



Concordia University

Concordia Institute for Information Systems Engineering

INSE 6250/4/UU - Quality Methodologies for Software (Winter 2007)

Time: Tuesday, 20h30 – 23h00
Classroom: H-400
Office hours: Wednesday, 14h00 – 16h00 or by appointment (Office: EV.7.630)

Instructor: Dr. J. Bentahar
E-mail: bentahar@ciise.concordia.ca
Tel: 514-848-2424 Ext. 5382

Course Web: <http://www.ciise.concordia.ca/~bentahar/inse6250.html>
You can use this web site to get lecture notes, useful links, assignments, and other useful information. It is highly advised to visit the web site regularly.

Textbooks: 1) *Quality Software Project Management*
Robert T. Futrell, Donald F. Shafer, Linda I. Shafer
Prentice Hall, 2002
ISBN: 0-13-091297-2

Suggested Readings This book covers several issues related to software quality. Some important chapters are: process overview, selecting software developments life cycles, developing the software requirements specification, introduction to software engineering, software metrics, and validation and verification.

2) *The B-Book: Assigning Programs to Meanings*
J. R. Abrial
Cambridge University Press, 1996
ISBN: 0-521-49619-5

Suggested Readings This book is a reference in the B method for software specification. It covers a comprehensive breadth of this method and its applications.

3) *Understanding Formal Methods*
Jean Francois Monin
Springer, 2003
ISBN: 1-85233-247-6

Suggested Readings This book presents an overview of formal methods for software development. A presentation of logical tools and set-theoretic specifications (Z, VDM, and B) is included.

Suggested Readings 4) *Principles of Model Checking*
J-P. Katoen
Formal Methods and Tools Group, University of Twente
Available online from the course web page

Description: This course presents the main quality methodologies used within software engineering process. It introduces software engineering and the main quality issues for software. The following issues will be covered: software engineering, quality methodologies, design for six sigma (DFSS) for Software, software metrics, meta-models, software modeling, specification languages, software verification, etc. Students will discover various concepts and techniques developed in recent research about software engineering and quality methodologies and learn to apply them through lectures, readings, assignment, and team project. Several materials from different sources will be used, particularly scientific papers.

Prerequisites: INSE 6210, COMP 5541 or equivalent.
Good knowledge in mathematics will be helpful.

Requirements:

- One group assignment
- One in-class final exam (closed book)
- One team project (2~3 members, presentation + report)

Grading:

- One assignment: 23%
- Final exam: 45%
- Project (presentation + report): 16% + 16% = 32%

The assignment will be graded based on **clarity** and **comprehensiveness**. The project will be graded based on **originality**, **relevance of the proposed solution**, and **contribution**. In-class exam will test students' knowledge and ability to understand, analyze, and synthesize concepts.

Important dates:

- Project proposal: February 13, 2007
- Assignment: February 13, 2007
- Final exam: March 27, 2007
- Project presentation: April 10, 2007
- Project report: April 10, 2007

Useful Links:	The Software Quality Page http://www.swquality.com/users/pustaver/index.shtml
	Software QA and Testing Resource Center http://www.softwareqatest.com/
	The B-Method http://www.b-core.com/ONLINEDOC/BMethod.html
	Formal Methods http://vl.fmnet.info/
	Software Quality Institute (SQI) http://lifelong.engr.utexas.edu/sqi/index.cfm

Submission:	All assignments are at the beginning of class. Late assignments will incur a penalty of 20% deduction (up to 100%). No points will be given to the assignment submitted 5 days after the due date.
--------------------	--

Policies:	Cheating and plagiarisms will be very seriously considered and handled according to the Concordia Academic Code of Conduct (can be found in the graduate student handbook) without exception. Please note the schedule of the exams. A makeup test will be given only in the case of a serious illness or emergency. You must contact the instructor before the exam. Only written and proved documentations are accepted for verification purposes.
------------------	--

Tentative Schedule: The table below provides a brief summary of some of the material that will be covered during the term. The schedule may change slightly.

Lecture	Topic	Event
1	Introduction to Software Process and Software Engineering	
2	Introduction to Quality Methodologies for Software	
3	DFSS for Software Part I (define overview, project context, initial analysis and design, ...)	
4	DFSS for Software Part II (multi generation planning, project management, risk analysis, software metrics, ...)	
5	Introduction to Formal Methods for Software	
6	The Z-Language	Project Proposal + Assignment
7	The B-Method	
8	VDM and SDL Languages	
9	Formal Verification (Logical Specification)	
10	Model Checking and Binary Decision Diagrams.	
11	Software Modeling	Final Exam
12	Failure Modes, Effects Analysis, and Defensive Programming	
13	Project	Project