# V/W-Modell

# Was Ist das?

**V-Modell:**  
Das V-Model sorgt für eine klare Strukturierung in der Softwareentwicklung. Es läuft gleich wie das Wasserfallmodell ab (Top-Down-Prinzip):  
  
Es wird zu jeder Entwurfsphase eine Testphase durchgeführt (Bottom-Up-Prinzip). So kann während der Entwicklung sichergestellt werden, dass, das System den Anforderungen entspricht

**Aufbau:**  
Auf der linken Seite sind die Entwurfsphasen abgebildet, also die Anforderungen an das Projekt. Sie laufen von oben nach unten ab ganz unten sind die Anforderungen detaillierter als oben.  
  
  
Auf der rechten Seite sind die Testphasen. Bei dieser Phase wird die Funktionalität und die Anforderungen des Systems überprüft  
  
A diagram of a software development

Description automatically generated

## Die Phasen des V-Modells

* **Entwurfsphase**
  + Top-Down-Prinzip
* **Implementierung**
  + Produkt entwickelt.
* **Validierungsphase**
  + Bottom—Up-Prinzip

**Entwurfsphasen:**

1. **Anforderungsanalyse:** In dieser Phase werden die Anforderungen an das System festgelegt.  
   ❔ Was soll das fertige Produkt am Ende können? Bis wann soll die Software fertig sein?
2. **Systementwurf:** Hier wird das Systemdesign erstellt. Es wird festgelegt, wie das System umgesetzt werden soll und welche Schritte dafür nötig sind.  
   ❔ Wie können die gesammelten Anforderungen umgesetzt werden?
3. **Architektur:** In der Architekturphase werden die Zuständigkeiten, Schnittstellen und Abhängigkeiten beschrieben.  
   ❔ Wer ist für welche Aufgaben zuständig?
4. **Komponentenentwurf:** In dieser Phase werden die einzelnen Komponenten nochmal detaillierter beschrieben. Einzelheiten wie das Backend oder Datenbanktabellen gehören in diesem Schritt dazu, um alle Komponenten möglichst genau festzulegen.  
   ❔ Wie sehen die einzelnen Komponenten genau aus?

**Implementierung**

Wie genau die Implementierungsphase umgesetzt wird, ist im V-Modell nicht vorgeschrieben. Es können **verschiedene Techniken und Methoden** der modernen Software-Entwicklung genutzt werden.

### **Validierung**

### Deutlich konkreter wird es auf der **rechten Seite des V**, auf der es um **Tests und Validieren** des entwickelten Systems geht. Die **Tests auf der rechten Seite beziehen** sich immer auf die jeweils **gleiche Ebene auf der linken Seite**.

### **Komponententest:** Die kleinste Ebene der Software wird getestet.

### **Integrationstest:** Auf der nächsthöheren Ebene wird getestet, ob verschiedene Module gemeinsam wie gewünscht funktionieren und beispielsweise Daten korrekt ausgetauscht werden.

### **Systemtest:** Spätestens hier kommt der Kunde ins Spiel, der das System als Ganzes intensiven Testläufen unterzieht.

### **Abnahmetest:** Der finale Test sollte in einer Umgebung stattfinden, die der späteren Produktivumgebung möglichst nah ist. Auch die Nutzer sollten möglichst den Endnutzern entsprechen, um ein möglichst aussagekräftiges Testergebnis zu erhalten.

### **W-Modell**

## Das W-Modell ist der Nachfolger des V-Modells

## **Für was ist das geeignet?**

**V-Modell**

## Das V-Modell bietet eine breite Anwendbarkeit für Softwareprojekte unterschiedlicher Größen und Branchen, was es zu einem vielseitigen Modell macht. Es erleichtert die Organisation und Durchführung von Projekten in verschiedenen Bereichen, einschließlich der Wirtschaft.

## 

## Speziell in Entwicklungsprojekten wurden Varianten des V-Modells an verschiedene Branchenanforderungen angepasst, um eine bessere Passung zu gewährleisten.

## 

## Allerdings ist das V-Modell linear ausgerichtet und daher nicht immer ideal für komplexe Projekte geeignet. In solchen Fällen bieten agile Methoden oft effektivere Alternativen, da sie flexibler auf Veränderungen reagieren können.

