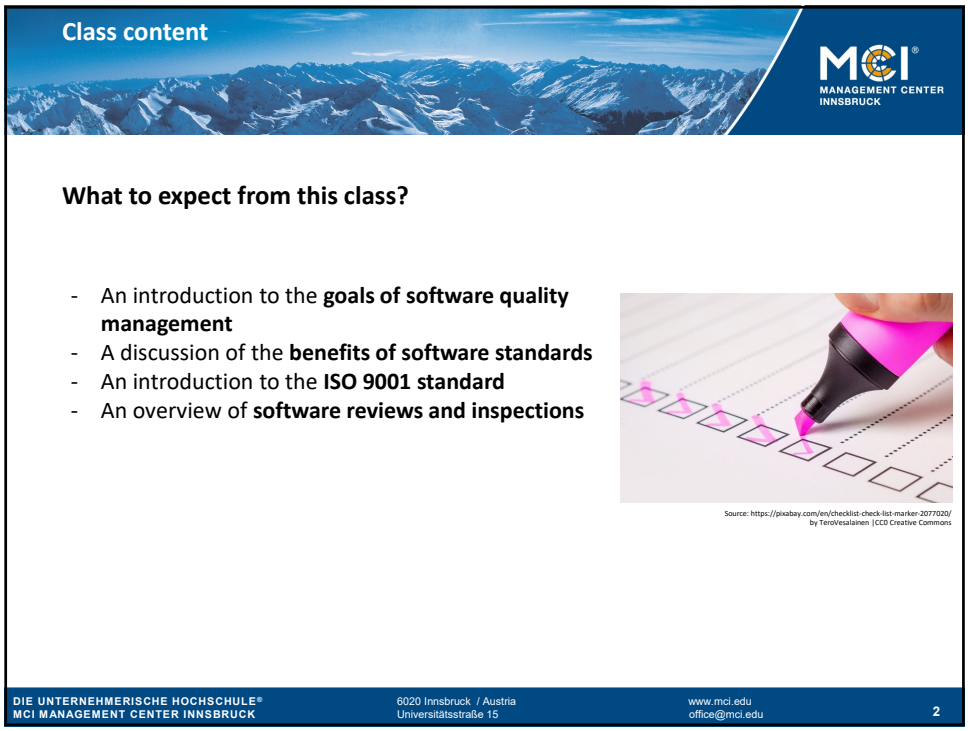

MCI
MANAGEMENT CENTER
INNSBRUCK
SYSTEMS PLANNING
Software Quality Management
DIE UNTERNEHMERISCHE HOCHSCHULE®
MCI MANAGEMENT CENTER INNSBRUCK
6020 Innsbruck / Austria
Universitätsstraße 15
www.mci.edu
office@mci.edu
1

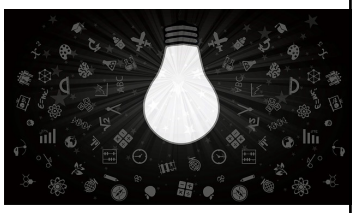

Class content
MCI
MANAGEMENT CENTER
INNSBRUCK
What to expect from this class?
- An introduction to the goals of software quality management
- A discussion of the benefits of software standards
- An introduction to the ISO 9001 standard
- An overview of software reviews and inspections
Source: https://pixabay.com/en/checklist-check-list-marker-2077020/
by TeroVesalainen |CCO Creative Commons
DIE UNTERNEHMERISCHE HOCHSCHULE®
MCI MANAGEMENT CENTER INNSBRUCK
6020 Innsbruck / Austria
Universitätsstraße 15
www.mci.edu
office@mci.edu
2

Class learning goals

MCI MANAGEMENT CENTER INNSBRUCK

What should be your learning outcome?

- Understand the **goal of software quality management**
- Know the **benefits of standards**
- Be familiar with the **principles of the ISO 9001 standard**
- Know **the goal of software reviews and inspections**



Source: https://pixabay.com/en/learning-hint-school-subject-3245792/ by harishs |CC0 Creative Commons

Software Quality Management

MCI MANAGEMENT CENTER INNSBRUCK

Software Quality

- Problems with software quality were **initially discovered in the 1960s**
- Software was **slow, unreliable, difficult to maintain, and hard to reuse**
- **New techniques** and **better software** testing led to improvements

Source: https://pixabay.com/en/label-quality-satisfaction-1289350/ by GDJ |CC0 Creative Commons

Quality management for software systems

Software quality management **has 3 concerns**:

- **Processes at the organizational level**
  Processes and standards that will lead to high quality software (documentation, design, and code)
- **Processes at the project level**
  Check that organizational processes have been followed
- **Quality management at the project level**
  Establish **a quality plan** for the project which sets out the **quality goals** for the project and defines what **processes and standards** are to be used

