

# L<sup>A</sup>T<sub>E</sub>X-Vorlage für diverse Ausarbeitungen .. oder so ähnlich

## PROGRAMMENTWURF

der Vorlesung „Advanced Software Engineering“

an der

Dualen Hochschule Baden-Württemberg Karlsruhe

von

**Max Mustermann**

Abgabedatum 1. April 2090

Matrikelnummer

4711

Kurs

tinfl17b3

Bearbeitungszeitrum

5. & 6. Semester

Gutachter der Studienakademie

Mirko Dostmann

# Inhaltsverzeichnis

<b>Inhaltsverzeichnis</b>	<b>ii</b>
<b>Abbildungsverzeichnis</b>	<b>iii</b>
<b>Codeverzeichnis</b>	<b>iv</b>
<b>Abkürzungsverzeichnis</b>	<b>v</b>
<hr/>	
<b>1 Domain Driven Design</b>	<b>1</b>
1.1 Analyse der Ubiquitous Language . . . . .	1
1.2 Analyse und Begründung der verwendeten Muster . . . . .	1
<b>2 Clean Architecture</b>	<b>2</b>
2.1 Schichtenarchitektur . . . . .	2
<b>3 Programming Principles</b>	<b>3</b>
3.1 SOLID . . . . .	3
3.2 GRASP (insb. Kopplung/Kohäsion) . . . . .	3
3.3 DRY . . . . .	3
<b>4 Refactoring</b>	<b>4</b>
4.1 Identifizieren von Codesmells . . . . .	4
<b>5 Entwurfsmuster</b>	<b>5</b>
5.1 Begründung des Einsatzes . . . . .	5
5.2 Unified Modeling Language (UML) Vorher . . . . .	5
5.3 UML Nachher . . . . .	5
<hr/>	
<b>A Anhang</b>	<b>I</b>
A.1 Löwenmann . . . . .	I
A.2 SQL Snippet . . . . .	II

# Abbildungsverzeichnis

A.1 Löwe . . . . .	I
--------------------	---

# Liste der Algorithmen

A.1	SQL - Snippet	II
-----	---------------	----

# Abkürzungsverzeichnis

<b>UML</b>	Unified Modeling Language . . . . .	ii
------------	-------------------------------------	----

# 1. Domain Driven Design

## 1.1 Analyse der Ubiquitous Language

## 1.2 Analyse und Begründung der verwendeten Muster

### 1.2.1 Analyse

### 1.2.2 Begründung

## 2. Clean Architecture

### 2.1 Schichtenarchitektur

#### 2.1.1 Planung

#### 2.1.2 Entscheidung anhand von Kriterien

## 3. Programming Principles

### 3.1 SOLID

#### 3.1.1 Analyse

#### 3.1.2 Begründung

### 3.2 GRASP (insb. Kopplung/Kohäsion)

#### 3.2.1 Analyse

#### 3.2.2 Begründung

### 3.3 DRY

#### 3.3.1 Analyse

#### 3.3.2 Begründung



# 4. Refactoring

## 4.1 Identifizieren von Codesmells

### 4.1.1 Code Smell 1

Begründung

Fix

### 4.1.2 Code Smell 2

Begründung

Fix

# 5. Entwurfsmuster

5.1 Begründung des Einsatzes

5.2 UML Vorher

5.3 UML Nachher

# A. Anhang

## A.1 Löwenmann



Abbildung A.1: Löwe

## A.2 SQL Snippet

```
1  #####
2
3  — Die Tabellengößen aus der Postgre_size Tabelle abfragen,
4  SELECT
5      relname as "Table",
6      pg_size_pretty(pg_total_relation_size(relid)) As "Size",
7      pg_size_pretty(pg_total_relation_size(relid) - pg_relation_size(relid)) as "
      External Size"
8  FROM pg_catalog.pg_statio_user_tables as cat
9  —,wenn der Name der Tabelle in dieser Liste steht
10 where cat.relname LIKE any('{a,
11                               by,
12                               c,
13                               d}'::text[])
14 ORDER by pg_total_relation_size(relid) desc ;
15
16 #####
```

Algorithmus A.1: SQL - Snippet