



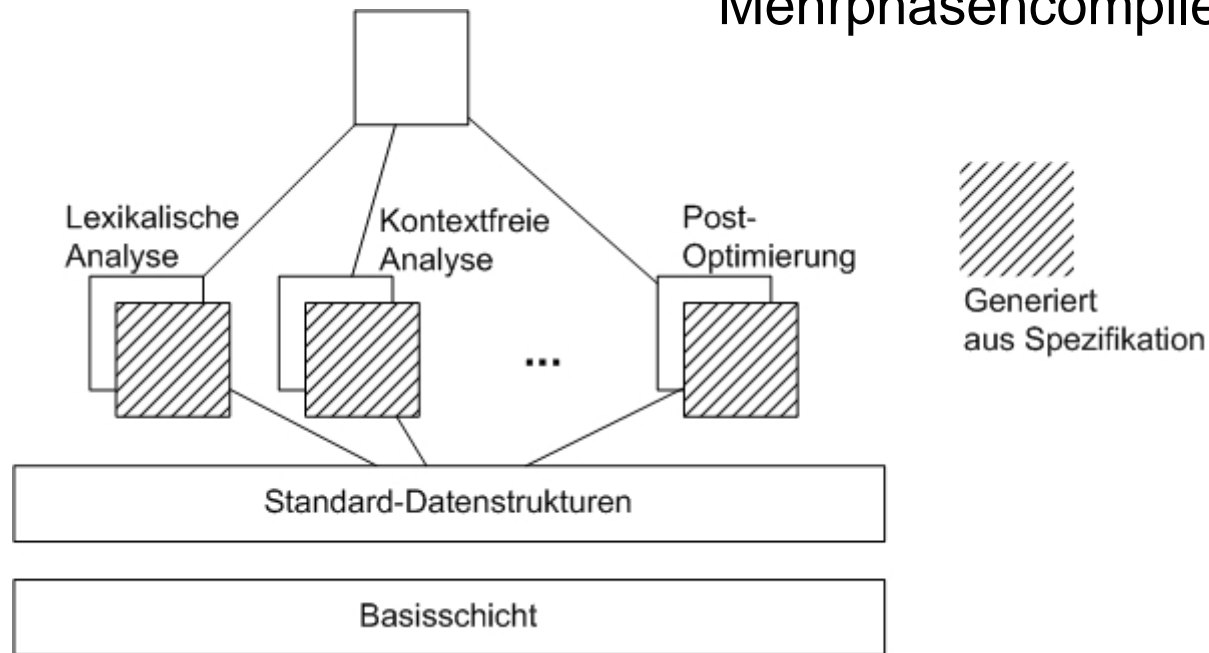
Übung „Einführung in die Softwaretechnik“

