

Primäre Ziele:

- Rüstzeug für die erfolgreiche Durchführung Ihres Software-Projekts vermitteln
 - Modell für zukünftige ähnliche Projekte bieten
- Praktikum und Vorlesung bilden eine Einheit

Primäre Ziele sind **nicht**:

- Vollständige Darstellung aller Themengebiete der Softwaretechnik¹
- Vermittlung von spezifischen Kenntnissen für die Entwicklung des Anwendungssystems X

¹Wird im Hauptstudium nachgereicht.

Szenario:

- Erstellung eines großen Softwaresystems
 - hier: mehrere Mitarbeiter über einen langen Zeitraum
 - hier **nicht**: ein „Freizeit“-Programmierer
- für einen Auftraggeber
 - hier: Individualsoftware
 - hier **nicht**: Standardsoftware (Commercial off the Shelf)
- im Rahmen eines Projekts
 - hier: einmalige Zielverfolgung
 - hier (eher) **nicht**: Weiterentwicklung existierender Software
(aber: die Software wird von uns später weiterentwickelt; Wartbarkeit ist erklärtes Ziel)

- Softwaretechnik ist nicht nur Programmieren.

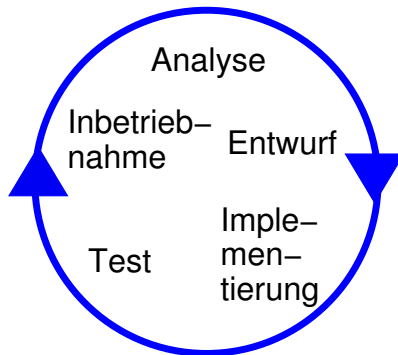
- Softwaretechnik ist nicht nur Programmieren.
- Softwaretechnik ist auch Programmieren.

Angestrebte Resultate

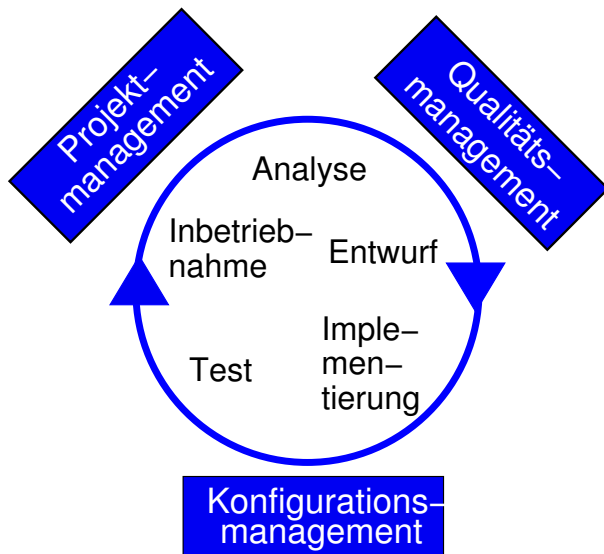
- Softwaretechnik ist nicht nur Programmieren.
- Softwaretechnik ist auch Programmieren.
- Software-Entwicklung produziert Dokumente.
- Der Quellcode ist **ein** Dokument **unter vielen**.

- Softwaretechnik ist nicht nur Programmieren.
- Softwaretechnik ist auch Programmieren.
- Software-Entwicklung produziert Dokumente.
- Der Quellcode ist **ein** Dokument **unter vielen**.

Sie sollen nicht nur ein richtiges System bauen.
Sie sollen ein System auch richtig bauen.



