Fachinformatiker für Anwendungsentwicklung

Dokumentation zur Projektarbeit

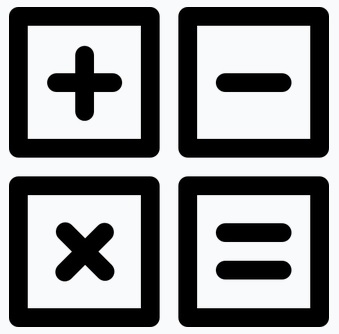
Taschenrechner mit GUI

Erstellung eines Taschenrechners mit grafischer Benutzeroberfläche

Prüfungsbewerber:

Benjamin Wilke

Ausbildungsbetrieb:



Inhaltsverzeichnis

[Inhaltsverzeichnis II](#_Toc98261444)

[Abbildungsverzeichnis III](#_Toc98261445)

[Tabellenverzeichnis III](#_Toc98261446)

[Verzeichnis der Listings III](#_Toc98261447)

[Abkürzungsverzeichnis III](#_Toc98261448)

[1. Einleitung 2](#_Toc98261449)

[1.1 Projektumfeld 2](#_Toc98261450)

[1.2 Projektziel 2](#_Toc98261451)

[1.3 Projektbegründung 2](#_Toc98261452)

[1.4 Projektschnittstellen 2](#_Toc98261453)

[1.5 Projektabgrenzung 2](#_Toc98261454)

[2. Projektplanung 2](#_Toc98261455)

[2.1 Projektphasen 2](#_Toc98261456)

[2.1.1 Projektwoche Ablauf (Idealfall) 2](#_Toc98261457)

[2.1.2 Hauptaufgaben 2](#_Toc98261458)

[2.1.3 Teilaufgaben 3](#_Toc98261459)

[2.2 Abweichungen vom Projektantrag 3](#_Toc98261460)

[2.3 Ressourcenplanung 3](#_Toc98261461)

[2.4 Entwicklungsprozess 3](#_Toc98261462)

[3. Analysephase 3](#_Toc98261463)

[3.1 Ist-Analyse 3](#_Toc98261464)

[3.2 Wirtschaftlichkeitsanalyse 3](#_Toc98261465)

[3.2.1 Make or Buy Entscheidung 3](#_Toc98261466)

[3.2.2 Projektkosten 3](#_Toc98261467)

[3.2.3 Amortisationsdauer 3](#_Toc98261468)

[3.3 Nutzwertanalyse 3](#_Toc98261469)

[3.4 Anwendungsfälle 4](#_Toc98261470)

[3.5 Qualitätsanforderungen 4](#_Toc98261471)

[3.6 Lastenheft/Fachkonzept 4](#_Toc98261472)

[Benutzer / Zielgruppe 4](#_Toc98261473)

[Ziele des Anbieters 4](#_Toc98261474)

[Ziele und Nutzen des Anwenders 5](#_Toc98261475)

[Anforderung 1 5](#_Toc98261476)

[Mockup 5](#_Toc98261477)

[4. Entwurfsphase 6](#_Toc98261478)

[4.1 Zielplattform 6](#_Toc98261479)

[4.2 Architekturdesign 6](#_Toc98261480)

[4.3 Entwurf der Benutzeroberfläche, Mockup 6](#_Toc98261481)

[4.4 Datenmodell 6](#_Toc98261482)

[4.5 Taschenrechnerlogik, Klassendiagramm 7](#_Toc98261483)

[4.6 Maßnahmen zur Qualitätssicherung 7](#_Toc98261484)

[4.7 Pflichtenheft/Datenverarbeitungskonzept 7](#_Toc98261485)

[5. Implementierungsphase 7](#_Toc98261486)

[5.1 Implementierung der Datenstrukturen 7](#_Toc98261487)

[5.2 Implementierung der Benutzeroberfläche 8](#_Toc98261488)

[5.3 Implementierung der Taschenrechnerlogik 8](#_Toc98261489)

