissertation proposal and the dissertation itself assess all skills, but particularly skills 5 and 6. Conference collaborations and the Group Software Development module assesses skill 7.
2. Self-management skills: to manage one’s own time; to meet deadlines; to work with others having gained insights into the problems of team-based systems development.	Transferable skills 1- 6 are developed through the use of on-line conferencing, written assignments and individual research	
3. IT Skills in Context (to use software in the context of problem-solving investigations, and to interpret findings)	Acquisition of 3 is developed throughout the programme, both in terms of the mode of delivery, which requires the use of IT skills and the use of particular tools in particular modules, for example, the use of CASE tools and software development environments.	
4. Problem formulation: To express problems in appropriate notations.	Acquisition of 4 , 5 and 6 is through individual investigative, problem-solving and research tasks	
5. Progression to independent learning: To gain experience of, and to develop skills in, learning independently of structured class work. For example, to develop the ability to use on-line facilities to further self-study.		
6. Comprehension of professional literature: to read and to use literature sources appropriate to the discipline to support learning activities.		
7. Working with Others**:** to be able to work as a member of a team; to be aware of the benefits and problems which teamwork can bring.	Acquisition of 7 is encouraged throughout the programme by the use of student conferencing. It is particularly emphasised in the Group Software Development module.	

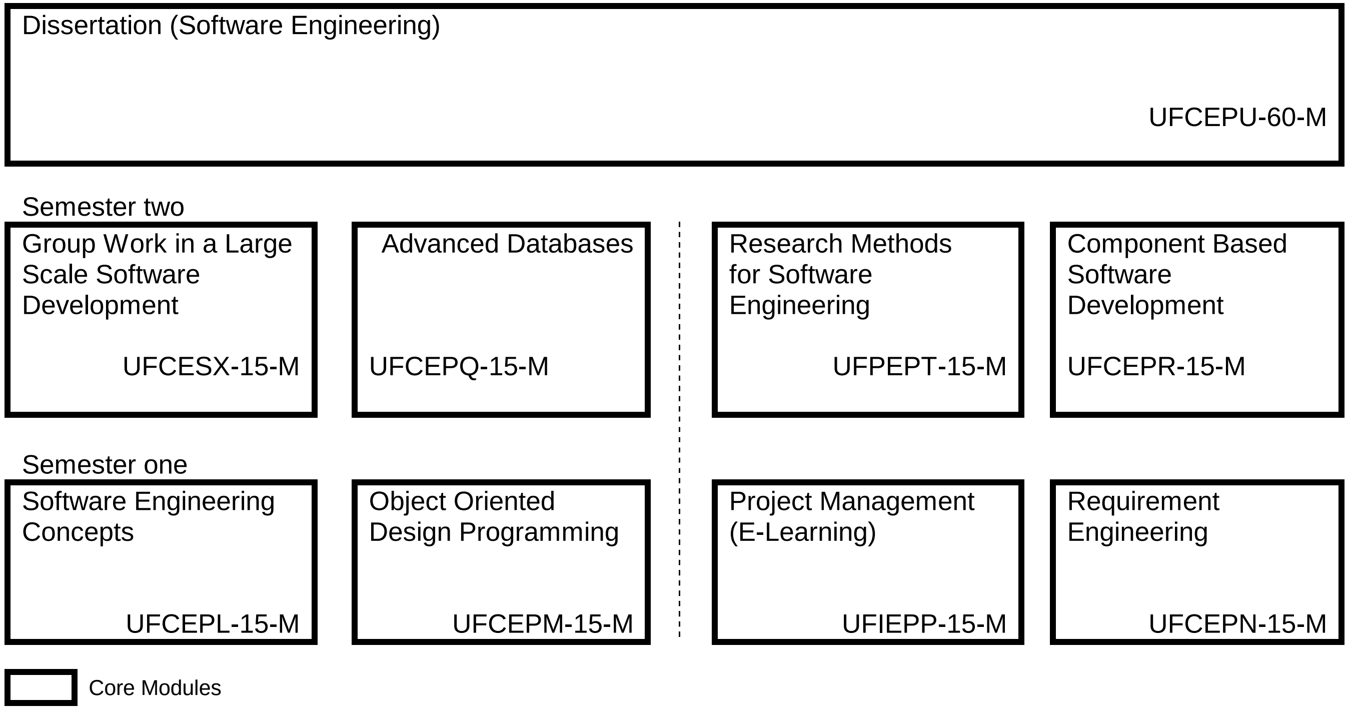
Section 4: Programme Structure

Programme Structure for **MSc Software Engineering** **For October 2004**

This structure is indicative and subject to change

PLEASE NOTE: REFER TO THE FACULTY ON-LINE INFORMATION SYSTEM FOR UP-TO-DATE STRUCTURE INFORMATION

http://www.cems.uwe.ac.uk/exist/index.xql



Dissertation (Software Engineering)
UFCEPU-60-M

Semester two

Group Work in a Large Scale Software Development
UFCESX-15-M

Advanced Databases
UFCEPQ-15-M

Research Methods for Software Engineering
UFPEPT-15-M

Component Based Software Development
UFCEPR-15-M

Semester one

Software Engineering Concepts
UFCEPL-15-M

Object Oriented Design Programming
UFCEPM-15-M

Project Management (E-Learning)
UFIEPP-15-M

Requirement Engineering
UFCEPN-15-M

▭ Core Modules

Part-time students in year one take modules to the left of the dotted line and in year two to the right of the dotted line

Section 5: Entry Requirements

The university's requirements for entry to a postgraduate programme apply. An honours degree in Computer Science, Software Engineering or a closely related discipline or extensive relevant professional experience is required. The University's general entry requirements are described in Section D of the Academic Regulations. A copy of the Academic Regulations is available from the University web site.

