

Dokumentationskonzept

1. Quellcode-Dokumentation (intern)

1.1 Bestandteile

- Quellcode selbst
- javadoc Dokumentation
- Block- und Inline-Kommentare des Quellcodes

1.2 Qualitätsstandards & Vorgehensweisen

1.2.1 Java Source Files Dateiaufbau

Java source files sollten folgenden Aufbau haben:

- Beginnende Kommentare
Die Quellcode-dateien sollten mit einem Kommentar beginnen, der Informationen zum Klassennamen, zur Version, zum Datum und zum Urheberrecht enthält.
- Paket und Import Statements
Die Package-Statements beschreiben die Paket-Zugehörigkeit, die Import-Statements die importierten Klassen.
- Klassen- und Interfacedeklarationen
Sie sollten mit einem JavaDoc-Kommentar beginnen, der nähere Informationen zur Klasse enthält. Dabei sollte man nie mehr als eine public Klasse pro Datei deklarieren (Ausnahme: Helferklassen).
Danach sollte das class- oder interface-Statement, die statischen Klassenvariablen, die Instanzvariablen, alle in der Reihenfolge public, protected und privat folgen.
Danach kommen der Konstruktor und die nach Funktionalität geordneten Methoden.

1.2.2 Bezeichner

- Selbstsprechende Bezeichner wählen
- Substantive für Klassen (1. Buchstabe groß)
- Verben für Methoden (kleingeschrieben)
- Substantive oder verständliche Abkürzung bei den Attributen (kleingeschrieben)
- Substantive für Objekt-namen (kleingeschrieben)
- Bei zusammengesetzten Bezeichnern beginnt jedes weitere Wort mit einem Großbuchstaben

1.2.3 Zeilenlänge

Die Zeilen sollten nicht länger als 80 Zeichen sein.

1.2.4 Zeilenumbruch

Falls eine Zeile doch länger als 80 Zeichen ist, sollte man sie nach folgenden Regeln umbrechen (laut Sun).

- Umbrüche nach Kommas
- Umbrüche vor Operatoren
- Ziehe Umbrüche auf hohen Ebenen, denen auf niedrigen Ebenen vor
- Die umgebrochen Zeile wird im Einzug mit dem vorhergehenden Ausdruck angepasst. Sollte dies zu unübersichtlichen Code führen, verwendet man acht Leerzeichen
- Innerhalb if-Statements verwendet man acht Leerzeichen

1.2.5 Kommentare

- Implementierungskommentare
 - beginnen mit // oder stehen in /* ... */
 - sollten den Code näher beschreiben
 - sollten sparsam eingesetzt werden
- Javadoc-Kommentare
 - sind spezielle Kommentare, die in /** ... */ geschrieben werden
 - sind zur Beschreibung von Klassen, Variablen oder Methoden zu verwenden
 - die folgenden speziellen Befehle, die mit @ beginnen, sind zu verwenden:
 - @author Gibt den Autor an
 - @param Beschreibung der Übergabeparametern
 - @return Beschreibung der Rückgabeparametern
 - @exception Beschreibung der möglichen Ausnahmen
 - @see Verweis auf anderen Klassen

1.3 Verantwortlichkeiten

Für die Quellcode-Dokumentation sind die Entwickler zuständig.

2. Entwurfsdokumentation (extern)

2.1 Bestandteile

- Design-Beschreibung
- Online-Hilfe
- Benutzerhandbuch

2.2 Qualitätsstandards & Vorgehensweisen

2.2.1 Design-Beschreibung

Die Design-Beschreibung dokumentiert die vom Team bezüglich der Anwendung getroffenen Entscheidungen. Vor allem legen wir sehr großen Wert auf die Dokumentation der Paket- und Klassenstruktur, damit die zukünftigen Erweiterungen ohne Probleme bewerkstelligt werden können.

2.2.2 Online-Hilfe & Benutzerhandbuch

Das Benutzerhandbuch in Kombination mit der Online-Hilfe entsteht parallel zur Entwicklung der Anwendung. Dem Anwender sollten alle Funktionen des Produkts umfassend und verständlich erklärt werden. Das Handbuch sollte gut strukturiert und vollständig sein. Es soll außerdem die Informationen zu Systemvoraussetzungen, Installation und Pflege des Programms enthalten.

2.3 Verantwortlichkeiten

Für die Entwurfsdokumentation sind die Zuständige für Modellierung (Design-Beschreibung), der technische Assistent (Online-Hilfe) und die Zuständige für Dokumentation & Qualitätssicherung (Benutzerhandbuch) verantwortlich.

Bemerkung: Bei der Zusammenstellung des Dokumentationskonzepts haben wir uns an die in der Aufgabenstellung empfohlenen Quellen gehalten.