

Projektarbeit

im Modul Software-Architektur
an der Hochschule für Technik und Wirtschaft des Saarlandes
im Studiengang Praktische Informatik
der Fakultät für Ingenieurwissenschaften

Cloud-Native Architekturen

vorgelegt von
Tristan Gläs und Carolin Becker

betreut und begutachtet von
Prof. Dr. Markus Esch

Saarbrücken, 22. März 2021

Selbständigkeitserklärung

Ich versichere, dass ich die vorliegende Arbeit (bei einer Gruppenarbeit: den entsprechend gekennzeichneten Anteil der Arbeit) selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ich erkläre hiermit weiterhin, dass die vorgelegte Arbeit zuvor weder von mir noch von einer anderen Person an dieser oder einer anderen Hochschule eingereicht wurde.

Darüber hinaus ist mir bekannt, dass die Unrichtigkeit dieser Erklärung eine Benotung der Arbeit mit der Note „nicht ausreichend“ zur Folge hat und einen Ausschluss von der Erbringung weiterer Prüfungsleistungen zur Folge haben kann.

Saarbrücken, 22. März 2021

Tristan Gläs und Carolin
Becker

Zusammenfassung

Kurze Zusammenfassung des Inhaltes in deutscher Sprache, der Umfang beträgt zwischen einer halben und einer ganzen DIN A4-Seite.

Orientieren Sie sich bei der Aufteilung bzw. dem Inhalt Ihrer Zusammenfassung an Kent Becks Artikel: <http://plg.uwaterloo.ca/~migod/research/beck00PSLA.html>.

*We have seen that computer programming is an art,
because it applies accumulated knowledge to the world,
because it requires skill and ingenuity, and especially
because it produces objects of beauty.*

— Donald E. Knuth [5]

Danksagung

Hier können Sie Personen danken, die zum Erfolg der Arbeit beigetragen haben, beispielsweise Ihren Betreuern in der Firma, Ihren Professoren/Dozenten an der htw saar, Freunden, Familie usw.

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 1 |
| 1.1 | L ^A T _E X installieren und einrichten | 1 |
| 1.1.1 | Unter Windows | 1 |
| 1.1.2 | Unter Linux | 1 |
| 1.2 | Entwicklungsumgebungen | 1 |
| 1.3 | Werkzeuge | 2 |
| 1.4 | Struktur und Gebrauch der Vorlage | 2 |
| 1.4.1 | Struktur der Vorlage | 2 |
| 1.4.2 | Gebrauch der Vorlage | 3 |
| 2 | Beispiele | 5 |
| 2.1 | Abkürzungen | 5 |
| 2.2 | Beispiel für BibLaTeX | 5 |
| 2.3 | Referenzierungen | 5 |
| 2.3.1 | Beispieltext | 6 |
| 2.4 | Dateien einbinden | 6 |
| 2.5 | Tabellen | 6 |
| 2.5.1 | Einfache Tabelle | 6 |
| 2.5.2 | Erweiterte Tabellenbefehle | 7 |
| 2.6 | Abbildungen | 8 |
| 2.6.1 | Wrapfigure | 8 |
| 2.6.2 | Subfigures | 9 |
| 2.6.3 | Qualitätsunterschiede | 9 |
| 2.7 | Quellcode einbinden | 11 |
| 2.8 | Mathematische Ausdrücke | 11 |
| 2.9 | To-Do-Notes | 13 |
| 3 | Einleitung | 17 |
| 4 | Cloud Native | 19 |
| 4.1 | Cloud Computing | 19 |
| 4.2 | Definition Cloud Native | 19 |
| 4.3 | Microservices | 19 |
| 4.4 | Container | 19 |
| 4.5 | Abgrenzungen | 19 |
| 4.6 | Einsatzgebiete | 19 |
| 4.7 | Eigenschaften | 19 |
| 4.8 | Design Prinzipien | 19 |
| 4.9 | Vor- und Nachteile | 19 |
| 5 | Fallbeispiel | 21 |
| 5.1 | Beschreibung | 21 |
| 5.2 | Anforderungen | 21 |
| 5.3 | Architekturentwurf | 21 |

| | |
|---|-----------|
| 5.4 Implementierung des Prototyps | 21 |
| 6 Diskussion | 23 |
| 6.1 Skalierbarkeit | 23 |
| 7 Fazit | 25 |
| Literatur | 27 |
| Abbildungsverzeichnis | 29 |
| Tabellenverzeichnis | 29 |
| Listings | 29 |
| Abkürzungsverzeichnis | 31 |
| A Erster Abschnitt des Anhangs | 35 |

