

Lastenheft  
Nachrichtenkommunikation für das THW

na17b

# Inhaltsverzeichnis

<b>1</b>	<b>Ausgangssituation</b>	<b>1</b>
<b>2</b>	<b>Zielsetzung und Produkteinsatz</b>	<b>1</b>
<b>3</b>	<b>Use-Cases</b>	<b>1</b>
	/LC0000/ Auswahl der Rolle . . . . .	1
	/LC0100/ Ausgehende Nachrichten . . . . .	2
	/LC0200/ Eingehende Nachrichten . . . . .	3
	/LC0300/ Kontrolle durch Sichter . . . . .	3
	/LC0400/ Ausdrucken eines Vordruckes . . . . .	3
<b>4</b>	<b>Nichtfunktionale Anforderungen</b>	<b>4</b>
	/LN0000/ Skalierbarkeit . . . . .	4
<b>5</b>	<b>Funktionale Anforderungen</b>	<b>4</b>
	/LF0000/ Ähnlichkeit zum Papierformular . . . . .	4
	/LF0100/ Kommunikation zwischen Computern . . . . .	4
	/LF0200/ Listenanordnung für Vierfachvordrucke mit Filterfunktion . . . . .	4
	/LF0300/ Verschiedene Rollen . . . . .	4
	/LF0400/ Archivierung . . . . .	4
	/LF0500/ Speicherung im RDF-Format . . . . .	5
	/LF0600/ Datensicherheit . . . . .	5
	/LF0700/ Dokument zu PDF rendern . . . . .	5
	/LF0800/ Postausgang . . . . .	5
<b>6</b>	<b>Qualitätsmatrix</b>	<b>6</b>
<b>7</b>	<b>Lieferumfang und Abnahmekriterien</b>	<b>6</b>
<b>8</b>	<b>Vorprojekt</b>	<b>7</b>
	<b>Glossar</b>	<b>8</b>

## 1 Ausgangssituation

Die Grundlage für dieses Projekt bildet der sogenannte Vierfachvordruck, ein internes Kommunikationsdokument des technischen Hilfswerks (THW). Es wird in der mobilen Einsatzzentrale des THW, der Fachgruppe Führung und Kommunikation (FGr FK) eingesetzt. Die FGr FK nutzt das Dokument, um ein- und ausgehende Nachrichten abzufassen und so z.B. Einsatzaufträge für Einheiten, eingehende Lagemeldungen oder Materialanforderungen abzuarbeiten. Der Vierfachvordruck ist eine Papier-Vorlage mit dreifachem Durchschlag. Zwei der Durchschläge werden an die zuständigen Personen innerhalb der FGr FK verteilt. Der dritte Durchschlag dient der Protokollierung. Im Angesicht heutiger Technologien erscheint dieses Verfahren nicht mehr zeitgemäß. Mithilfe einer Software wäre es möglich, den Prozess digital durchzuführen. Dadurch müssten keine handschriftlichen Dokumente verfasst und verteilt werden und das Verfahren könnte beschleunigt werden. Alle Nachrichten und Nachrichtenverläufe ließen sich leicht in einer Datenbank archivieren. Ein weiterer Vorteil wäre der Entfall von Mehraufwand durch schlecht lesbare Handschrift. Außerdem ließe sich die Verwaltung verschiedener Dokumente am Arbeitsplatz übersichtlicher gestalten, etwa durch ein digitales Postfach.

## 2 Zielsetzung und Produkteinsatz

Ziel des Projekts ist es, eine Anwendung zu schreiben, welche es erlaubt, einen Vierfach-Vordruck abzufassen, anzuzeigen und zu archivieren sowie den damit verbundenen Workflow zu durchlaufen, welcher lediglich durch die Software ergänzt aber nicht ersetzt werden soll. Zur maximalen Kompatibilität soll es sich um eine Browseranwendung handeln, welche ihre Daten in einer versionierten RDF-Datenbank auf Serverseite ablegt. Die Archivierung muss dabei die rechtliche Verbindlichkeit sicherstellen, d.h. nachträgliche Änderungen an den Dokumenten sollen erkennbar sein und verhindert werden. Die Software soll innerhalb der Fachgruppe Führung/Kommunikation eingesetzt werden um den Austausch von ein- und ausgehenden Nachrichten zu beschleunigen und zu vereinfachen.

