Waru mist die Kommunikation von Softwarearchitekturen so wichtig?

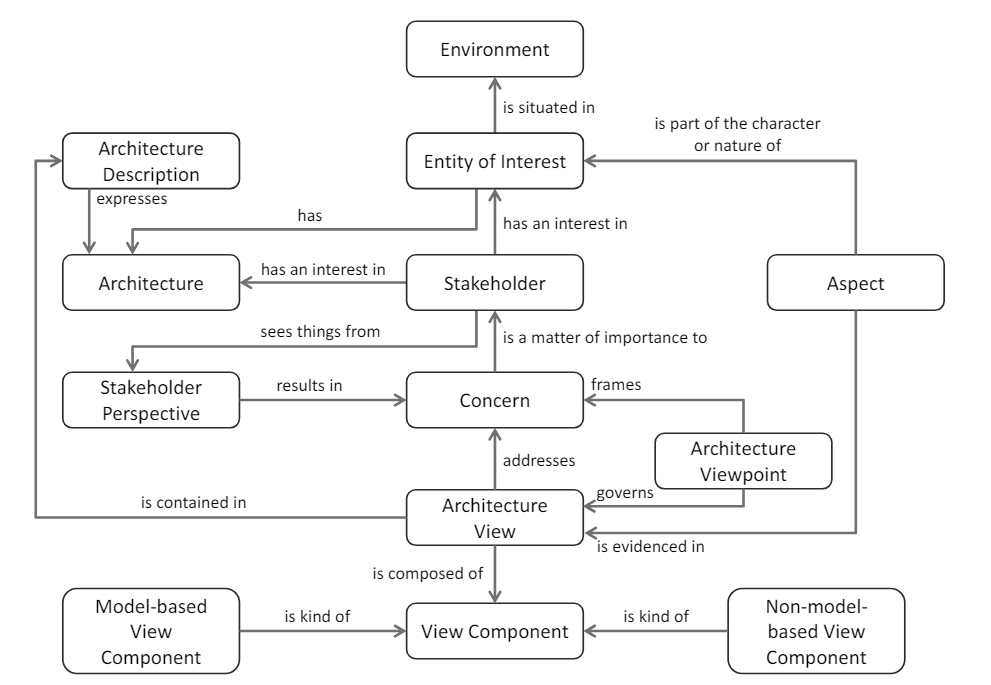
* Verstandnis und Zusammenarbeit
* Einheitliche Vision
* Risikominimierung
* Bessere Entscheidungsfindung
* Kunden- und Stakeholder-Engagement
* Dokumentation und Nachvollziehbarkeit

Arten der Kommunikation Softwarearchitekturen

* Dokumentation
* Architekturdiagramme
* Prasentationen
* Prototypen und Proof-of-Concepts
* Code und Komentare
* Workshops und Brainstorming-Sitzungen

ISO/IEC/IEEE 42010-2022

* “Systems and Software Engineering – Architectural Description”
* Norm, die sich mit der Architekturbeschreibung von Systemen befasst
* Legt Grundsatze und Best Practices fest fur Architekturbeschreibungen
* Defieniert Schlusselkonzepte
* Bietet Richtlinien fur die Auswahl und Anpassung von Architekturansatzen und -methoden



**Stakeholder**

= role, position, individual or organization having a right, share, claim or other interest in an architecture entity or its architecture that reflects their need and expectations