Aktivitäten bei der Softwareentwicklung



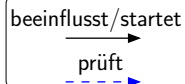
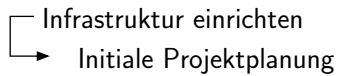
Infrastruktur einrichten

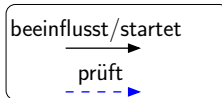
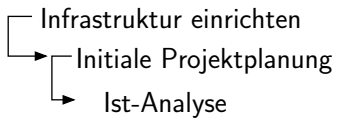
beeinflusst/startet

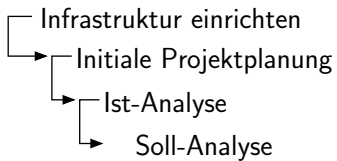


prüft

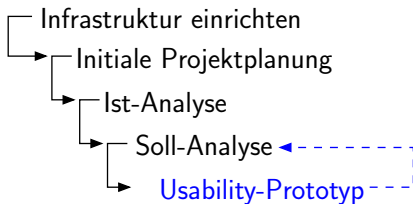




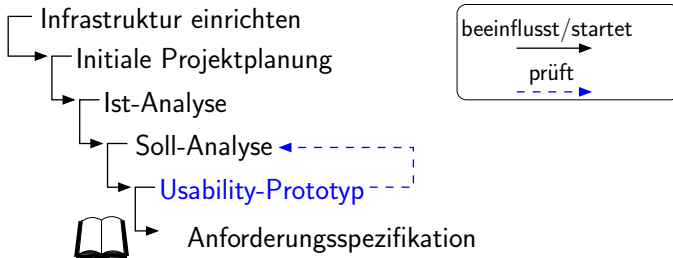


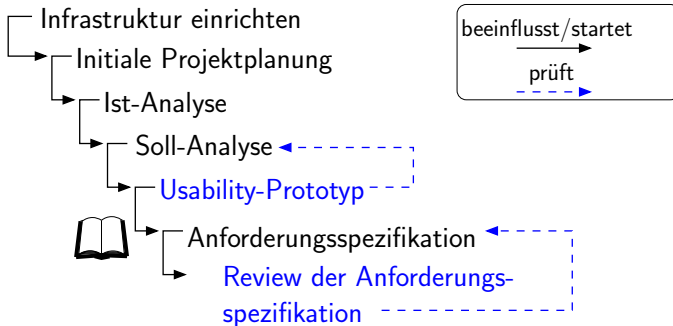


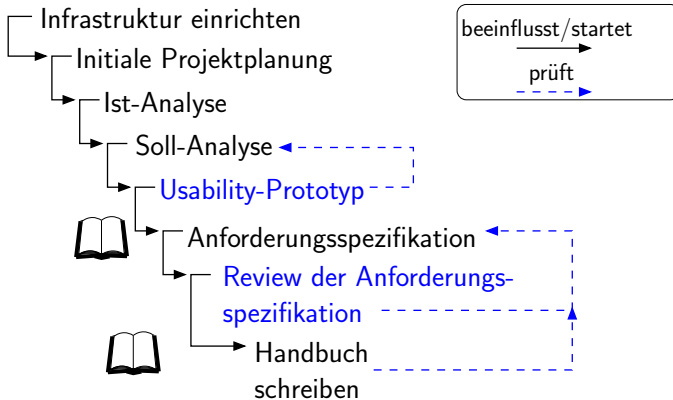
beeinflusst/startet
 prüft

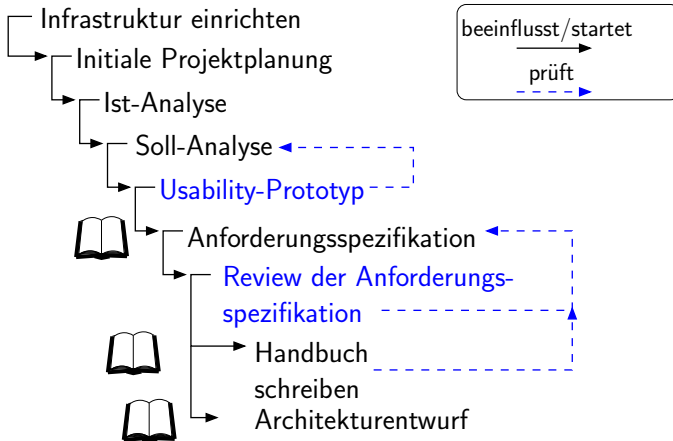


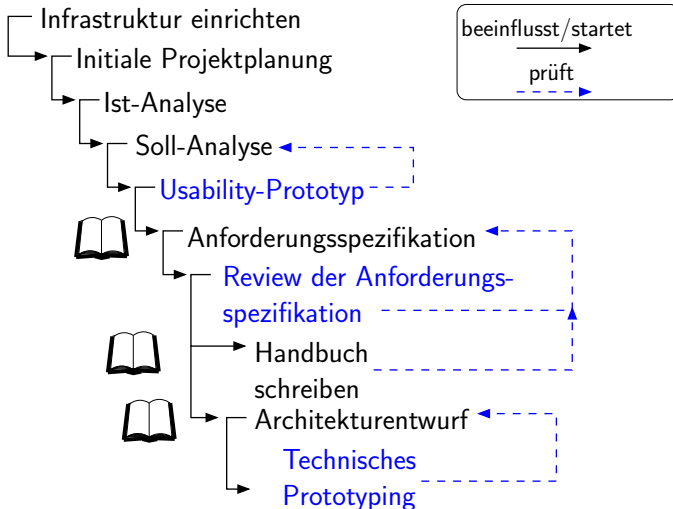
