PROGRAMME SPECIFICATION

<table>
<thead>
<tr>
<th>Awarding body/institution:</th>
<th>Queen Mary, University of London</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching institution (if different from above):</td>
<td>Queen Mary, University of London</td>
</tr>
<tr>
<td>Name of the final award and Programme title:</td>
<td>MSc Software Engineering</td>
</tr>
<tr>
<td>Duration of Study/Period of Registration</td>
<td>1 year Full-Time 2 years Part-Time</td>
</tr>
<tr>
<td>UCAS code:</td>
<td>G4S4</td>
</tr>
<tr>
<td>QAA Benchmark Group</td>
<td>Computer Science</td>
</tr>
<tr>
<td>Academic Department/s involved in programme delivery</td>
<td>School of Electronic Engineering &amp; Computer Science</td>
</tr>
</tbody>
</table>

If accredited by a professional/statutory body, please give the name, date of last accreditation visit, approximate date of next visit and details of exemptions that will be given to QMUL graduates. Accredited by the British Computer Society in 2010. CITP Further Learning (Exemption) and CEng*/CSci (Partial Fulfilment) *conditional on taking module AMCM055 Software Risk Assessment

Criteria for admission to the programme

Candidates should normally possess (or shortly expect to obtain) a good Honours degree (first or upper-second class honours) with a substantial computer science component (at least half) or equivalent industrial experience. Candidates should also have good programming skills for undertaking the practical elements of the course.
### Aims of the programme

The programme aims to teach architecture alternatives for software design (patterns of software design and component technologies) and for information handling (structured information, databases), the key issues of interactive system design, leading to the ability to identify issues and trade-offs in the design of interaction and to be able to invent and evaluate alternative solutions to design problems. Graduates should have an understanding of the mathematical foundations of software and the practical application of these techniques as well as management of software project risks and trade-offs between different quality attributes. They will also have gained essential transferable skills for team work and research.

### Learning outcomes for the programme

Students completing the course will develop some of the following capabilities: design complex software systems, with informed selection between alternative architectures understand mathematical foundations of software and apply advanced tools to demonstrate properties of systems or software design for usability and assess usability, using models of user, interfaces and dialogue understand and apply risk management principles to software projects demonstrate skills in scholarship, communications (written, verbal), planning and research relevant to both industrial and academic practice.

### Teaching, learning and assessment strategies

Modules are taught through a combination of lectures, seminars, coursework and design-based laboratories. Teaching is interactive and student participation is encouraged. Lectures and seminars are used to introduce principles, methods and techniques. Courseworks develop the students' own skills in research, analysis, design, implementation and validation. Design-based laboratories provide students with the guidance and help required to achieve this.

Individual projects are undertaken during the summer months under the supervision of an academic member of staff with whom there are weekly consultancy meetings. These are used for students to report on their progress, discuss research and design issues and plan their future work. This develops and re-inforces students' ability to communicate technical ideas clearly and effectively.

The Research Methods modules are assessed on the basis of written coursework alone, including the literature review and project specification. All the other taught modules are assessed through a combination of written examination and practical coursework. The project is examined on the basis of a written report and oral presentation.
Programme structure(s) and requirements, levels and modules

Students must take 8 taught modules (15 credits Level 7 each) and the dissertation (60 credits Level 7). The dissertation is normally undertaken during the summer months.

**Semester A**
Core modules:
- AMCM056 Research Methods I
- AMCM048 Advanced Program Design

Two of:
- AMCM058 Advanced Database Systems & Technology
- AMCM333 Algorithms & Complexity
- AMCM054 Distributed Systems and Security
- AMCM042 Foundations for Information Retrieval
- AMCM043 Structured Documents and XML

**Semester B**
Core module:
- AMCM057 Research Methods II

At least one of:
- AMCM049 Specification and Verification
- AMCM055 Software Risk Assessment

Further options:
- ELEM006 Multimedia Systems
- AMCM052 Entrepreneurship in Information Technology
- AMCM051 The Semantic Web
- AMCM016 Interactive Systems Design
- AMCM060 Computability
- AMCM061 Computational Genomics
- AMCM306 High Performance Computing
- AMCM069 Advanced Study Project

Core module:
- AMCM025 Dissertation

Notes:
- Options in semesters 1 and 2 ensure students have flexibility to tailor the course to their interests and also allow students to avoid courses that they have already covered in previous degree courses.
- If AMCM042 Foundations of Information Retrieval is taken, AMCM043 Structured Documents and XML must also be taken.
- The dissertation will be undertaken in a suitable area of software engineering and supervised by an experienced member of academic staff.
## Quality assurance mechanism
(please include details of: SSLC meetings, student feedback mechanisms, personal tutor arrangements, programme induction, programme review and monitoring.)

1. Programme induction on core Research Methods I module, with small group meetings.
2. Covered by the School of Electronic Engineering & Computer Science SSLC; graduate student representative
3. Students feedback questionnaires for taught courses and peer review.
4. Personal supervisor for project.

## Employers Links
Please provide details of any links with employers e.g.
- Details of advisory panels that include current or potential employers;
- Organisations that regularly employ graduates from this programme and the roles that graduates undertake.
- Student prizes donated by organisations that may offer employment to graduates from this programme.

If there are no links with employers consider the learning outcomes and transferable skills and explain how these might be used to inform employers about the qualities and skills a graduate from this programme might be expected to have.

The School of Electronic Engineering & Computer Science has a wide range of industrial contacts secured through research projects and consultancy, our Industrial Experience programme and our Industry Panel.

The Industry Panel works to ensure that our courses are state of the art and match the changing requirements of this fast moving industry. The Panel includes representatives from a variety of Computer Science oriented companies ranging from SMEs to major blue-chips. These include: Microsoft Research, Royal Bank of Scotland, BT Labs, Oaklodge Consultancy, Intel Research, The Usability Company, Hewlett Packard Labs and Arclight Media Technology Limited.

Recent graduates have found employment as programmers, Systems Analysts, Software Engineers, database developers, IT consultants and web developers with well known multinational companies throughout the UK and Europe, the Americas and Asia.

Merril Lynch, Microsoft, Nokia, Barclays Capital, Logica, JPMorgan and Bear Sterns are among the organizations that have recently employed graduates of EECS programs.
Knowledge and skills relevant to employment

This program provides as core an advanced unit in Java programming designed to build on the programming experienced of students in their first degree. In addition to the Java language, the course unit examines the use and reuse of patterns in Object Oriented development, and emphasizes the modular approach essential to working professionally as a member of a software development team. Further core units provide the techniques and tools for developing complex, modular software systems, the importance of software reliability in system development and the assessment and management of risk associated with large scale software projects. Other optional course units enable the further development of database systems knowledge from a first degree, the development of XML, XSLT and an understanding of the Semantic Web and the technologies underpinning Web 2.0.

Career Opportunities

Graduates of this program will be particularly well suited to joining organisations involved in the development of large scale software projects, either as software, database or web developers and/or involved in the assessment and management of project risk. They might equally play a role in the development of ecommerce systems and the management of projects involving the interconnection of a large number of software modules and database system components.