

# Softwaretechnik 2021

Bearbeitet von: Jonny Lam

## Aufgabe 1-2: Zielkonflikte in der Softwareentwicklung

Eine Softwareentwicklung gilt allgemein hin als erfolgreich, wenn

- die Kosten niedrig waren,
- die verstrichene Zeit bis zur Fertigstellung kurz war,
- die Qualität der Software hoch ist und
- der Umfang der Software groß ist, also viele Funktionen implementiert wurden. Bearbeiten Sie die folgenden Teilaufgaben:

a) Sind alle vier Ziele gleichermaßen erreichbar? Was passiert, wenn man versucht eines der Ziele (z.B. die Qualität) zu optimieren?

Nein, nicht alle vier Ziele sind gleichermaßen erreichbar.

Wenn man ein Projekt planen will, muss man Kompromisse eingehen, angenommen ein Kunde will sein Produkt schon in einer Woche fertig haben, aber die Zeit reicht nicht aus, um damit allein fertig zu werden. So kann man für diese Woche einen Mitarbeiter einstellen und sich die Arbeit aufteilen. Durch den Mitarbeiter haben wir also die **Zeit und Qualität** des Produktes eingehalten, doch die **Kosten** wurden höher (da man den Mitarbeiter auch bezahlen muss).

Will man jetzt z.B. das Projekt doch allein machen, so könnte man den Umfang verringern, aber dann hätte man **Umfang gegen Zeit** getauscht.

Um hingegen **keine extra Kosten zu generieren bei gleichbleibender Qualität und Umfang**, kann man alternativ den Kunden um **mehr Zeit** bitten.

b) Nennen und erläutern Sie jeweils ein Beispiel für Art von Software, bei der eines der vier Ziele den Vorrang haben sollte.

**Individualsoftware:** Hier sollte die Qualität der Software hoch sein, da man nur einen Kunden hat mit denen man auch über die Anforderungen kommunizieren kann.

**Standardsoftware oder eigener Web-Dienst:** Hier sollte der Umfang im Vordergrund stehen, da man (potenziell) Millionen BenutzerInnen und KundInnen hat, diese dann aus dem Grund auch verschiedene Anforderungen an das Programm haben.

c) Warum kann man bei Softwareprojekten die Ziele niedriger Kosten und geringer Zeitaufwand zusammenfassen und so die Ziele der Softwareentwicklung als ein Dreieck darstellen?

Niedrige Kosten und geringer Zeitaufwand kann man zusammenfassen, da die beiden Punkte sich nicht gegenseitig ausschließen. Man kann niedrige Kosten haben und einen geringeren Zeitaufwand, wenn das Projekt z.B. für ein einfaches Produkt sein soll, dann braucht man logischerweise nicht so viel Zeit dafür und man hat **weniger Stunden** daran gearbeitet was auch zu **niedrigeren Kosten** führt, da man meistens pro Stunde bezahlt wird.

d) Untersuchen Sie das Verhältnis zwischen Kosten und Zeitaufwand bei einem Softwareprojekt genauer. Nehmen Sie an, ein/e Softwareentwickler/in würde für eine Software mit definiertem Umfang und definierter Qualität 1 Jahr Entwicklungszeit benötigen. Wie lange würden zwei, fünf und hundertfünfzig Entwickler/innen für das gleiche Projekt benötigen? Begründen Sie Ihre Antworten.

Es ist nicht leicht zu sagen, wie hoch der **Kosten und Zeitaufwand** mit mehr Mitarbeitern sein würde.

Die Frage ist ob sich das Projekt gut aufteilen lässt in zwei, fünf oder 150 Entwickler/innen. Die Entwickler/innen sollen ja nicht all an einer Aufgabe arbeiten.

Dazu muss man noch die Zeit einberechnen, in dem zwischen den Entwickler/innen kommuniziert wird. Je mehr Entwickler/innen, desto länger die benötigte Zeit für die Kommunikation.

Wenn ein Entwickler 1 Jahr braucht, müssten theoretisch zwei Entwickler ein halbes Jahr brauchen, um den gleichen Kosten und Zeitaufwand zu haben.

150 Entwickler hätten dann 2 Tage Zeit um auf das gleiche Kosten/Zeit Verhältnis wie ein Entwickler zu kommen.

Ich denke, dass sich 150 Entwickler nicht lohnen würden, da allein die Zeit für die Kommunikation, das Einteilen der Aufgaben länger als 1 Tag dauern wird.

Man kann leider nicht genau sagen wie lange zwei, fünf oder 150 Entwickler/innen für das Projekt brauchen werden.

Es kommt auf das Projekt an.

Auf jedenfall denke ich, aber dass das Projekt schneller fertiggestellt wird, doch nicht für den gleichen Kostenaufwand.

### Aufgabe 1-3: Begriffsklärung

a) Formulieren Sie für jeden Begriff eine prägnante Erklärung in eigenen Worten und Veranschaulichen Sie ihn jeweils mit einem selbstgewählten Beispiel. Wenn es zwischen zwei Begriffen eine Beziehung gibt, erklären Sie diese ebenfalls.

**Klasse:** Abstraktes Modell für Objekte.

**Vererbung:** Mithilfe der Vererbung kann man neue Klassen auf Basis einer anderen Klasse erstellen.

**Attribut:** Wird benutzt um Eigenschaften, Zustände oder Beziehungen von Objekten zu beschreiben.

**Operation:** Beschreibt einen Schritt, wo ein Befehl ausgeführt wird, z.B. die Addition

**Methode:** Teil eines Objektes, eine Art „Algorithmus“. Das kann bei einem Auto z.B. „fahr()“ oder „brems()“ sein.

**Exemplar:** ein konkretes Objekt.

**Objekt:** Wird aus einer Klasse während der Laufzeit erstellt.

**Bibliothek:** Sammlung von Programmen, die man in seinem Code einbinden und benutzen kann.

**Spezifikation:** Sichergehen von Vorbedingung und Nachbedingung.

**Implementierung:** Erstellung einer Software.

**Verifikation:** Beweis der Korrektheit einer Software.

**Geheimnisprinzip:** Daten vor Zugriff von außen schützen. Meistens durch Datenkapselung.

b) Erklären Sie den Unterschied zwischen einer Klasse und einer Menge von Objekten.

Eine Klasse ist ein „Bauplan“ eines Objektes, eine Klasse kann z.B. ein Auto, Haus oder eine Flasche sein. Eine Klasse beinhaltet Eigenschaften wie z.B. Größe, Farbe, Breite, Baujahr usw. Ein Objekt kann aus einer Klasse erstellt werden, dieses Objekt besitzt die Eigenschaften der jeweiligen Klasse. Dem Objekt kann man weitere Eigenschaften hinzufügen, im Fall vom Haus z.B. Garage oder beim Auto z.B. Spoiler.

c) Welche in der ersten Vorlesung vom 2021-04-12 genannten softwaretechnischen Aktivitäten werden von den unter a) genannten Begriffen nicht abgedeckt? Nennen Sie zwei möglichst eindeutige Beispiele.

**Anforderungsermittlung:** Entscheidung der Priorität von Kosten, Zeit, Qualität und Aufwand.

**Management:** Kontrolle der Einhaltung von Kosten, Zeit, Qualität und Aufwand.