
Softwareentwicklung

Softwarelebenszyklus

- Idee
- Entwicklung (development)
- Anwendung (use) und Wartung (maintenance)
- letzte Anwendung

Softwareentwicklungsschritte

zyklisch wiederholt (und oft überlappend):

- Analyse (analysis)
- Entwurf (design) !
- Implementierung (implementation) !!
- Verifikation (verification) !
- Testen (testing) !
- Validierung (validation)

Schritte beim Programmieren

zyklisch wiederholt:

- Planen
- Editieren
- Übersetzen
- Testen
- Debuggen

Programmierwerkzeuge

- Editor
- Compiler und Interpreter
- Debugger
- Integrierte Entwicklungsumgebung

Softwarequalität (1)

Brauchbarkeit entsprechend Bedürfnissen der Anwender

- Zweckerfüllung
- Bedienbarkeit
- Effizienz

Zuverlässigkeit je nach Anwendungsbereich, Kostenfaktor

- Bewährtheit
- formale Korrektheit
- Fehlerresistenz

Softwarequalität (2)

Wartbarkeit als wichtiger Kostenfaktor

- Einfachheit
- Lesbarkeit
- Lokalität
- Faktorisierung

Festlegung von Softwareeigenschaften

- informelle Beschreibung
- Anwendungsfälle (use cases)
- Testfälle
- formale Spezifikation

Arten von Softwareeigenschaften

- funktionale Eigenschaften
- nichtfunktionale Eigenschaften