## 

**Stärken**

**V-Modell:**

* Strukturiertes Vorgehen: Das V-Modell bietet eine klare Struktur und definierte Abläufe für den Softwareentwicklungsprozess. Dadurch wird sichergestellt, dass alle relevanten Phasen durchlaufen werden und nichts übersehen wird.

* Verknüpfung von Anforderungen und Tests: Das V-Modell verbindet Anforderungen mit den entsprechenden Tests. Dadurch wird gewährleistet, dass das entwickelte System den Anforderungen entspricht und korrekt funktioniert.

* Frühe Fehlererkennung: Durch die eng mit den Entwicklungsphasen verknüpften Testphasen können Fehler frühzeitig erkannt und behoben werden.

* Nachvollziehbarkeit und Dokumentation: Das V-Modell legt großen Wert auf die Dokumentation jeder Phase. Dadurch wird die Nachvollziehbarkeit des Entwicklungsprozesses und des Systems selbst erleichtert. Es ermöglicht auch eine bessere Zusammenarbeit zwischen verschiedenen Projektbeteiligten.

**Schwächen:**  
**V-Modell:**

* Hoher Dokumentationsaufwand: Das V-Modell erfordert umfassende Dokumentation in jeder Phase, was viel Zeit und Ressourcen beansprucht. Daher könnte es eher für kleinere Projekte geeignet sein.

* Geringe Agilität: Im Vergleich zu agilen Methoden wie Scrum oder Kanban betont das V-Modell weniger Flexibilität und folgt fest vorgegebenen Phasen, was seine Anpassungsfähigkeit einschränken kann.
* Hohe Abhängigkeit von Testaktivitäten: Unzureichend geplante oder durchgeführte Testphasen können zu Verzögerungen führen und die Effektivität des Modells beeinträchtigen, da das V-Modell stark auf die Testaktivitäten angewiesen ist

## Vorteile und Nachteile des V-Modells

Falls du jemals in die Verlegenheit kommst, deinen Chef vom V-Modell überzeugen zu wollen, kannst du folgende **Vorteile** erwähnen:

* Da bereits **früh Tests konzipiert** werden, werden **unvollständige Spezifikationen** in einem frühen Stadium **erkannt**.
* Das Modell ist insgesamt **einfach aufgebaut** und benötigt wenig Schulung für alle Beteiligten.
* Da bereits vor der Implementierung sowohl Spezifikation als auch Tests erarbeitet werden, ist **während der Entwicklung** vergleichsweise **wenig** [**Kommunikation**](https://projekte-leicht-gemacht.de/softskills/kommunikation/) **zwischen Entwicklern und Kunde** nötig.
* **Teammitglieder**, die für das **Testen verantwortlich** sind, werden bereits **in frühen Projektphasen eingebunden**.
* Die **Testabdeckung** sowohl einzelner Komponenten als auch des gesamten Systems **ist hoch**.

Oft als „altes“ Modell bezeichnet, führen **Kritiker** eine Reihe von **Nachteilen** auf:

* Das Modell fordert eine **sehr ausführliche Dokumentation** auf allen Ebenen.
* Das lineare Vorgehen ist **starr und wenig flexibel**.
* Das V-Modell wird besonders von Fürsprechern [agiler Methoden](https://projekte-leicht-gemacht.de/blog/projektmanagement/agil/agile-methoden-ueberblick/) kritisiert, da es den **Prozess der Software-Entwicklung zu stark vereinfacht** und Entscheider in falscher Sicherheit wiegen kann.
* Es unterscheidet sich nicht grundlegend vom Wasserfallmodell, weshalb die gleichen Nachteile gelten, [wie in diesem Artikel beschrieben](https://projekte-leicht-gemacht.de/blog/projektmanagement/klassisch/wasserfallmodell/).

**Anwendungsgebiete**

**V-Modell:**

Das V- Modell eignet sich am besten für IT-Projekte, wegen der durchgehenden Testphasen.

<https://projekte-leicht-gemacht.de/blog/projektmanagement/klassisch/v-modell/>

<https://de.wikipedia.org/wiki/V-Modell>

<https://studyflix.de/wirtschaft/v-modell-6716>

<https://asana.com/de/resources/v-model>

A screenshot of a computer

Description automatically generated

A white paper with black text

Description automatically generated

A diagram of a brick wall

Description automatically generated

A diagram of a software development

Description automatically generated with medium confidence

A white paper with black text

Description automatically generated