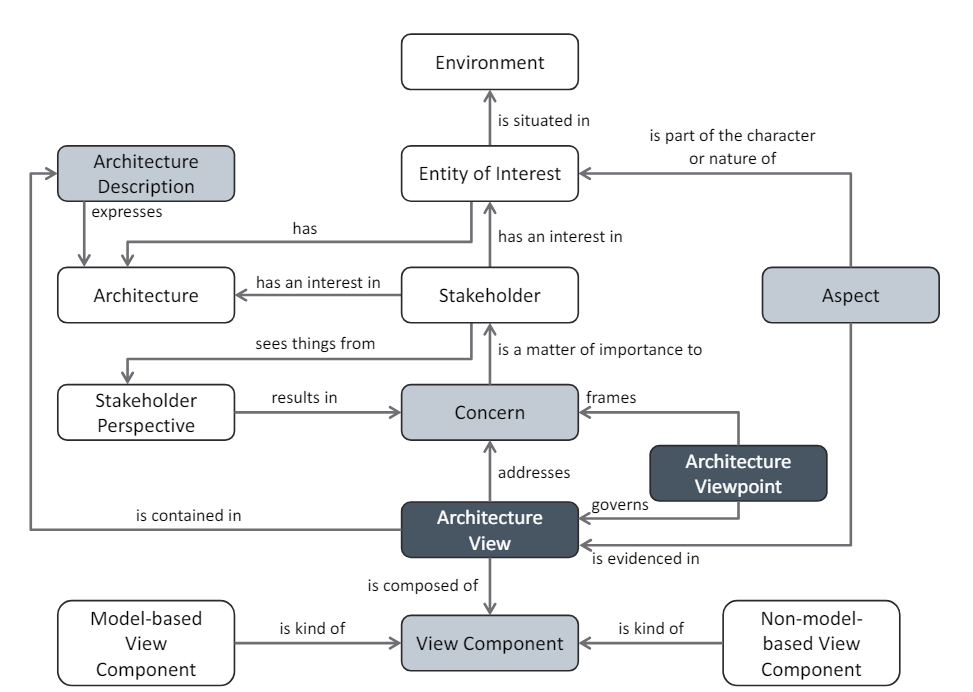
A diagram of a software company

Description automatically generated

A diagram of a company

Description automatically generated

**Views und Viewpoints**

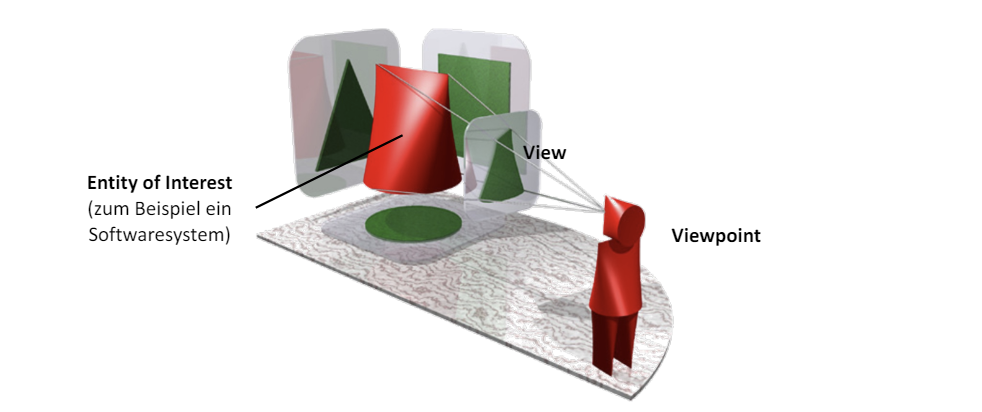


Concern = ein Intersesse, das sich auf das System, seine Entwicklung oder seinen Betrieb bezieht

Aspects = bestimmte Perspektiven oder Blickwinkel, unter denen das System analysiert und beschrieben wird

Viewpoint = defienerter Satz von Konventionen und Regeln zur Darstellung und Beschreibung bestimmter Concerns eines Systems

Architecture View = eine Darstellung eines Systems aus der Perspektive eines bestimmten Architecture Viewpoints



**Sichtenmodelle**

= Strukturierte Darstellungen eines Systems, die verschiedene Aspekte der Architektur aus unterschiedlichen Perspektiven oder Sichten zeigen

**4+1 Sichtenmodell nach Philippe Kruchten**

**A diagram of a system

Description automatically generated**

**Siemens’ Four View Model**

* Conceptual View/ Konzeptionelle Sicht
* Module View/ Modul- oder logische Sicht
* Execution View/ Laufzeitsicht
* Physical View/ Physikalische Sicht

**Sichtenmodell nach arc42**

* Kontextabgrenzung
* Bausteinsicht
* Laufzeitsicht
* Verteilungssicht

Ebenen in arc 42:

