

- **Anforderungsanalyse**
 - Funktionale Anforderungen mit Use Cases
 - Qualitätsanforderungen und Randbedingungen
 - Domänenmodell
- **Design**
 - Softwarearchitektur
 - Use Case Realisierung (Statische und dynamische Modelle)
- **Implementation**
 - Quellcode (inkl.Javadoc)
- **Testing**
 - Unit-Tests
 - Integrations- und Systemtests

0.1 Überblick Anforderungsanalyse

- **User Research**
 - Personas
 - Szenarien
 - Contextual Inquiry
- Sketching und Prototyping
- **Use Cases**
 - Ableiten und Modellieren
 - Detaillierung (UML-Use-Case-Diagramm, Use-Case-Spezifikation, UI-Sketching)
- **Qualitätsanforderungen, Randbedingungen** erheben
- **Domänenmodell**
 - Konzeptuelles UML-Klassendiagramm
- **objektorientierte Analyse(OOA)**
 - Objekte/Konzepte in dem Problembereich zu finden und zu beschreiben

0.2 Überblick Design

- **Softwarearchitektur**
 - UML-Paketdiagramm
 - UML-Deploymentdiagramm
- **Use-Case-Realisierung und Klassendesign**
 - UML-Klassendiagramm
 - UML-Sequenzdiagramm
 - UML-Kommunikationsdiagramm
 - UML-Zustandsdiagramm
 - UML-Aktivitätsdiagramm
- Entwurf **Design Patterns**
- **Objektorientierte Design (OOD)**
 - Geeignete Softwareobjekte und ihr Zusammenwirken definieren

0.3 Überblick Implementation

- **Code**
 - Umsetzung Design in entspr. OOP-Sprache
- **Refactoring**
 - Code smells aufdecken und verbessern
- **laufende Dokumentation**
 - vom Quellcode

0.4 Überblick Testing

- **Unit-Tests**
 - Laufendes Design und Implementierung
- **Teststufen Integration und System**
 - Planung, Design und Durchführung
- **Dokumentation**
 - Testkonzept und Test