

Proseminar Wintersemester 2007 / 2008

# DOXYGEN

source code documentation generator tool

Johannes Blatz

Medizinische Informatik / 4. Semester

Wilhelm- Busch- Str. 2

76684 Östringen

Email: [johannes.blatz@onlinemed.de](mailto:johannes.blatz@onlinemed.de)

Universität Heidelberg / Hochschule Heilbronn

## Inhaltsverzeichnis

Einleitung .....	3
Hintergrund zu Doxygen.....	4
Alternative Dokumentationssysteme .....	4
Stand der Entwicklung .....	4
Installation.....	5
Installationsvoraussetzungen .....	5
Durchführen der Installation .....	5
Empfehlenswertes Plug-In .....	7
Plug-In: Graphviz.....	7
Einrichten und erste Schritte mit Doxygen – der doxywizard.....	8
Interessante Zusatzmöglichkeiten .....	14
Graphviz Plug-In einbinden .....	16
Die fertige Dokumentation.....	20
Erweiterte Funktionen .....	21
Präsentation der Suchfunktion und von Graphviz .....	23
Das Dokumentieren .....	24
Doxygen- Tags.....	26
HTML- Tags .....	27
Alternativen.....	28
Diskussion.....	29
Bekannte Beispiele, in denen Doxygen eingesetzt wird .....	30
Quellenverzeichnis.....	31
Downloadverzeichnis.....	31

## Einleitung

Diese Ausarbeitung soll Sie in das Dokumentationstool Doxygen, ein Open Source Dokumentationssystem für annähernd alle wichtigen Programmiersprachen, einführen. Es folgen Informationen zu Installation, Verwendung, Nutzen und eventuelle Vor- bzw. Nachteile der Software.

Die Zielgruppe dieser Arbeit sind Menschen, die sich mit Softwareentwicklung und deren Dokumentation befassen. Aus diesem Grund wird ein grundsätzliches Softwareentwicklungsverständnis vorausgesetzt. Alle aufgeführten Beispiele sind in der Programmiersprache Java verfasst.

Dieses Dokument ist als Projektarbeit eines Seminars für Entwicklungswerkzeuge entstanden und führt Softwareentwickler in den Umgang mit dem Dokumentationstool Doxygen ein. Ziel ist es, alle wichtigen Grundfunktionen zu erläutern und deren Nutzen anhand von Beispielen schlüssig darzustellen.

Das Referat befasst sich mit der zum Zeitpunkt der Veröffentlichung dieses Dokumentes aktuellen Version 1.5.4 der Doxygen- Software. Es ist nicht auszuschließen, dass die Bedienung der Software in zukünftigen Versionen abweichen kann.

Man sollte deshalb zu jeder Version der Software die offiziellen Benutzerhandbücher studieren.

## Hintergrund zu Doxygen

Für nahezu jede Programmiersprache existiert ein eigenes Dokumentationssystem, teilweise gibt es sogar mehrere parallel. Doxygen stellt ein Dokumentationssystem dar, das in allen gängigen Programmiersprachen eingesetzt werden kann. Der große Vorteil hierbei liegt darin, dass jedes



Abbildung 1: Doxygen- Logo (Dimitri van Heesch)

Dokumentationssystem seine eigene Grammatik, Richtlinien und einen eigenen Stil besitzt, die untereinander meist inkompatibel sind. Mit Doxygen können eine Grammatik und ein Stil in allen Entwicklungsaufgaben eingesetzt werden. Ferner wird die Übersichtlichkeit der Dokumentationen verschiedener Projekte erhöht, da jedes Dokumentationssystem seine spezifische Ausgabe und Logik besitzt. Durch den Einsatz eines einzigen Systems sieht die Dokumentation eines Java-, C++- oder Web- Projekts immer gleich aus, wodurch die Analyse deutlich vereinfacht wird.

## Alternative Dokumentationssysteme

Neben Doxygen existiert eine Reihe weiterer Dokumentationssysteme mit oftmals hoher Verbreitung. Eines der bekanntesten Beispiele ist Javadoc. Die meisten dieser Systeme arbeiten nur mit einer spezifischen Programmiersprache zusammen, im Falle von Javadoc nur mit Java. Doxygen unterstützt Javadoc, wodurch der einfache Javadoc- Stil in allen Programmiersprachen angewendet werden kann, auch weit über die Grenzen von Java hinaus.

## Stand der Entwicklung

Das Projekt wurde 1997 gestartet und verfolgt, wie fast jedes Internetprojekt, die These, dass die Entwicklung niemals abgeschlossen ist und somit der Beta- Status niemals verlassen werden kann. Es findet sich immer Raum für neue Innovationen und Verbesserungen.

## Installation

Doxygen ist für alle gängigen Plattformen wie Windows, Linux, Solaris, BeOS, IRIX, OS/2, FreeBSD und Mac OS X verfügbar. In dieser Ausarbeitung wird die Stand- Alone- Version für Windows beschrieben, die eine eigene Benutzeroberfläche (GUI) mit Assistenten besitzt, die derzeit nicht für alle Plattformen verfügbar sind. Alle vorhandenen Versionen können unter (1) des Downloadverzeichnisses heruntergeladen werden.