* Makroarchitektur
* Mittlere Ebene
* Microarchitektur

**UML – Unified Modeling Language**

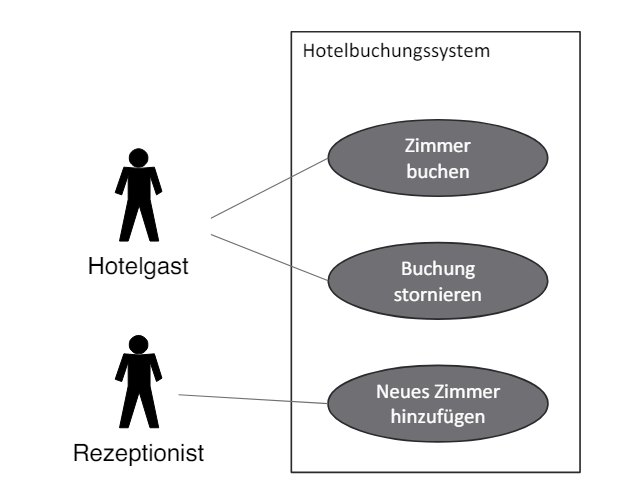
Mit der UML steht eine graphische Modellierungssprache zur Verfugung, welche zur Verfugung, welche zur Dokumentation von Softwaresystem verwendet warden kann

Arten von Diagramme:

* Strukturdiagramme
  + Klassendiagramm, Komponentendiagramm, Verteilungsdiagramm, Kompositionsstrukturdiagramm, Paketendiagramm, Objektdiagramm, Profildiagramm
* Verhaltensdiagramme
  + Aktivitatsdiagram, Anwendungsfalldiagramm, Interaktionsubersichtdiagramm, Kommunikationsdiagramm, Sequenzdiagramm, Zeitverlaufsdiagramm, Zustandsdiagramm

**UML Anwendungsfalldiagramm**

* Modellierung von funktionalen Anforderungen eines Systems aus Anwender-Perspektive

****

**UML Klassendiagramm**

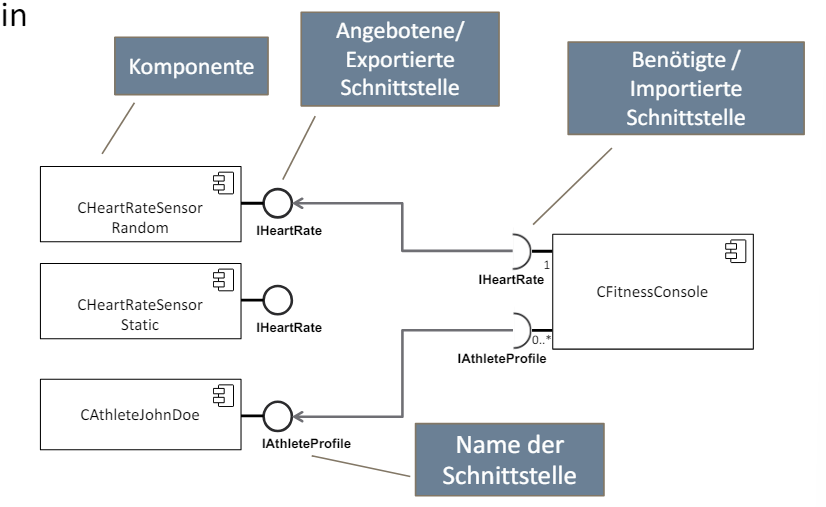
* Grafische Darstellung von Klassen, Schnittstellen, Attributen und Methoden sowie deren Beziehungen zueinander
* A diagram of a company

