

Qualitätssicherungskonzept

1 Dokumentationskonzept

- Interne Dokumentation _____ 2
 - Dokumentation der Funktionen _____ 2
 - Programmierstandards _____ 2
- Externe Dokumentation
 - Dokumente _____ 3
 - Was muss in jedem Dokument vorhanden sein? _____ 3

2 Testkonzept

- Einleitung _____ 3
- PHPUnit _____ 4
- Komponententest _____ 4
- Integrationstest _____ 4
- Systemtest _____ 4
- Oberflächentest _____ 4
- Abnahmetest _____ 4
- Dokumentation _____ 5

3 Organisatorische Festlegungen

- Treffen _____ 5
- Übersicht und Kontakt _____ 5

Qualitätssicherungskonzept

1 Dokumentationskonzept

Interne Dokumentation

Unter der internen Dokumentation versteht man die Kommentierung innerhalb des Quellcodes. Wir haben uns grundsätzlich darauf geeinigt die Dokumentation auf Englisch zu verfassen. Außerdem werden wir uns, da wir mit PHP programmieren, an die Standards des phpDocumentors halten. Zu finden sind diese unter:

http://manual.phpdoc.org/HTMLSmartyConverter/HandS/phpDocumentor/tutorial_manual.pkg.html

Dokumentation der Funktionen

Sämtliche Funktionen bzw. Klassenmethoden sollen in einem vorangestellten Dokumentationsblock bezüglich Funktion, zu übergebende Parameter und Rückgabewerte beschrieben werden. Sollte eine Parameter mehrere Typen zulassen, so müssen alle Möglichkeiten, mit | getrennt, angegeben werden. Folgendes Beispiel soll die Funktionen-Dokumentation veranschaulichen:

Programmierstandards

Die folgenden Standards sollen eingehalten werden:

- Einrückungen:

Damit der Quellcode möglichst übersichtlich und leicht zu lesen ist, werden folgende Regeln festgelegt:

- i. Jeder Befehl steht auf einer separaten Zeile.
- ii. Die Zeilenlänge sollte 80 Zeichen nicht überschreiten (Ausnahmen sind möglich, bedingt durch z.B. einen langen hash-Code in einer if- Anweisung).
- iii. Unterbefehle wie zu in if-Blöcken, werden um 4 Leerzeichen eingerückt.
- iv. Hinter Kommas und runden Klammern steht grundsätzlich ein Leerzeichen.
- v. Öffnende und schließende geschweifte Klammern solcher Funktions- Blöcke stehen auf separaten Zeilen.

- Bezeichnungen:

Alle Variablennamen werden Englisch verfasst, des Weiteren gelten folgende Punkte:

- i. Variablen- und Funktionsnamen beginnen stets mit kleinem Buchstaben. Sollten die Namen mehrere Wörter enthalten, so wird der Anfangsbuchstabe jedes neuen Wortes groß geschrieben.
- ii. Klassennamen beginnen mit großen Buchstaben, außerdem gelten die gleichen Regeln wie in Punkt i.
- iii. Konstantennamen werden vollständig aus Großbuchstaben gebildet. Sollten sie aus mehreren Wörtern bestehen, so werden diese Wörter mit Unterstrichen getrennt.
- iv. Die Bezeichner sind so zu wählen, dass die möglichst nicht mit anderen

Qualitätssicherungskonzept

Bezeichnern verwechselt werden können. Des Weiteren müssen sie selbsterklärend sein, d.h. Auskunft über z.B. die Funktion der bezeichneten Variable geben. Dies bezieht mit ein, dass die Bezeichner nicht zu kurz zu wählen sind und keine Ziffern enthalten sollten, da Ziffern grundsätzlich keinen Aufschluss auf die Funktion der Variablen geben.

- Eine PHP-Datei enthält immer nur eine einzige Klasse.

Externe Dokumentation

Dokumente

Alle Dokumente werden ab sofort mit OpenOffice erstellt und in PDF umgewandelt. Hintergrund ist das sich nicht jeder mit Latex auskennt. Dies zu erlernen würde zu viel Zeit beanspruchen.

Was muss in jedem Dokument vorhanden sein?

In jedem Dokument müssen Name, Abnehmer, Datum und Dokumenttitel vorhanden sein. Name des Verfassers, Abnehmer und Datum wird in der Kopfzeile Rechtsbündig versehen. Der Dokumententitel steht im Dokument (nicht in Kopf- oder Fußzeile) zentriert.