1 Einleitung

1.1 \LaTeX installieren und einrichten

1.1.1 Unter Windows

Als LaTeX-Distribution unter Windows steht *MikTeX* zu Verfügung, die als freie Software im Internet erhältlich ist. *MikTeX* unterstützt Windows XP, Vista und Windows 7. Neben *MikTeX* wird noch ein PostScript-Interpreter benötigt, z.B. GhostScript, zu finden auf Chip.de.

Wichtig: Bei *MikTeX* unbedingt Vollinstallation auswählen, sonst sind eventuell benötigte Packages nicht vorhanden.

1.1.2 Unter Linux

Unter Linux existiert die LaTeX-Distribution *texlive*, die als aktuelle Version aus den Paketquellen geladen werden kann (unter Ubuntu mit `apt-get install texlive-full`). Auch hier ist ganz wichtig, die volle Distribution zu laden, damit alle Packages zur Verfügung stehen.

1.2 Entwicklungsumgebungen

Hat man die passende Distribution installiert, bieten sich vielerlei Möglichkeiten an ein LaTeX-Projekt anzugehen oder einzelne Dokumente zu editieren. Unter Windows können dies folgende sein:

TeXnicCenter Umfangreiche Entwicklungsumgebung mit Projektorganisation und Autovervollständigung

TeXLipse Eclipse-Plugin, das alle Vorteile der Eclipseumgebung mit LaTeX verbindet

TeXmaker Einfacher LaTeX-Editor mit Pdf-Direktvorschau

Unter Linux stehen bereit:

Gummi Ebenfalls einfacher LaTeX-Editor mit Direktvorschau

TeXLipse Auch für Linux erhältlich

Kile Umfangreiche Entwicklungsumgebung, ähnlich wie TeXnicCenter

Nach der Installation muss die Entwicklungsumgebung eingerichtet werden; dazu finden sich viele Anleitungen im Internet, die genau erklären, welche Distribution auf welche Weise eingerichtet wird. Insbesondere sollte der PDF-Viewer festgelegt werden, damit bei Gummi und TeXmaker die Direktvorschau funktioniert. Manchmal kommt es vor, dass die Ausgabe nach dem Kompilieren Umlaute und Sonderzeichen nicht richtig darstellt. Unter Linux hängt dies mit den unterschiedlichen Zeichensätzen zusammen, die unterstützt werden. Um diese Vorlage zu verwenden ist es notwendig, den verwendeten Zeichensatz des Editors bzw. der Entwicklungsumgebung auf den in diesem Dokument verwendeten Zeichensatz umzustellen: UTF-8 ohne BOM (Byte Order Mark).

1.3 Werkzeuge

JabRef Ein Literaturverwaltungsprogramm, welches das *BibTeX*-Format einsetzt und mit Hilfe einer graphischen Oberfläche das Anlegen von Literaturverzeichnissen vereinfacht.

1.4 Struktur und Gebrauch der Vorlage

Die vorliegende Vorlage für Abschlussarbeiten besteht aus einer internen Struktur, die grundsätzlich nicht verändert werden sollte.

1.4.1 Struktur der Vorlage

htw-i-mst-config.tex Enthält alle zu ladenden Packages, Styleparameter für Hyperlinks, Codelistings und Literaturverzeichnis sowie globale Parameter für Tabellen und Beschriftungen. Im Besonderen befinden sich hier die Variablen für den eigenen Namen, Titel, Datum der Arbeit, den betreuenden Professor etc.

htw-i-mst-vorlage.tex Dies ist die Hauptdatei, in der alle notwendigen *.tex-Dateien eingebunden werden, die zu dem Dokument gehören. Es empfiehlt sich die interne Struktur *nicht* zu verändern. Eigene Kapitel werden an der dafür markierten Stelle eingebunden.