[6. Abnahmephase 8](#_Toc98261490)

[7. Einführungsphase 9](#_Toc98261491)

[8. Dokumentation 9](#_Toc98261492)

[9. Fazit 9](#_Toc98261493)

[9.1 Soll-/Ist-Vergleich 9](#_Toc98261494)

[9.2 Lessons Learned 9](#_Toc98261495)

[9.3 Ausblick 9](#_Toc98261496)

[Literaturverzeichnis 9](#_Toc98261497)

[Eidesstattliche Erklärung 9](#_Toc98261498)

[Anhang i](#_Toc98261499)

[A. Detaillierte Zeitplanung i](#_Toc98261500)

[B. Lastenheft (Auszug) i](#_Toc98261501)

[C. Use-Case-Diagramm i](#_Toc98261502)

[D. Pflichtenheft (Auszug) i](#_Toc98261503)

[E. Oberflächenentwürfe i](#_Toc98261504)

[F. Screenshots der Anwendung i](#_Toc98261505)

[G. Entwicklerdokumentation (Auszug) i](#_Toc98261506)

[H. Testfall und sein Aufruf an der Konsole i](#_Toc98261507)

[I. Klassendiagramm i](#_Toc98261508)

[J. Benutzerdokumentation (Auszug) i](#_Toc98261509)

# Abbildungsverzeichnis

# Tabellenverzeichnis

# Verzeichnis der Listings

# Abkürzungsverzeichnis

GUI…...*Graphical User Interface*

MVC….*Model View Controller*

UML….*Unified Modeling Language*

# Einleitung

## Projektumfeld

Projekt innerhalb einer Projektwoche einer Umschulung.

## Projektziel

Die Entwicklung eines Taschenrechners mit GUI inklusive entsprechender Dokumentation wie Lasten- und Pflichtenheft sowie IHK Dokumentation.

## Projektbegründung

Praktische Umsetzung der bisher erlernten Inhalte und Methoden.

## Projektschnittstellen

## Projektabgrenzung

Es wird eine einfache GUI und Grundrechenarten implementiert.

# Projektplanung

## Projektphasen

Für das Projekt sind 5 Tage inklusive Präsentation angesetzt.

### Projektwoche Ablauf (Idealfall)

**Allgemeiner Arbeitsablauf**

Tägliche Dokumentation (Lastenheft, Pflichtenheft, IHK Doku)

Tag 01: Projektantrag & Anforderungsanalyse und GUI Mockup

Tag 02: UML (Anwendungsfall- und Klassendiagramm)

Tag 03 & 04: Implementierung & Test

Tag 5: Präsentation des Programmes mit offene Feedbackrunde

Tägliches Feedbackgespräch ca. 10-15 Min.

### Hauptaufgaben

|  |  |
| --- | --- |
| Phase | Zeit in Tagen |
| Anforderungsanalyse | 0,5 |
| Entwurf | 1 |
| Implementierung | 2 |
| Qualitätsmanagement | 0,5 |
| Dokumentation + Präsentation | 1 |

### Teilaufgaben

|  |  |
| --- | --- |
| Phase | Zeit in Stunden |
| Anforderungsanalyse | 4 |
| Soll-Analyse durchführen | 2 |
| Lastenheft erstellen | 2 |
| Entwurf | 8 |
| GUI Mockup erstellen | 2 |
| UML (Anwendungsfalldiagramm) | 3 |
| UML (Klassendiagramm) | 3 |
| Implementierung | 16 |
| Erstellen der GUI | 3 |
| Programmieren der Funktionen | 13 |
| Qualitätsmanagement | 4 |
| Software Tests manuell und mit JUnit | 4 |
| Dokumentation + Präsentation | 8 |
| Lastenheft | 3 |
| Pflichtenheft | 2 |
| IHK Dokumentation | 3 |
| Gesamt | 40 |

## Abweichungen vom Projektantrag

## Ressourcenplanung

HomeOffice Büro, PC, Entwicklungsumgebung Eclipse, Office Paket, Internet

## Entwicklungsprozess

Beim Entwicklungsprozess wurde nach der Wasserfallmethode vorgegangen, die teilweise durch agile Prozesse unterstützt wurde.