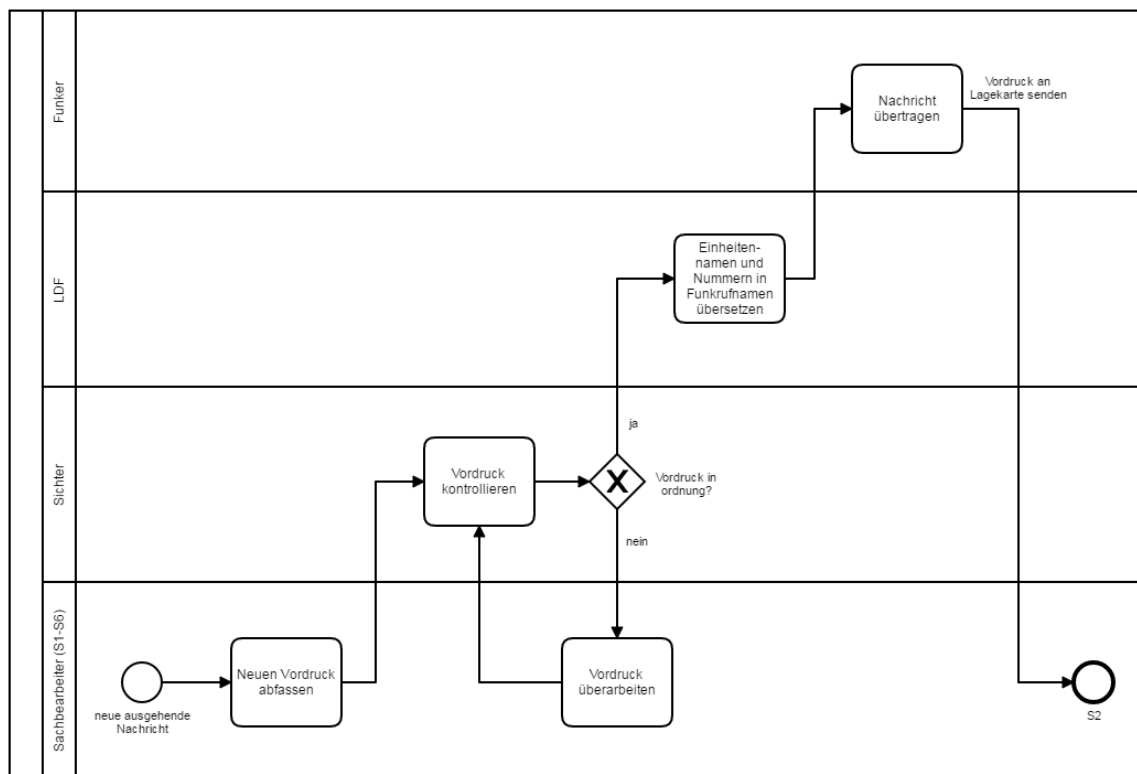
## 3 Use-Cases

### /LC0000/ Auswahl der Rolle

Das System bietet dem Nutzer die Möglichkeit eine der folgenden Rollen auszuwählen

