

Qualitätsziele

February 8, 2023

Beschreibung

Funktionale Eignung

Zu häufigen Qualitätsaspekten oder auch Qualitätsmerkmale zählt zunächst die **Funktionale Eignung**. Das ist die Voraussetzung, dass die Software die erwarteten Anforderungen erfüllt.

Zuverlässigkeit

Die **Zuverlässigkeit** ist wichtig, wenn die Leistung der Software über einen bestimmter Zeitraum entsprechend den Anforderungen erfüllt werden muss. Gerade bei einer großen Nutzerbasis wie es bei einem Flughafen der Fall ist, ist dieser Aspekt von hoher Bedeutung.

Kompatibilität

Findet eine Interaktion mit anderen Systemteilen statt, so ist wichtig, dass **Kompatibilität** herrscht. Das bedeutet, dass sie problemlos Informationen austauschen können oder auf der entsprechenden Hard- und Softwareumgebung laufen.

Benutzbarkeit

Da mit Anwender leicht mit dem Programm umgehen können und es möglichst wenig menschliche Fehler gibt, ist eine hohe **Benutzbarkeit** wichtig.

Leistungseffizienz

Damit die Anwendung keine unnötigen Kosten verursacht, ist die **Leistungseffizienz**, also der Umgang mit Ressourcen wie Speicherplatz und Bearbeitungszeit wichtig.

Sicherheit

Um Fehlverhalten aufgrund von nicht autorisierten Zugriffen zu verhindern ist die **Sicherheit** der Anwendung wichtig. Trotzdem muss der Zugang zu den Daten für diejenigen, die darauf zugreifen dürfen nicht verwehrt sein.

Wartbarkeit

Tauchen während der Anwendungszeit Fehler aufgrund einer Veränderung der Umgebung auf, sollen diese möglichst leicht entfernt werden können. Dazu ist ein gut **Wartbarkeit** wichtig. Dafür ist eine Wiederverwendbarkeit von Programmteilen hilfreich.

Übertragbarkeit

Falls sich Hard- oder Software ändert, ist es gut, wenn die Anwendung eine stabile **Übertragbarkeit** aufweist, damit die Anwendung problemlos auch auf anderen Rechnern laufen kann.

Bemerkungen

Außerdem gibt es noch die *Qualitätsziele*, die ein angestrebte Qualitätsaspekte darstellen, die in der Anwendung relevant sind. Das kann zum Beispiel "Robustheit gegen hohe Netzlast" sein. Häufig finden Wechselwirkungen zwischen verschiedenen Qualitätsaspekten statt, sodass sie sich gegenseitig ausschließen. Dadurch können nie alle Qualitätsziele vollständig erfüllt werden (Zeit- vs. Verbrauchverhalten, Sicherheit vs. Benutzbarkeit, Leistungseffizienz vs. Wartbarkeit, ...). In diesen Fällen muss überprüft werden, welches Ziel die höhere Priorität hat.

Die Gesamtheit aller Qualitätsziele bildet die *Qualitätsanforderung*. Um diese überprüfen zu können müssen alle Qualitätsziele konkret festgehalten werden. Teilweise kann der Erfolg eines Qualitätsaspektes in *Qualitätsmetriken* gemessen werden diese können während der Laufzeit gemessen werden (Transaktionen pro Stunde, Anzahl paralleler Benutzer, MTBF (mean time between failures), LOC (lines of code), Anzahl der Accounts mit bestimmter Zugriffsberechtigung, ...) oder durch Analyse des Codes festgestellt werden (Anzahl der Schnittstellen, Anzahl an Geschäftsprozessen, ...). Es ist wichtig, das bei der Planung der Metriken festgehalten wird, wie genau gemessen wird, da gerade bei statischen Analysen die zum Beispiel die Zählweise unterscheiden kann.

Ein Beispiel für eine feste Metrik für die Metrik der Komplexität ist die McCabe-Metrik. Sie zählt die Anzahl der if-Anweisung, der Schleifen und der Anzahl der Zweige.

In einem *Qualitätsmodell* kann der Kunde zunächst abstrakte Qualitätsziele definieren, diese auf seinen angewendet konkretisieren und schließlich messbare Indikatoren und Metriken finden. Ein solches Modell sollte sich im Lastenheft wiederfinden. Dabei kann es helfen, *Qualitätsszenarien* zu entwickeln, die festlegen wie das System in einer bestimmten Situation sich zu verhalten hat. Diese Szenarien lassen sich in drei Typen unterteilen: Verwendungsszenarien im Fall der Standardverwendung des Systems, Änderungsszenarien, also Abweichungen von der Standardanwendung, und Stressszenarien, im Fall von unerwarteten Veränderungen.

Zusätzlich gibt es, nachdem die funktionalen Anforderungen und Qualitätsanforderungen geklärt sind, noch die *constrains*. Sie sind weitere Bedingungen des Kunden, beispielsweise an das Format von Daten oder das Einhalten der DSGVO.

Qualitätsmerkmale nach ISO/IEC 25010

Funktionale Eignung

- Funktionale Vollständigkeit
- Funktionale Korrektheit
- Funktionale Angemessenheit

Leistungseffizienz

- Zeitverhalten
- Verbrauchsverhalten
- Kapazität

Zuverlässigkeit

- Reife
- Verfügbarkeit
- Fehlertoleranz
- Wiederherstellbarkeit

Sicherheit

- Vertraulichkeit
- Integrität
- Nachweisbarkeit
- Verantwortlichkeit
- Authentizität

Kompatibilität

- Koexistenz
- Interoperabilität

Wartbarkeit

- Modularität
- Wiederverwendbarkeit
- Analysierbarkeit
- Änderbarkeit
- Testbarkeit

Benutzbarkeit

- Verständlichkeit
- Erlernbarkeit
- Bedienbarkeit
- Schutz vor Fehlbedienung
- Attraktivität der Nutzungsoberfläche
- Barrierefreiheit

Übertragbarkeit

- Anpassbarkeit
- Installierbarkeit
- Austauschbarkeit