# Analysephase

## Ist-Analyse

## Wirtschaftlichkeitsanalyse

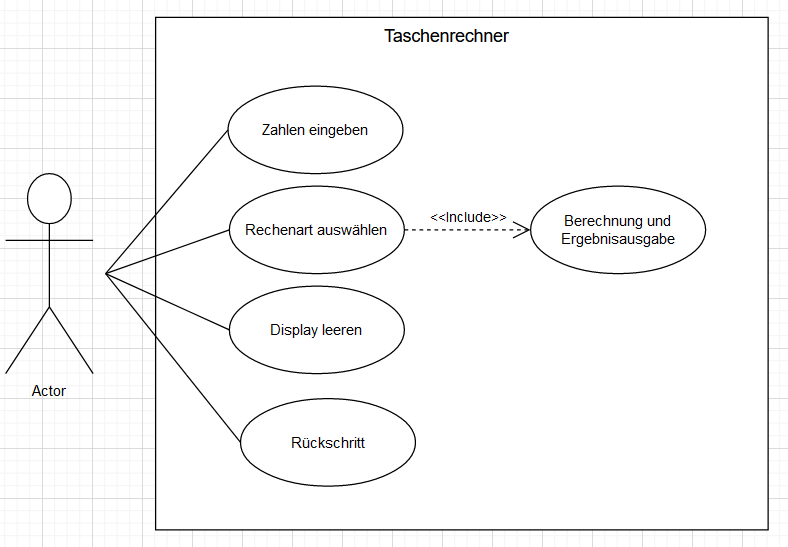
### Make or Buy Entscheidung

### Projektkosten

### Amortisationsdauer

## Nutzwertanalyse

## Anwendungsfälle



## Qualitätsanforderungen

Das Programm soll einfach und intuitiv bedienbar sein. Nach der Entwicklung wird das Programm durch einen Blackboxtest getestet.

## Lastenheft/Fachkonzept

Auszüge aus dem Lastenheft:

## Benutzer / Zielgruppe

Die Zielgruppe sind Anwender eines Taschenrechners.

## Ziele des Anbieters

Es soll ein Programm entwickelt werden, dass einen Taschenrechner mit den Grundrechenarten darstellt. Dabei sollen auch die Funktionen wie Leeren und Rückschritt möglich sein. Die Funktion „leeren“ soll ganz einfach den Taschenrechner auf den Startzustand zurücksetzen und die Funktion „Rückschritt“ die letzte Eingabe rückgängig machen.

Die Anordnung der Buttons für das Programm entsprechen einem gewöhnlichen, einfachen Taschenrechner.

Optional: Beachten der Mathematischen Regeln wie Punkt vor Strichrechnung bei Eingabe von mehr als einer Operation.

## Ziele und Nutzen des Anwenders

-Grundrechenoperationen durchführen: Addition, Subtraktion, Division und Multiplikation.

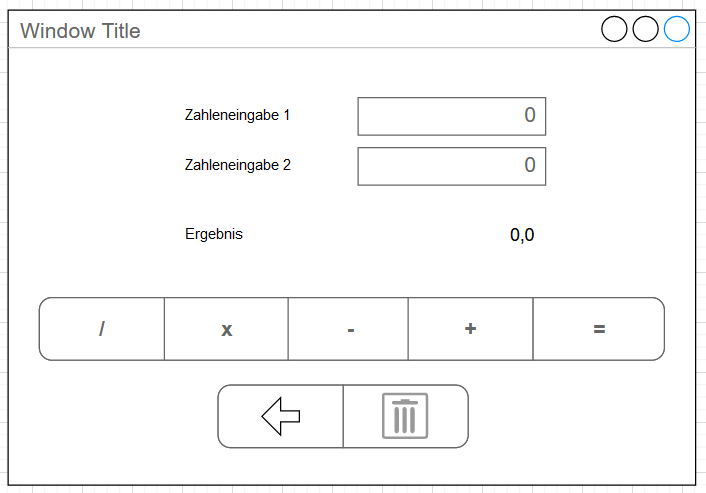
-Auf Startzustand zurücksetzen (Clear)