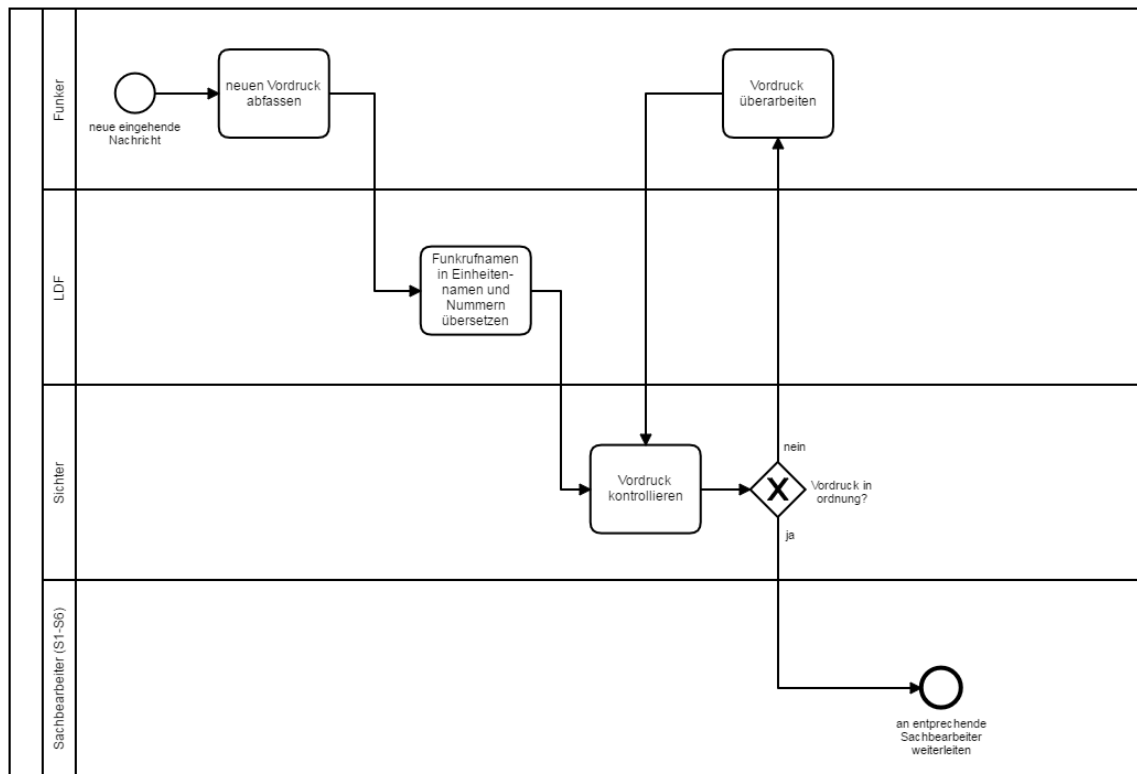
- Sachbearbeiter (S1-S6)
- Sichter
- Leiter der Fernmeldezentrale (LDF)
- Funker

## /LC0100/ Ausgehende Nachrichten



Ein Sachbearbeiter erhält eine neue ausgehende Nachricht und fasst einen neuen Vordruck ab. Dieser Vordruck wird von einem Sichter kontrolliert und an den LDF übermittelt. Der LDF übersetzt die Einheiten-namen und Nummern in Funkrufnamen bevor der Vordruck an den Funker zur Übertragung weitergeleitet wird. Im Anschluss daran erhält Sachbearbeiter S2 (Lagekarte) den Vordruck, womit der Vorgang abgeschlossen wird.

## /LC0200/ Eingehende Nachrichten



Ein Funker erhält eine neue eingehende Nachricht und fasst einen neuen Vordruck ab. Der LDF übersetzt die Funkrufnamen in Einheiten-namen und Nummern bevor der Vordruck zur Kontrolle an einen Sichter gesendet wird. Anschließend wird der Vordruck an die entsprechenden Sachbearbeiter übermittelt

## /LC0300/ Kontrolle durch Sichter

Ein Sichter erhält einen Vordruck zur Kontrolle und stuft diesen als unzureichend ein. Der Vordruck wird dann zurück an den Verfasser gesendet, welcher diesen überarbeitet und zur erneuten Kontrolle an den Sichter schickt. Wird ein Vordruck vom Sichter erfolgreich abgenommen durchläuft er den weiteren Workflow.

## /LC0400/ Ausdrucken eines Vordruckes

Das System bietet dem Nutzer die Möglichkeit, sich einen verfassten oder erhaltenen Vordruck anzuschauen und auszudrucken.