Bibliography.bib Zentrale Datei für die Literaturangaben, welche man z.B. mit JabRef verwalten kann.

Chapters/ Ablageort für alle selbst angelegten Kapitel der Arbeit. Die Aufteilung in eigene Dateien erleichtert die Übersicht über den Quellcode.

Graphics/ Ablageort für alle im Dokument benötigten Grafikdateien. Gerne darf man hier Unterverzeichnisse zur besseren Strukturierung anlegen.

Examples/ Dieser Ordner enthält die in dieser Vorlage beigelegten LaTeX-Beispiele, welche vor der Abgabe der Arbeit selbstverständlich gelöscht werden sollten.

Frontbackmatter/ In diesem Ordner sind all jene Dateien abgelegt, die – außer dem Kern-text in *Chapters/* – die Gesamtheit der Abschlussarbeit ausmachen.

Titlepage.tex Definiert die Titelseite der Abschlussarbeit. Diese Datei muss normalerweise nicht verändert werden.

Abbreviations.tex Hier werden alle Abkürzungen hinterlegt, die im Dokument verwendet werden.

Abstract.tex Eine kurze Zusammenfassung der Abschlussarbeit wird in diese Datei eingefügt.

Acknowledgements.tex Dort finden Danksagungen ihren Platz.

ConfidentialityClause.tex Beinhaltet den Sperrvermerk und ist nur zu verwenden, falls dies beispielsweise vom beteiligten Unternehmen gefordert wird.

Contents.tex Enthält wichtige Eintragungen in die *Table-of-Contents*. Diese Datei muss normalerweise nicht geändert werden.

Declaration.tex Enthält die Selbständigkeitserklärung. Diese Datei darf nicht geändert werden.

Colophon.tex Enthält einen Hinweis auf die Urheber dieser Vorlage. Diese Datei darf nicht geändert werden.

ListOfs.tex Enthält die Einträge für die Tabellen- und Abbildungsverzeichnisse etc. und muss gewöhnlich nicht verändert werden.

1.4.2 Gebrauch der Vorlage

Grundsätzlich ist nicht viel zu tun, um die Vorlage für Abschlussarbeiten zu verwenden. Man entpackt den Hauptordner in das gewünschte Verzeichnis und nutzt die Dateien so, wie in Abschnitt 1.4.1 beschrieben. Danach werden *alle* Dateien gespeichert und die Hauptdatei, *htwsaar-i-mst-vorlage.tex*, mehrfach kompiliert (LaTeX benötigt mehrere Durchgänge um z.B. Referenzen richtig zuzuordnen). Hat man Änderungen in *Bibliography.bib* bzw. *Bibliography.tex* vorgenommen oder neue Zitate z.B. mittels `\cite` eingefügt, muss erst mit *BibLaTeX* und anschließend mit dem entsprechenden LaTeX-nach-PDF-Compiler übersetzt werden.

2 Beispiele

2.1 Abkürzungen

Um Abkürzungen zu verwenden, muss über `\usepackage{acronym}` das benötigte Package geladen werden. Danach kann man lange Begriffe ganz bequem abkürzen:

So muss man nicht ständig Wireless Local Area Network (WLAN) ausschreiben, auch Transmission Control Protocol (TCP) lässt sich abkürzen. Würde man im Text WLAN oft verwenden, kann man sie, wie hier, nur als Abkürzung anzeigen lassen - oder bei Bedarf die Erklärung mitliefern (Wireless Local Area Network (WLAN)). Weiteres Beispiel könnte die Gang of Four (GoF) sein.

Weitere Informationen sind im Acronym-Manual zu finden.

2.2 Beispiel für BibLaTeX

BibLaTeX ist ein Package, das einem die Arbeit mit Zitaten bzw. Quellenangaben erleichtern kann. Mit JabRef (Abschnitt 1.3) ist es möglich **.bib*-Dateien zu erstellen, in denen alle Angaben zu Autor, Buchtitel, Erscheinungsdatum usw. hinterlegt werden, welche zum passenden Zeitpunkt abgerufen werden können. Das Literaturverzeichnis wird mittels `\printbibliography` ausgegeben.

