

---

# **Software Engineering 2**

## **Projektdokumentation**

**Projekt:  
Entwicklung eines Software-Systems  
zur Simulation der Steuerung eines Fahrstuhls**

---

# Inhaltsverzeichnis

1	Einführung	4
<b>I</b>	<b>Benutzerhandbuch</b>	<b>6</b>
1	Systemvoraussetzungen	8
2	Installation	9
3	Anwendung	10
<b>II</b>	<b>Entwicklerhandbuch</b>	<b>11</b>
1	Übersicht	13
2	Performance	14
2.1	Performanceoptimierung im Algorithmus . . . . .	14
2.2	Performanceoptimierung in der Benutzeroberfläche . . . . .	14
3	Klassendiagramme	15
4	API	16
<b>III</b>	<b> Projektdokumentation</b>	<b>17</b>

1	Anforderungs- und Problemanalyse	19
1.1	Methoden . . . . .	19
1.2	Diagramme . . . . .	20
2	Software-Entwurf	22
3	Qualitätssicherung	23
4	Software-Test	24
5	Team-Organisation	25
5.1	Gruppenarbeit. . . . .	25
5.2	Verwendete Werkzeuge . . . . .	25
5.3	Resümee. . . . .	25
6	Glossar	26

# 1. Einführung

Im Rahmen der Belegarbeit im Modul SOFTWARE ENGINEERING 2 ist ein Software-System zu implementieren, welches die Steuerung eines Fahrstuhls simuliert. Dieses Software-System soll in der Zukunft als Anschauungsmaterial im Lehrbetrieb verwendet werden. Studierenden soll damit ermöglicht werden, die Zusammenhänge zwischen real existierenden Automaten und der Thematik der Zustandsdiagramme zu erfahren.

Die Dokumentation des Projektes gliedert sich in folgende Teildokumentationen:

## **Benutzerhandbuch**

Im Benutzerhandbuch werden Anweisungen für die korrekte Verwendung der Software gegeben. Sie kann Mitarbeitern oder Studierenden zur Verfügung gestellt werden, welche die Anwendung verwenden möchten. Neben den Hinweisen zur Verwendung sind die Systemvoraussetzungen sowie Installationsanweisungen enthalten.

## **Entwicklerhandbuch**

Um Eine Weiterentwicklung der Anwendung zu ermöglichen werden im Entwicklerhandbuch die internen Zusammenhänge und Strukturen dokumentiert. Enthalten sind die Klassendiagramme, sowie die Auflistung der Funktionen der API.

## Projektdokumentation

Dieser Teil der Dokumentation wird sich mit Organisation der Projektarbeit beschäftigen. Analysiert werden die Herangehensweisen, verwendete Werkzeuge und verschiedene Entscheidungen die während der Projektarbeit getroffen worden. Ziel ist es die Zusammenarbeit und die Projektrealisierung zu reflektieren und entsprechende Schlüsse zu ziehen.

Ebenfalls behandelt dieser Abschnitt Themen der Analyse, der Qualitätssicherung und des Software-Test.

Die einzelnen Teile der Dokumentationen sind in sich abgeschlossen und können unabhängig voneinander verwendet werden.

## Konventionen

Folgende Konventionen werden im Dokument verwendet:

Quellcode in Monospace

Innerhalb des Quellcodes camelNotation u.s.w.

Ausformulieren  
der Kon-  
ventionen,  
welche  
weiteren  
Konventio-  
nen sind  
sinnvoll.

**Teil I.**

**Benutzerhandbuch**



# **1. Systemvoraussetzungen**



## **2. Installation**

### **3. Anwendung**

**Teil II.**

**Entwicklerhandbuch**



# **1. Übersicht**

## 2. Performance

### 2.1. Performanceoptimierung im Algorithmus

### 2.2. Performanceoptimierung in der Benutzeroberfläche

hier muss erklärt werden wie der Algorithmus mit den Zählern funktioniert und ggf. warum das  $O(1)$  ist

hier muss erklärt werden wie ReactJS funktioniert und warum es die Performance verbessert

### **3. Klassendiagramme**

## 4. API



**Teil III.**

# **Projektdokumentation**

kleine Einleitung

Seitennummerierung zurücksetzen!