## 4 Nichtfunktionale Anforderungen

### /LN0000/ Skalierbarkeit

Das System der Vierfachvordrucke bietet die Möglichkeit, dass auch mehrere Personen eine Rolle übernehmen, also dass es zum Beispiel zwei oder mehr Funker gibt. Dies ist notwendig, da unterschiedliche Notlagen unterschiedlich stark besetzte Teams erfordern um bewältigt zu werden. Demnach ist es wichtig, dass mit der Software umsetzbar ist, dass eine bestimmte Position zu unterstützenden Zwecken mehrfach besetzt werden kann. Daraus folgt, dass mindestens elf Personen gleichzeitig in der Applikation arbeiten können. Zu achten ist dabei vor allem auf Konflikte bei der gleichzeitigen Bearbeitung von Vordrucken.

## 5 Funktionale Anforderungen

### /LF0000/ Ähnlichkeit zum Papierformular

Der simulierte Vierfachvordruck sollte in Form und Formulierung dem Papierformular gleichen. Somit soll der gewohnte Arbeitsablauf gewährleistet werden.

### /LF0100/ Kommunikation zwischen Computern

Zur Weiterleitung des (ausgefüllten) Dokuments von Mitarbeiter zum Mitarbeiter müssen Computer untereinander kommunizieren können. Gegenüber äußeren unbefugten Zugriffen sollten die Computer sowie der Kommunikationsweg geschützt sein.

### /LF0200/ Listenanordnung für Vierfachvordrucke mit Filterfunktion

In dem Fall, dass zeitgleich mehr als ein Vierfachvordruck bei einem Mitarbeiter ankommt, sollte eine Liste zum Abrufen der eingegangenen Dokumente existieren. Somit kann jedes Dokument nacheinander bearbeitet werden, wodurch es auch zu keinem Informationsverlust kommen kann. Zudem soll eine Filterfunktion existieren, bei der nach einem Namen beziehungsweise Kürzel gesucht werden kann. Daraufhin sollen in der Liste nur Nachrichten von dem entsprechenden Absender angezeigt werden.

### /LF0300/ Verschiedene Rollen

Zum Zwecke der besseren Übersicht und Nachbildung des Workflows ist die Aufteilung in verschiedene Rollen, welche mit den entsprechenden Rollen des aktuellen Workflows übereinstimmen, notwendig. Realisiert wird das, indem beim Start der Applikation die jeweilige, gerade besetzte Position angegeben wird und dann im weiteren Arbeitsverlauf auch nur die Arbeitsschritte, z.B. das Ausfüllen bestimmter Felder des Vierfachvordrucks, von dieser Person ausgeführt werden können, wohingegen die anderen gesperrt sind. Dadurch wird die Integrität der einzelnen Rollen gewährleistet. Dabei wird die gerade besetzte Rolle stetig durch die Farbgebung der Oberfläche deutlich gemacht, nah an der Farbgebung der Kopie des Vierfachvordrucks gehalten, die man an dieser Position erhalten würde. Die Farbgebung kann sich allerdings auch variabel gestalten, z.B. wenn der Vierfachvordruck zur Korrektur nochmal zurückgeschickt wurde, wird er mit der Farbe des Arbeitsschritts anstelle der Farbe der Rolle dargestellt.

### /LF0400/ Archivierung

Es ist von immenser Bedeutung für das Technische Hilfswerk, nach einiger Zeit wieder auf die Angaben eines Vierfachvordrucks zugreifen zu können. Momentan wird dies ermöglicht, indem eine Kopie des Zettels im Archiv geordnet abgelegt wird. Da wir das manuelle Ausfüllen eines Vierfachvordrucks mit unserer Applikation abschaffen wollen, müssen wir dafür sorgen, dass die

Archivierung trotzdem noch aktiv gehalten werden kann. Das heißt, es ist unbedingt erforderlich, dass die gespeicherten Dokumente ausdrückbar sind. Dabei sollte die Darstellungsform der ursprünglichen Zettel möglichst beibehalten werden, um die Archivierung einheitlich zu halten. Außerdem sollen die Eingaben auch noch längerfristig auf einem Server gespeichert werden, um u.a. einen noch schnelleren Zugriff zu ermöglichen. Dieser Server ist derselbe wie der, über den unser Programm später laufen soll, ist somit also nur lokal zugreifbar.