## Installationsvoraussetzungen

Die hier vorgeführte Installationsroutine ist für alle Windowsversionen ab Windows 95 bis Windows Vista gültig und getestet.

## Durchführen der Installation

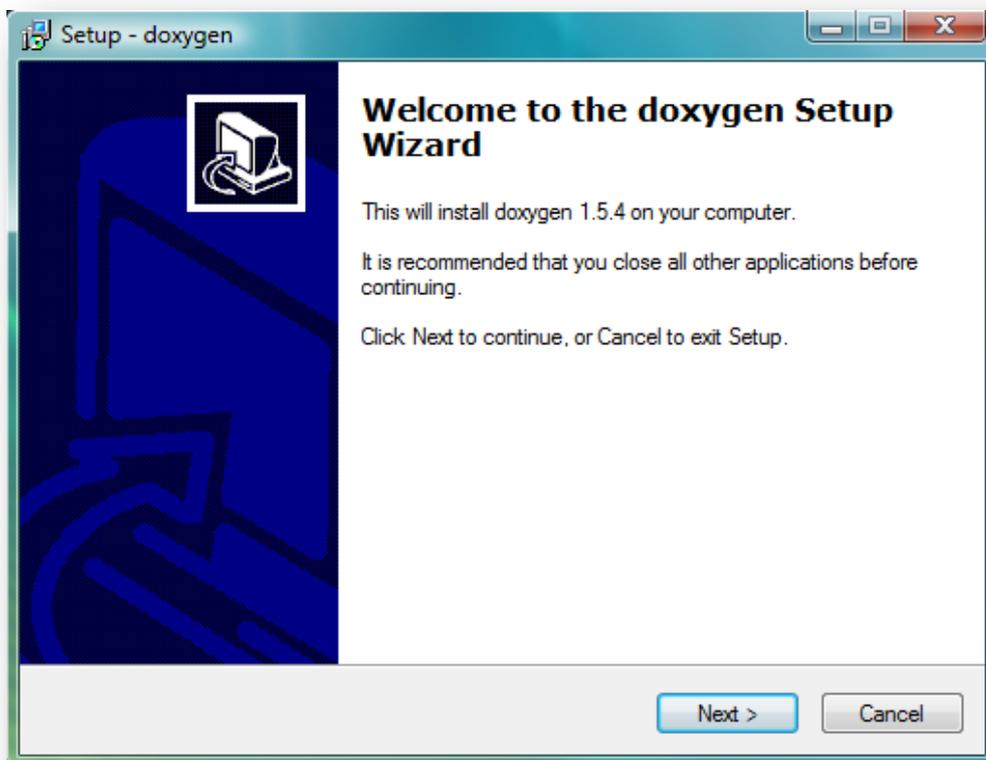


Abbildung 2: Startbildschirm des Setup- Programms (Screenshot)

Die Version für Windows besitzt ein eigenes Setup- Programm, das die Installation stark vereinfacht.

Doxygen wird unter der GNU GENERAL PUBLIC LICENSE in der Version 2 veröffentlicht.

Standardmäßig bringt Doxygen bereits einige Beispielprojekte mit sich, die den Einstieg deutlich vereinfachen.

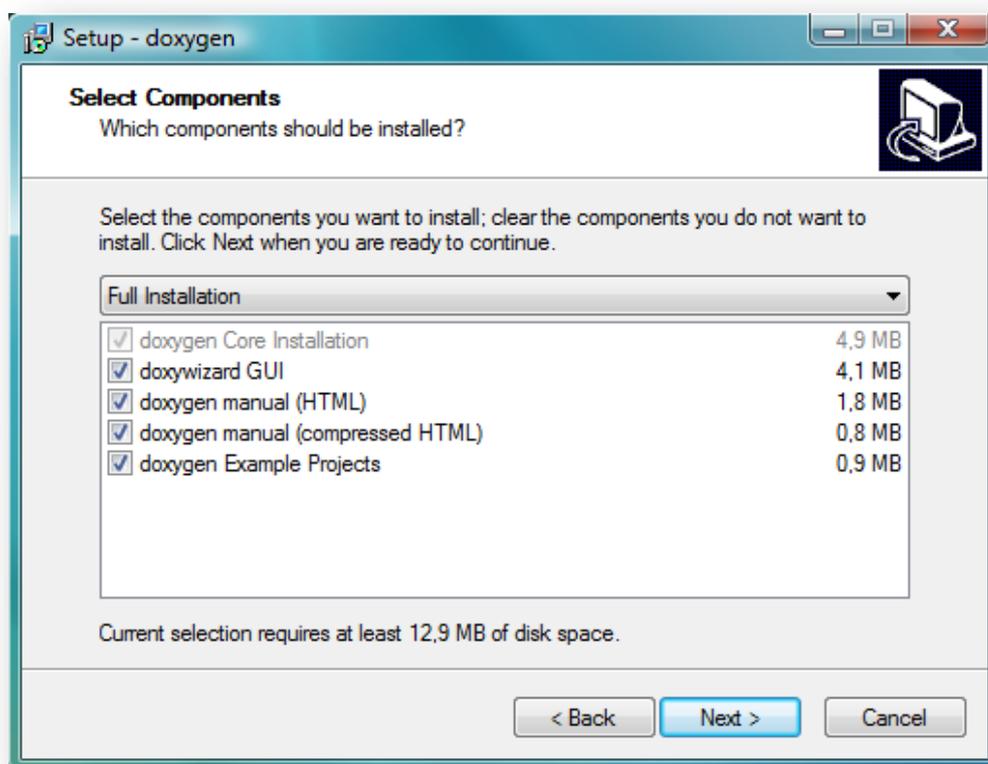


Abbildung 3: Mitgelieferte Pakete des Setup- Programms (Screenshot)