Im Allgemeinen wird im Literaturverzeichnis auch nur jene Literatur aufgenommen, die auch in der **.tex*-Datei referenziert wird. Danach ist es wichtig nicht nur mit *PdfLaTeX*, sondern auch mit *BibLaTeX* zu kompilieren, damit die zitierten Einträge in die verschiedenen Hilfsdateien aufgenommen werden können.

Einige Zitate

In diesem Satz könnten wir auf [4] verweisen, ebenso auf das wichtige Werk [3]. Wenn uns das nicht genug ist, sollten wir das anmerken, was in [6] geschrieben wurde. Im Zweifelsfall verweisen wir auf eine einzelne Seite, wie in [1, S. 112] zu finden.

Üblicherweise wird auch der Name des Autors bzw. der Autoren genannt, also beispielsweise bei einem Verweis auf Knuth [4] oder auch bei mehreren Autoren Cormen u. a. [2]. LaTeX stellt Mechanismen zur Verfügung, auch dies automatisiert zu erledigen.

2.3 Referenzierungen

Mit Referenzierungen kann ich ganz bequem auf Textpassagen, Kapitel, Sections oder Abbildungen im weiteren Text verweisen. Dies ist ein Verweis auf Abschnitt 2.3.1, der sich auf Abschnitt 2.3.1 befindet.

Auch ein Verweis auf Tabelle 2.1 auf Seite 7 ist möglich.

Man sollte beachten, dass man sein Dokument, wenn es Referenzierungen enthält, mehrmals kompiliert, da sonst manche Verweise nicht aufgelöst werden können.

2.3.1 Beispieltext

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

2.4 Dateien einbinden

Damit man nicht alle Einstellungen, Optionen, Packages und Texte, Abbildungen etc. in einer Datei unterbringen muss, werden zwei Befehle bereitgestellt, um externe **.tex*-Dateien einzubinden: `\include{PFAD}` und `\input{PFAD}`. Mit dem erstem Befehl wird eine neue Seite angelegt, danach kommen die Inhalte aus der angegebenen Datei; mit dem zweiten Befehl wird keine neue Seite angelegt – der Inhalt der angegebenen Datei wird direkt an die betroffene Stelle eingefügt.

Wichtig: Der *Pfad* wird sinnigerweise *relativ* angegeben, wobei als Stammverzeichnis jenes Verzeichnis angesehen wird, in dem die **.tex*-Datei mit der *Document*-Umgebung abgelegt ist (in diesem Fall ist es *htwsaar-i-mst-config.tex*).

2.5 Tabellen

2.5.1 Einfache Tabelle

In LaTeX lassen sich Tabellen unterschiedlicher Ausprägung einfach erzeugen. Das allgemeine Format einer Tabelle sieht aus wie folgt:

Listing 2.1: Allgemeines Format

```
\begin{table}
  \caption{BESCHRIFTUNG}
  \begin{tabular}{FORMATIERUNG}
    TABELLENINHALT
  \end{tabular}
\end{table}
```

Eine Beispieltabelle (Tabelle 2.1) könnte also so aussehen:

Listing 2.2: Tabelle 2.1

```
\begin{table}
  \caption{Beispiel 1}
  \begin{tabular}{lrcr}
    \toprule
    \textbf{Name} & \textbf{Vorname} & \textbf{Matrikelnummer} & \textbf{Lieblingsspeise} \\
    \midrule
    Jackson & Michael & 123456 & Erdbeereis \\
    Springsteen & Bruce & 234567 & Schwedisches Lakritz \\
    & & & 
  \end{tabular}
\end{table}
```



```

        Bach & Anna, Magdalena & 3456789 & Frankfurter
        Kranz \\
        Schumann & Clara & 4567890 & Bisquitt\"ortchen \\
        \bottomrule
    \end{tabular}
    \label{tab:beispieltabelle1}
\end{table}