-Schritt zurück gehen

### Anforderung 1

Es muss eine grafische Oberfläche genutzt werden können. Diese ist in der folgenden Übersicht dargestellt.

### Mockup



**Anforderung 2**

Der Anwender soll Zahlen eingeben können, die anschließen mit den Grundrechenarten berechnet werden und schließlich angezeigt werden können. Außerdem kann der Rechner auf den Startzustand zurückgesetzt werden. Zusätzlich kann die letzte Eingabe rückgängig gemacht werden.

# Entwurfsphase

## Zielplattform

Systemvoraussetzungen:

Windows 7 SP1 und höher  
Linux Kernel 6 und höher

MacOS X 10.11 und höher  
Minimum: Oracle Java SE 11

Programmiersprache: Java

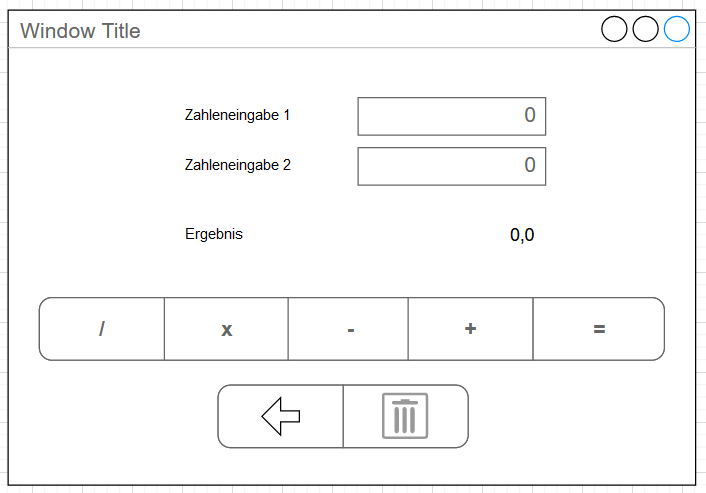
Entwicklungsumgebung (IDE): Eclipse

## Architekturdesign

Das Programm wird im MVC (Model View Controller) -Observer Pattern geschrieben. So können einfach Änderungen an der Benutzeroberfläche vorgenommen werden, ohne dass die Funktionalität beeinträchtigt wird. Andersherum können problemlos Funktionen geändert oder erweitert werden, ohne die Benutzeroberfläche zu ändern.

