

# Wochenbericht III - Praktikum "Verteilte Systeme": Überarbeitung der arc42-Kapitel 1 bis 6

Manh-An David Dao

May 11, 2025

## 1 Mitglieder des Projektes

Mitglied des Projektes	Aufgabe	Fortschritt	Zeiteinsatz	Check
Manh-An David Dao	arc42: Ergänzung Kapitel 2.1 & überarbeitung Kapitel 6	80%	2.5h	ok
Jannik Schön	arc42: Überarbeitung Kapitel 5	70%	1h	ok
Marc Siekmann	repo cleanup, arc42 Korrektur Kapitel 1, Anlegen arc42 Middleware	80%	2h	ok
Phillip Patt	arc42: Überarbeitung Kapitel 4, Roboterarm initialisieren & Organisatorisches	70%	4h	nok

## 2 Zusammenfassung der Woche

Neustrukturierung der Gruppenarbeit mit Korrektur. Neue Erkenntnisse bzgl. Middleware und Grundstruktur des Systems.

In dieser Woche fand die Bearbeitung

Wesentliche Pull request/commits des Projektes waren:

## 3 Bearbeitete Themen und Schlüssel Erkenntnisse

### 3.1 arc42 Middleware, Cleanup, Korrektur Kapitel 1

Nach der vergangenen Vorlesung haben wir uns entschieden eine Middleware und eine Applikation jeweils als eigenständiges Produkt anzufertigen. Dies hat den Vorteil, dass Entscheidungen bzgl. Kommunikation, Security etc. gekapselt werden (Skript S.72). So kann die eigentliche Applikation ebenfalls unabhängig entwickelt werden und muss sich nur an dem gegebenen Interface der Middleware richten. Das Gesamtsystem ist demnach ein 2-Schichten-System. Daher wurde ein weiteres arc42 Dokument eingerichtet, in der eine eigene Architektur entwickelt wird. Dazu wurde eine Grundlage mithilfe Note-bookML für die ersten drei Kapitel geschaffen.

### 3.2 arc42 Kapitel 2.1 und 6

In dieser Woche wurde das Kapitel 2.1 (technische Randbedingung) überarbeitet und um eine Skizze ergänzt.<sup>1</sup> Die Skizze dient der Visualisierung des Systems und stellt insbesondere jene Komponenten dar, die für uns fest vorgegeben sind und auf die wir keinen Einfluss nehmen können. Ziel war es, ein gemeinsames Verständnis darüber zu schaffen, welche Systemteile als unveränderlich zu betrachten sind und wo unser tatsächlicher Handlungsspielraum liegt.

Im Rahmen von Kapitel 6 (Laufzeitsicht) entschieden wir uns, ein im Skript (vgl. S.101) erwähntes Verhaltensmuster mit einem Sequenzdiagramm darzustellen. Das Diagramm wurde überarbeitet, um die Interaktionen zwischen den Komponenten verständlicher und visuell präziser abzubilden.<sup>2</sup>

### 3.3 arc42 Kapitel 5

Kapitel 5 wurde überarbeitet. So wurde die Bausteinsicht der aktuellen Erkenntnisse von Kapitel 5 aus der Vorlesung angepasst. Dabei wurde darauf geachtet, dass die Bausteinsicht der Logik einer „Three-Layer Architektur“ (s. VS-Skript S. 68) erfüllt. Die Implementierung hilft in der Zukunft bei der Wartung, dem Testen und der Übersichtlichkeit des Quellcodes. Auch sind die verschiedenen Schichten austauschbar.

### 3.4 arc42 Kapitel 4 & Roboterarm initialisieren

Um die Kritik der vorigen Woche bzgl. Zeitmanagement zu verarbeiten wurde ein Issueboard<sup>3</sup> auf Gitlab eingerichtet. Des Weiteren wurden die nächsten Schritte als Issues formuliert um bessere asynchrone Arbeit zu ermöglichen.