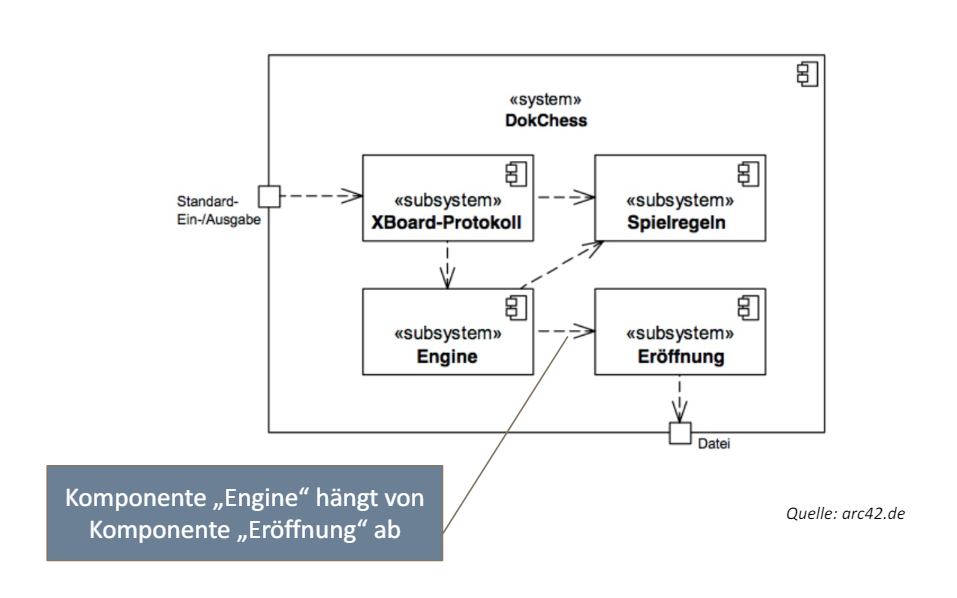
  Description automatically generated
* A diagram of a black and white diagram

  Description automatically generated

**UML Komponentendiagramm**

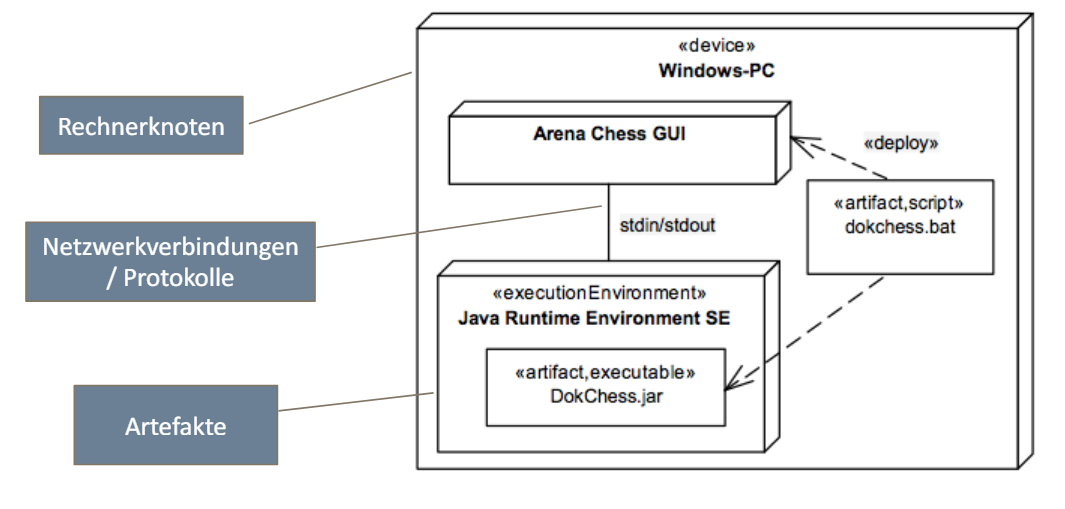
* Graphische modellierung von Komponenten und Subkomponente sowie deren Verdrahtung



