

Dokumentationskonzept

Mit den ersten greifbaren Ergebnissen der Planungsphase beginnt die Dokumentation und mit Beendigung der Hilfe bzw. des Handbuchs ist Sie vielleicht auch das Letzte, das am Projekt getan wird. Somit wird klar, dass die Dokumentation ein grosser und vor allem wichtiger Bestandteil der Softwareentwicklung ist. Aufgrund der breit gefächerten Anwendungszielgruppe von Laien bis Experten und Wartungspersonal und deren unterschiedlichen Wissen, ist es unumgänglich die einzelnen Schichten für den jeweiligen Nutzer verständlich zu dokumentieren. Im Folgenden werden nun, die für unser Projekt geltenden Standards und Vorgehensweisen vereinbart und erläutert, dazu betrachten wir folgende 2 Sichten:

1. interne bzw. Quellcodenahe Sicht - hier steht die Quellcodedokumentation im Vordergrund, um unbeteiligten Programmierern die Wartung, Pflege und Erweiterung des Produktes zu ermöglichen:

- 1.1. guter code
- 1.2. Quellcodedokumentation
- 1.3. (javadoc Extraktion)
- 1.4. JUnit - Test - Dokumentation

2. externe Projektdokumentation - dies ist die Sicht des Kunden, die Funktionalität und Benutzbarkeit sind vorrangig:

- 2.1. Dokumentenbeschreibung
- 2.2. Designbeschreibung
- 2.3. Handbuch
- 2.4. Hilfe
- 2.5. integrierte Dokumentation

1.1. Unter "guten Code" versteht man eine Reihe von Standards und Normen die für eine klare Struktur und ein vereinfachtes Verständnis des Quellcodes sorgen sollen. Von immenser Bedeutung sind hier aussagekräftige problemnahe Bezeichner für Klassen, Interfaces, Methoden, Variablen usw.. Desweiteren sind Strukturen wie Schleifen einzurücken, damit klar wird, welche Anweisungen zu welchen Strukturen gehören. Bei besonders langen Strukturen ist ebenso das Ende der selbigen deutlich zu machen. Werden Variablen immer wieder mit der selben Belegung verwendet, so bedeutet guter Code, dass an deren Stelle Konstanten eingeführt werden.

Akteure: Implementierer

Verantwortlicher: Implementierer

1.2. Da wir in Java programmieren, verwenden wir eine Quellcodedokumentation gemäß den Konventionen von javadoc. Für Klassen, Methoden und Interfaces ist ein Kommentarkopf zu erstellen der die Funktion wiedergibt und die jeweils nötigen Tags enthält. Nach den Konventionen für guten Code sollten sich die Bezeichner selbst erklären, aber speziell Parameter, Rückgaben und Exceptions sind per Tag zu erläutern. (@author, @param, @return, @throws, ... siehe Beispiel in Anlage) Unübersichtliche oder komplizierte Codepassagen werden vom Implementierer mit kurzen, präzisen Blockkommentaren versehen, um anderen Programmierern Einblick zu gewähren. Bei Schleifen und Case-Strukturen sind ebenfalls alle "Ausprägungen" mit einem kurzen Kommentar zu versehen.

Akteure: Implementierer

Verantwortlicher: Dokumentierer

1.3. Wenn dies alles ordnungsgemäß gemacht worden ist, extrahiert man jetzt mit javadoc die Dokumentationen und erhält für jede Klasse und jedes Package eine eigene Dokumentationsdatei! Für wichtige Klassen sollte man überlegen, diese

Doku in die Hilfe einzubauen, um das Verständnis zu erleichtern!

Akteure: Dokumentierer + Implementierer

Verantwortlicher: Dokumentierer

- 1.4.** Die Testdokumentation in unserem Projekt gestaltet sich wie eben beschrieben, da alle Testfälle Programme sein werden. Damit besitzen sie natürlich ihren eigenen Quellcode und können, wie in 1.2. bzw 1.3. beschrieben, mit javadoc gehandhabt werden. (Ein Testkonzept wird im späteren Verlauf noch erarbeitet.)

Akteure: Tester, Implementierer, Dokumentierer

Verantwortlicher: Dokumentierer
- 2.1.** Bereits in der Planungs- und Definitionsphase ist bsw. auf eine Legende zu Artefakten zu achten und zu speziellen Entwürfen eine kurze Dokumentation zu verfassen. Der das Projekt begleitende Glossar sollte ebenfalls so früh wie möglich begonnen werden, um Unklarheiten zu beseitigen. In diesem Glossar werden alle Begriffe, die für das Projekt von Bedeutung sind erfasst.

Akteure: alle

Verantwortlicher: Dokumentierer
- 2.2.** Mit der Designstudie wird die Grundlage für das Projekt geschaffen - in einem frühen Stadium wird das gesamte Konzept und die damit verbundene Architektur dargelegt. Dem Kunden ist es somit von beginn an möglich, Einfluss auf die Entwicklung des Projektes zu nehmen, indem er konzeptionelle Fehler aufzeigt und sich einen Eindruck von den grundlegenden Eigenschaften des Systems zu verschaffen. Ebenso ist es eine gute Möglichkeit projektfremden Programmierern einen guten Einblick zu bieten.

Akteure: alle

Verantwortlicher: Projektleiter
- 2.3.** Von grösster Bedeutung bei der externen Betrachtung ist das Handbuch. Es ist direkt an den Benutzer gerichtet und muss daher alle notwendigen Informationen enthalten, um Diesem die uneingeschränkte Bedienung zu ermöglichen. Ziel ist es mit klar erklärten Beispielen alle Funktionen aufzuzeigen und häufig auftretende Probleme im Voraus zu beseitigen.

Akteure: Implementierer, Dokumentierer, Tester, Projektleiter

Verantwortlicher: Projektleiter
- 2.4.** Zum fertigen Produkt wird abschliessend eine Hilfe beigefügt, die durch Tooltips einen zwar oberflächlichen, aber leicht verständlichen Überblick der Funktionalität gibt! Die Tooltips sind selbsterklärend und sorgen für eine effiziente Nutzung. Desweiteren ist es Standard eine Installationsanleitung oder Readme zu verfassen, die sich aber nur an den Administrator richtet und diesen über die technischen Voraussetzungen und Hinweise in Kenntnis setzt. Dabei ist es im Gegensatz zur Hilfe ausreichend auf eine technisch-verständliche Sprache zu achten.

Akteure: Implementierer, techn. Assi, Tester, Dokumentierer

Verantwortlicher: Dokumentierer
- 2.5.** Unter integrierter Dokumentation versteht man die ständige entwurfsnahe Einarbeitung von anfallenden Änderungen in die bereits angefertigten Dokumente und Konzepte. Vor allem das Entwurfskonzept ist somit einer ständigen Entwicklung unterworfen. Änderungen im geplanten Programmierablauf müssen dabei so deutlich gemacht werden, das für unbeteiligte Programmierer der Zusammenhang weiter leicht erkennbar bleibt.

Akteure: Implementierer, Tester, Dokumentierer, Projektleiter

Verantwortlicher: Projektleiter