```

Mit `\caption{Beispiel 1}` bekommt unsere Tabelle eine Beschriftung am Tabellenkopf. `l|r|c|r` legt die Textausrichtung der einzelnen Spalten fest: `l` bedeutet linksausgerichtet, `r` rechtsausgerichtet und `c` zentriert. Durch `|` werden Spaltenlinien gezogen. `\toprule`, `\midrule` und `\bottomrule` erzeugen Kopf-, Mittel- und Abschlusslinie in der Tabelle. Als Spaltentrenner wird das `&` genutzt, Zeilentrenner ist der doppelte Backslash (`\\`). Am Ende kann die Tabelle auch mit einem Label versehen werden (`\label{tab:beispieltabelle1}`), über welches diese referenziert wird.

2.5.2 Erweiterte Tabellenbefehle

Um Tabellen in LaTeX flexibler zu gestalten gibt es weitere Befehle bzw. zusätzliche Pakete, die einem das Leben leichter machen (Tabelle 2.2). Hierzu ein weiteres Beispiel:

Listing 2.3: Tabelle 2.2

```

\begin{table}
    \centering
    \caption{Beispiel 2}
    \begin{tabular}{llll}
        \hline
        Author & Title & Year & \\
        \hline
        \hline
        \multirow{3}{*}{Stanislaw Lem} & Solaris & 1961 & \\
        & Roboterm\"archen & 1967 & \\
        & Der futurologische Kongress & 1971 & \\
        \hline
        \multirow{3}{*}{Isaac Asimov} & Ich, der Robot & & \\
        1952 & & & \\
        & Der Tausendjahresplan & 1966 & \\
        & Doctor Schapirows Gehirn & 1988 & \\
        \hline
    \end{tabular}
    \label{tab:beispieltabelle2}
\end{table}

```

Tabelle 2.1: Beispiel 1

| Name | Vorname | Matrikelnummer | Lieblingsspeise |
|-------------|-----------------|----------------|----------------------|
| Jackson | Michael | 123456 | Erdbeereis |
| Springsteen | Bruce | 234567 | Schwedisches Lakritz |
| Bach | Anna, Magdalena | 3456789 | Frankfurter Kranz |
| Schumann | Clara | 4567890 | Bisquittörtchen |

2 Beispiele

Tabelle 2.2: So sollte man es nicht machen! Beispiel für einen schlechten Tabellenstil

| Author | Title | Year |
|---------------|-----------------------------|------|
| Stanislav Lem | Solaris | 1961 |
| | Robotermärchen | 1967 |
| | Der futurologische Kongress | 1971 |
| Isaac Asimov | Ich, der Robot | 1952 |
| | Der Tausendjahresplan | 1966 |
| | Doctor Schapirows Gehirn | 1988 |

Mit `\centering` wird die Tabelle zentriert ausgerichtet, analoge Befehle für rechts- bzw. linksausrichtung sind z.B. `\raggedleft` und `\raggedright`.

Eine weitere Form der Tabellen ist das Package *tabularx*, das variable Spaltenbreiten unterstützt, und *booktabs*, welches mit horizontalen Linien besser arbeiten kann.

2.6 Abbildungen

LaTeX unterstützt generell die Formate **.jpeg*, **.png* und **.pdf*. Handelt es sich z.B. um Strichgrafiken oder skalierbare Farbflächen, sollte **.pdf* die erste Wahl sein, da sich in diesem Format Vektorgrafiken ohne Qualitätsverlust darstellen bzw. skalieren lassen.



Abbildung 2.1: Erstes Bild, Völklinger Hütte

2.6.1 Wrapfigure

Abbildung 2.1 ist zwar ganz nett anzusehen, aber vielleicht sähe es eleganter aus, wenn die Abbildung von unserem Textabschnitt umflossen werden würde. Diese Art von Abbildungen sollte jedoch sparsam und mit großer Sorgfalt eingesetzt werden, da es zu unschönen Darstellungen kommen kann. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an.

Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.



