

Qualitätssicherungskonzept

Web Annotation mit Fragment Ids

Gruppe: swp12-9

Inhaltsverzeichnis

1. Dokumentationskonzept.....	2
1.1. Quelltexte.....	2
1.2. Änderungsdokumentation.....	4
1.3. Modellierungsdokumentation.....	4
1.4. Externe Dokumentation.....	4
2. Testkonzept.....	5
2.1. Komponententest.....	5
2.2. Integrationstest.....	6
2.3. Systemtest.....	6
3. Allgemeine Festlegungen.....	7

1. Dokumentationskonzept

Eine ausführliche Dokumentation der zu erstellenden Software und wichtiger Arbeitsschritte spielt eine entscheidende Rolle. Sie sichert zum einen den projektinternen Austausch aller Beteiligten und zum anderen die externe Kommunikation - beispielsweise zum Auftraggeber.

Des Weiteren gewährleistet sie die Möglichkeit für interessierte Entwickler (unter Umständen auch Nutzer) die Funktionalitäten und den Aufbau des Programms besser zu verstehen und vereinfacht ein eventuelles Vorhaben der Weiterentwicklung in erheblichen Maße. Gerade für Open Source Projekten sind einheitliche Richtlinien zur Beschreibung aller Projektkomponenten von hoher Bedeutung.

1.1. Quelltexte

Die Dokumentation sämtlicher implementierter Quelltexte und Testroutinen unserer Software geschieht auf Basis von phpdoc bzw. Javadoc. Zusätzlich wird gefordert, dass sämtliche Bezeichnungen und Kommentare in englisch verfasst werden - Umlaute werden äquivalent umgeformt.

Explizit wird dabei unterschieden zwischen:

Grundlegendes

- der Beginn einer jeder Klasse wird mit den folgenden Informationen eingeleitet:
 - Vor- und Nachname des Autors / der Autoren
 - Versionsnummer und Datum der Bearbeitung
 - kurze Beschreibung der Klasse

- komplexe Codeinhalte sind durch Kommentare separat zu beschreiben, besonders Verschachtelungen sind explizit zu kommentieren, d.h. jede Verzweigung

- Jede Klassenvariable, Methode und Funktion wird mit einem entsprechenden Kommentar über Zweck bzw. Inhalt dokumentiert
- Anweisungsblöcke jeder Kontrollstruktur werden mit geschweiften Klammern, unabhängig von der Anzahl der Anweisungen, umschlossen

Namens- und Benennungsgrundsätze

Alle Namenszuweisungen geschehen unter dem Aspekt der „Selbsterläuterung“, d.h. anhand der Bezeichnungen kann der Zweck nachvollzogen werden.

- Klassen beginnen mit Großbuchstaben
- Variablen und Methoden/Funktionen beginnen mit Kleinbuchstaben
- Konstantennamen bestehen ausschließlich aus Großbuchstaben
- zusammengesetzte Namen für Klassen, Variablen oder Funktionen/Methoden beginnen fortlaufend nach dem ersten Wort mit einem Großbuchstaben

Umbrüche und Einrückungen

Zur Gewährleistung von Übersichtlichkeit und Überschaubarkeit des Codes gelten folgende Festlegungen:

- Anweisungen (inklusive Funktions-/Methodenköpfen oder Kontrollstrukturen) enden immer mit einem Zeilenumbruch
- überschreitet eine Zeile 100 Zeichen so ist im Allgemeinen eine Aufteilung in zwei Zeilen notwendig
- zwischen Codeblöcken wie z.B. Methoden oder nach Kontrollstrukturen wird eine Leerzeile eingebettet
- Anweisungen werden zur verbesserten Übersicht eingerückt, eine Einrückung ist einen Tabulator lang

1.2. Änderungsdokumentation

Jede abgeschlossene (Teil-)Arbeit am Code muss ins Repository eingepflegt werden. Um die angewandten Änderungen/Neuerungen/Erweiterungen nachvollziehbar zu machen, müssen jegliche Commits kommentiert werden.

1.3. Modellierungsdokumentation

Modellierungsschritte werden mit Hilfe von Diagrammen, wie UML- und Sequenzdiagrammen, oder Use-Case-Szenarios verdeutlicht. Dabei wird besonders auf Übersichtlichkeit und Kompaktheit der einzelnen Entwürfe geachtet.

1.4. Externe Dokumentation

Um die Anwendung der Software auch ohne Verständnis des Quellcodes zu garantieren wird projektbegleitend ein Screencast in englischer Sprache erstellt. Dieser stellt eine Schritt-für-Schritt Anleitung zur Nutzung unserer Software dar.

Der Screencast wird auf einer für unser Projekt angelegten Website zu sehen sein. Auf dieser werden sich neben allgemeinen Informationen auch Inhalte, wie Hintergrundinformationen bzw. Ideen zum Entwurf und der technischen Umsetzung oder Beispiele getreu dem Motto „Eat-Your-Own-Dogfood“ (Markierung/Annotation markanter Inhalte auf der eigenen Website etc.) zu finden sein.

