

# **Vorgehensmodelle zur Projektentwicklung**

**Ausarbeitung für die mündl. PRE Matura 2015**

Samuel Hammer

28. Mai 2015

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>3</b>
1.1	Phasenmodelle . . . . .	4
1.2	Agile Softwareentwicklung . . . . .	4
<b>2</b>	<b>Wasserfallmodell</b>	<b>5</b>
2.1	Vorteile des Wasserfallmodells . . . . .	6
2.2	Nachteile des Wasserfallmodells . . . . .	7
<b>3</b>	<b>Inkrementalmodell</b>	<b>8</b>
<b>4</b>	<b>Evolutionsmodelle</b>	<b>9</b>
4.1	Rapid Prototyping . . . . .	9
<b>5</b>	<b>Spiralmodell</b>	<b>10</b>

# 1 Einleitung

IT-Projekte können je nach konkretem Anwendungsfall unterschiedlich groß bzw. komplex werden. Um unabhängig von diesen Faktoren die Übersichtlichkeit, Organisation und Struktur eines Projektes zu gewährleisten, gibt es verschiedene Vorgehensmodelle. Die existierenden Vorgehensmodelle lassen sich grob in zwei Kategorien unterteilen:

- Klassische Phasenmodelle
- Vorgehensmodelle zur agilen Softwareentwicklung

Abhängig vom Anwendungsfall, können verschiedene Vorgehensmodelle verschiedene Vor- und Nachteile für das Projekt haben. In der Praxis gibt es also in den seltensten Fällen ein Modell, das für die Durchführung des Projektes hundertprozentig passt.

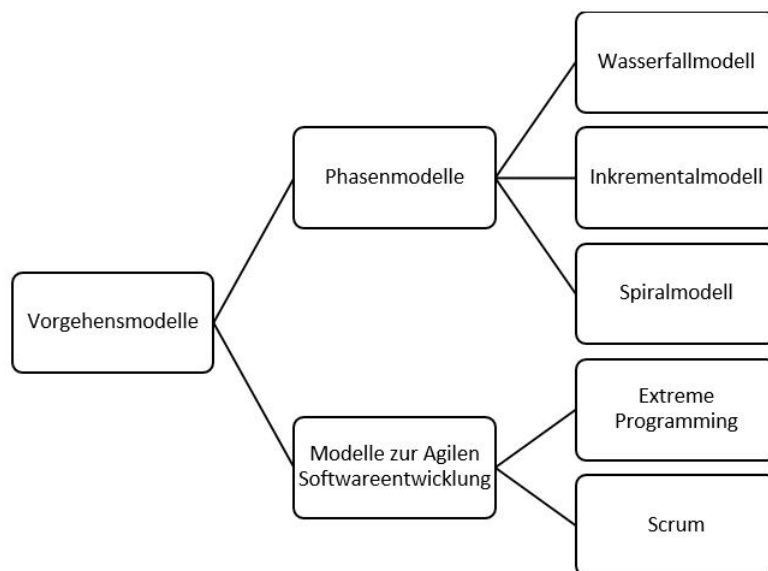


Abbildung 1.1: Übersicht über Vorgehensmodelle

## 1.1 Phasenmodelle

Phasenmodelle sind „standardisierte Projektstrukturen für die Erstellung des Projekterzeugnisses“.<sup>1</sup> Da allerdings nicht auf jedes Projekt die selben Phasen zutreffen, gibt es verschiedene Abwandlungen. Für Softwareprojekte wird in der Regel folgende Phasenstruktur verwendet:

1. Anforderungen
2. Analyse
3. Design
4. Entwicklung
5. Test

Wenn diese verschiedenen Phasen des Projektes komplett sequentiell und unabhängig voneinander stattfinden, so spricht man vom klassischen Wasserfallmodell. (Siehe Kapitel 2) In der Praxis hat sich das strikte vorgehen von oben nach unten (also wie ein Wasserfall) allerdings nicht bewährt, da dieses Modell es nicht erlaubt einzelne Phasen zu wiederholen. Für diesen Fall wurden iterative Modelle entwickelt. Hier können dann einzelne Phasen wiederholt werden falls sich beispielsweise während des Projektes die Anforderungen verändern.

## 1.2 Agile Softwareentwicklung

Agile Softwareentwicklung gibt es seit den frühen 1990er Jahren. Mit steigender Komplexität der Softwareprojekte, wurde auch der Aufwand in Bezug auf das Projektmanagement immer größer. Da klassische Phasenmodelle aufgrund ihrer geringen Flexibilität nicht mehr praktikabel waren, wurde das sogenannte „Agile Manifest“ verfasst. Das Agile Manifest ist ein Werk, in dem Agile Werte definiert werden, um Softwareprojekte einfacher und besser zu gestalten. Ein Auszug aus dem Agilen Manifest besagt:

„Wir erschließen bessere Wege, Software zu entwickeln, indem wir es selbst tun und anderen dabei helfen. Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:

