

Teaching Quality of Software at FCUL

Ana Paula Afonso

Universidade de Lisboa

TEQUA'07 Workshop

Structure of Presentation

- Context
- **Which topics** should be taught
- **When** Software Quality is taught
- **How** Software Quality is taught
- Conclusions

Context

- SQE teacher not a SQE researcher
 - Information Systems Education Area in Departamento Informática at FCUL
- Evolution of this course
 - PDS – Software Development Processes (00/01, 01/02, 02/03)
 - PQSI – Quality Processes in Computing Systems (03/04, 04/05, 05/06)
 - QS – Quality of Software (06/07)

What can a teacher use to teach Software Quality Engineering (SQE)?

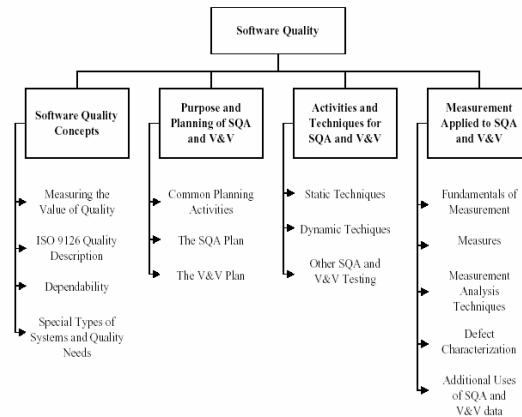
- Support on crucial levels: theory, practice, tools and experience
- **Theory Level**
 - **SWEBOK** - Guide to the Software Engineering Body of Knowledge, IEEE Computer Society
 - **SE2004** - Software Engineering 2004
formerly known as Computing Curriculum Software Engineering (CCSE)
IEEE Computer Society, ACM
 -

What can a teacher use to teach SQE?

- Main contribution of SWEBOK
 - Create an internationally guide of SWEBOK
- For SQE education SWEBOK offers:
 - Recognize Software Quality as a separate Knowledge area (KA)
 - Recommend Software Quality KA contents and breakdown topics
 - Recommend references, standards and further reading

What can a teacher use to teach SQE? SWEBOK

- Software Quality Knowledge area breakdown of topics



What can a teacher use to teach SQE? SE2004

- **Main contributions of SE2004**
 - provides recommendations about what should constitute an undergraduate software engineering education.
 - **What** every SE graduate must **know** - SEEK (Software Engineering Education Knowledge).
 - **Ways** that this **knowledge** and the **skills** fundamental to software engineering can be **taught** in various contexts - Curriculum

What can a teacher use to teach SQE? SE2004 – Software Quality KA

Units and Topics

| Reference | | k,c,a | E,D,O | Hours | Related Topics |
|-----------|---|-------|-------|-------|-----------------------------------|
| QUA | Software Quality | | | 16 | |
| QUA.cc | <i>Software quality concepts and culture</i> | | | 2 | |
| QUA.cc.1 | Definitions of quality | k | E | | |
| QUA.cc.2 | Society's concern for quality | k | E | | |
| QUA.cc.3 | The costs and impacts of bad quality | k | E | | |
| QUA.cc.4 | A cost of quality model | c | E | | MGT.pp.4 |
| QUA.cc.5 | Quality attributes for software (e.g. dependability, usability, etc.) | k | E | | MAA.rva.5,VAV.t st.9,QUA.pda.5 |
| QUA.cc.6 | The dimensions of quality engineering | k | E | | |
| QUA.cc.7 | Roles of people, processes, methods, tools, and technology | k | E | | |
| QUA.std | <i>Software quality standards</i> | | | 2 | PRF.pr.5 |
| QUA.std.1 | The ISO 9000 Quality Management Systems | k | E | | |
| QUA.std.2 | ISO/IEEE Standard 12207 Software Life Cycle Processes | k | E | | |
| QUA.std.3 | Organizational implementation of standards | k | E | | |
| QUA.std.4 | IEEE software quality-related standards | | D | | |

What can a teacher use to teach SQE? SE2004 - Software Quality KA

| | | | | | |
|-----------|--|---|---|---|---------------------|
| QUA.std | <i>Software quality standards</i> | | | 2 | PRF.pr.5 |
| QUA.std.1 | The ISO 9000 Quality Management Systems | k | E | | |
| QUA.std.2 | ISO/IEEE Standard 12207 Software Life Cycle Processes | k | E | | |
| QUA.std.3 | Organizational implementation of standards | k | E | | |
| QUA.std.4 | IEEE software quality-related standards | | D | | |
| QUA.pro | <i>Software quality processes</i> | | | 4 | |
| QUA.pro.1 | Software quality models and metrics | c | E | | VAV.fnd.4,QUA.pda.5 |
| QUA.pro.2 | Quality-related aspects of software process models | k | E | | PRO.imp.3 |
| QUA.pro.3 | Introduction/overview of ISO 15504 and the SEI CMMs | k | E | | PRF.pr.5 |
| QUA.pro.4 | Quality-related process areas of ISO 15504 | k | E | | PRF.pr.5 |
| QUA.pro.5 | Quality-related process areas of the SW-CMM and the CMMIs | k | E | | |
| QUA.pro.6 | The Baldrige Award criteria as applied to software engineering | | O | | |
| QUA.pro.7 | Quality aspects of other process models | | O | | |
| QUA.pca | <i>Process assurance</i> | | | 4 | |