Source: https://pixabay.com/en/quality-stamp-guarantee-business-1714288/ by TheDigitalArtist |CC0 Creative Commons

Software Quality Management

MCI MANAGEMENT CENTER INNSBRUCK

Quality assurance, quality control, quality management

Quality assurance
- Definition of processes and standards that should lead to high quality products
- Introduction of quality processes into the manufacturing process

Quality control
- Seek out products that do not meet the required quality level
- Not really used in software engineering

Quality management
- Check whether procedures, processes and standards are met
- Usually taken on by a **separate QM team** which is different from the development team

Source: https://pixabay.com/en/check-hook-check-mark-yes-agree-3183216/ by geralt |CC0 Creative Commons

DIE UNTERNEHMERISCHE HOCHSCHULE®
MCI MANAGEMENT CENTER INNSBRUCK
6020 Innsbruck / Austria
Universitätsstraße 15
www.mci.edu
office@mci.edu
6

MANAGEMENT CENTER INNSBRUCK

Quality management process

Source: Sommerville, I. (2011). Software Engineering, 9th Ed. Boston, MA, USA: Pearson Education Inc.

Software Quality Management

MANAGEMENT CENTER INNSBRUCK

Quality management team

- Optimally quality management is not associated with any particular group **(i.e. QM team),** but rather **has organization-wide responsibility**
- If **QM lies with the project management** there may arise the temptation to sacrifice quality to meet goals
- However, in **small organizations** it is often impossible to have an explicit QM team

Source: https://pixabay.com/en/hand-united-together-people-unity-1917895/ by truthseeker08 |CC0 Creative Commons

MCI MANAGEMENT CENTER INNSBRUCK

Quality planning

- Developing a **quality plan for a project**
- Describe the **desired software qualities and how they are to be assessed**
- Without definition, there may be **conflicting assumptions** about the desired quality
- **Agile methods** use a **less formal approach** to quality management i.e. the customer usually evaluates the quality through testing

Source: https://pixabay.com/en/teamwork-team-gear-gears-drive-2198961/ by geralt | CC0 Creative Commons

Software Quality Management

MCI MANAGEMENT CENTER INNSBRUCK

Quality culture

- Standards and processes are important but quality managers should also aim to develop **a 'quality culture'**
- **Everyone** responsible for software development should be **committed to produce high quality software**
- They should **themselves develop new approaches for quality improvement**
- There are also **intangible qualities** that cannot be measured by standards (e.g. readability of code, elegance, etc.)

Source: https://pixabay.com/en/books-door-entrance-culture-1655783/ by ninocare | CC0 Creative Commons

MANAGEMENT CENTER INNSBRUCK

Software Quality

- Fundamentals of quality management were established by **manufacturing**
- The idea was that **products could be completely specified** (Crosby, 1979)
- Consequently one can **check a product against its specification**
- A small **tolerance was allowed** ('almost right')
- This notion of quality **cannot be applied to digital systems!**

Source: https://pixabay.com/en/check-hook-check-mark-yes-agree-3183213/ by geralt | CC0 Creative Commons

MANAGEMENT CENTER INNSBRUCK

Software Quality

The idea of tolerance **is not applicable** to software/digital systems because:

- It is **difficult to write complete and unambiguous specifications**
- Specifications usually **integrate requirements from several classes of stakeholders** and hence are full of compromise
- It is **impossible to measure certain quality characteristics** (e.g. maintainability)

Source: https://pixabay.com/en/quality-control-quality-control-571149/ by geralt | CC0 Creative Commons

Software Quality

The QM team has to use their judgment to decide if an acceptable level of quality has been achieved. Some questions asked may include:

- Have **programming and documentation standards** been followed in the development process?
- Has the software been properly **tested**?
- Is the software sufficiently **dependable** to be put into use?
- Is the **performance** of the software **acceptable** for normal use?
- Is the software **usable**?
- Is the software **well-structured and understandable**?

Source: https://pixabay.com/en/question-question-mark-help-2309040/ by 3dman_eu |CC0 Creative Commons

Note: you may have a separate testing group for checking software against its requirements!