Ein Inhaltsverzeichnis, sowie die Nummerierung muss erstellt werden. Bei kurzen Dokumenten (wie Briefen, Bekanntmachungen) kann das Inhaltsverzeichnis ausgelassen werden.

2 Testkonzept

Einleitung

Bei der Entwicklung von Software ist ein Testkonzept erforderlich. Den bei zunehmendem Umfang des Projektes können immer mehr schwerwiegende Fehler auftreten, welche dann nur unter viel Aufwand zu finden und unter bestimmten Umständen nur schwer zu beheben sind. Aus dem Grund ist es wichtig frühestmöglich mit den Tests während der Entwicklung anzufangen umso auftretende Fehler schneller zu erkennen und dadurch leichter diese Fehler zu beheben.

Um diese Tests überschaubar und strukturiert zu gestalten, werden diese in Testphasen eingeteilt:

- Komponententest
- Integrationstest
- Systemtest
- Oberflächentest
- Abnahmetest

Qualitätssicherungskonzept

PHPUnit

PHPUnit ist ein Framework zum Testen von PHP-Skripten. Dieses Framework ist geeignet für automatisierte Tests einzelner Klassen und Methoden.

Komponententest

Der Komponententest ist die Testphase in der die einzelnen Klassen und Methoden auf ihre Funktionalität getestet werden, um so frühzeitig Fehler zu erkennen und diese zu beheben. Hierbei dient uns als Hauptwerkzeug das Framework PHPUnit, mit dem wir die PHP-Komponenten unseres Projektes testen werden.

Hierbei wird eine Testumgebung erzeugt, mit geeigneten Testklassen.

Diese Testklassen sind wie folgt zu bezeichnen:

Testklassen heißen „Test“ gefolgt vom Klassennamen

Diese Testklassen werden in geeignete Testpakete (Testsuiten) zusammengefasst.

Diese Testpakete sind wie folgt zu bezeichnen:

Testpakete heißen „Testsuit“ gefolgt vom Paketnamen

Integrationstest

Der Integrationstest ist die Testphase in der das Zusammenwirken zusammenhängender Klassen getestet wird. Dies dient hierbei frühzeitig Fehler unter den Klassen zu finden und zu beheben.

Systemtest

Der Systemtest ist die Testphase in der alle Komponenten des Programmes zusammengefügt werden. In dieser Phase wird nun aus der Sicht des Anwenders getestet, auf die Funktionalität, ob alle Anforderungen umgesetzt wurden und auf das problemlose Abläufen des Programms und letzte Fehler überprüfen.

Oberflächentest

Der Oberflächentest ist die Testphase zu testen der Benutzeroberfläche, auf ihr Funktionalität und aufkommende Fehler bei der Bedienung dieser Oberfläche.

Abnahmetest

Der Abnahmetest ist die letzte Testphase, nach Fertigstellung des Projektes, die durch den Kunden erfolgt. Dem Kunden wird hier das fertige Programm vorgestellt und er testet dann, ob das fertige Programm seinen Vorgaben entspricht, zu seiner Zufriedenheit funktioniert und er mit der Umgebung zurechtkommt.

Qualitätssicherungskonzept

Dokumentation

Auftretende Fehler müssen dokumentiert werden, wenn möglich die Ursache und Lösung mit notiert werden. Dies unterstützt dann bei Auftreten gleichartiger Fehler das Finden und die Behebung der Fehler.

3 Organisatorische Festlegungen

Treffen

Jede Woche findet ein Treffen ohne Betreuer statt. In diesen Meetings wird der aktuelle Stand des Projekts und Probleme besprochen. Sollte es ein Problem geben, dass keiner aus der Gruppe lösen kann, muss zusätzlich ein Treffen mit den Betreuern organisiert werden.

Bei Sonderfällen wird auch ein Treffen im Skype ausgerufen, in dem dringende Angelegenheiten besprochen werden.

Mitglieder, die nicht zum Treffen erscheinen konnten, werden frühestmöglich über die wichtigsten Details informiert.

Übersicht und Kontakt

Jedes Mitglied muss sich bei Github zu registrieren, da mithilfe dieser Plattform die Verwaltung des Projektes abläuft. In Github gibt es ein Wiki, dass die Mitglieder über den Aufbau des Projekts informieren soll. Jeder im Projekt kann über Github andere über Probleme und Schwierigkeiten informieren.

Jedes Mitglied ist dazu verpflichtet regelmäßig ins Github zu schauen.