What can a teacher use to teach SQE? SE2004 - Software Quality KA

| | | | | | |
|-----------|---|---|---|---|-------------------------------|
| QUA.pca | <i>Process assurance</i> | | | 4 | |
| QUA.pca.1 | The nature of process assurance | k | E | | |
| QUA.pca.2 | Quality planning | a | E | | MGT.pp |
| QUA.pca.3 | Organizing and reporting for process assurance | a | E | | |
| QUA.pda.4 | Techniques of process assurance | c | E | | |
| QUA.pda | <i>Product assurance</i> | | | 4 | |
| QUA.pda.1 | The nature of product assurance | k | E | | |
| QUA.pda.2 | Distinctions between assurance and V&V | k | E | | VAV |
| QUA.pda.3 | Quality product models | k | E | | |
| QUA.pda.4 | Root cause analysis and defect prevention | c | E | | PRO.con.6 |
| QUA.pda.5 | Quality product metrics and measurement | c | E | | VAV.fnd.4,QUA.c.c.5,QUA.pro.1 |
| QUA.pda.6 | Assessment of product quality attributes (e.g. useability, reliability, availability, etc.) | c | E | | |

What can a teacher use to teach SQE?

- **Practice Level**
 - Quality models: McCall, Boehm, ISO/IEC 9126
 - Software Product Quality Measurement and evaluation standards
 - Software Product Quality support processes standards
 - Software Measurement standards
 - Life Cycle Processes standards
 -
- **Tools** - some
- **Experience** – usually partially published, often not shared, kept as notes of courses materials

Structure of Presentation

- Context
- Which topics should be taught
- **When Software Quality is taught**
- How Software Quality is taught
- Conclusions

When Software Quality is taught

- Undergraduate course in Informatics Engineering
 - Object Oriented Development (2 year)
 - Software Engineering (3 year)
- Graduate Programme Informatics Engineering
 - Specialization areas
 - Software Engineering
 - Information Systems



Structure of Presentation

- Context
- Which topics should be taught
- When Software Quality is taught
- **How Software Quality is taught**
- Conclusions

Theory classes - What topics

- The software quality challenge, Software quality definition
- Software quality factors: McCall and ISO/IEC 9126 models
- Components of a software quality assurance system
 - Pre-project components: development and quality plans
 - Software project life cycle components
 - Integration of quality activities
 - Reviews
 - Software testing
 - Infrastructure components
 - Configuration management
 - Documentation control
 - Management components
 - Project progress control
 - Risk management
 - Software quality metrics
 -
 - SQA Standards, system certification components
 - SEI CMM, ISO 9001, 9000-3
 - IEEE 1012, ISO/IEC 12207
 - Human component

Theory classes - What topics

PART - I

- The software quality challenge, Software quality definition
- Software quality factors: McCall and ISO/IEC 9126 models
- Components of a software quality assurance system
 - Pre-project components: development and quality plans
 - Software project life cycle components
 - Integration of quality activities
 - Reviews
 - Software testing
 - Infrastructure components
 - Configuration management
 - Documentation control
 - Management components
 - Project progress control
 - Risk management
 - Software quality metrics
 -
 - SQA Standards, system certification components
 - SEI CMM, ISO 9001, 9000-3
 - IEEE 1012, ISO/IEC 12207
 - Human component

PART - II

- Case Studies: Exposure to real situations
- Three seminars with invited speakers from the industry.

Practice classes - What topics

- The Personal Software Process (Humphrey)
 - The software engineer's job
 - Time management
 - Tracking time
 - Product and Period Planning
 - Product size
 - Managing your time
 -

How they are taught Theoric and Praticce Classes

- Each theoric classe have three parts:
 - Reading:
 - Self-study materials that students work
 - Exposure and discussion:
 - The topic is presented with more detail and discussed with students
 - Evaluation
 - The student elaborate a resume and/or answer a question about the topic learned
- **Praticce classes:** Each student group presents a part of the PSP

Evaluation Rules

- Questions and resumes (10%)
- PSP exercises (10%)
- Project (40%)
- Final Exam (40%)

Bibliography

▪ Fundamental Books

- Daniel Galin, Software Quality Assurance, Addison Wesley, 2004
- Watts Humphrey, Introduction to the Personal Software Process, Addison Wesley, 1997

▪ Complementary Books

- Steve McConnell, Software Project Survival Guide, Microsoft Press, 1998
- Sami Zahran, Software Process Improvement: Practical Guidelines for Business Success, Addison Wesley, 1998
- Stephen H. Kan, Metrics and Models in Software Quality Engineering, Addison Wesley, 1995
- A. Goldberg e K. Rubin, Succeeding with Objects: Decision Frameworks for Project Management, Addison Wesley, 1995
- Terry Winograd, Bringing Design to Software: Expanding Software Development to Include Design, Addison Wesley, 1996
- Mark Paulk, Charles Weber et al, The Capability Maturity Model: Guidelines for Improving the Software Process, Addison Wesley, 1995

Some conclusions

- It is fundamental to incorporate topics about Software quality in the Computing curriculum
- The market are asking for students with those skills
- Teaching SQE it still a difficult role, however not without support
- It is necessary to shared teaching experiences
- The maturation process of SQE probably evolves a Body of Knowledge in Software Quality Engineering
- ? What ways can we promote early on students a sensibility for quality and a culture to quality?



Questions and Sugestions?

Ana Paula Afonso
apa@di.fc.ul.pt

*Departamento de Informática
Faculdade de Ciências, Universidade de Lisboa*