Software Quality Management

MCI MANAGEMENT CENTER INNSBRUCK

Software Quality

- **Subjective quality** of software systems is **largely based on non-functional characteristics**
- If the **functionality is not exactly** what the user expects, **he/she will find a way around this**
- Yet, if the software is **unreliable or too slow**, then it is practically **impossible to achieve any goals** with it
- Boehm et al. (1978) suggest **15 important software quality attributes**

Source: https://pixabay.com/en/checklist-list-check-mark-business-1919328/ by Tumisu |CC0 Creative Commons

DIE UNTERNEHMERISCHE HOCHSCHULE®
MCI MANAGEMENT CENTER INNSBRUCK
6020 Innsbruck / Austria
Universitätsstraße 15
www.mci.edu
office@mci.edu

MCI MANAGEMENT CENTER INNSBRUCK

Software Quality Attributes (Boehm et al., 1978)

Safety	Understandability	Portability
Security	Testability	Usability
Reliability	Adaptability	Reusability
Resilience	Modularity	Efficiency
Robustness	Complexity	Learnability

Source: Sommerville, I. (2011). Software Engineering, 9th Ed. Boston, MA, USA: Pearson Education Inc.

Note: you cannot optimize for all of these attributes e.g. **improving robustness may lead to loss of performance.** The **quality plan** needs to **define** which **attributes** are important.

MCI MANAGEMENT CENTER INNSBRUCK

Manufacturing vs. Software Engineering

- In **manufacturing** there is a **clear link between process and product quality**
- Once the **process is well defined** it will consistently **output high-quality products**
- **However, software is not manufactured but rather designed!**
- There is a **high influence** of individual **skills** and **creativity**
- Still, **best practices encapsulated in standardized processes** can help produce high quality software

Source: https://pixabay.com/en/flat-design-symbol-icon-www-2126877/ by janjf93 | CC0 Creative Commons

MCI MANAGEMENT CENTER INNSBRUCK

Software Standards

- An important part of quality assurance is the **definition** or **selection** of standards
- Once standards have been selected, project-specific processes have to be defined to **monitor** the use of these standards

2 types of software engineering standards
- **Product standards**
  Applies to the software product being developed i.e. documentation standards, coding standards, etc.
- **Process standards**
  Define the processes to be followed during development e.g. definition of specification, design and validation processes, etc.

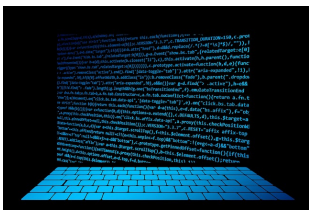
Software Quality Management

MCI MANAGEMENT CENTER INNSBRUCK

Software Standards and Engineers

Although engineers usually agree about the general need for standards, **they often find reasons** why a certain standard is not necessary

QM should therefore take the **following steps**:
- **Involve software engineers** in the selection of product standards
- **Review and modify** standards regularly to reflect changing technologies
- **Provide software tools** to support standards



Source: https://pixabay.com/en/programming-keyboard-computer-3460032/
by geralt | CCO Creative Commons

MCI MANAGEMENT CENTER INNSBRUCK

Software Standards and Engineers

Different types of software need different standards. **QM and PM** need to **decide** early in the project which standards

- To **use without change**
- To **modify**
- To **ignore**

Also, **new standards** may have to be created in response to customer or project requirements

Source: https://pixabay.com/en/network-cables-cable-patch-494649/ by blickpixel | CCO Creative Commons

MCI MANAGEMENT CENTER INNSBRUCK

The ISO 9001 Standard Framework

- The ISO 9000 standards are an i**nternational set of standards** that can be used in the development of all quality management systems in all industries
- **ISO 9001** is the **most general** of these standards
- It was originally **developed in 1987** with its most recent **revision in Sept. 2015**
- It **sets out general principles**, **describes quality processes and lays out organizational standards and procedures** that should be defined
- All these should be documented in the **organizational quality manual**

Source: https://pixabay.com/en/iso-standard-symbol-global-154533/ by OpenClipart-Vectors | CCO Creative Commons

MANAGEMENT CENTER INNSBRUCK

The ISO 9001 Standard Framework

Source: Sommerville, I. (2011). Software Engineering, 9th Ed. Boston, MA, USA: Pearson Education Inc.

MANAGEMENT CENTER INNSBRUCK

The ISO 9001 Standard Framework

Source: Sommerville, I. (2011). Software Engineering, 9th Ed. Boston, MA, USA: Pearson Education Inc.