## Empfehlenswertes Plug-In

### Plug-In: Graphviz

Graphviz ist eine Software zur Visualisierung von Graphen, die in Doxygen Zusammenhänge von Klassen und Methoden übersichtlich darstellen kann. Veröffentlicht wurde Graphviz unter der GPL GENERAL PUBLIC LICENSE in der Version 1.0 und ist für alle wichtigen Plattformen verfügbar. Beschrieben wird hier, wie bereits bei Doxygen angekündigt, nur die Windows- Version 2.16.1.

Verfügbar sind alle Versionen als Download unter (2) des Downloadverzeichnisses

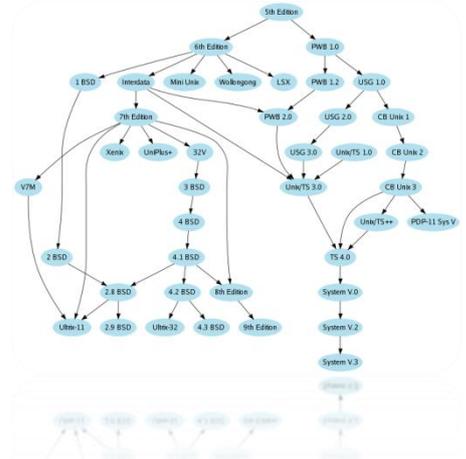


Abbildung 4: Beispielgraph (Ian Darwin und Geoff Collyer)

## Einrichten und erste Schritte mit Doxygen – der doxywizard

In den folgenden Abschnitten wird der Einstieg in Doxygen anhand eines einfachen Java- Beispiels erklärt. Das Beispiel hat den Namen „DoxygenTestProjekt“ und besitzt nur eine Klasse „Test“.