### **/LF0500/ Speicherung im RDF-Format**

Die vom Verfasser im Vierfachvordruck eingegebenen Informationen, sollen im RDF Datenmodell mithilfe des im QuitStore verfügbaren SPARQL-Endpoints auf einem lokalen Server abgespeichert werden. Demnach werden die Eingaben, z.B. 'Dokument hat Absender: Nathanael Arndt' in folgender Form dargestellt:

Subjekt Vierfachvordruck (repräsentiert durch URI)  
Prädikat 'hatAbsender'  
Objekt 'Natanael Arndt' (repräsentiert durch URI oder Name)

Die anderen Stationen, die bisher Kopien des Vierfachvordrucks erhalten haben, müssen dann auf die Eintragungen des Verfassers zugreifen können. Dies wird direkt durch den SPARQL-Endpoint ermöglicht, welcher eine Oberfläche für gezielte SPARQL-Anfragen an das Datenmodell bietet.

### **/LF0600/ Datensicherheit**

Die Vordrucke werden nach ihrer Erstellung und Durchlaufens des weiteren Prozesses abgespeichert. Damit eine Archivierung der Vordrucke sinnvoll ist, muss sichergestellt werden, dass sie von niemandem nachträglich verändert werden können. Ansonsten verliert das Archiv jegliche juristische Relevanz und das THW die Möglichkeit nachzuvollziehen wo und warum Fehler aufgetreten sind. Demnach muss die Eindeutigkeit der gespeicherten Daten gewährleistet sein.

### **/LF0700/ Dokument zu PDF rendern**

Um die manuelle Archivierung zu gewährleisten, sollten die digital getätigten Eingaben ausdrückbar sein. Dabei sollte möglichst Einheitlichkeit beibehalten werden, d.h. alle Eingaben müssen so in eine digitale Version des Vierfachvordrucks geschrieben werden, wie es der Verfasser auch tun würde. Die Vorlage sollte dabei möglichst in ein PDF konvertiert werden, um gutes Ausdrucken zu gewährleisten. Das Ausdrucken sollte außerdem so benutzerfreundlich wie möglich vonstatten gehen, also mit abgespeicherter, aber veränderbarer Vorkonfiguration welche auf einen Knopfdruck abgeschickt werden kann.

### **/LF0800/ Postausgang**

Jeder Verfasser einer Nachricht sollte in der Lage sein selbstverfasste Nachrichten einsehen zu können. Somit ist es möglich, dass auch nach Versand einer Nachricht diese eingesehen werden kann.

## 6 Qualitätsmatrix

Qualitätskriterium	Gewichtung
Zuverlässigkeit (Reliability)	Sehr Hoch
Gebrauchstauglichkeit (Usability)	Sehr Hoch
Funktionalität (Functional Suitability)	Hoch
(IT-)Sicherheit (Security)	Hoch
Wartbarkeit (Maintainability)	Hoch
Effizienz (Performance Efficiency)	Hoch
Portabilität (Portability)	Mittel
Kompatibilität (Compatibility)	Mittel


