

Qualitätssicherungskonzept

Inhaltsverzeichnis

1. Dokumentationskonzept	2
1.1 Bezeichner	2
1.2 Einrückungen	2
1.3 Kommentare	2
1.4 Schlüsselwort „this“ für globale Variablen	3
1.5 Automatische Quellcodedokumentation mit javadoc und doxygen	3
2. Testkonzept	3
2.1 Allgemeines Testkonzept	3
2.2 JUnit Tests	3
2.3 Oberflächentests mit Selenium	3
2.4 Komponententests	3
2.5 Integrationstests	4
2.6 Systemtests	4
3. Organisatorische Festlegung	4

1. Dokumentationskonzept

Das Dokumentationskonzept bezieht sich auf allgemein anerkannte Richtlinien für die Softwareentwicklung im Softwareentwicklungsprozess mit Java.

1.1 Bezeichner

Bei Bezeichnerrichtlinien wird zwischen Variablen, Methoden, Konstanten und Klassen unterschieden.

Entsprechende Namen sind nach Aufgabe zu wählen.

Standardbezeichnersprache ist Englisch.

- **Klassenbezeichner** beginnen mit einem Großbuchstaben.
Wenn mehrere Worte gewählt werden, muss jedes Wort auch mit Großbuchstaben beginnen.
z.B. CreateNewPage(...)
- **Variablen- und Methodennamen** mit kleinem Buchstaben, allerdings wird auch jedes weitere Wort im Namen großgeschrieben.
z.B. tempParameter = ...;
- **Konstantenbezeichner** werden grundsätzlich großgeschrieben. Mehrere Wörter werden mit einem Unterstrich getrennt.
z.B. TEMP_PARAMETER = ...;

1.2 Einrückungen

Einrückungen dienen der besseren Lesbarkeit des Quelltextes. Es ist zu Empfehlen die festgelegten Richtlinien sofern möglich (z.B. Eclipse) in der Entwicklungsumgebung zu definieren. Es ist dann möglich über die Formatierungsfunktion (Menü: Bearbeiten/Format) den Quelltext automatisch einzurücken.

- Jeder Unterbefehl ist innerhalb einer Methode um vier Leerzeichen einzurücken
- Neue Befehle sind in neuen Zeilen zu schreiben.
- Die maximale Zeilenlänge beträgt 80 Zeichen.
- In Funktionsköpfen kommt nur nach einem Komma ein Leerzeichen
- Öffnende und schließende geschweifte Klammern des Funktionsrumpfes stehen immer in neuen Zeilen

1.3 Kommentare

Kommentare dienen der Nachvollziehbarkeit des Quelltextes. Dokumentationssprache ist Deutsch.

- Funktionen und Methoden sind immer zu dokumentieren
- Bei eigenen Algorithmen ist deren Funktionsweise und Zweck zu dokumentieren
- Einzeilige Kommentare beginnen mit //
- Mehrzeilige Kommentare werden mit /* ... */ umschlossen

1.4 Schlüsselwort „this“ für globale Variablen

Zur Verbesserung der Verständlichkeit des Quellcodes ist bei der Verwendung von definierten globalen Variablen das Schlüsselwort `this`. Voranzustellen.

Dadurch können lokale von globalen Variablen unterschieden werden.

z.B.

`this`.tempVariable für globale Variablen

tempVar für lokale Variablen

1.5 Automatische Quellcodedokumentation

Zur schnellen Einarbeitung und Übersichtlichkeit ist aus den Quellcodedatei eine Dokumentation zu erstellt.

Dies wird mit dem Tool „Javadoc“ umgesetzt.

Bei der Ausführung wird eine Übersicht über alle Klassen- und

Methodendokumentationen in Form einer HTML Übersicht erstellt.

Entwicklungsdokumentation mit Übersicht der einzelnen Klassendiagramme wird mit doxygen und der Erweiterung graphviz realisiert.

2. Testkonzept

2.1 Allgemeines Testkonzept

Um die weiterhin komplette Funktionsfähigkeit des OLAT-PA Moduls unter OLAT 7 zu gewährleisten wird hauptsächlich mit Hilfe des Junit- und Selenium Frameworks getestet.

2.2 JUnit Tests

Mit Hilfe von Unit-Tests werden die geänderten Klassen bzw. Methoden automatisch getestet.

Dazu werden eigene Testklassen für die zu testende Klasse geschrieben. Die Testklasse startet die zu testende Klasse und prüft nach der Ausführung das Ergebnis, welches dem vorgegeben entsprechen muss.

2.3 Oberflächentests mit Selenium

Das Selenium ist ein Testframework speziell für Webanwendungen. Es wird für das Testen aus Benutzersicht eingesetzt, da es die Oberfläche von OLAT testet.

Mit Hilfe von Selenium-IDE und einem Firefox-Plugin werden Testfälle definiert.

Ebenfalls wie bei Unit-Tests können die Ergebnisse geprüft werden.

2.4 Komponententests

Zu Testen sind hierbei die einzelnen Klassen und Methoden. Hierfür werden parallel zur Implementierung der einzelnen Stories Testbeispiele festgelegt um die Klassen und Methoden hinsichtlich ihrer Funktionalität zu prüfen. Hierfür kann auch das

Testframework junit verwendet werden, da dies besonders für des automatisierte Testen einzelner Klassen und Methoden geeignet ist.

2.5 Integrationstests

Der Integrationstest überprüft das Zusammenspiel der einzelnen Komponenten und setzt somit auch einen erfolgreichen Komponententest der Stories voraus. Getestet wird regelmäßig zu einem festgesetzten Zeitpunkt mit dem aktuellen Stand der Erweiterung.

2.6 Systemtests

Es wird das Verhalten der Erweiterung bestehend aus allen Komponenten aus Nutzersicht bezüglich der festgelegten Anforderungen überprüft. Da mit Selenium IDE bereits aus Anwendersicht getestet wird, wird der Systemtest ebenfalls regelmäßig zu einem bestimmten Zeitpunkt durchgeführt.

3. Organisatorische Festlegungen

- An den beiden wöchentlichen Teamtreffen wird das weitere Vorgehen sowie die Arbeitsteilung besprochen sowie über vorliegende Ergebnisse diskutiert.
- Erste Ergebnisse sind spätestens in der Vorwoche des Abgabetermins auf den Teamtreffen zu präsentieren.
- Die Kommunikation außerhalb der Teamtreffen erfolgt hauptsächlich per Mail.
- Die Ergebnisse der Teamtreffen werden protokolliert und anschließend per Mail dem Projektteam schriftlich zur Verfügung gestellt.
- Die Zusammenführung der einzelnen Ergebnisse obliegt dem jeweiligen Phasenverantwortlichen. Ihm werden alle Einzelergebnisse zu einer internen Deadline zugeschickt.
- Die Umsetzung oben genannter Dokumentationsrichtlinien obliegt jedem Programmierer selbst und wird durch den Verantwortlichen für Dokumentation&QS überprüft.
- Alle Gruppenmitglieder machen sich vor Beginn der Implementierung mit den festgelegten Dokumentationsrichtlinien vertraut.
- Die Verwaltung der Quelltexte findet über Mercurial statt.
- Die Änderungen an den Quelltexten werden nach Prüfung auf Buildfähigkeit auf einen zentralen Server committet.
- Treten während der Implementierung Fehler auf, werden diese mit korrekter nachvollziehbarer Beschreibung dokumentiert.