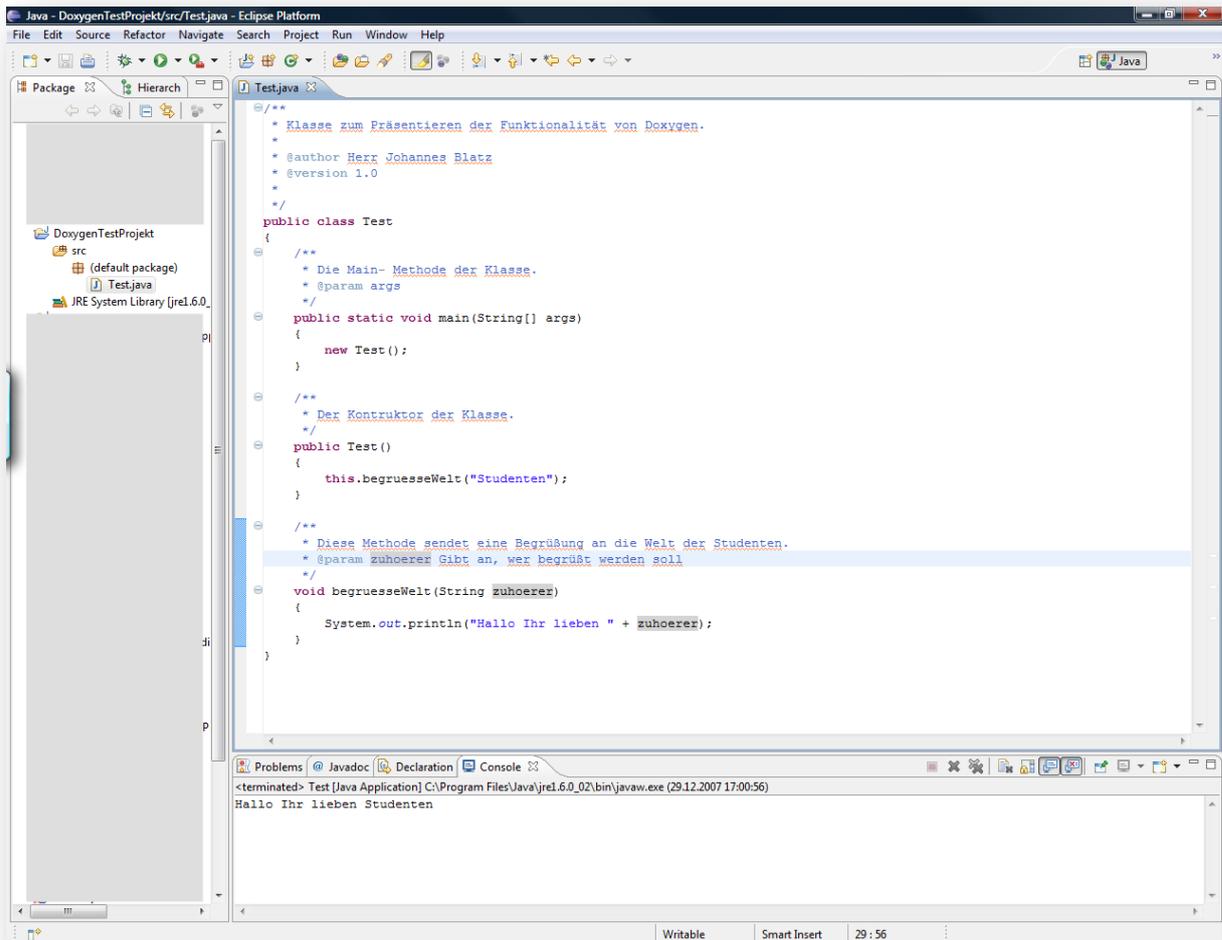


Abbildung 5: Einfaches Java- Beispiel zur Dokumentationsdemonstration (Screenshot)

Das Beispiel dient ausschließlich der Demonstration der einfachsten Funktionen.

Die Einrichtung von Doxygen erfolgt über eine eigene Oberfläche, die in dieser Form derzeit nicht für alle Plattformen zur Verfügung steht.

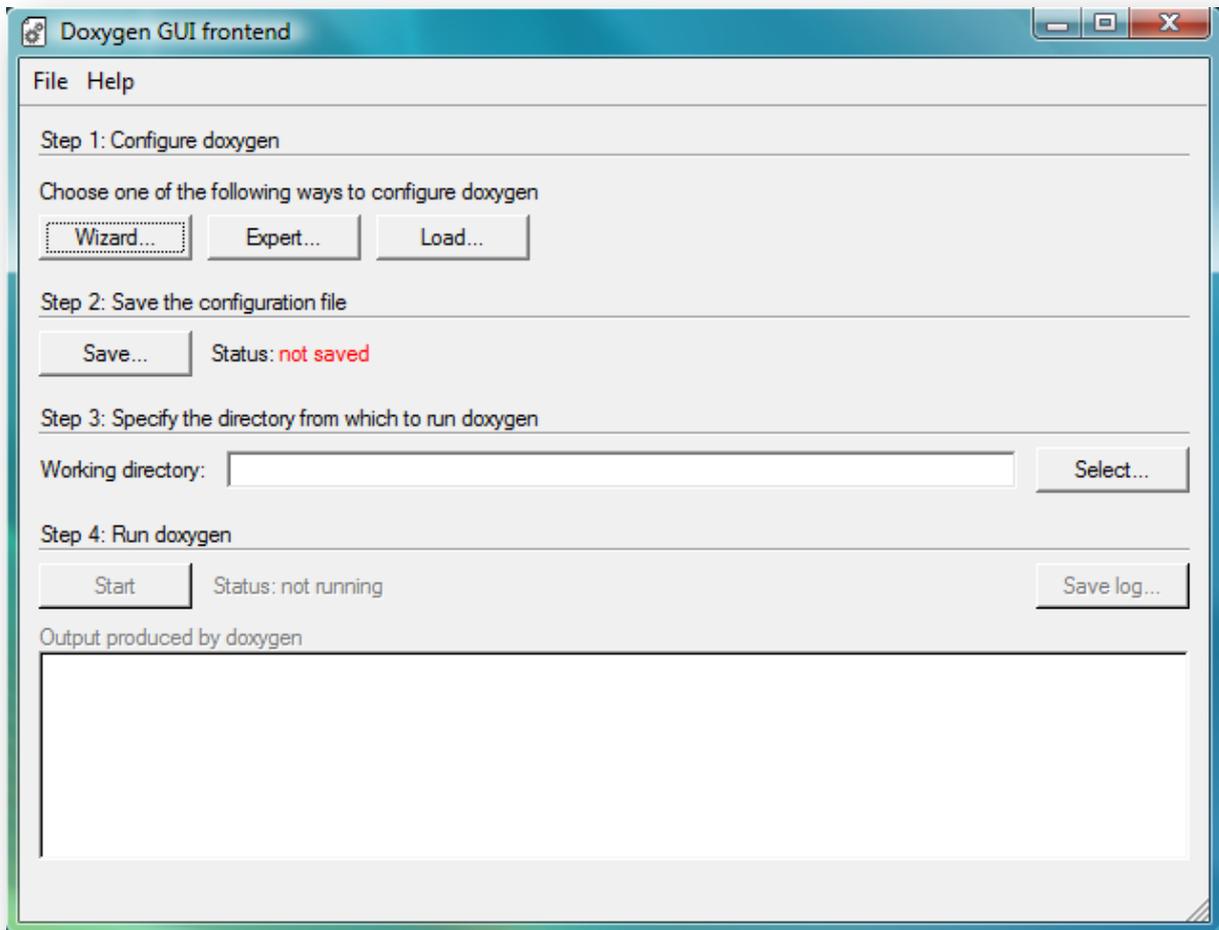


Abbildung 6: Die Benutzeroberfläche von Doxygen (Screenshot)

Doxygen arbeitet grundsätzlich unter allen Plattformen mit einer Konfigurationsdatei, in der gespeichert ist, wie Doxygen bei der Erstellung der Dokumentation vorgehen soll. Bei der Erstellung dieser Konfigurationsdatei hilft der „doxywizard“, zu dem man gelangt, indem man auf „Wizard“ klickt.