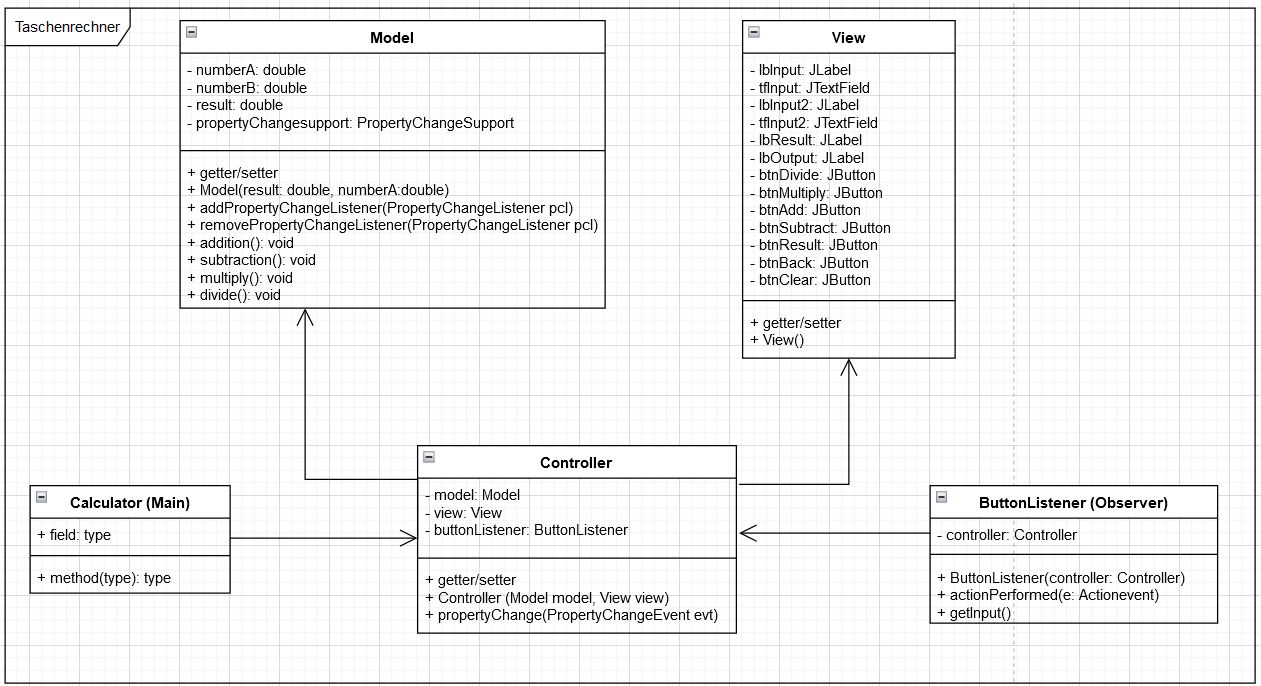
## Entwurf der Benutzeroberfläche, Mockup

Es wird eine einfache übersichtliche GUI erstellt, die im folgenden Mockup zu sehen ist:



## Datenmodell

## Taschenrechnerlogik, Klassendiagramm

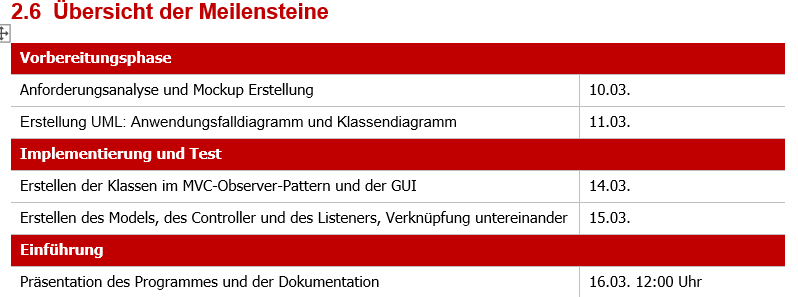


## Maßnahmen zur Qualitätssicherung

Es werden Anwendertests im Blackbox-Verfahren durchgeführt.

## Pflichtenheft/Datenverarbeitungskonzept

Auszug aus dem Pflichtenheft:

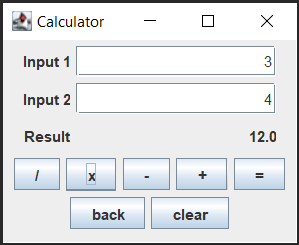


# Implementierungsphase

## Implementierung der Datenstrukturen

## Implementierung der Benutzeroberfläche

Screenshot der GUI:



## Implementierung der Taschenrechnerlogik

Das Ergebnis von zwei eingegebenen Zahlen kann mit einem Tastendruck auf die entsprechende Rechenoperation angezeigt werden. Ohne Neueingabe das Ergebnis der anderen Operationen angezeigt werden.

Der Rechner kann auf den Startzustand zurückgesetzt werden.

Die letzte Eingabe kann rückgängig gemacht werden.

