SIMON FRASER UNIVERSITY
Senate Committee for Undergraduate Studies
NEW COURSE PROPOSAL

Course Number: CMPT 473-3

Course Title: Software Quality Assurance
Short Course Title: Software Quality Assurance

Course vector: 3 lecture

Course Description (for Calendar). Attach a course outline to this proposal.

Factors in software quality include functionality, reliability, usability, efficiency, maintainability, and portability. Techniques for assessing the quality of software with respect to such factors, and methods for improving the quality of both software products and software-development processes.

Prerequisite: CMPT 373.

Corequisite: none

Course(s) to be dropped if this course is approved: none

Rationale for Introduction of this Course:

This course will be required in the proposed Software Systems program for the Surrey campus.

Scheduling and Registration Information:

Indicate effective **semester/year** course would be first offered and planned **frequency** of offering thereafter.
Fall 2009, initially offered twice annually in Surrey

Waiver required: no

Will this be a required or elective course in the curriculum?
Required in the Software Systems program.

What is the probable enrolment when offered?

40 students.

Which of your present CFL faculty have the expertise to offer this course?

Toby Donaldson, John Edgar, Anne Lavergne, Uwe Glaesser, Dirk Beyer, Rob Cameron

Are there any proposed student fees associated with this course other than tuition fees? (if so, attach mandatory supplementary fee approval form)

no

Resource Implications:

Note: Senate has approved (S.93-11) that no new course should be approved by Senate until funding has been committed for necessary library materials. Each new course proposal must be accompanied by a library report and, if appropriate, confirmation that funding arrangements have been addressed.

Campus where course will be taught:

Surrey.

Library report status _____

Provide details on how existing instructional resources will be redistributed to accommodate this new course. For instance, will another course be eliminated or will the frequency of offering of other courses be reduced; are there changes in pedagogical style or class sizes that allow for this additional course offering?

See attached Software Systems Curriculum document.

Any outstanding resource issues to be addressed prior to implementation: space, laboratory equipment, etc.

See attached Software Systems Curriculum document.

Approvals

1. **Departmental approval** indicates that the Department has approved the content of the course, and has consulted with other Departments and Faculties regarding proposed course content and overlap issues.

   _____
   Chair, Dept./School Date

   _____
   Chair, Faculty Curriculum Committee Date

2. **Faculty approval** indicates that all the necessary course content and overlap concerns have been resolved, and that the Faculty/Department commits to providing the required Library funds.

_____ Date: _____
Dean or Designate

List which other Departments and Faculties have been consulted regarding the proposed course content including overlap issues. *Attach documentary evidence of responses.*

_____
_____
_____
_____

Other Faculties approval indicates that the Dean(s) or designate of other Faculties affected by the proposed new course support(s) the approval of the new course.

_____ Date: _____

_____ Date: _____

3. SCUS approval indicates that the course has been approved for implementation subject, where appropriate, to financial issues being addressed.

Course approved by SCUS (Chair of SCUS)

_____ Date: _____

Approval is signified by date and appropriate signature.

Proposed CMPT 473 Course Outline

Factors in software quality include functionality, reliability, usability, efficiency, maintainability, and portability. This course examines techniques for assessing the quality of software with respect to such factors, and methods for improving the quality of both software products and software-development processes. Hands-on experience will familiarize students with state-of-the-art methods and tools.

Topics:
- Introduction to software quality
- Quality management systems (ISO 9000, ISO 9126), certification
- Quality factors (functionality, reliability, usability, efficiency, maintainability, portability)
- Metrics for quantifying factors of software quality
- Design validation: software component metrics, structure analysis
- Introduction to software verification and static program analysis
- System testing: recovery, security, stress, and performance testing
- Component testing: unit and integration testing, validation testing, debugging, instrumentation, profiling
- Methods to derive test cases: black box, white box, class-based, automatic
- Bug-tracking systems, change management

Grading:
Assignments 75%, final exam 25%

Textbook:
Metrics and Models in Software Quality Engineering (2nd Edition), Stephen H. Kan, Addison-Wesley, 2003