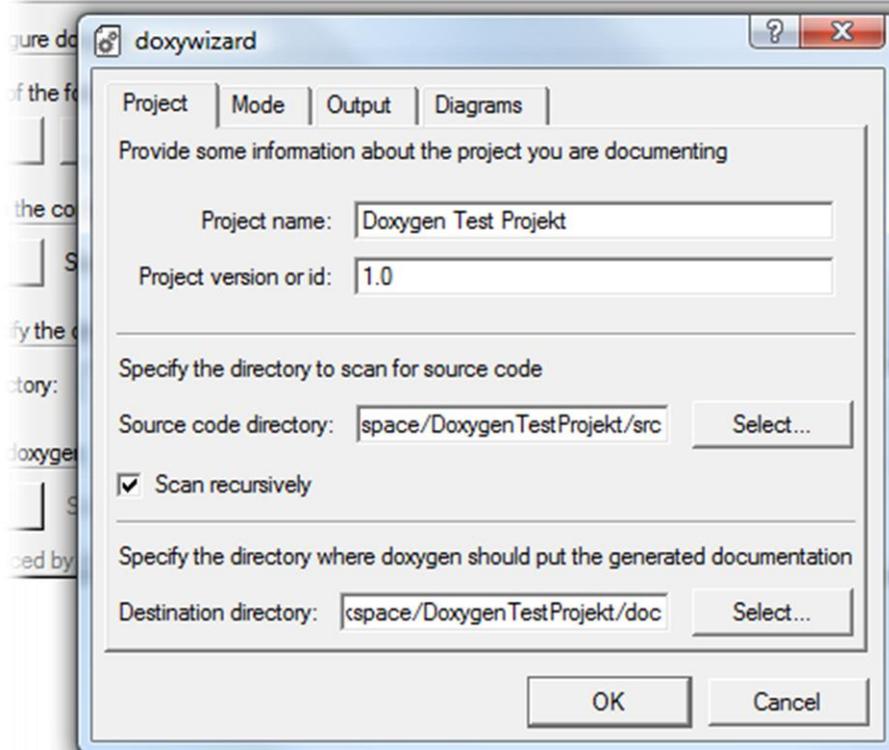


Abbildung 7: Der doxywizard, Registerkarte „Project“ (Screenshot)

Es folgt in der Registerkarte „Project“ die Eingabe des Projektnamens und dessen Version. Diese Angaben werden sich später auch in der Überschrift der generierten Dokumentation wiederfinden.

Darunter wird das Verzeichnis angegeben, in welchem sich die Quellcode- Dateien befinden. Wird mit Unterverzeichnissen gearbeitet, sollte der Haken bei „Scan recursively“ gesetzt werden.

Daraufhin wird ausgewählt, in welches Verzeichnis die Dokumentation ausgegeben und gespeichert werden soll. Es wird empfohlen hierfür ein eigenes Verzeichnis innerhalb des Projektverzeichnisses anzulegen, beispielsweise im Stil von Javadoc mit dem Namen „doc“.

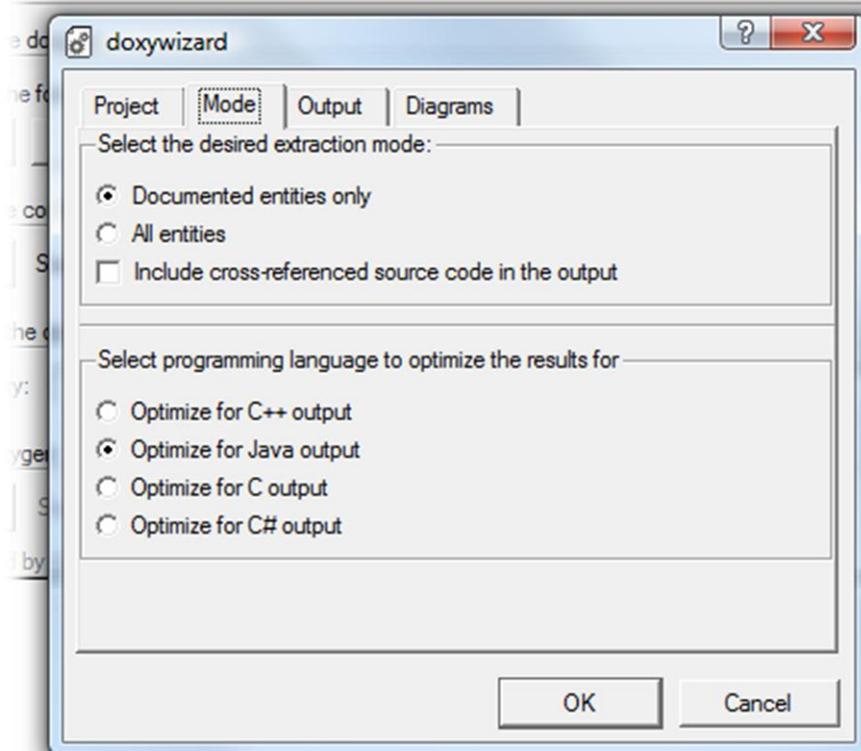


Abbildung 8 Der doxywizard, Registerkarte „Mode“ (Screenshot)

Auf der folgenden Registerkarte „Mode“ kann festgelegt werden, für welche Art von Code die Dokumentation optimiert werden soll. In unserem Beispiel handelt es sich um Java- Code. Im Allgemeinen unterscheiden die Optimierungen für eine der Programmiersprachen kaum.