****

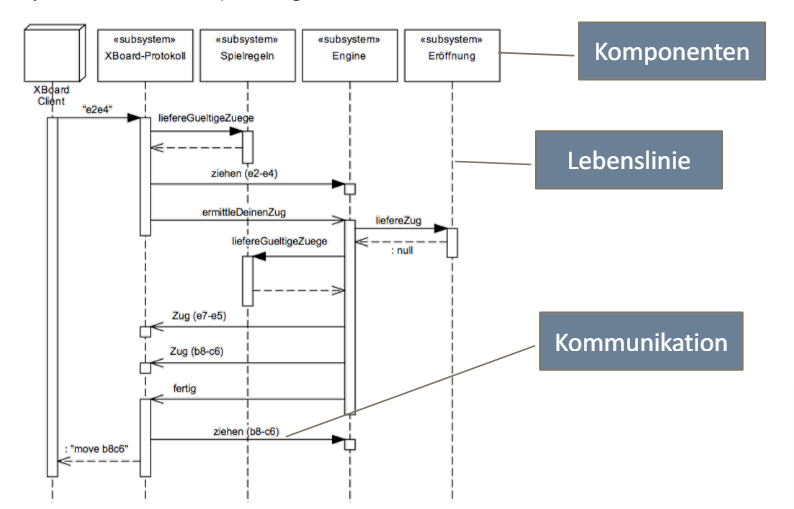
**UML Verteilungsdiagramm**

* Graphische Modellierung der Verteilung von Komponenten auf Rechnerknoten

****

**UML Sequenzdiagramm**

* Graphische Modellierung zur Beschreibung des Austauschs von Nachrichten zwischen Objekten

****

**UML Aktivitatsdiagramm**

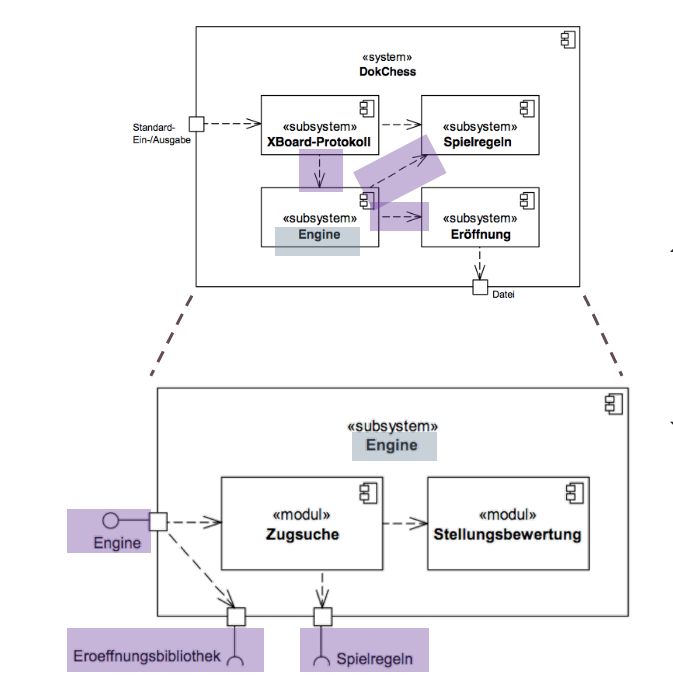
* Stellt die Vernetzung von Aktionen und deren Verbindungen mit Kontroll- und Datenflussen graphisch dar

