**Technische Dokumentation**

für

**<Projekt>**

Version <X.X>

verfasst von

|  |  |  |
| --- | --- | --- |
| <Name> |  | <Email> |
| <Name> |  | <Email> |
| <Name> |  | <Email> |
| <Name> |  | <Email> |
| <Name> |  | <Email> |

|  |  |
| --- | --- |
| Dozent: |  |
| Lehrveranstaltung: |  |
|  |  |
| Datum: |  |
|  |  |

Inhaltsverzeichnis

[Inhaltsverzeichnis iii](#_Toc434394800)

[Historie der Dokumentversionen iv](#_Toc434394801)

[1 Einführung 1](#_Toc434394802)

[2 Funktionalität des Systems 3](#_Toc434394803)

[3 Qualitätsanforderungen 5](#_Toc434394804)

[4 Architektur 7](#_Toc434394805)

[5 Komponenten (Statische Sicht) 9](#_Toc434394806)

[5.1 Komponenten 9](#_Toc434394807)

[5.1.1 Komponente <Name> 9](#_Toc434394808)

[5.2 Schnittstellen 9](#_Toc434394809)

[5.2.1 Schnittstelle<Name> 9](#_Toc434394810)

[5.3 Regeln 10](#_Toc434394811)

[6 Kommunikation (Dynamische Sicht) 11](#_Toc434394812)

[7 Technologie 14](#_Toc434394813)

[Quellen XV](#_Toc434394814)

Historie der Dokumentversionen

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Autor (en)** | **Änderungsgrund/Bemerkungen** |
| *0.1* | *00/00/00* | *<Hauptautor genügt>* | *<Informationen über die Revision (z.B. Ersterstellung, Aktualisierung, bereit zur Abnahme, abgenommen, …)>* |
|  |  |  |  |
|  |  |  |  |

<In dieser Vorlage finden Sie Texte, die von den "<>" Symbolen beschränkt sind. Dieser Text (in Kursivschrift) soll Sie durch die Vorlage führen und Erklärungen zu den verschiedenen Abschnitten in diesem Dokument liefern.

Bitte stellen Sie sicher, dass Sie alle Kommentare vor dem Absenden/Einreichen des Dokuments löschen.>

# Einführung

<Was ist die Zielaufgabe des Projektes? Konzeptuelle Sicht - sehr knapp nur zum Verständnis >

# Funktionalität des Systems

<Beschreibung der Funktionalität des Systems im Überblick, z. B. mit Hilfe eines Use-Case-Diagramms>

# Qualitätsanforderungen

<Beschreibung der Qualitätsanforderungen an das System in Form von Szenarien. (Referenz auf gegebenes Material möglich) Diese Szenarien sollen durch entsprechende Taktiken in der Architektur umgesetzt werden. >

# Architektur

<Struktur/Architektur als Überblick dokumentiert (Logische Sicht, Konzeptuelle Sicht), dabei werden die wichtigen Architekturentscheidungen begründet:

Klassifikation der Architektur nach einem Architekturstil/Architekturmuster. Begründung der Wahl des Stils/Musters (Stärken/Schwächen) Welche Szenarien werden durch die Architektur (Stil/Muster) unterstützt? (ggf. auch Zuordnung der Szenarien zu Taktiken)>

# Komponenten (Statische Sicht)

<Modul-/Implementierungssicht, Physikalische Sicht >

## Komponenten

<Überblick durch Komponentendiagramm mit Schnittstellen und Bezug zu einer Mehrschichtenarchitektur >

### Komponente <Name>

<Die Komponente und ihre Hauptfunktionen sollten klar werden, dabei werden die wichtigen Designentscheidungen begründet (Diskussion Stärken/Schwächen). Ggf. werden Modifikationen gegenüber dem gewählten Architekturstil/Architekturmuster angegeben

Für die jeweiligen Komponenten bietet ggf. ein Klassendiagramm Übersicht.

Bitte bei verwendeten (Entwurfs-)Mustern, diese beschreiben und deren Verwendung begründen, d.h. hier mehr ins Detail gehen.>

## Schnittstellen

### Schnittstelle<Name>

**<** Schnittstellen soll sehr detailliert beschrieben werden:

* Funktionsaufruf und Parameterleiste
  + Namen und Zweck der Funktionen/Methoden
  + Name, Bedeutung und Wertebereiche von Übergabe- und Rückgabeparametern
  + Verhalten der Komponente bei Nutzung einer Methode. Dieses Verhalten kann sich nur über eine Schnittstellen der Komponente zeigen.
* Ausnahmebehandlung,
* Übertragungsrichtung  
  ggf. Unterscheidung Kontrollfluß und Datenfluß
* ggf. Synchronisationsmechanismus
* wichtige Designentscheidungen begründen (Diskussion Stärken/Schwächen)

Schnittstellen können durch eine Schnittstellenspezifikation beschrieben werden, z. B. gemäß IDL von Corba:

interface Name

{ attribute Typ Attribut-Name;

...

void Operation()...;

...

};

Signatur einer Operation:

[oneway] <Ergebnistyp> <Operationsname>   
 (in Typ1 Parameter1,   
 out Typ2 Parameter2,   
 inout Typ3 Parameter3, ...)

[raises (Ausnahme1, ..., AusnahmeM)]

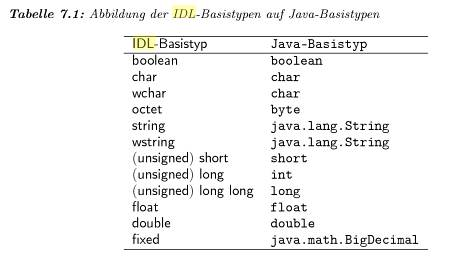
[context (Kontextname1,..., KontextnameK)]

Komplexe Typen als struct definiert werden

struct Typname{

Typ Attribut-Name;

};



(Quelle : Oliver Haase, Kommunikation in verteilten Anwendungen: Einführung in Sockets, Java RMI, CORBA und Jini, Oldenbourg Wissenschaftsverlag, 2008)

>

## Regeln

<Welche Regeln/ generellen Prinzipien gelten für die Kommunikation/Interaktion der Komponenten? Regeln auflisten, begründen, indem Sie auf Qualitätsanforderungen bezogen werden.>

# Kommunikation (Dynamische Sicht)

< Beschreiben Sie in zwei Sequenzdiagrammen interessante Abläufe der Funktionalität des Systems>

# Technologie

< Beschreiben Sie die gewählten Technologien, so dass klar ist, wie die Komponenten und deren Interaktion umgesetzt wurde, z. B. Programmiersprachen, Kommunikationsprotokolle, Synchronisation, Verteilung auf Hardware etc. Ggf Verteilungsdiagramm

Bewerten Sie, wie sich die Technologie für die Architektur eignet>

Quellen