## 7 Lieferumfang und Abnahmekriterien

Das fertige Produkt umfasst eine voll funktionale Anwendung, welche innerhalb einer mobilen Einsatzzentrale die Arbeit mit dem Vierfachvordruck ersetzen kann. Aufgrund der großen Verantwortung des THW ist die Zuverlässigkeit der Software besonders wichtig. Um diese zu gewährleisten muss die Software und deren Module vor ihrer Auslieferung und während der Entwicklung vollumfänglichen, automatisierten Tests unterzogen werden. Wofür ein Testingframework Anwendung findet, welches Modultests und Integrationstest durchführt. Eine Testcoverage von 100 % muss gewährleistet sein. Eine weitere Komponente ist das zuverlässige Speichern, Abfragen und Darstellen der ausgefüllten Vordrucke. Hierzu wird ein Datenbanksystem aufgesetzt, welches das RDF-Modell verwendet, SPARQL als Querylanguage unterstützt und die Quit-Store Speicher-Architektur nutzt. Die Darstellung orientiert sich stark am Original des Vierfachvordrucks und erfolgt über einen Browser. Um die Verlässlichkeit weiter zu steigern wird darüber hinaus das Prinzip der kontinuierlichen Integration angewandt. Somit kann eine gemeinsame, einfach erreichbare und verlässliche Codebasis für das gesamte Team geschaffen werden. Ein weiteres wichtiges Abnahmekriterium ist die hohe Gebrauchstauglichkeit, wobei insbesondere die schnelle Zugänglichkeit, die klare Trennung der Rollen und die Möglichkeit des effizienten und schnellen Arbeitens zu gewährleisten ist. Aus der geforderten Usability folgt auch eine entsprechende Performanz, welche das unterbrechungsfreie, parallele Arbeiten von bis zu elf Nutzern (sechs Sachgebetsbearbeiter, zwei Funker, ein Sichter, ein Lagekartenführer und ein Stabsleiter) garantiert, wobei als Hardware ein Raspberry Pi vorgesehen ist. Um die Akzeptanz weiter zu steigern, soll die Software in der Lage sein den Nutzern durch ihre Funktionalität weitere Aufgaben abzunehmen oder zumindest zu erleichtern. Dazu seien bspw. das Drucken der rechtlich belastbaren Unterlagen und die verschiedenen Sichten mit ihren Zugriffsbeschränkungen genannt. Im Hinblick auf die mögliche Weiterentwicklung ist auch die Wartbarkeit durch eine ausführliche, wohl strukturierte Dokumentation, sowie einheitlich formatierten und gut leserlichen Code möglichst komplikationsfrei zu gestalten. Die Kompatibilität muss in sofern gegeben sein, dass die Webseite auf unterschiedlichen Geräten gut erkenn- und bedienbar sein soll. Durch die feste Hardware ist die Portabilität weniger wichtig, trotzdem sollte die Möglichkeit bestehen ohne großen Aufwand auf ein anderes System zu migrieren.



## 8 Vorprojekt

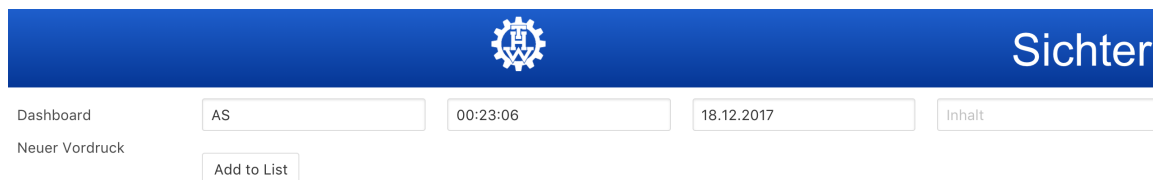
Der Fokus im Vorprojekt liegt auf der Erstellung der Benutzeroberfläche zur Erstellung eines Vierfachvordrucks. Die GUI wird über den Browser aufgerufen und ermöglicht in der Startseite die Auswahl der entsprechenden Rolle über ein Dropdown-Menü. In derselben Übersichtsseite sind die für die ausgewählte Rolle relevanten Informationen ersichtlich. Jede Rolle hat einen eigenen View. Beim Erstellen eines neuen Vordrucks soll das Design des Formulars im Vorprojekt weitestgehend fertig umgesetzt sein. Zu Demonstrationszwecken sollen einige voreingestellte Vordrucke zur Verfügung stehen. Neu erstellte Vordrucke bleiben zunächst nur temporär bestehen und gehen beim Verlassen der Seite verloren. Dadurch sollen Testnutzer in der Lage sein, einen ersten Überblick über den Prozess des Verfassens eines Vierfachvordrucks zu erhalten, ohne sich Gedanken über die Integrität der Daten machen zu müssen. Der Workflow liegt hierbei zunächst nicht im Fokus; dieser wird erst im fertigen Projekt ersichtlich sein, da hierzu ein funktionsfähiges Backend benötigt wird.