Abbildung 2.2: Völklinger Hütte, *.jpg

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

2.6.2 Subfigures

Es ist ebenso möglich, mehrere Abbildungen nebeneinander zu setzen, wie in Abbildung 2.3 zu sehen ist. Eine separate Referenzierung ist auch möglich: Abbildung 2.3a.



(a) Erstes ...



(b) ... und zweites Bild

Abbildung 2.3: Mehrere Abbildungen nebeneinander

2.6.3 Qualitätsunterschiede

Leider haben die unterschiedlichen Grafikformate bedingt durch die unterschiedlichen Kompressionsverfahren einige Schwächen, insbesondere die Umwandlung in das *JPG*-Format erzeugt unangenehme Artefakte im Bild. Abbildung 2.4 zeigt die Unterschiede zwischen *PDF-Format* und *JPG-Format* im Vergleich.

2 Beispiele



(a) *PDF-Format*



(b) *JPG-Format*

Abbildung 2.4: Beide Formate im Vergleich



Abbildung 2.5: *PNG-Format*



Abbildung 2.6: *JPG-Format*

Wenn eine *.pdf-Datei nicht infrage kommt, beispielsweise bei Screenshots, ist unbedingt das PNG-Format vorzuziehen. Den Unterschied machen Abbildung 2.5 und Abbildung 2.6 deutlich.

„Faustregeln“ im Umgang mit Abbildungen:

- Diagramme bzw. alles, was Linien usw. enthält: *PDF* (im Vektorformat).
- Screenshots bzw. alles, was größere gleichfarbige Flächen enthält: *PNG*.
- Der Rest (in der Regel Fotos): *JPEG*.

2.7 Quellcode einbinden

Das Package *lstlisting* ermöglicht es, Quellcode ansprechend in das Dokument einzubinden. Man kann Quellcode einzeilig einbinden mittels `\lstinline|Quellcode|`. Dabei ist darauf zu achten, dass der Befehl einmal mit `{ }` und einmal mit `| |` aufgerufen werden kann, je nachdem, welche Zeichen im angegebenen Quelltext genutzt werden. Es ist auch möglich eine eigene Umgebung für Quelltext zu schaffen:

Listing 2.4: Erstes Listing

```
private Umgebung(int i, int k)
{
    System.out.println("Eine Funktion mit " + i + "und" + k ".");
}
```

Wer Quelltext aus externen Dateien einbinden möchte, geht wie folgt vor:

Listing 2.5: Externer Quellcode

```
public class HalloWelt {
    public static void main(String[] args) {
        System.out.println("Hallo Welt!");
    }
}
```

Wie genau der Quellcode formatiert und gefärbt ist, ist in *htw Saar i.mst.config.tex* hinterlegt, wobei für verschiedene Sprachen auch eigene Styles angelegt werden können (hier z.B. für Java).

2.8 Mathematische Ausdrücke

Mathematische Ausdrücke sind eine kleine Kunst für sich. Am allereinfachsten kann man eine Formel, wie $a + b = c$ in den Fließtext einbinden, wobei LaTeX die Höhe der Ausdrücke der Zeile anpasst, wie hier zu sehen $\sum_{y=0}^x a$. In einer Umgebung sieht das schon anders aus:

$$\sum_{y=0}^x a \quad (2.1)$$

Griechische Buchstaben:

$$\alpha\beta\gamma\delta\epsilon\zeta\eta\theta\iota\kappa\lambda\mu\nu\xi\pi\omega\rho\sigma\tau\nu\phi\chi\psi\omega \quad (2.2)$$

2 Beispiele

Brüche:

$$\text{Ergebnis} = \frac{a}{b} \quad (2.3)$$

$$\frac{\sin \alpha^2 + \cos \alpha^2}{1} = 1 \quad (2.4)$$

$$\frac{\frac{-9x}{2y}}{3z+2} \quad (2.5)$$

Text innerhalb von Formeln:

$$\sum_{y=1}^n y = \frac{n * (n + 1)}{2} \quad \text{Gaußsche Summenformel} \quad (2.6)$$

Hoch- bzw. Tiefstellungen:

$$x_{i,j}^2 \quad (2.7)$$

$$x_{i,j}^2 \quad (2.8)$$

$$x_{n_0} \quad (2.9)$$

Matrizen: Matrizen werden innerhalb der mathematischen Umgebung als wiederum neue Umgebung eingebunden. Wie bei Tabellen auch werden Zeilen durch `\\` und Spalten durch `&` getrennt.

