Confirmation that this version of the module specification has been approved by the School Learning and Teaching Committee:

19 December 2014………………………………………………….(date)

**MODULE SPECIFICATION**

1. **Title of the module**

CO661 Theory and Practice of Concurrency.

1. **School or partner institution which will be responsible for management of the module**

School of Computing

1. **Start date of the module**

Academic year 2015/16

1. **The number of students expected to take the module**

30-40 students

1. **Modules to be withdrawn on the introduction of this proposed module and consultation with other relevant Schools and Faculties regarding the withdrawal**

None

1. **The level of the module (e.g. Certificate [C], Intermediate [I], Honours [H] or Postgraduate [M])**

H

1. **The number of credits and the ECTS value which the module represents**

15 credits (7.5 ECTS)

1. **Which term(s) the module is to be taught in (or other teaching pattern)**

Autumn or Spring

1. **Prerequisite and co-requisite modules**

Pre-requisites:

CO519 Theory of Computing

CO539 Web Development

CO545 Functional and Concurrent Programming

1. **The programmes of study to which the module contributes**

Computer Science (BSc)

1. **The intended subject specific learning outcomes**

Upon successful completion of the module students will:

* 1. Have a critical understanding of the principles of concurrent programming, as well as its advantages and challenges;
  2. Be able to reason on the properties of a distributed process (e.g., safety and liveness), and compare the behaviour of different processes.
  3. Be able to design and implement processes satisfying given properties.
  4. Be able to apply the acquired knowledge to real scenarios e.g. application-level protocols, Web services.
  5. Be familiar with advanced concepts of Web Services.

1. **The intended generic learning outcomes**

Upon successful completion of the module students will have developed the   
 following skills:

* 1. Systematic and rigorous reasoning,
  2. Application of abstract concepts to concrete scenarios,
  3. Ability of presenting and discussing state of the art topics.

1. **A synopsis of the curriculum**

This module is aimed at introducing the principles of concurrency theory (1,2,3) and demonstrating how these can be applied to design and implement distributed applications (4). Advanced concepts of Web services will be studied and placed in the perspective of these principles (5,6).

1. Message passing primitives for concurrency: synchronous versus asynchronous message passing, the actor model.
2. Reasoning on processes: temporal logic, safety and liveness properties, bisimulation.
3. Channel passing and mobility.
4. Design and implementation of application–level protocols.
5. Web services: from stateless services to distributed business processes (also known as service orchestrations).
6. Transaction protocols on the Web: two-phase commit, long running transactions.
7. **Indicative Reading List**

*Milner, R. 1989 Communication and Concurrency, Prentice-Hall,*

*Armstrong, J, Virding, R, Williams,W.M, Wilkstrom, 1996 C. Concurrent Programming in ERLANG, Prentice-Hall,*

*Lynch, N.A. 1996 Distributed Algorithms. Morgan Kaufmann Publishers In (Section 7)*

1. **Learning and Teaching Methods, including the nature and number of contact hours and the total study hours which will be expected of students, and how these relate to achievement of the intended module learning outcomes**

22 hours of lectures (learning outcomes: 11.1, 11.2, 11.5, 12.1, 12.2)

8 hours of supervised terminal classes (learning outcomes: 11.2, 11.3, 12.1, 12.2)

10 hours of student’s seminars (learning outcomes: 11.4, 12.3)

60 hours on assessed coursework (learning outcomes: 11.2, 11.3, 11.4, 12.1, 12.2)

50 hours of study and revision (learning outcomes: 11.1, 11.2, 11.5, 12.1)

150 hours total

1. **Assessment methods and how these relate to testing achievement of the intended module learning outcomes**

This module will be assessed by 50% examination, 50% coursework.

* 2-hour written examination (50%): testing learning outcomes (11.1, 11.2, 11.5, 12.1).
* 1 practical assessment (20%): testing learning outcomes (11.2, 12.1).
* 1 practical assessment (20%): testing learning outcomes (11.3, 12.2).
* Seminar discussions (10%): testing learning outcomes (11.4, 11.5, 12.3).

1. **Implications for learning resources, including staff, library, IT and space**

The module will require labs for the terminal sessions held in 8 weeks of the term. The terminal sessions will require installation of open software and can be staffed by the core lecturing team and sessional teachers.

1. **The School recognises and has embedded the expectations of current disability equality legislation, and supports students with a declared disability or special educational need in its teaching. Within this module we will make reasonable adjustments wherever necessary, including additional or substitute materials, teaching modes or assessment methods for students who have declared and discussed their learning support needs. Arrangements for students with declared disabilities will be made on an individual basis, in consultation with the University’s disability/dyslexia support service, and specialist support will be provided where needed.**
2. **Campus(es) or Centre(s) where module will be delivered:**

Canterbury