beeinflusst/startet
 prüft

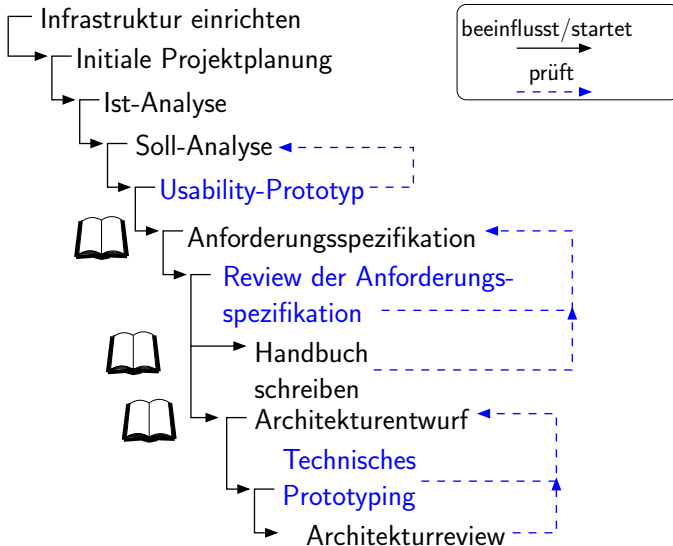


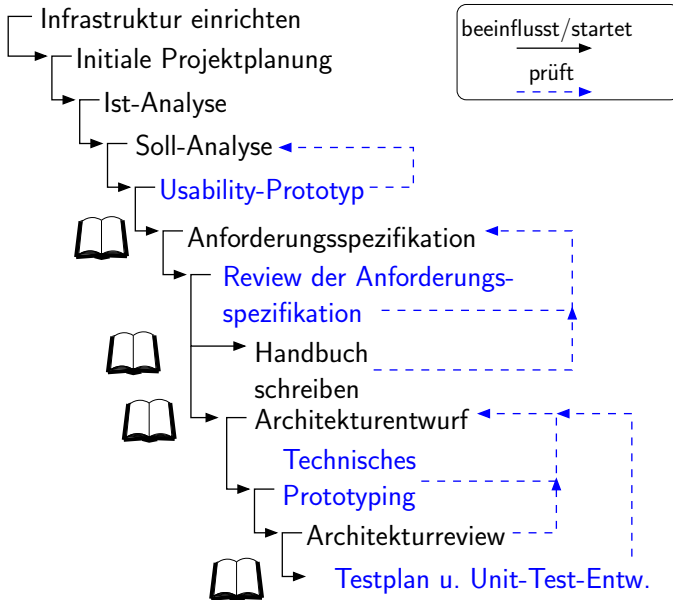


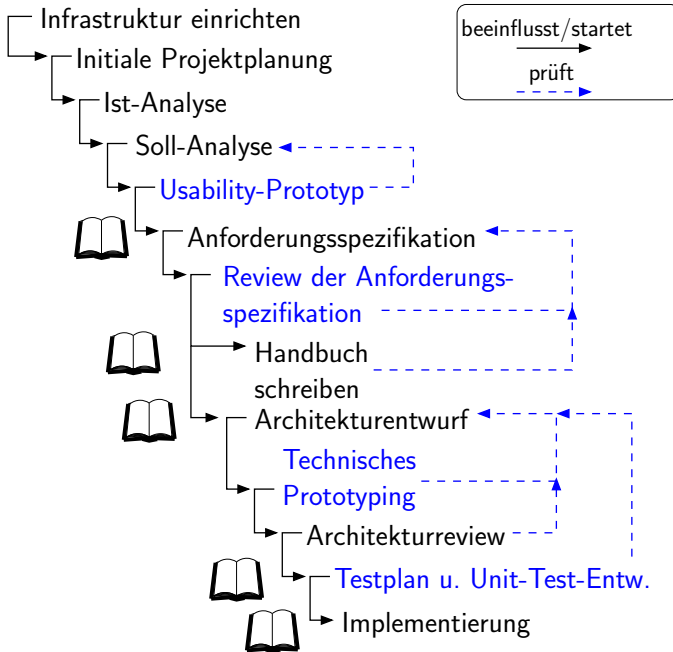


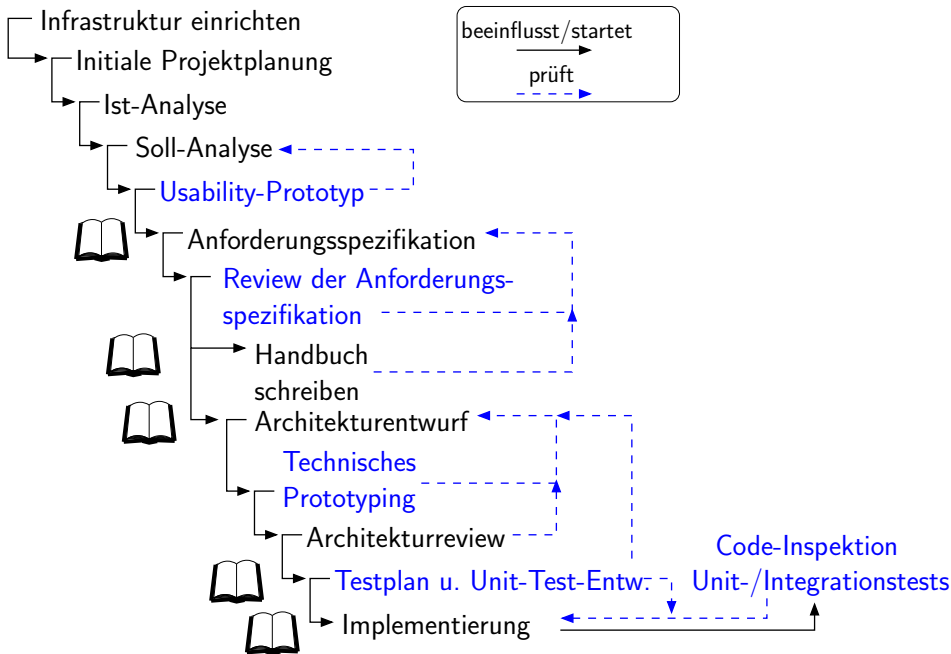


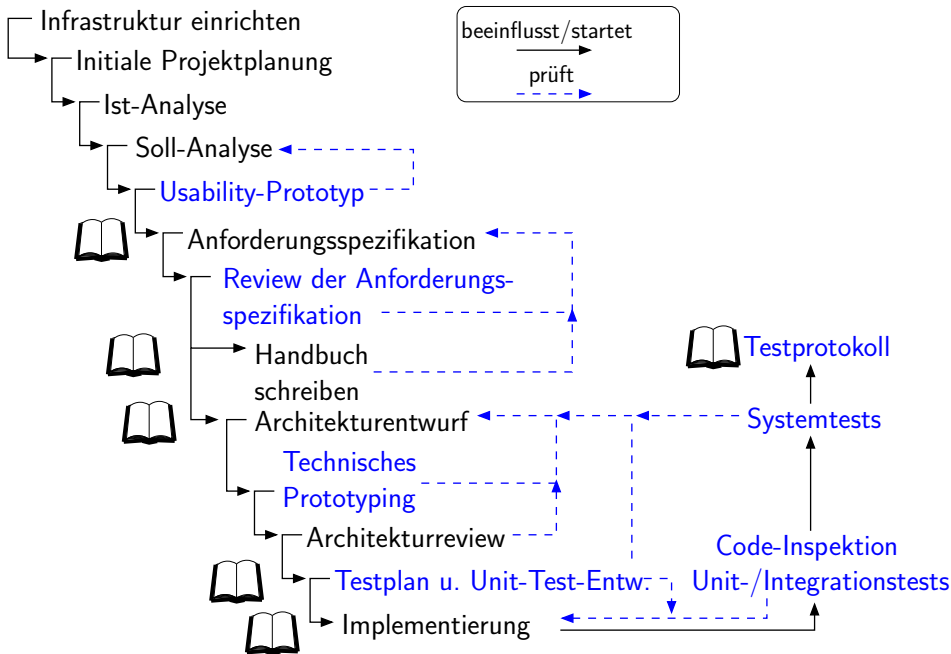


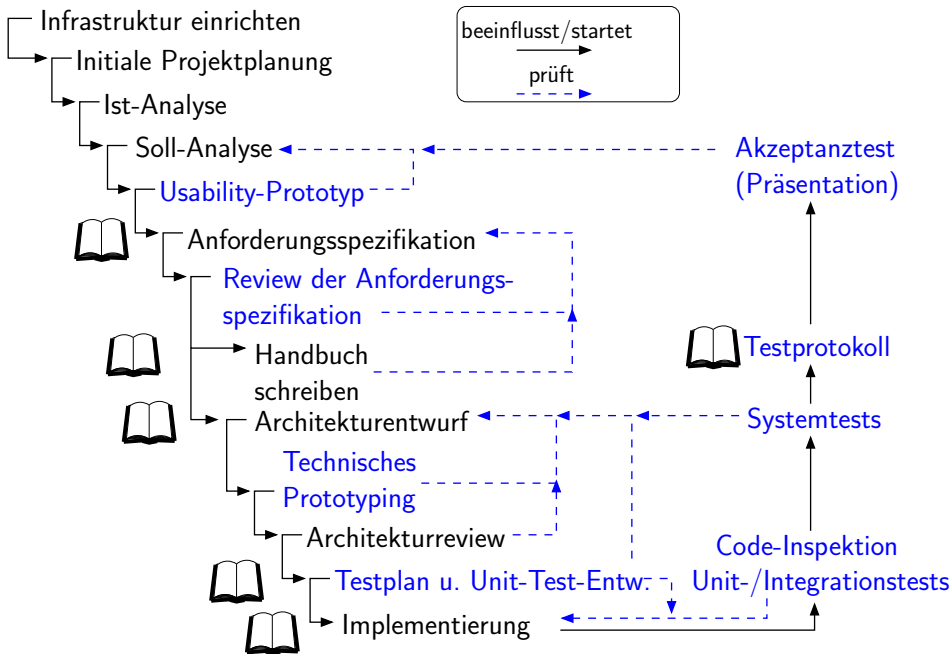


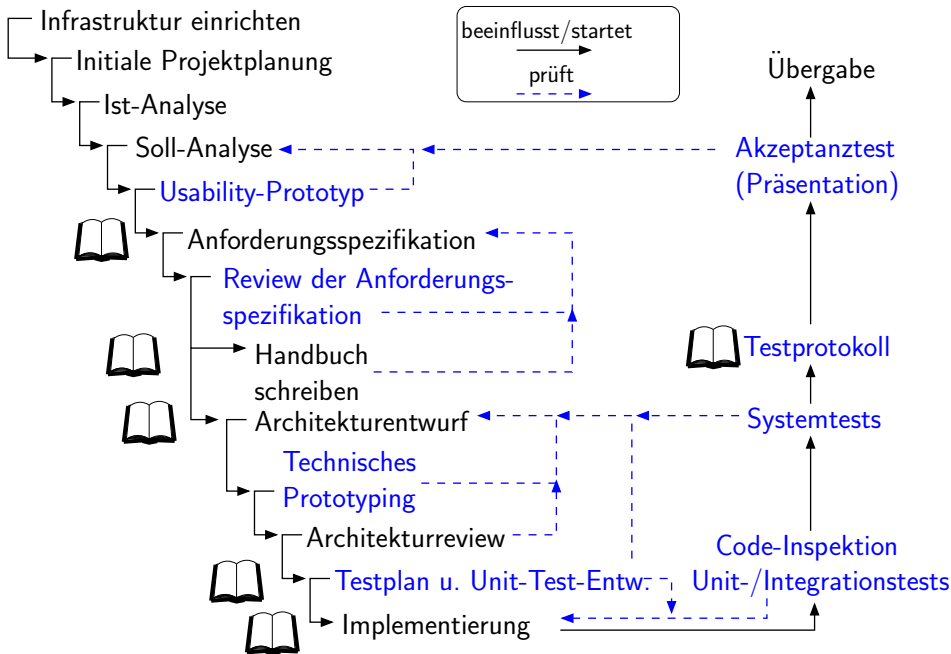












Vorläufiger Zeitplan

5.12.2006	Initialer Projektplan
23.1.2007	Anforderungsspezifikation mit Angebot
23.1-6.2.2007	Review der Anforderungsspezifikation
12.-16.2.2007	Vorlesung <i>Datenbankgrundlagen</i>
27.2.-6.3.2007	Prüfung zur Vorlesung