The screenshot shows the 'Sichter' interface with a blue header bar containing a logo and the title 'Sichter'. Below the header, there is a sidebar with 'Dashboard' and 'Neuer Vordruck'. The main area is titled 'Nachrichtenübersicht' and contains a table with four columns: 'Verfasser', 'Datum', 'Uhrzeit', and 'Kurzinhalt'. The table lists three messages from authors AS and LS on December 16, 2017.

Verfasser	Datum	Uhrzeit	Kurzinhalt
AS	17.12.2017	17:03:12	Baum umgestürzt auf Hauptstraße.
AS	16.12.2017	22:57:21	Keller vollgelaufen nach Unwetter.
LS	16.12.2017	22:13:54	Noteinsatz nach schwerem Gewitter.

Abbildung 1: Startseite



The screenshot shows the 'Sichter' interface with a blue header bar. Below the header, there is a sidebar with 'Dashboard' and 'Neuer Vordruck'. The main area contains a form with four input fields: 'AS', '00:23:06', '18.12.2017', and 'Inhalt'. Below the form is a button labeled 'Add to List'.

Abbildung 2: Erstellen eines neuen Vordrucks

## Glossar

**Continuous Integration** Der Begriff Continuous Integration (CI) beschreibt eine Softwareentwicklungspraxis, in der alle beteiligten Entwickler eines Projektes häufig (ein- bis mehrmals am Tag) ihren aktuellen Arbeitsstand (meistens verwirklicht in einem dezidierten Branch) in den Hauptstand (Masterbranch) integrieren.

Um dieses Ziel zu erreichen werden meist folgende Tools verwendet:

- Version Control System (z.B. Git)
- automatisierte Tests (z.B. Unit- oder Integrationstests)
- CI-Server (z.B. Github CI)

Die direkten Vorteile der CI sind zum einen der minimale Integrationsaufwand sowie die Fähigkeit, jederzeit eine stabile Version veröffentlichen zu können. Der minimale Integrationsaufwand ergibt sich durch die Häufigkeit, mit der die Entwickler ihren Arbeitsstand integrieren. Durch die Bedingung, dass vor jeder Integration alle automatisierten Testfälle ohne Fehlermeldungen durchlaufen müssen ergibt sich ein relativ stabiler Hauptstand, der jederzeit als Grundlage für neue Features dienen kann. Bugs die nicht durch Tests erkannt wurden und erst im Release auffallen, können dank der kompakten Natur kurzer Integrationszyklen schnell erkannt und behoben werden. Kommunikation ist insofern ein wichtiger Aspekt der CI, als dass jederzeit alle Entwickler den Zustand der Software kennen können sollten. Hierbei hilft der Einsatz eines CI-Servers, der beispielsweise jederzeit wenn Code in den Hauptstand integriert werden soll, den kompletten Build Prozess nachvollzieht, alle Tests durchlaufen lässt und entsprechend alle auftauchenden Fehlermeldungen an das Team weitergibt.

**CSS** CSS steht für Cascading Style Sheets und ist eine Gestaltungssprache, über die sich die Darstellung von Informationen steuern lässt. Im Gegensatz zu HTML wird sich dabei nicht auf den Inhalt bezogen, sondern allein auf das Design.

**Deployment** Bezeichnet den Prozess der Installation und Konfiguration von Software auf Computern. Dieselbe Software kann mehrmals 'deployed' werden, z.B. wenn eine neue Version der Software herausgegeben wurde.

**Framework** Ein Framework stellt dem Softwareentwickler einen Rahmen mit festen Regeln und wiederverwendbare Strukturen zur Verfügung. Insbesondere definiert es den Kontrollfluss der Anwendung und die Schnittstellen der Klassen. Im Gegensatz zur Arbeit mit Klassenbibliotheken werden nicht bereits vorhandene Klassen und Funktionen verknüpft und verwendet um den Ansprüchen des Projekts zu genügen, stattdessen werden selbständig Funktionen und Klassen in einem vorgegebenen Rahmen implementiert, welche dann vom Framework genutzt werden um vordefinierte Aufgaben zu erledigen. Woraus sich auch ergibt das Frameworks stark Anwendungsspezifisch sind.