MCI MANAGEMENT CENTER INNSBRUCK

The ISO 9001 Standard Framework

- Some software customers demand their **supplier should be ISO 9001 certified** as this **gives them confidence** that the company has an approved quality management system in place
- Yet, this **does not mean that ISO 9001 certified companies develop better** software than those without this certification
- **Agile methods** usually avoid endless documentation and rather focus on the code being developed. Also, these **companies feel opposed** to what they see as bureaucratic **overhead of standards**

MCI MANAGEMENT CENTER INNSBRUCK

Reviews and inspections

- QA activities that **check the quality of project deliverables**
- Usually involves a **group of people who examines** the software
- It is based on the **documents that have been produced** during the development process
- It is checked for **consistency** and **completeness**
- It is also used to help **discover problems and omissions**
- The conclusion of the reviewing process should be **formally recorded** as part of the QM process

Source: https://pixabay.com/en/fax-white-male-3d-model-isolated-1889009/ by 3dman_eu | CC0 Creative Commons

Reviews and inspections

- The purpose of reviews is to **improve software** quality, **not to assess performance** of people
- It is inevitable that **mistakes made by individuals** will be **revealed to the whole team**
- Project managers therefore, need to be **sensitive**
- They should have developed a working culture that provides **support without blame**

Source: https://pixabay.com/en/thumb-feedback-confirming-3150402/ by gerait |CC0 Creative Commons

The review process

The review process is normally structured into **three phases**:

- **Pre-review activities**
  Planning and review preparation; team; time and place; documents; etc.
- **The review meeting**
  'walk through' the document with the reviewing team; should be short i.e. approx. 2 hours
- **Post-review activities**
  Issues and problems raised during the review should be addressed; bug-fixing, refactoring, etc.

Source: https://pixabay.com/en/startup-meeting-brainstorming-594090/ by StartupStockPhotos |CC0 Creative Commons

Review

Review team
The review team usually consists **of three to four people** with at least **one senior** who will take the responsibility for **making significant technical decisions**

Agile methods
- In agile the review process is usually **informal**
- **Scrum** uses a **review** meeting after each iteration
- **XP** uses **pair programming** for continuous reviewing

Source: https://pixabay.com/en/white-male-3d-model-isolated-3d-2064870/ by 3dman_eu |CC0 Creative Commons

Program inspections

- 'peer reviews' where **team members collaborate** to find bugs
- Usually **line-by-line code review**
- Aims at identifying **logical errors**, **anomalies** in the code that might indicate an erroneous condition, or **features** that have been **omitted**
- Looks at both **program code** and **design models**
- During inspection checklists **taken from books** or **from knowledge** of common defects may be used

Source: https://pixabay.com/en/certified-genuine-authentic-1726359/ by TheDigitalArtist |CC0 Creative Commons

MANAGEMENT CENTER INNSBRUCK

Software Measurement Metrics

- Tries to **derive a numeric value or profile** for an attribute of a software component
- May be used to **draw conclusions about the quality of software or assess the effectiveness of software processes, tools and methods**
- **Long-term** goal is to **use measurement in place of reviews**
- Numbers could **highlight areas of the software** that could be improved
- However, there is still quite **a long way to go for this ideal solution**

Source: https://pixabay.com/en/drive-car-traffic-road-information-44276/ by Clker-Free-Vector-Images | CC0 Creative Commons

MANAGEMENT CENTER INNSBRUCK

What you should have taken away from this class:

- **Software quality management** is **concerned** with ensuring that **software has a low number of defects** and that it reaches the required standards of maintainability, reliability, portability, and so on. It includes **defining standards** for processes and products and **establishing processes** to check that these standards have been followed.
- Software standards are important for **quality assurance** as they represent an identification of **'best practice'**. When developing software, standards provide a solid foundation for building good quality software.
- You should document a **set of quality assurance procedures** in an **organizational quality manual**. This may be based on the generic model for a quality manual suggested in **the ISO 9001 standard.**
- **Reviews** of the software process deliverables **involve a team of people** who check that quality standards are being followed. Reviews are the most widely used technique for assessing quality.
- In a **program inspection** or **peer review**, a small team systematically checks the code. They read the code in detail and look for possible errors and omissions. The problems detected are then discussed at a **code review meeting**.
- **Software measurement** can be used to **gather quantitative data** about software and the software process. You may be able to use the values of the **software metrics** that are collected to make inferences about product and process quality.

DIE UNTERNEHMERISCHE HOCHSCHULE®
MCI MANAGEMENT CENTER INNSBRUCK
6020 Innsbruck / Austria
Universitätsstraße 15
www.mci.edu
office@mci.edu

References

- Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M., Macleod, G., & Merrit, M. (1978). *Characteristics of Software Quality*. Amsterdam, The Netherlands: North-Holland.
- Crosby, P. (1979). *Quality is free.* New York, NY, USA: McGraw-Hill.
- Ince, D. (1994). *ISO 9001 and Software Quality Assurance.* Reading, MA, USA: McGraw-Hill.
- Stalhane, T., & Hanssen, G. K. (2008). The Application of ISO 9001 to Agile Software Development. *Proceedings of the PROFES International Conference on Product Focused Software Process Improvement.*

Thank you for your attention!

mentoring the motivated.