- **Menschen und Interaktionen** mehr als Prozesse und Werkzeuge
- **Funktionierende Software** mehr als umfassende Dokumentation
- **Zusammenarbeit mit dem Kunden** mehr als Vertragsverhandlung
- **Reagieren auf Veränderung** mehr als das Befolgen eines Plans

Das heißt, obwohl wir die Werte auf der rechten Seite wichtig finden, schätzen wir die Werte auf der linken Seite höher ein.“<sup>2</sup>

---

<sup>1</sup>[3]

<sup>2</sup>[2]

## 2 Wasserfallmodell

Das Wasserfallmodell ist eines der ältesten linearen Vorgehensmodellen. Es ist nicht iterativ, d.h. wenn man nach dem Wasserfallmodell vorgeht, ist es prinzipiell nicht möglich einzelne Projektphasen zu wiederholen. Ursprünglich stammt das Wasserfallmodell aus der Bau- und Produktionsbranche. In anbetracht der Tatsache, dass zum damaligen Zeitpunkt noch kein eigenes brauchbares Modell zur Softwareentwicklung existierte, wurde das Wasserfallmodell schlicht und einfach für IT-Projekte adaptiert.

Bei diesem Modell muss erst jede Phase abgeschlossen werden, bevor zur nächsten übergegangen werden kann. Daraus resultiert eine starke Unflexibilität, was das Wasserfallmodell vor allem bei Softwareprojekten sehr umständlich und beinahe unbrauchbar macht. Wenn beispielsweise in einem Projekt bereits mit der Programmierung begonnen wurde und der Kunde im Nachhinein eine Änderung an der Spezifikation vornehmen möchte, ist dies laut Wasserfallmodell nicht mehr möglich. Vor allem bei langläufigen

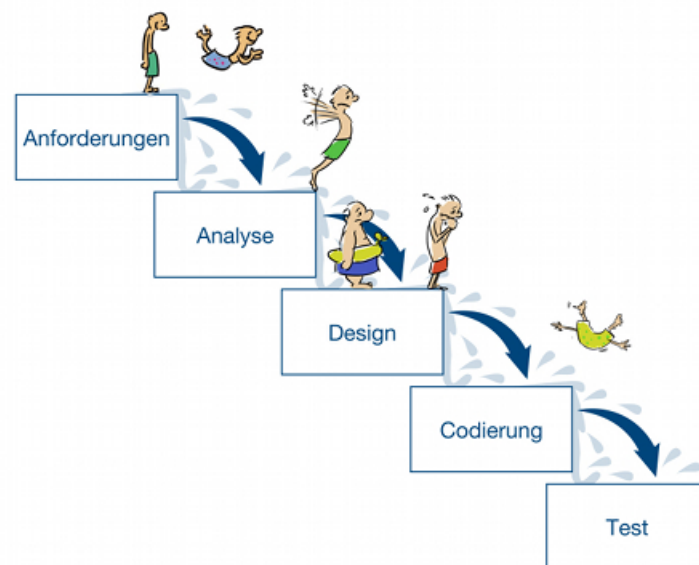


Abbildung 2.1: Das Wasserfallmodell[6]

Projekten ist der Einsatz des Wasserfallmodells sehr problematisch, da sich über lange Zeiträume oft die technischen Gegebenheiten und Umgebungsbedingungen ändern

können. Sollte bei weit fortgeschrittenem Projektverlauf ein Fehler in einer früheren Phase entdeckt werden, ist nicht selten ein Projektabbruch die einzige Möglichkeit den Schaden einigermaßen zu begrenzen.

Um die oben genannten Probleme aus der Welt zu schaffen, oder zumindest zu verbessern wurde das Erweiterte Wasserfallmodell nach Royce entwickelt. Bei dieser Version