**Git** Git ist eine freie Software zur verteilten Versionsverwaltung.

**HTML** HTML steht für Hypertext Markup Language und ist eine Sprachkonvention, mit der digitale Dokumente für die Interpretation durch Web-Browser semantisch strukturiert werden. HTML wurde entwickelt, um Informationen digital austauschen zu können, ohne diese mehrmals umformen und anpassen zu müssen. Ein HTML-Dokument besteht aus einem

Doctype-Tag (Dokumenttypdeklaration), Head (Meta Informationen) und Body (der Inhalt) .

**JavaScript** JavaScript ist eine Skriptsprache, die ursprünglich dazu diente Benutzerinteraktionen mit Webseiten auszuwerten und daraufhin Inhalte anzupassen, nachzuladen oder zu erzeugen. Dieser Anwendungsbereich wurde inzwischen durch unzählige Bibliotheken stark erweitert. JavaScript ist eine dynamisch typisierte, objektorientierte, klassenlose Skriptsprache .

**Markdown** Markdown ist eine einfache Auszeichnungssprache, und lässt sich u.a. direkt in HTML übersetzen. Die Ausdrücke sind einfach zu lesen, zu schreiben und zu bearbeiten. Verwendet wird Markdown z.B. auf GitHub oder auch für Jekyll .

**Model-View-Controller** MVC ist ein Muster zur Implementierung von User-Interfaces, welches Quellcode in drei unabhängigen Komponenten organisiert. Dies soll die interne Repräsentation von Informationen und deren Verarbeitung von der Art wie diesem dem Nutzer präsentiert werden trennen. Dies erhöht die Wiederverwendbarkeit von Code und erlaubt die parallele Entwicklung der verschiedenen Teile. Das Model ist hierbei die zentrale Komponente, welche das Problem beschreibt. Es managt direkt die Daten und Logik des Programms. Die View-Komponente besteht aus der Darstellung der Daten z.B. durch Tabellen oder Diagramme und ermöglicht es verschiedene Views für verschiedene Nutzer zu definieren. Der Controller akzeptiert User-Input der dann in die entsprechenden Befehle für Model und View umgewandelt wird .

**QuitStore** Software die Git-Versionierung für sogenannte ‘Named Graphs’ (ein Schlüsselkonzept des Semantic Web das bewirkt, dass eine Menge an RDF Ausdrücken über eine URI identifiziert werden) ermöglicht .

**RDF** Kurz für Resource Description Framework. RDF ist ein Datenmodell, welches alle Daten in Triples der Form Subjekt, Prädikat, Objekt dargestellt. Ein Subjekt Leipzig wäre beispielsweise durch das Prädikat hatHochschule mit dem Objekt Universität Leipzig verbunden. Durch diese Schreibweise bildet sich eine Baumstruktur, in der Entitäten durch Relationen miteinander verknüpft sind .

**Skriptsprache** Eine Programmiersprache, die sich vor allem durch Interpreterausführung und oftmals entfallenden Deklarationszwang auszeichnet .

**SPARQL** Kurz für Semantic Protocol and RDF Query Language. SPARQL ist eine Anfragesprache, mit der man Informationen aus einem RDF Datenmodell extrahieren und modifizieren kann .

**Unit Test** Unit-Tests dienen dazu, diskrete Einheiten des Quellcodes auf ihre Funktionalität hin zu überprüfen. Hierzu werden im Vorfeld Testfälle mit möglichst umfassenden Eingabeparametern und den zu erwartenden Output definiert. Der Test besteht dann daraus, dass der Output, den die Einheit aus dem definierten Input produziert, mit dem erwarteten Output verglichen wird. Testing-Frameworks helfen beim erstellen und automatisierten Durchführen dieser Tests .