Software Quality

- Views
- Implications
- Parameters

Software Quality

- Software Engineering
  - "The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software."
    - IEEE
- Quality
  - Not a single idea - many aspects

Software Quality

- Popular View
  - In everyday life, usually thought of as intangible, can be felt or judged, but not weighed or measured
  - "I know it when I see it" - implies that it cannot be controlled, managed, or quantified
  - Often influenced by perception rather than fact
    - For example, a Cadillac may be spoken of as a “quality” car, in spite of the fact that its reliability and repair record is much the same as a Chevrolet

Software Quality

- Professional View
  - In a profession such as software development, there is an ethical imperative to quality
  - Quality is not just a marketing and perception issue, it is a moral and legal requirement - we have a professional responsibility associated with the software we create
  - Professionals must be able to demonstrate, and have confidence, that they are using "best practices"

Software Quality

- Professional View
  - In practical terms, therefore, product quality must be measurable in some way
  - Product quality is spoken of in terms of
    - conformance to requirements, including timeliness and cost
    - fitness for use, does it actually do the job?
    - freedom from errors and failures, is it reliable and robust?
    - customer satisfaction, are users happy with it?

Software Quality

- Why does it matter?
  - McConnell's "Code Complete" suggests an industry average of 15–50 errors per 1,000 lines of delivered code
  - Thousands of bugs in commercial software
  - If we still use the software, what harm is there in a few errors?

Software Quality

- Therac-25
  - Radiation therapy machine
  - Patients were given massive overdoses of radiation
    - Approximately 100 times the intended dose
  - Primarily blamed on bad software design and development practices
    - Software designed so that it was realistically impossible to automate testing

Software Quality

- Software Quality
  - Software quality is normally spoken of in terms of several different dimensions called quality parameters, roughly split into two groups
    - Technical Quality Parameters
    - User Quality Parameters

Software Quality

- Technical Quality Parameters
  - Correctness, reliability, capability, performance, maintainability
  - These are open to objective measures and technical solutions (focus of this course)
- User Quality Parameters
  - Usability, installability, documentation, availability
  - These often require subjective analysis and non-technical solutions

Technical Quality Parameters

- Correctness: lack of bugs and defects
  - measured in terms of defect rate
    (# bugs per line of code)
- Reliability: does not fail or crash often
  - in terms of failure rate (# failures per hour)
- Capability: does all that is required
  - in terms of requirements coverage
    (% of required operations implemented)

Technical Quality Parameters

- Maintainability: easy to change and adapt to new requirements
  - in terms of change logs (time and effort to add a new feature) and impact analysis (# lines affected by a new feature)
- Performance: fast and small enough
  - in terms of speed and space usage (seconds of CPU time, MB of memory, etc.)

User Quality Parameters

- Usability: sufficiently convenient for the intended users
  - in terms of user satisfaction (% of users happy with interface and ease of use)
- Installability: convenient and fast to install
  - in terms of user satisfaction (# install problems reported per installation)

User Quality Parameters

- Documentation:
  well documented
  - in terms of user satisfaction
    (% of users happy with documentation)
- Availability:
  easy to access and available when needed
  - in terms of user satisfaction
    (% of users reporting access problems)

Customers vs. users

- Pre-internet era: *customers* = *users*
- Internet era:
  If someone pays, they are the customer
- Consequence:
  - Facebook's users are **not** Facebook's customers
  - "Customer satisfaction"?