Section 6: Assessment Regulations

The University's Modular Assessment Regulations apply to this programme.

Section 7: Student Learning: Distinctive Features and Support

This programme is distinctive in that it is one of only very few full masters degrees in Software Engineering to be offered by distance learning by a UK university. The programme makes extensive use of UWE's virtual learning environment (Blackboard) to provide an integrated entry point to the modules on offer. For ease of use, all modules on the programme are offered with a common look and feel. Technical support is well-established and the reliability of both hardware and software well-proven. All modules have their space on Blackboard. For each module, this space will include all the resources needed to complete the module. A separate on-line space is provided for information relating to the programme as a whole.

Academic Support

A named module leader is responsible for each module on the programme and the module leader is responsible for providing academic support for his/her module. In addition, the programme is managed by a programme leader. The programme leader is also available to provide academic support across the modules.

Pastoral Care

The faculty offers pastoral care through its Student Advisers, a team of staff who provide a comprehensive, 9 - 5 (UK time) support service. This service can be accessed by email or telephone. Where appropriate, support can also be given via an instant messaging service. All students on the programme are allocated to the same Adviser, who is trained to provide advice on matters commonly of concern, including regulatory and other matters. The Adviser will, when necessary, advise the student to seek advice to from other professional services including the university's Centre for Student Affairs or from members of academic staff.

Progression to Independent Study

All of the modules on this programme require students to carry out independent study, such as research for projects and assignments. An extensive range of on-line journals and reference material is available to students of UWE. These materials are accessed through the University's library web-site.

The progression to independent study will also be assisted by the nature of the support offered in individual modules. Typically, module leaders will provide a plan for the module indicating the activities to be carried out and the forms of learning to be undertaken during the delivery of the module, with a view to encouraging students to plan ahead and to take responsibility for managing their time and resources.

Student Collaboration

One of the great potential benefits of E-learning is the enhanced possibilities for collaborative work. In classroom-based programmes it is comparatively rare for students to have legitimate access to the work of fellow students. In this programme students will post some of their work on conferences and examine the work of others. Some of this work, including the review of the work of others, will contribute to the assessment. Conference contributions will also be used for discussion of key topics

Section 8 Reference Points/Benchmarks

In designing this programme, the faculty has drawn upon the following external reference points:

1. The QAA Framework for Higher Education Qualifications in England, Wales and Northern Ireland
2. The QAA Benchmark Statement for Computing
3. UWE's Learning & Teaching Strategy

The QAA Framework for Higher Education Qualifications in England, Wales and Northern Ireland describes the attributes and skills expected of Masters graduates. It is our view that the learning outcomes of this programme are fully consistent with the qualification descriptor in the Framework, and hence that graduates will be able to demonstrate that they meet the expectations of the Framework. For example, students will be introduced to contemporary research in Requirements Engineering and be required to analyse and critique that work throughout their period of study; Students will develop "a comprehensive understanding of techniques applicable to their own research or advanced scholarship", both through the study of the Research Methods module, though the development of their dissertation and elsewhere.; students will "deal with complex issues both systematically and creatively, make sound judgements in the absence of complete data, and communicate their conclusions clearly to specialist and non-specialist audiences" especially in the context of the group software development module.

The **QAA Subject Benchmark Statement for Computing**

The QAA Subject Benchmark Statement Computing applies most closely to this programme Although the published benchmark statement refers to Bachelor's Honours degrees rather than to an MSc programme, there is a proposed benchmark standard for taught masters in computing. The proposed standard is built on the Bachelor's Honours statement and has been used to inform the development of this programme.

The defining principles of the study of Computing according to the draft benchmark are:

- Knowledge and understanding of aspects of computer systems and their use;
- A combination of theory and practice, with practice being guided by theoretical considerations
- A strong emphasis on applications with usability being important
- The use of specification, design, implementation and maintenance being important features which characterise the lifecycle and contribute fundamentally to the concept of process
- An understanding of and attention to the many and varied aspects of quality and
- An understanding of professional, social and ethical issues related to Computing.

The design of this award has followed these principles. In particular, the proposed statement suggests that, irrespective of the discipline from which a student enters the programme, a masters programme must *"add value* to undergraduate degrees" and "be characterised by an ethos of advanced work and scholarship".

Immediately upon entry to this programme, students will be required to make use of knowledge and skills appropriate to an honours graduate in computing. They will be required to use skills of software development and design to analyse and extend problems presented in the first semester; they will use critical evaluation to discuss journal papers at an advanced level; they will use their understanding of key aspects of computing to extend their knowledge in new directions. Ultimately, they will also be required to demonstrate original thinking in the development of their level M dissertation.

UWE's Learning & Teaching Strategy

This initiative chimes well with UWE's stated Learning & Teaching Strategy as articulated at : http://www.uwe.ac.uk/info/landt/

In its strategic objectives, the university has stated its commitment to:

- Open-minded and systematic enquiry into their (the staff's) practice and its effects;
- A student-centred approach, that encourages students to take responsibility for aspects of their learning, and teachers to take responsibility for facilitating that learning;
- Enthusiasm for teaching supported by continual development of their subject through an appropriate combination of research, professional practice, consultancy and engagement with the national Subject Centres;
- Curriculum development and the critical appraisal of curriculum content and delivery as a central aspect of the teaching role;
- The expansion of opportunity and removal of unnecessary barriers to learning;
- The development of student skills for research and inquiry as a major part of student learning;
- The development of skills that will equip students as lifelong learners.

The nature of this programme has required the staff involved to reflect extensively on their teaching practice. The section in on Staff Development provides more detail of the way that this reflection has been encouraged. The remote delivery of the programme directly contributes to an expansion of opportunities for learning.

Part 2: Module Specifications

(See Appendix A)