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad (2.10)$$

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} \quad (2.11)$$

Fallunterscheidung:

$$f(x) = \begin{cases} 0, & \text{falls } x < 0 \\ 1, & \text{falls } x \geq 0 \end{cases} \quad (2.12)$$

2.9 To-Do-Notes

Um bei einer längeren Arbeit nicht den Überblick zu verlieren, an welcher Stelle es nötig ist weiter zu arbeiten, bietet es sich an, kleine Notizen einzufügen. Das Package *todonotes* stellt eine elegante Lösung bereit, um differenziert und vielfarbig jene Abschnitte zu kennzeichnen, die einer weiteren Bearbeitung bedürfen.

Beispiel für To-Do-Notes

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: Dies ist ein Blindtext? oder Huardest gefburn? Kjift? mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie Lorem ipsum dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.

A very long todonote that certainly will fill more than a single line in the list of todos. Just to make sure let's add some more text ...

Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: Dies ist ein Blindtext? oder Huardest gefburn? Kjift? mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie Lorem ipsum dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.



1: Erste Nummer...

2: Zweite Nummer...

Nachfolgend wird noch eine Liste aller To-Dos auf einer separaten Seite ausgegeben.

Plain todonotes.

Plain todonotes.

Todonote that is only shown in the margin and not in the list of todos.

A note with no line back to the text.

A very long todonote that certainly will fill more than a single line in the list of todos ...

Liste der noch zu erledigenden Punkte

| | |
|--|----|
| Plain todonotes. | 13 |
| Plain todonotes. | 13 |
| A very long todonote that certainly will fill more than a single line in the list of todos. Just to make sure let's add some more text | 13 |
| A note with no line back to the text. | 13 |
| A short entry in the list of todos | 13 |
| Abbildung: A figure I have to make | 13 |
| 1: Erste Nummer... | 13 |
| 2: Zweite Nummer... | 13 |

3 Einleitung

- Cloud Computing wird immer populärer daher Cloud Native - Man braucht viele Ressourcen - Relativ neues gebiet

Cloud-Native ist ein relativ neuer Ansatz in der Informatik bei dem auf Cloud-Technologien basierende Infrastrukturen verwendet werden um skalierbare Softwaresysteme zu entwickeln.

4 Cloud Native

4.1 Cloud Computing

Nist Definition und eigene Erklärung Erklärung warum das wichtig ist für cloud native

4.2 Definition Cloud Native

Cncf definition und Erklärung der Begriffe bis auf microservices und container

4.3 Microservices

definition warum braucht man das im cloud native bereich

4.4 Container

definition warum braucht man das im cloud native bereich

4.5 Abgrenzungen

gegenüber monolith gegenüber virtualisierung

4.6 Einsatzgebiete

vielleicht in einleitung

4.7 Eigenschaften

Skalierbarkeit/Elastizität Infrastructure is fluid Updates und Tests Sicherheit Globale Ebene

