

Phase 10

1. Beschreiben Sie den Software-Lebenszyklus

Der Software-Lebenszyklus ist ein Prozess, der die Phasen der Entwicklung und Wartung von Softwareprodukten beschreibt. Es gibt verschiedene Modelle, die den Software-Lebenszyklus beschreiben können, aber im Allgemeinen umfasst er die folgenden Phasen:

1. **Anforderungsanalyse:** In dieser Phase werden die Anforderungen an die Software festgelegt. Das umfasst die Erfassung und Analyse von Kundenbedürfnissen, die Identifizierung von Funktionen und Features und die Festlegung von Zielgruppen. Also kurz gesagt: Was soll das System leisten?
2. **Analyse und Entwurf:** In dieser Phase wird die Architektur und Struktur der Software entwickelt. Dabei werden die Funktionen und Features in ein konkretes Konzept umgesetzt und es werden Pläne für die Programmierung, das Datenmanagement und die Benutzeroberfläche erstellt. Also wie soll das System arbeiten?
3. **Implementierung:** In dieser Phase wird die Software programmiert und erstellt. Das umfasst das Schreiben von Quellcode, das Testen von Softwaremodulen und die Integration von Komponenten.
4. **Testen:** In dieser Phase wird die Software auf Fehler und Funktionalität getestet. Das umfasst das Durchführen von manuellen und automatisierten Tests, um sicherzustellen, dass die Software den Anforderungen entspricht.
5. **Auslieferung:** In dieser Phase wird die Software offiziell freigegeben und an den Kunden oder Benutzer übergeben. Das umfasst die Installation und Konfiguration der Software auf dem Zielcomputer oder in der Zielumgebung.
6. **Nutzung:** In dieser Phase wird die Software auf Fehler und Schwachstellen überwacht und es werden Updates und Patches bereitgestellt, um diese zu beheben. Außerdem können Änderungen und Verbesserungen an der Software vorgenommen werden, um sie an veränderte Anforderungen oder Bedürfnisse der Benutzer anzupassen.

Lebenszyklusphase	Projektphase
Anforderungsanalyse	1, 5
Analyse und Entwurf	3, 6, 7, 8, 9

Implementierung	2, 4
Testen	
Auslieferung	
Nutzung	

2. Vergleichen Sie die Vorgehensmodelle

1. Wasserfall-Modell:

Das Wasserfall-Modell ist ein sequentielles Modell, das aus verschiedenen Phasen besteht. Die Phasen folgen in einem linearen Prozess, beginnend mit der Anforderungserfassung, gefolgt von Entwurf, Implementierung, Testen und Inbetriebnahme. Jede Phase muss abgeschlossen sein, bevor die nächste beginnt. Es ist ein eher starres Modell und eignet sich gut für Projekte mit klaren und stabilen Anforderungen.

2. V-Modell:

Das V-Modell ist eine Weiterentwicklung des Wasserfall-Modells. Wie beim Wasserfall-Modell wird jede Phase in einer linearen Abfolge durchgeführt. Der Unterschied besteht darin, dass das V-Modell detaillierte Testpläne enthält, die von der Anforderungserfassung bis zur Abnahme reichen.

3. Spiralmodell:

Das Spiralmodell ist ein iteratives Modell, bei dem jede Phase in kleinen Zyklen durchlaufen wird. Es besteht aus vier Phasen. In jeder Phase werden Risiken identifiziert und Maßnahmen ergriffen, um diese zu minimieren. Das Spiralmodell ist flexibler als das Wasserfall-Modell und eignet sich für Projekte mit hohem Risiko oder hoher Komplexität.

4. Kanban:

Kanban ist ein agiles Modell, das sich auf die Visualisierung von Aufgaben konzentriert. Es basiert auf einem Pull-System, bei dem Arbeitsschritte nur aufgenommen werden, wenn der Bedarf besteht. Das Kanban-Board visualisiert den Arbeitsablauf und zeigt den Fortschritt der einzelnen Aufgaben an. Es ist ideal für Teams, die schnell auf Veränderungen reagieren müssen.

5. Scrum:

Scrum ist ein weiteres agiles Modell, das auf Iterationen und inkrementeller Entwicklung basiert. Das Team organisiert sich in Sprints von 1-4 Wochen, in denen es ein bestimmtes Ziel erreicht. Scrum ist ideal für Teams, die schnell und flexibel auf sich ändernde Anforderungen reagieren müssen.

Fazit:

Insgesamt gibt es bei allen Modellen Vor- und Nachteile. Die Wahl des Modells hängt von der Art des Projekts, den Anforderungen, dem Budget und der verfügbaren Zeit ab. Es gibt kein "bestes" Modell - jedes Modell hat seine Stärken und Schwächen.

3. Wählen Sie eines der Vorgehensmodelle

Rollenverteilung im Projekt:

Jupp: Anbinder für Datenbank

Pascal: Dokumentenersteller,

Henning: Programmierer

Jonas: Product-Owner,

Chris: GUI-Entwickler

Philipp: Ersteller für Testpläne

Niklas: Scrum-Master

Zeitplan:

24	11	13.03.2023	14.03.2023	15.03.2023	16.03.2023	17.03.2023	18.03.2023	II	C	
25	12	20.03.2023	21.03.2023	22.03.2023	23.03.2023	24.03.2023		I	C	22.03.2023
26	13	27.03.2023	28.03.2023	29.03.2023	30.03.2023	31.03.2023	01.04.2023	II	A	
27	14	03.04.2023	04.04.2023	05.04.2023	06.04.2023			I	A	07.04.2023
	15									
28	16	17.04.2023	18.04.2023	19.04.2023	20.04.2023	21.04.2023	22.04.2023	II	A	22.04.2023
29	17	24.04.2023	25.04.2023	26.04.2023	27.04.2023	28.04.2023		I	B	
30	18		02.05.2023	03.05.2023	04.05.2023	05.05.2023	06.05.2023	II	B	01.05.2023
31	19	08.05.2023	09.05.2023	10.05.2023	11.05.2023	12.05.2023		I	B	09.05.2023
32	20	15.05.2023	16.05.2023	17.05.2023				II	C	18.05.2023 19.05.2023
33	21	22.05.2023	23.05.2023	24.05.2023	25.05.2023	26.05.2023		I	C	
34	22		30.05.2023	31.05.2023	01.06.2023	02.06.2023	03.06.2023	II	A	29.05.2023
35	23	05.06.2023	06.06.2023	07.06.2023	08.06.2023	09.06.2023		I	A	
36	24	12.06.2023	13.06.2023	14.06.2023	15.06.2023	16.06.2023	17.06.2023	II	B	12.06.2023
37	25	19.06.2023	20.06.2023	21.06.2023	22.06.2023	23.06.2023		I	B	
38	26	26.06.2023	27.06.2023	28.06.2023	29.06.2023	30.06.2023	01.07.2023	II	C	
39	27	03.07.2023	04.07.2023	05.07.2023	06.07.2023	07.07.2023		I	C	03.-05.07.2023 21.08.2023

Bis Ende KW 12: Phase 11 Modul 1&2

Bis Ende KW 20: Phase 11 (Komplett)

Bis Ende KW 21: Phase 12 (Testphase)

Bis Ende KW 26: Gesamt Auswertung der Phasen

Bis Ende KW 27: Präsentation an den Kunden