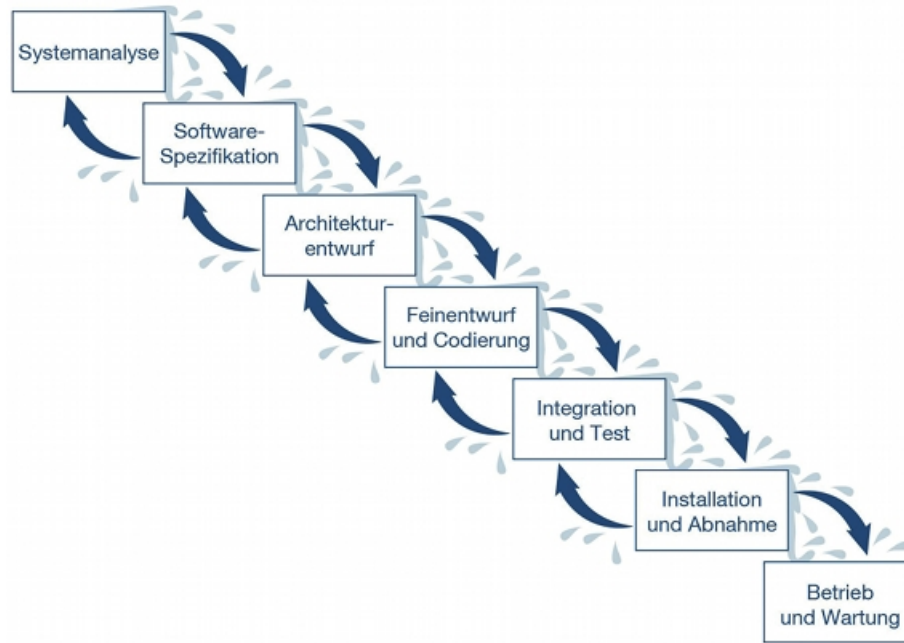


Abbildung 2.2: Das Erweiterte Wasserfallmodell nach Royce[6]

des Wasserfallmodells sind Rücksprünge erlaubt. Nichtsdestotzotz müssen nach einem Rücksprung auf eine vorhergehende Phase, alle danach folgenden Phasen wiederholt werden. Diese Änderung verbessert das Modell zwar drastisch, allerdings werden dadurch noch lange nicht alle grundlegenden Fehler ausgemerzt.

## 2.1 Vorteile des Wasserfallmodells

Da das Wasserfallmodell ein sehr altes und rudimentäres Vorgehensmodell ist, hat es, auf heutige Projekte bezogen, nicht viele Vorteile.

Ein entscheidender Vorteil ist allerdings die Einfachheit des Modells. Es lässt sich in der Praxis sehr leicht auf ein Projekt anwenden und vermittelt schnell und einfach den Eindruck von Übersicht und Kontrolle. Entgegen aller Kritik, kann das Wasserfallmodell also durchaus sinnvoll sein, solange sich das Projekt im sehr kleinen Rahmen bewegt.

## 2.2 Nachteile des Wasserfallmodells

Wie bereits in Kapitel 2 ausführlich erklärt, ist das Wasserfallmodell aufgrund der fehlenden Flexibilität und der Tatsache, dass der gesamte Projektverlauf linear sein muss, in der Praxis für heutige Projekte nahezu unbrauchbar. Werden Fehler erst spät im Projektverlauf entdeckt, führt das zu sehr kostspieligen Änderungen. Des Weiteren tritt durch die lange Laufzeit eines solchen Projektes, der sogenannte "Return On Investment"<sup>1</sup> erst sehr spät nach Beginn des Projektes ein.

Auch ist es beim Wasserfallmodell oft schwierig, die einzelnen Phasen voneinander abzugrenzen. Übergänge, die in der Theorie klar definiert sind, können in der Praxis oftmals eher fließend sein.

---

<sup>1</sup>"Der Begriff Return on Investment (...) ist eine betriebswirtschaftliche Kennzahl zur Messung der Rendite einer unternehmerischen Tätigkeit, gemessen am Gewinn im Verhältnis zum eingesetzten Kapital." [4]

## **3 Inkrementalmodell**



## **4 Evolutionsmodelle**

### **4.1 Rapid Prototyping**

## 5 Spiralmodell

# Literatur

- [1] *Agile Softwareentwicklung*. Abgerufen am: 27.05.2015. Wikipedia. URL: [http://de.wikipedia.org/wiki/Agile\\_Softwareentwicklung](http://de.wikipedia.org/wiki/Agile_Softwareentwicklung).
- [2] Kent Beck u. a. *Manifesto for Agile Software Development*. 2001. URL: <http://www.agilemanifesto.org/>.
- [3] *Projektphase*. Abgerufen am: 27.05.2015. Wikipedia. URL: <http://de.wikipedia.org/wiki/Projektphase>.
- [4] *Return On Investment*. Abgerufen am: 28.05.2015. Wikipedia. URL: [http://de.wikipedia.org/wiki/Return\\_on\\_Investment](http://de.wikipedia.org/wiki/Return_on_Investment).
- [5] *Vorgehensmodell zur Softwareentwicklung*. Abgerufen am: 27.05.2015. Wikipedia. URL: [http://de.wikipedia.org/wiki/Vorgehensmodell\\_zur\\_Softwareentwicklung](http://de.wikipedia.org/wiki/Vorgehensmodell_zur_Softwareentwicklung).
- [6] *Wasserfall-Modell*. Abgerufen am: 28.05.2015. Scrum-kompakt.de. URL: <http://www.scrum-kompakt.de/grundlagen-des-projektmanagements/wasserfall-modell/>.
- [7] *Wasserfallmodell*. Abgerufen am: 28.05.2015. Wikipedia. URL: <http://de.wikipedia.org/wiki/Wasserfallmodell>.
- [8] Harald Wittek. *Projektentwicklung 5. Semester Projektphasenmodelle*. HTBLVA Spengergasse.

# Abbildungsverzeichnis

1.1	Übersicht über Vorgehensmodelle . . . . .	3
2.1	Das Wasserfallmodell[6] . . . . .	5
2.2	Erweitertes Wasserfallmodell . . . . .	6