4.8 Design Prinzipien

twelve factor app

4.9 Vor- und Nachteile

die von der Powerpoint

5 Fallbeispiel

5.1 Beschreibung

5.2 Anforderungen

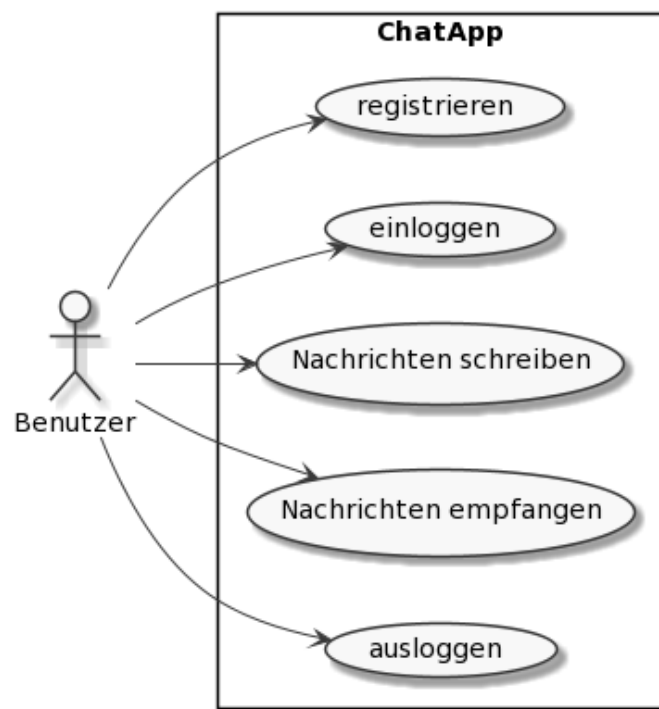


Abbildung 5.1: Use-Case Diagramm

5.3 Architekturentwurf

5.4 Implementierung des Prototyps

6 Diskussion

6.1 Skalierbarkeit

7 Fazit

TeXnicCenter Umfangreiche Entwicklungsumgebung mit Projektorganisation und Autovervollständigung

TeXLipse Eclipse-Plugin, das alle Vorteile der Eclipseumgebung mit LaTeX verbindet

TeXmaker Einfacher LaTeX-Editor mit Pdf-Direktvorschau

Literatur

- [1] Jon Bentley. *Programming Pearls*. Addison–Wesley, 1999.
- [2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest und Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2001.
- [3] Gunter Dueck. *Duecks Trilogie: Omnisophie – Supramanie – Topothesie*. <http://www.omnisophie.com>. Springer, 2005.
- [4] Donald E. Knuth. „Big Omicron and Big Omega and Big Theta“. In: *SIGACT News* 8.2 (1976), S. 18–24.
- [5] Donald E. Knuth. „Computer Programming as an Art“. In: *Communications of the ACM* 17.12 (1974), S. 667–673.
- [6] Ian Sommerville. *Software Engineering*. Boston, MA, USA: Addison-Wesley, 1992.

Abbildungsverzeichnis

| | | |
|-----|---|----|
| 2.1 | Erstes Bild, Völklinger Hütte | 8 |
| 2.2 | Völklinger Hütte, *.jpg | 9 |
| 2.3 | Mehrere Abbildungen nebeneinander | 9 |
| 2.4 | Beide Formate im Vergleich | 10 |
| 2.5 | PNG-Format | 10 |
| 2.6 | JPG-Format | 10 |
| 5.1 | Use-Case Diagramm | 21 |

Tabellenverzeichnis

| | | |
|-----|---|---|
| 2.1 | Beispiel 1 | 7 |
| 2.2 | So sollte man es nicht machen! Beispiel für einen schlechten Tabellenstil . | 8 |

Listings

| | | |
|-----|------------------------------|----|
| 2.1 | Allgemeines Format | 6 |
| 2.2 | Tabelle 2.1 | 6 |
| 2.3 | Tabelle 2.2 | 7 |
| 2.4 | Erstes Listing | 11 |
| 2.5 | Externer Quellcode | 11 |

Abkürzungsverzeichnis

WLAN Wireless Local Area Network

TCP Transmission Control Protocol

GoF Gang of Four

Anhang

A Erster Abschnitt des Anhangs

In den Anhang gehören „Hintergrundinformationen“, also weiterführende Information, ausführliche Listings, Graphen, Diagramme oder Tabellen, die den Haupttext mit detaillierten Informationen ergänzen.

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Kolophon

Dieses Dokument wurde mit der \LaTeX -Vorlage für Abschlussarbeiten an der htw saar im Bereich Informatik/Mechatronik-Sensortechnik erstellt (Version 2.1). Die Vorlage wurde von Yves Hary und André Miede entwickelt (mit freundlicher Unterstützung von Thomas Kretschmer, Helmut G. Folz und Martina Lehser). Daten: (F)10.95 – (B)426.79135pt – (H)688.5567pt