Die übrigen Einstellungen können in der Regel unverändert bleiben.

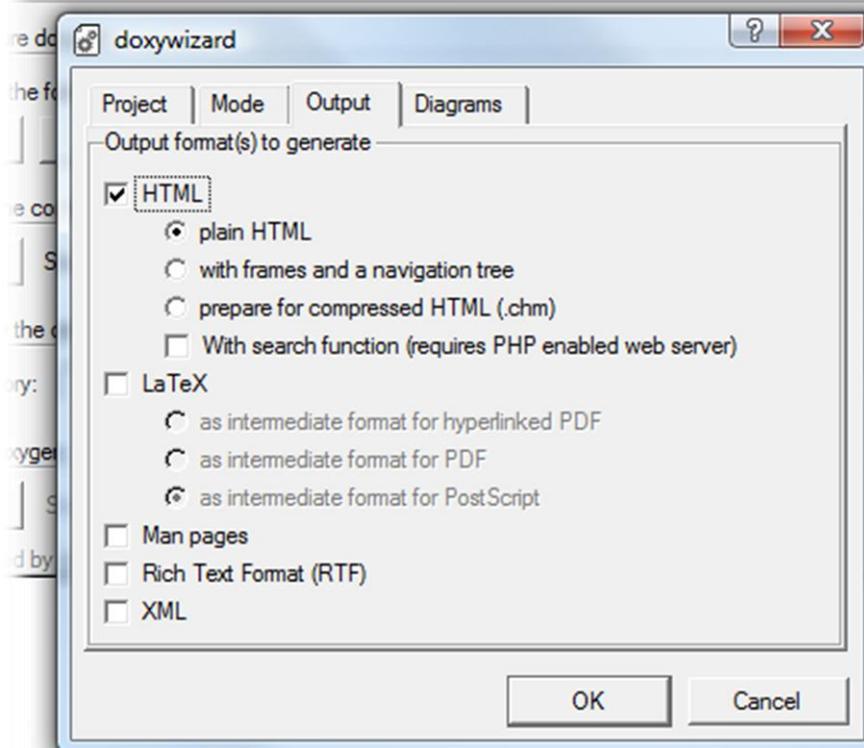


Abbildung 9 Der doxywizard, Registerkarte „Output“ (Screenshot)

In der Registerkarte „Output“ kann angegeben werden, in welcher Form die Dokumentation erstellt werden soll. Es werden einige Formate unterstützt, jedoch ist nicht jede Form zu empfehlen. Über die LaTeX- Ausgabe, kann mit einem LaTeX- Compiler später eine PDF- Datei generiert werden. Ebenfalls möglich ist die Ausgabe als XML- Datei und im RTF- Format, das mit allen gängigen Textverarbeitungsprogrammen geöffnet werden kann.

Die beste Wahl stellt jedoch das HTML- Format da, da hier am elegantesten innerhalb eines Browsers durch die Methoden und Klassen über Hyperlinks navigiert werden kann. Darüberhinaus lässt sich in die HTML- Ausgabe eine PHP- basierte Suche integrieren, auf die in den folgenden Kapiteln weiter eingegangen wird.

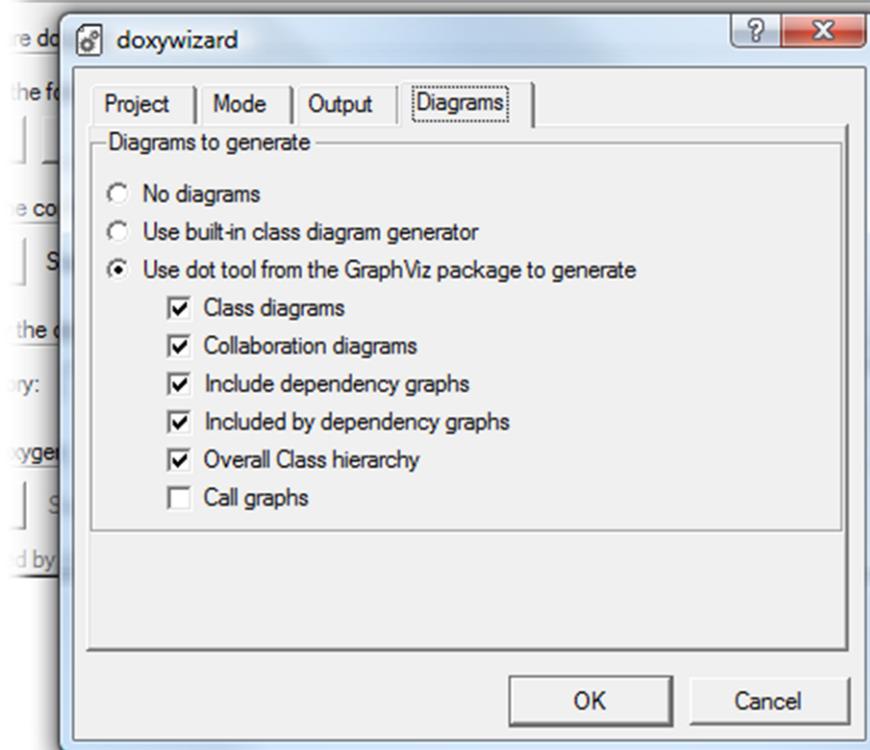


Abbildung 10 Der doxywizard, Registerkarte „Diagrams“ (Screenshot)