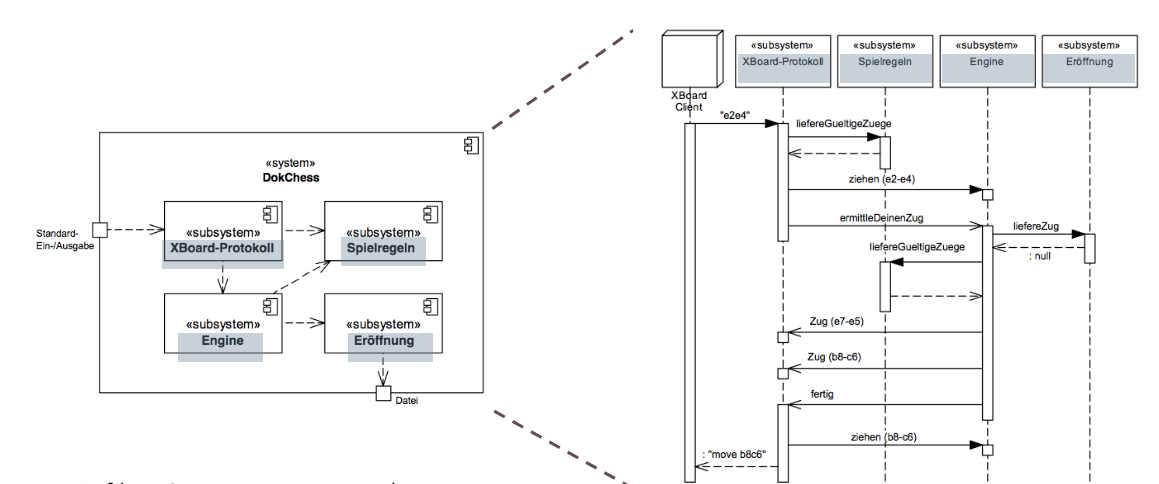
**A diagram of a company

Description automatically generated**

**Konsistenz zwischen Diagrammen**

* Auf konsistente Bennenung der Elemente zwischen der Elemente zwischen Diagrammen achten
* Auf Konsistenz der Schnittstellen achten





**Dokumentation von Architekturentscheidungen**

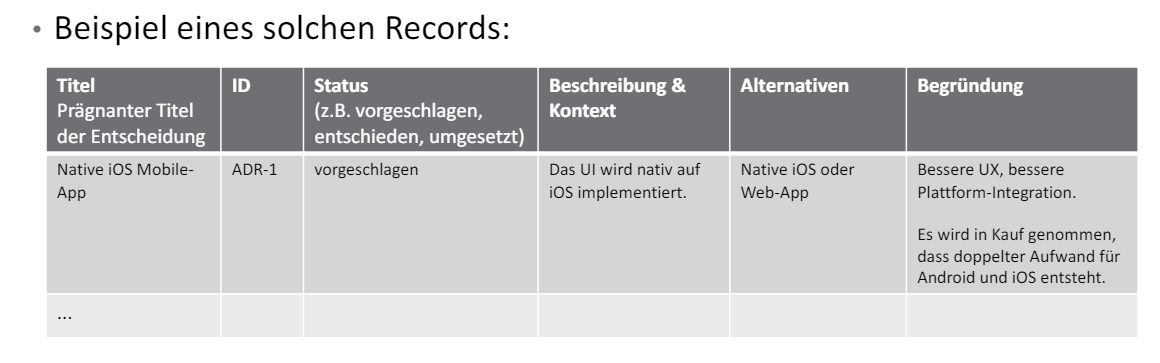
**Architekturentscheidungen ->** werdeninsbesondere Ist- und Sollzustand einer Architektur dokumentiert

Hauptarbeit eines Architekten ist es, Architektur-relevante Entscheidungen zu treffen

Architekturentscheidungen zu dokumentieren hilft dabei:

* Entscheidungen Team-ubergreifend sichtbar zu machen
* Entscheidungen im Team und im Unternehmen durchzusetzen
* Die Qualitat von Arhitekturentscheidungen zu messen
* Die Entstehung einer kunftigen Ist-Architektur nachzuvollziehen
* Dem Team Guidelines fur detaillierte Entscheidungen zu geben

**Architectural Decision Records(ADRs) =** Bewahrte Moglichkeit, Architekturentscheidungen zu dokumentieren

****

**Allgemeine Aspekte zur Dokumentation**

Weitere ubliche Dokumenttypen:

* Zentrale Architekturbeschreibung (arc42)
* Architekturuberclick
* Dokumentubersicht
* Ubersichtsprasentation
* Architekturttapete
* Handbuch zur Dokumentation
* Technische Informationen
* Dokumentation von externen Schnittstellen

Praxisregeln zur Dokumentation:

* Schreiben aus Sicht des Lesers
* Unnotige Wiederholungen vermeiden
* Mehrdeutigkeit vermeiden
* Standardisierte Organisationsstruktur
* Begrunden Sie wesentliche Entscheidungen schriftlich
* Uberprufung der Gebrauchstaulichkeit
* Ubersichtliche Diagramme
* Regelmassige Aktualisierung
* Eine Architekturbeschreibung sollte so einfach wie moglich sein, aber so umfangreich wie notwendig
* “You know you’ve achieved perfection in design, not when you have nothing more to add but when you have nothing more to take away”