Ausschnitt des Quelltextes aus der Klasse Controller:

Ein Bild, das Text, Elektronik, Screenshot enthält.

Automatisch generierte Beschreibung

# Abnahmephase

Die Anwendung wurde erfolgreich durch Probanden im Blackbox-Verfahren getestet.

# Einführungsphase

# Dokumentation

Es liegen ein Projektantrag, ein Lastenheft, ein Pflichtenheft und diese IHK-Dokumentation vor.

# Fazit

## Soll-/Ist-Vergleich

Das Projektziel wurde erreicht. Der Auftraggeber ist zufrieden. Bei der Projektplanung wurden Zeit, Kosten, Personal und Sachmittel eingehalten.

Kurz vor der Umsetzung wurde das Konzept noch leicht verändert, um eine einfachere Implementierung zu gewährleisten, um das Projektziel nicht zu gefährden.

Bei mehr Zeit hätten noch zusätzlich JUnit-Tests implementiert werden können.

Tatsächlicher Zeitaufwand:

Zeit Mockup 10.03.22 11:20 – 11:40 Uhr

Zeit Use Case 10.03.22 11:40 - 12:15 Uhr

Zeit Klassendiagramm 10.03.22 12:15 – 12:40 Uhr

Zeit Programmierung 10.03.22 12:40 – 14:00 Uhr View

Zeit Programmierung 10.03.22 18:00 – 18:40 Uhr View

Zeit Programmierung 11.03.22 10:00 – 12:30 Uhr View

Zeit Programmierung 11.03.22 12:30 - 13:00, 14:10 - Uhr Listener

Zeit Programmierung 11.03.22 14:10 - 15:00 Uhr Model

Zeit Programmierung 11.03.22 15:00 - 15:10 Uhr Controller //nur addition

Zeit Programmierung 14.03.22 8:20 – 16:45 Uhr Controller, Listener, Model bis hier

Zeit Programmierung 15.03.22 8:20 – 10:40 Uhr zu Ende programmiert

Zeit Dokumentation 15.03.2022 10:40 – 18:00 Uhr +Klassendiagramm

## Lessons Learned

Der Zeitaufwand für die Dokumentation ist nicht zu unterschätzen, besonders nebensächliche Sachen wie Formattierungen.

## Ausblick

# Literaturverzeichnis

# Eidesstattliche Erklärung

# Anhang

# Detaillierte Zeitplanung

# Lastenheft (Auszug)

# Use-Case-Diagramm

# Pflichtenheft (Auszug)

# Oberflächenentwürfe

# Screenshots der Anwendung

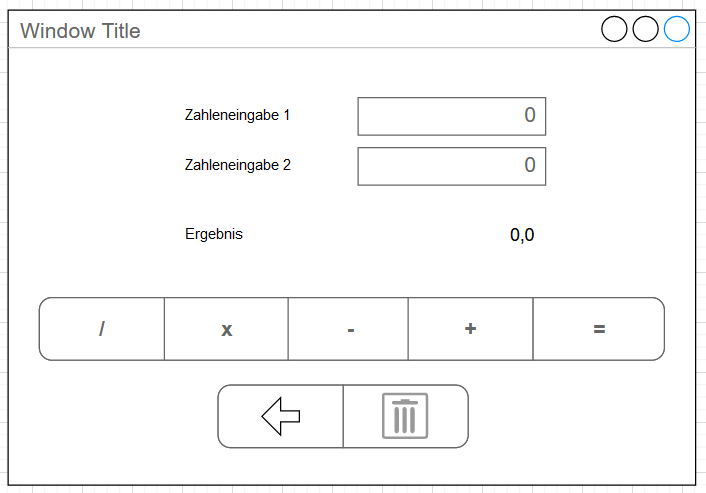
# Entwicklerdokumentation (Auszug)

# Testfall und sein Aufruf an der Konsole

# Klassendiagramm

# Benutzerdokumentation (Auszug)

Mockup



Anwendungsfalldiagramm