Die letzte Registerkarte „Diagrams“ befasst sich mit dem zuvor angesprochenen Plug-In „Graphviz“, das aus dem Quellcode und generierten Daten von Doxygen ein übersichtliches Diagramm für Methoden und Klassen erstellen kann. Die hier abgebildeten Einstellungen dürften für die meisten Ansprüche optimal sein.

Das Plug-In „Graphviz“ muss in Doxygen noch eingebunden und eingerichtet werden. Wie dies funktioniert, erfahren Sie im nächsten Kapitel.

## Interessante Zusatzmöglichkeiten

Klicken Sie auf der Doxygen- Benutzeroberfläche auf „Expert...“. Wählen Sie in dem nun geöffneten Fenster die Registerkarte „Project“ aus.

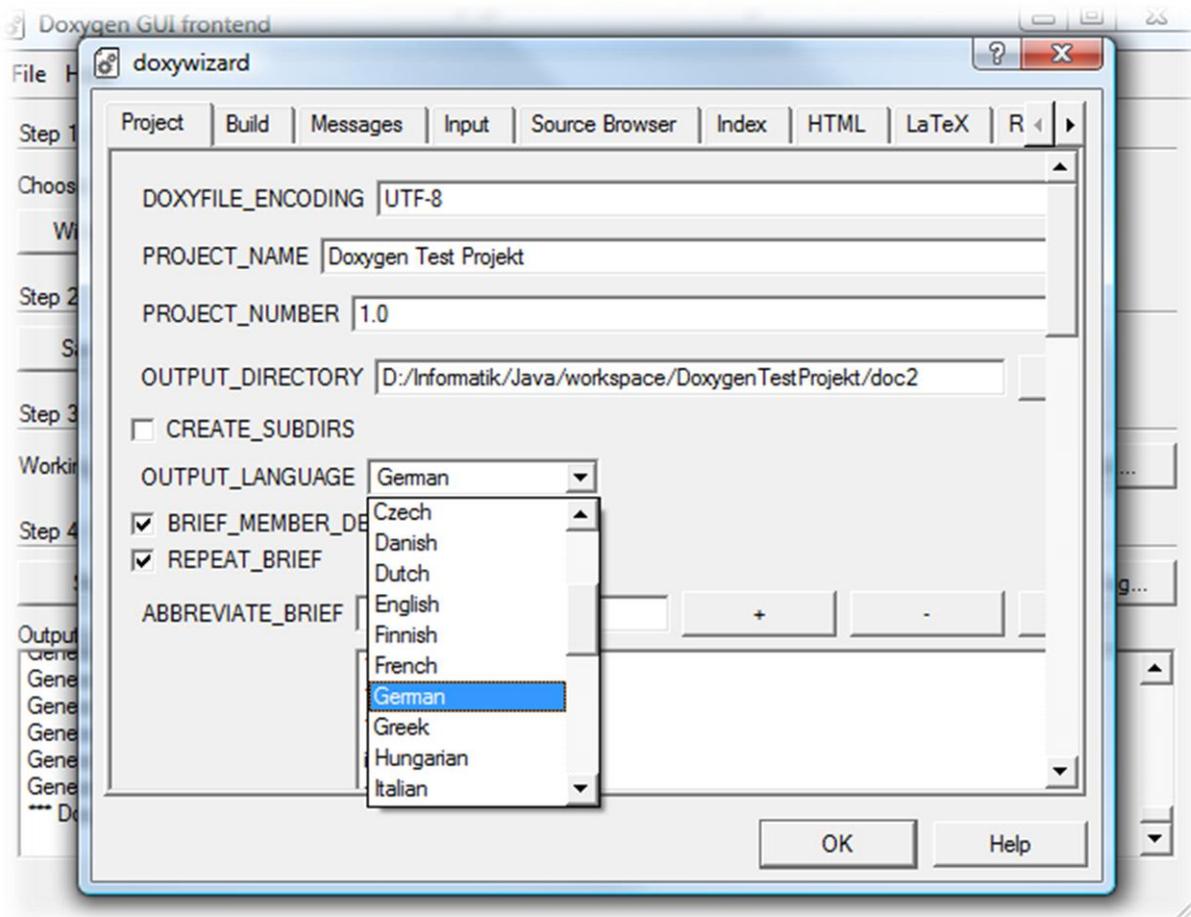


Abbildung 11: Der doxywizard im Expertenmodus, Registerkarte „Project“ (Screenshot)

Unter dem Punkt „OUTPUT\_LANGUAGE“ haben Sie die Möglichkeit eine Sprache auszuwählen, in welcher später die Menüs und die Struktur der Dokumentation ausgegeben werden soll. Standardmäßig ist Englisch eingestellt, was besonders für große und internationale Projekte zu empfehlen ist.