1.4.2007	Architekturentwurf
16.4.2007	Testplan mit Unit-Tests
16-30.4.2007	Walkthrough des Architekturentwurfs
Juni/Juli 2007	Code-Inspektion
20.7.2007	Testprotokoll
20.7.2007	Handbuch
20.7.2007	Endabgabe ²
23.7.-3.8.2007	Abschlusspräsentationen

Vorläufige Teilergebnisse müssen Sie evtl. in den Tutorien vorher abgeben.

²Implementierung (Source+lauffähig), Doku., Testplan, Whiteboxtests, Testprotokoll

- initialer Projektplan umfasst Aufgaben, Teilnehmer, Rollen, Verantwortlichkeiten und Pflichten, Risiken und deren Behandlung

Abgaben des inkrementellen Projektplans

- initialer Projektplan umfasst Aufgaben, Teilnehmer, Rollen, Verantwortlichkeiten und Pflichten, Risiken und deren Behandlung
- danach: inkrementelle Fortführung
 - Abgabedatum eine Woche nach Abgabe der abgeschlossenen Phase
 - Verantwortlicher ist Phasenleiter für die zu planende Phase
 - erste Woche jeder Phase dient Einarbeitung und Planung
 - Granularität der Arbeitspakete: Arbeitsumfang von 1-2 Personen mit max. 1-2 Zeitwochen
 - zu Beginn des Sommersemesters:
ein Plan für die verbleibenden Arbeiten

Abgaben des inkrementellen Projektplans

- initialer Projektplan umfasst Aufgaben, Teilnehmer, Rollen, Verantwortlichkeiten und Pflichten, Risiken und deren Behandlung
- danach: inkrementelle Fortführung
 - Abgabedatum eine Woche nach Abgabe der abgeschlossenen Phase
 - Verantwortlicher ist Phasenleiter für die zu planende Phase
 - erste Woche jeder Phase dient Einarbeitung und Planung
 - Granularität der Arbeitspakete: Arbeitsumfang von 1-2 Personen mit max. 1-2 Zeitwochen
 - zu Beginn des Sommersemesters:
ein Plan für die verbleibenden Arbeiten

Fertigstellung (Ist) wird dokumentiert durch individuelle Zeiterfassung aufgeschlüsselt in Aktivitäten (TimeTracker).