Anhand der Rückmeldungen aus der Vorlesung und den Besprechungen mit dem Professor wurde die arc42-Dokumentation Kapitel 4 - Lösungsstrategien überarbeitet. Es wurde ein erster Versuch unternommen das System funktional zu zerlegen, der MR<sup>4</sup> ist noch ein Draft und dient als Gesprächsgrundlage für das weitere vorgehen.

Das exemplarische Ansprechen der Roboter ist nicht weiter fortgeschritten.

## 4 Fachlicher Bezug

Die Überarbeitung der arc42-Dokumentation erfolgte auf Grundlage der Rückmeldungen aus den Besprechungen mit dem Professor sowie unter Berücksichtigung der Inhalte des begleitenden Skripts. Dabei wurde auf bestehende Kenntnisse aus dem Studium und der Vorlesung zurückgegriffen.

- **Verteilte Systeme – Skriptbezug**

Die Verwendung des Sequenzdiagramms geht auf einen Hinweis im Skript zu Verteilten Systemen zurück.<sup>5</sup>

- **arc42 – technische Randbedingung**

Die Ergänzung der technischen Randbedingung basiert auf einer methodischen Herangehensweise, wie sie im vorherigen Semester im Fach Software Engineering vermittelt wurde.

- **arc42 – Laufzeitsicht**

Die Zuordnung des Sequenzdiagramms basiert auf arc42-Dokumentation (Kapitel 6).<sup>6</sup>

---

<sup>1</sup>[https://git.haw-hamburg.de/infwgi246/vs\\_praktikum-2025-rose/-/merge\\_requests/12](https://git.haw-hamburg.de/infwgi246/vs_praktikum-2025-rose/-/merge_requests/12)

<sup>2</sup>[https://git.haw-hamburg.de/infwgi246/vs\\_praktikum-2025-rose/-/merge\\_requests/13](https://git.haw-hamburg.de/infwgi246/vs_praktikum-2025-rose/-/merge_requests/13)

<sup>3</sup>[https://git.haw-hamburg.de/infwgi246/vs\\_praktikum-2025-rose/-/boards](https://git.haw-hamburg.de/infwgi246/vs_praktikum-2025-rose/-/boards)

<sup>4</sup>[https://git.haw-hamburg.de/infwgi246/vs\\_praktikum-2025-rose/-/merge\\_requests/9](https://git.haw-hamburg.de/infwgi246/vs_praktikum-2025-rose/-/merge_requests/9)

<sup>5</sup>[https://github.com/scimbe/vs\\_script/blob/main/vs\\_script-first-v01.pdf](https://github.com/scimbe/vs_script/blob/main/vs_script-first-v01.pdf) S. 101

<sup>6</sup>[https://www.dokchess.de/06\\_laufzeitsicht/01\\_zugermittlung/](https://www.dokchess.de/06_laufzeitsicht/01_zugermittlung/)

## 5 Zeitaufteilung

- **Marc Siekmann**

Einarbeitung NotebookML: 20 Minuten

Repository Cleanup: 20 Minuten

arc42 Korrektur: 60 Minuten

Anlegen arc42 Middleware: 20 Minuten

Organisation, Rücksprache mit Prof, technische Probleme, Sonstiges: ca. 2h

Reviews: 30 Minuten

- **Manh-An David Dao**

Ergänzung Kapitel 2.1: 30 Minuten

Überarbeitung Kapitel 6: 60 Minuten

Reviews und Wochenbericht: 60 Minuten

- **Phillip Patt**

Organisatorisches: 120 Minuten

Überarbeitung Kapitel 4: 90 Minuten

Roboterarm initialisieren: 0 Minuten

Reviews und Wochenbericht: Zeitaufwand 30 minuten

## 6 Nächste Schritte

- Architektur Middleware und Applikation
- Ansteuerung Roboter mit ITS-Board