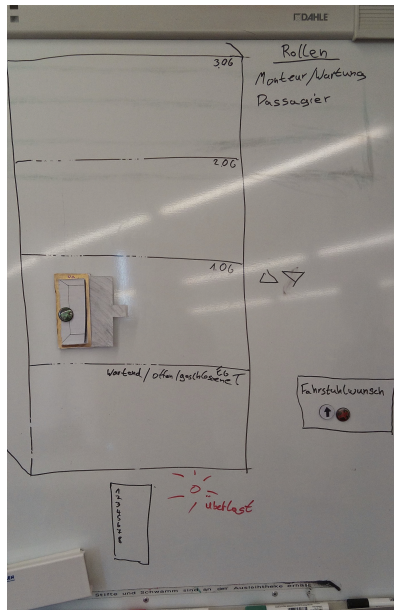
# 1. Anforderungs- und Problemanalyse

## 1.1. Methoden

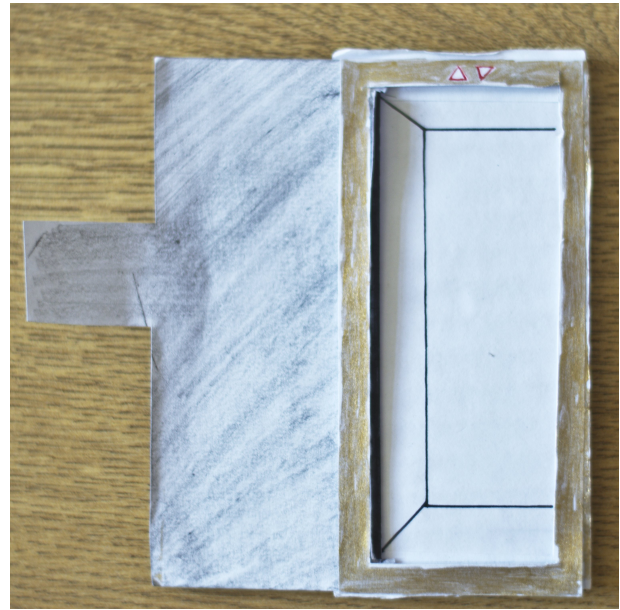
Aufgabe der Anforderungsanalyse in diesem Projekt war es herauszufinden welches Problem der Kunde, in unserem konkreten Fall unsere Professorin Frau Hauptmann mit der zu entwickelnden Software lösen möchte. Dafür wurden Interviews mit dem Kunden durchgeführt und entsprechende Ergebnisse mit Hilfe von Audioaufzeichnung, Mitschrift und Fotografien protokolliert. Zur detaillierten Beschreibung einzelner Abläufe des Systems wurden Kreativtechniken wie das Zeichnen verschiedener Szenarien an einem Whiteboard sowie die manuelle Simulation des Fahrstuhles mit einem aus Pappe gefertigten Modells durchgeführt. Grundlegende Fragen die im Laufe der Analyse geklärt werden mussten waren

- Wie viele Fahrstühle sollen verwendet werden können?
- Wie soll das Gebäude beschaffen sein?
- Welcher Algorithmus soll verwendet werden?
- Gibt es Schnittstellen zu anderen Systemen?

Weiterhin musste festgelegt werden ob die Priorität des Systems auf der Simulation oder auf einer möglichst realitätsnahen Umsetzung eines Liftes liegt. Im Laufe der Analyse und Modellierung entsprechender Anwendungsfälle wurde ersichtlich, dass das System sich aus zwei Teilsystemen, der **Fahrstuhlsteuerung** und der **Fahrstuhlsimulation** zusammensetzt, deren Anforderungen getrennt voneinander beschrieben werden mussten.



(a) Skizze der Fahrstuhlsimulation am Whiteboard



(b) Modell des Fahrstuhles

**Abbildung 1.1.:** Kreativtechniken zur Anforderungsanalyse

Eine Besonderheit des Systems ist die Umgebung in der es eingesetzt werden soll, der Lehrbetrieb an einer Hochschule. Daraus ergaben sich spezielle Anforderungen wie das Anzeigen der Zustandsübergänge die gesondert betrachtet werden mussten.

## 1.2. Diagramme

### Zustandsdiagramm

Die wesentliche Funktionalität sowie der Algorithmus des Systems lassen sich in dem Zustandsdiagramm abbilden. Um das bestmögliche Ergebnis zu erhalten haben wir verschiedene Versionen des Diagramms entworfen und diese in Gruppentreffen diskutiert und überarbeitet.



Zustandsdiagramm ggf. mehrere Versionen

## 2. Software-Entwurf

Für die technische Umsetzung der Zustände in ausführbaren Quellcode ergaben sich folgende Entwurfsmuster: \_\_\_\_\_

### **State Design Pattern**

Vorteile, Nachteile, warum haben wir uns dagegen entschieden?

### **Zustandstabelle**

### **Event Methode**

ist das eine korrekte Bezeichnung? Falls nein, wie heißt das bei uns?

Wir müssen aufpassen, dass wir eine klare Trennung zwischen Entwurfs-spezifischen Inhalten im Entwickler-handbuch und hier vornehmen.

### **3. Qualitätssicherung**

## **4. Software-Test**



## 5. Team-Organisation

### 5.1. Gruppenarbeit

Um das Zusammenarbeiten in der Gruppe einfacher zu gestalten wurden verschiedene Technologien eingesetzt. Zu finden von Terminen für Meetings wurde Doodle<sup>1</sup> verwendet. Für die Kommunikation in der Gruppe Slack<sup>2</sup>.

Hier muss  
hin warum  
wir uns für  
JS und Co  
entschieden  
haben...

### 5.2. Verwendete Werkzeuge

### 5.3. Resümee

---

1 [www.doodle.com](http://www.doodle.com)

2 <https://slack.com/>

## **6. Glossar**