Kriterien zum Erwerb des Leistungsnachweises I

- Berichtsheft über beide Semester
 - individuell für jeden Teilnehmer
 - gegengezeichnet von allen Gruppenmitgliedern
 - System zur Erfassung wird vorgegeben (TimeTracker)
 - muss spätestens drei Tage nach jeder Abgabe aktuell sein
- alle Inkremente/Versionen des Projektplans
- Angebot mit Kostenschätzung
- Anforderungsspezifikation (ggf. angepasst bei Änderungen)
 - Mindestanforderungen auf der Web-Seite
 - die Bewertung des abgegebenen Systems wird an Anforderungsspezifikation gemessen

Kriterien zum Erwerb des Leistungsnachweises II

- Architekturentwurf
- Programm-Code und Test-Code
- Benutzerhandbuch
- Qualitätssicherung
 - Reviews der Anforderungsspezifikation und des Architekturentwurfs (Abgabe der Protokolle)
 - Testplan und -protokoll
 - Code-Inspektion
- Abschlusspräsentation
 - Vorführung des lauffähigen Systems
 - Reflektion über den Projektverlauf
- Prüfung zur Vorlesung (inkl. Datenbankgrundlagen) im Wintersemester

- die Veranstaltung besteht aus
 - ① Vorlesung mit 6 Credit Points (CP)
 - ② Projekt mit 12 CP

→ Note zur Vorlesung macht ein Drittel der Endnote aus

- die Veranstaltung besteht aus
 - ① Vorlesung mit 6 Credit Points (CP)
 - ② Projekt mit 12 CP

→ Note zur Vorlesung macht ein Drittel der Endnote aus

- mündliche Prüfungsform:
 - Einzelgespräche
 - Aufgaben 30 min vorab
 - 15 min Gespräch über die Aufgaben (und ggf. darüber hinaus)

- die Veranstaltung besteht aus
 - ① Vorlesung mit 6 Credit Points (CP)
 - ② Projekt mit 12 CP

→ Note zur Vorlesung macht ein Drittel der Endnote aus

- mündliche Prüfungsform:
 - Einzelgespräche
 - Aufgaben 30 min vorab
 - 15 min Gespräch über die Aufgaben (und ggf. darüber hinaus)
- schriftliche Prüfungsform:
 - Klausur über 60 min
 - ab einer Interessentenzahl von mind. 20 Prüflingen

- Prüfung zur Vorlesung muss erfolgreich abgelegt werden
- Projekt muss erfolgreich beendet werden
 - Mindestkriterien erfüllt zum Abgabetermin
 - ausreichender individueller Beitrag
 - höchstens eine Abgabe, die mit 5,0 bewertet wurde

Gruppe insgesamt ist verantwortlich für das Gesamtergebnis.

Gruppe insgesamt ist verantwortlich für das Gesamtergebnis.

Text auf dem Leistungsnachweis:

„Die individuelle Leistung der Studentin/des Studenten ist nach meinen Feststellungen deutlich abgrenzbar und bewertbar.“

→ **Individuelles** Berichtheft erläutert den individuellen Beitrag!

Gruppe insgesamt ist verantwortlich für das Gesamtergebnis.

Text auf dem Leistungsnachweis:

„Die individuelle Leistung der Studentin/des Studenten ist nach meinen Feststellungen deutlich abgrenzbar und bewertbar.“

→ **Individuelles** Berichtheft erläutert den individuellen Beitrag!

Rollenverteilung bleibt dem Projekt überlassen.

Jedoch: Jede Phase hat einen Leiter. Jedes Mitglied übernimmt diese Rolle mindestens einmal.

Projektplan legt die Verantwortlichkeiten fest.

- ① Die Teilnahme an der Code-Inspektion ist für jeden Pflicht. Einzelheiten werden in den Übungsgruppen vereinbart.
- ② Kann jemand den eigenen Code nicht erläutern, erhält sie / er **keinen Leistungsnachweis** für das Software-Projekt.

Gewichtungen

Projektplan	8 %
Planung Entwurfsphase	2 %
Planung Sommer	4 %
Anforderungsspez./Angebot	22 %
Architekturentwurf	20 %
Implementierung	24 %
Testplan	5 %
Tests+Testprotokoll	5 %
Dokumentation	7 %
Abschlusspräsentation	3 %

In begründeten Fällen kann die individuelle Note von der Gruppennote abweichen.