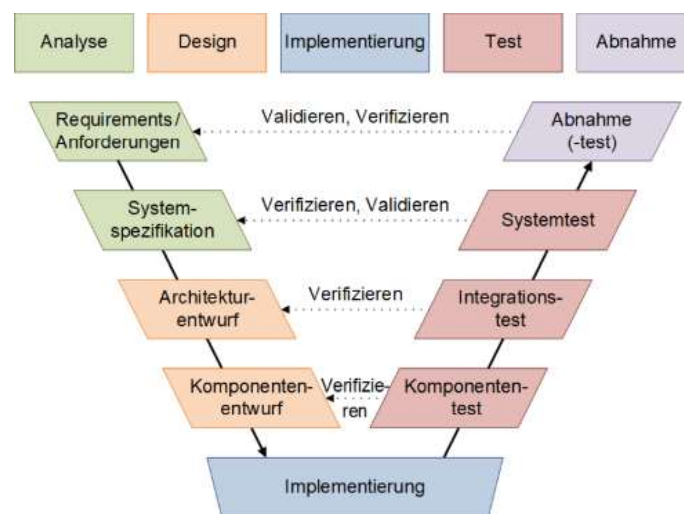


# Vorgehensmodelle Testing

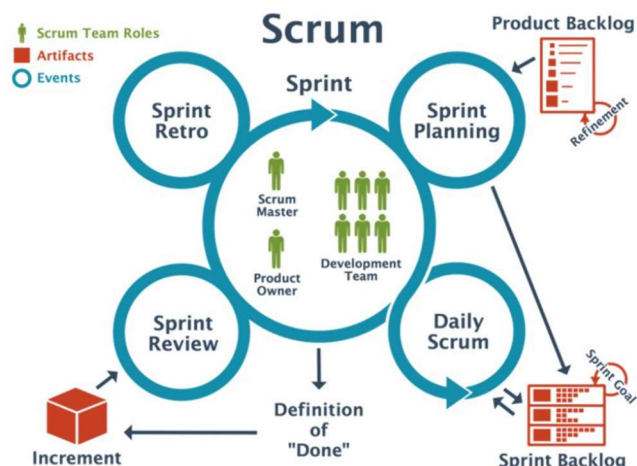
## V-Modell

Linker Ast: Entwicklungsschritte	Rechter Ast: Integrations- und Testarbeiten
<b>Anforderungsdefinition</b> mit den Wünschen und Anforderungen des Auftraggebers	<b>Komponententest</b> prüft, ob jeder elementare Softwarebaustein seine Vorgaben erfüllt.
<b>Funktionaler Systementwurf</b> wo die Anforderungen auf Funktionen und Dialoge abgebildet werden	<b>Integrationstest</b> prüft, ob Gruppen von Komponenten korrekt zusammenspielen.
<b>Technischer Systementwurf</b> , wo die technische Realisierung entworfen wird (System wird in Komponente unterteilt, Schnittstellen werden definiert)	<b>Systemtest</b> prüft, ob das System als Ganzes die Anforderungen erfüllt.
<b>Komponentenspezifikation</b> , wo jedes Teilsystem im Detail beschrieben wird.	<b>Abnahmetest</b> prüft, ob das System vom Kunden als korrekt akzeptiert wird.
<b>Programmierung</b> , wo jeder Baustein (Modul, Klasse, etc.) in einer Programmiersprache programmiert wird	



Tests werden parallel zum Entwickeln erstellt

## Scrum



1. Nach «Definition of Done» wird ein neues Inkrement so gesagt ein nächstes Ziel gesetzt.
2. Danach startet der Sprint für das **Inkrement**, wofür man ein Sprint **Review** macht.
3. Als nächstes kommt der **Sprint Retro**.
4. Im Anschluss macht man ein **Sprint Planning** mit dem **Product Backlog**.
5. Zum Schluss findet mit Hilfe des **Sprint Backlogs** das **Daily Scrum** statt.
6. Der Sprint wird vom **Scrum Master**, **Product Owner** und vom **Dev-Team** abgehalten

Die Software wird in Versionen entwickelt.