Zusätzlich zum o.g. wird eine Readme bzw. Install inklusive technischen Voraussetzungen der Software, einer Anleitung zu Installation und ähnlichem erstellt, um die Inbetriebnahme für andere Hosts zu erleichtern.

2. Testkonzept

Um stabile und funktionstüchtige Software zu einem Projekt zu realisieren, ist es ab einer gewissen Komplexität unumgänglich die einzelnen Komponenten und das zusammengefügte Konstrukt systematisch (schon im Laufe der Implementierung) zu testen. Um das breite Spektrum an Testmöglichkeiten zu schematisieren, werden die folgenden Kategorien unterschieden:

- Komponententest
- Integrationstest
- Systemtests

Zur Verwaltung und regelmäßigen Ausführung der Testfälle empfiehlt sich die Verwendung einer Fremdsoftware (z.B. Junit, PHPUnit).

Die Vorteile eines umfangreichen Testkonzepts sind im Wesentlichen die (weitest gehende) Fehlerfreiheit der einzelnen Module, sowie der vollständigen Software und die Qualitätssicherung der gesamten Funktionalität in dem Maße, wie sie im Pflichtenheft fixiert wurde.

2.1. Komponententest

Das Testen der einzelnen Komponenten beginnt schon frühzeitig im Laufe der Implementierungsphase. Dies beinhaltet programmierte Methoden, Klassen, Skripte etc. unter verschiedenen Einflussparametern unabhängig vom Gesamtkonstrukt auf zugewiesene Funktions- und Qualitätsmerkmale zu testen. Im Rahmen unseres Projekts wird ab einer festzulegenden Komplexität der einzelnen Komponenten und Module ein automatisiertes Testverfahren angestrebt.

Diese Automatisierung soll schnelle und regelmäßige Testdurchläufe ermöglichen und zu den einzelnen Testfällen eine Auswertung protokollieren. Daraus resultierend sollten Fehler im Sinne einer reibungslosen Umsetzung unseres Projekts frühzeitig und zuverlässig erkannt und gelöst werden.

2.2. Integrationstest

Integrative Tests dienen dazu, die Funktionalität bezüglich der Anbindung und Verknüpfung einzelner Komponenten zu sichern. Insbesondere diese Testphase wird in unserem Projekt zur Annotation auf Inhalte von Websites ein wesentlicher Bestandteil zu einer erfolgreichen Umsetzung der Softwarelösung sein. Beispielsweise bei einer Umsetzung mittels eines Browser-Plugins wird das Zusammenwirken der PHP- und JavaScript Skripte mit dem entsprechenden Browser eine elementare Bedeutung haben. In erster Linie sollte schon während der Implementierung achtsam auf mögliche Schnittstellen des Moduls geachtet werden. Darüber hinaus erlegt sich unser Team nach Fertigstellung einzelner Komponenten hinreichende Tests zur Verträglichkeit und Integration des Moduls im Gesamtkonstrukt auf.

2.3. Systemtest

Systemtests ermöglichen den Test des Gesamtprojekts aus der Sicht des Anwenders. Dabei kommt es weniger auf die einzelnen Komponenten und viel mehr auf die Funktionalität der Gesamtlösung an. Primär sind die im Pflichtenheft verankerten Funktionsfähigkeiten zu überprüfen. In unserem Projekt wird neben diesem Faktor, der hauptsächlich die Annotationsmöglichkeiten verschiedener Webinhalte prüft, die Verträglichkeit auf verschiedenen Betriebssystemen und falls möglich unter Verwendung unterschiedlicher Browser zu testen sein.

3. Allgemeine Festlegungen

Neben Dokumentations- und Testkonzept werden auch einige allgemeine Festlegungen getroffen, die den Erfolg des Projektes sichern sollen. Um sicher zu stellen dass das Team über den Verlauf des Projektes informiert ist, ist ein wöchentliches Treffen geplant. Dort werden Ergebnisse der Woche ausgewertet und Aufgaben für die nächste Woche verteilt.

Der Projektleiter dokumentiert das Treffen und verschickt die Aufzeichnung per Email an alle anderen Mitglieder der Gruppe. Zur weiteren Kommunikation ist ein email-Verteiler eingerichtet, mit dem alle Teammitglieder erreicht werden. Auf den wöchentlichen Treffen werden außerdem interne Deadlines festgelegt, zu welchen Arbeiten angefertigt werden. Bis zu dieser Deadline werden an den Verantwortlichen der jeweiligen Projektphase die Dokumente gesendet, sodass diese zusammengeführt werden können. Vor der Abgabe wird das fertige Dokument per Rundmail an alle verteilt, sodass jeder diese nochmal überprüfen kann. Für die Einhaltung der Deadlines sind der Phasen- sowie der Projektleiter verantwortlich.

Jeder ist zur Einhaltung des Dokumentations- und Testkonzepts verpflichtet, die beiden dafür Verantwortlichen überprüfen dies stichprobenartig. Sämtlicher Quelltext sowie alle erstellten Dokumente und Dokumentationen werden frei veröffentlicht.