Lösungshinweise zum Übungsblatt 04



Aufgabe 8 – Spannweite von Softwareentwicklungsprozessen

- Wohlverstandener Prozess: Entwicklung eines Mehrphasencompilers

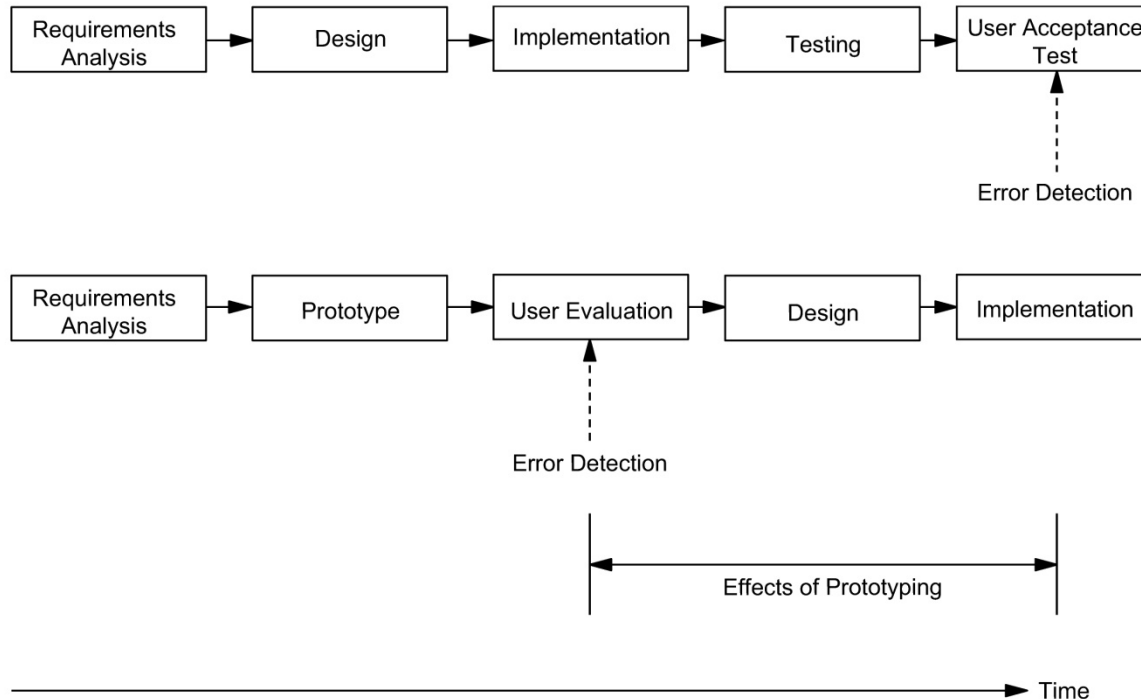


- Anforderungsspezifikation ist im wesentlichen Sprachdefinition
- Architektur durch Rahmenwerk vorgegeben
- Implementierung zum großen Teil automatisch generiert



Aufgabe 8 – Spannweite von Softwareentwicklungsprozessen

- Risikominimierung durch Rapid Prototyping



- Exploratives Prototyping (Anforderungen klären)
- Experimentelles Prototyping (Machbarkeit nachweisen)



Aufgabe 9 – Beispiele für Fehler

	Vollständigkeit	Korrektheit	Konsistenz mit anderen Dokumenten
Anforderungs-spezifikation	Anforderungen fehlen	Widersprüchliche Anforderungen	Anforderungen falsch erhoben
Entwurfs-spezifikation	Modul für Datenbankzugriff nicht eingeplant	Zyklische allgemeine Benutzbarkeit Nicht-eindeutige Modulnamen	Geforderte Anbindung an verschiedene Datenbanken wurde nicht berücksichtigt
Implementation	Methodenrumpf wurde nicht implementiert	Methoden wurden fehlerhaft implementiert (Compilerfehler, Laufzeitfehler, ...)	Implementierung entspricht nicht Entwurfsspezifikation Ein anderes Modul als im Entwurf vorgegeben wird verwendet
Testfälle/ Testsystem	z.B. bei Black-Box-Test: Nicht alle Äquivalenzklassen abgedeckt	Unit-Test, Stubs, Driver fehlerhaft implementiert	Testfälle entsprechen nicht funktionalen Anforderungen
Dokumentation	Installationshandbuch vergessen	Dokumentation widersprüchlich Verweise auf nicht existierende Kapitel	Benutzerdokumentation inkonsistent mit Systemfunktionalität



Aufgabe 10

	A	E	Impl
Korrektheit	B	B	(T), V
Vollständigkeit	-	B	B, T
Robustheit	-	-	B, T
Bediener- freundlichkeit	-	-	T
Verständlichkeit	B	B	B
Änderbarkeit	-	B	B

- $A \cong$ Anforderungsdefinition
- $E \cong$ Architekturspezifikation
- $Impl \cong$ Quelltext

- $V \cong$ Verifikation
- $T \cong$ Test
- $B \cong$ Begutachtung



Aufgabe 11

- **Objekte**
 - Abstraktion von Realität/Daten/Implementierung
 - Modularisierung, Information Hiding, Lokalität
 - keine Mehrfachverwendung, kein Festhalten von Ähnlichkeiten
- **Klassen**
 - Abstraktion von einzelnen Instanzen
 - keine Hierarchiebildung (\Rightarrow Vererbung)
 - Modularisierung
 - Mehrfachverwendung, Festhalten von Ähnlichkeiten
- **Vererbung**
 - Abstraktion von einzelnen Klassen
 - Strukturierung, Hierarchiebildung
 - Mehrfachverwendung, Festhalten von Ähnlichkeiten
- **Polymorphie**
 - Abstraktion vom „Datentyp“
 - statische Polymorphie, dynamische Polymorphie