In diesem Expertenmodus existiert ebenfalls die Registerkarte „Source Browser“, welche weitere interessante Optionen zur Verfügung stellt.

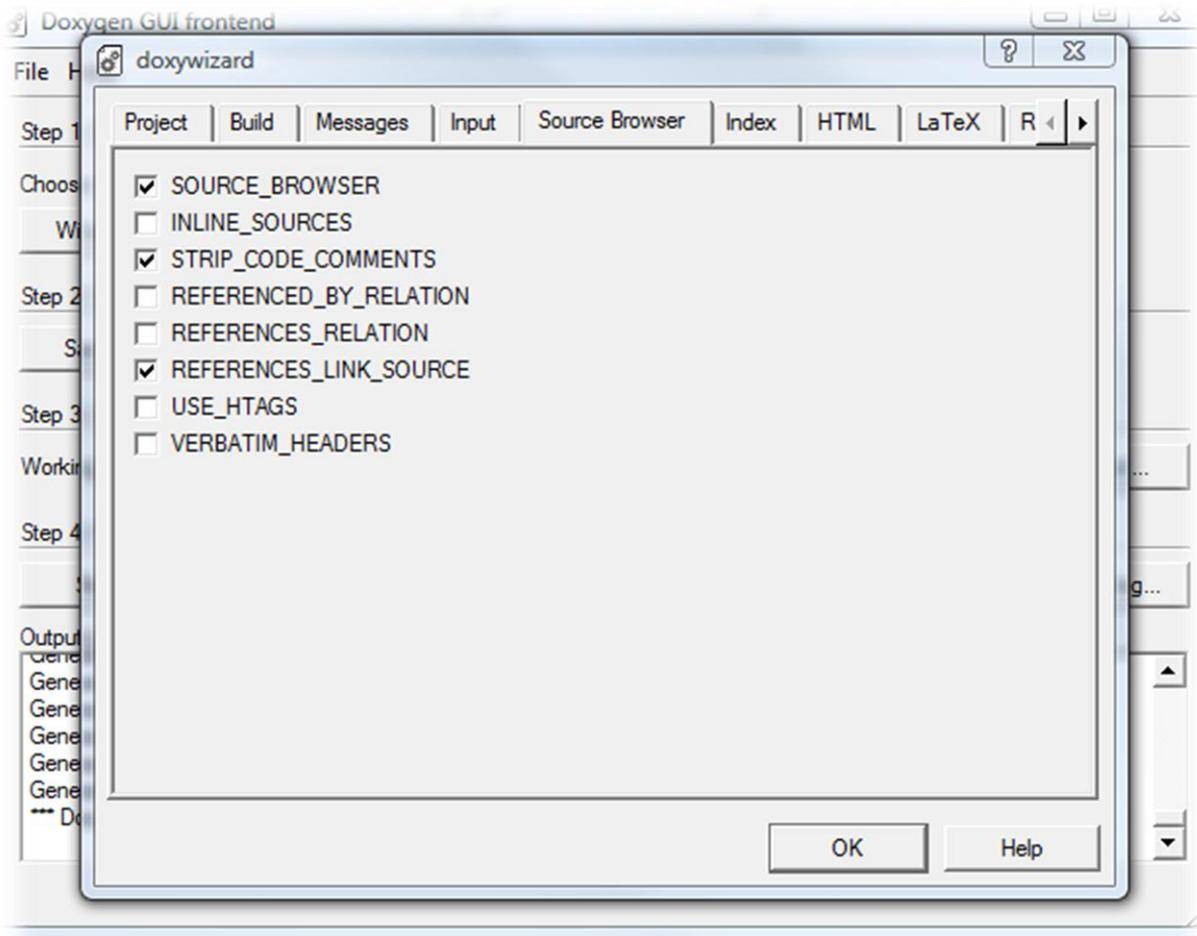


Abbildung 12: Der doxywizard im Expertenmodus, Registerkarte „Source Browser“ (Screenshot)

Das Setzen des Hakens vor „SOURCE\_BROWSER“ bewirkt, dass in die Dokumentation zusätzlich zur eigentlichen Dokumentation eine Liste mit den Quellcodedateien eingebunden wird. Diese Quellcodedateien liegen ohne Dokumentation vor, jedoch kann auf zuvor dokumentierte Einträge (Variablen, Methoden, etc.) geklickt werden, wodurch man direkt zur passenden Dokumentation des entsprechenden Eintrages geleitet wird.

Der Expertenmodus bietet im Allgemeinen sehr viele Einstellungsmöglichkeiten. Eine vollständige Auflistung mit Kurzerklärung ist im Handbuch zu Doxygen zu finden, das auf der Homepage des Projekts zum Download bereitsteht. Den Link zum Projekt finden sie unter (1) im Downloadverzeichnis.

## Graphviz Plug-In einbinden

Klicken Sie auf der Doxygen- Benutzeroberfläche auf „Expert...“. Wählen Sie in dem nun geöffneten Fenster die Registerkarte „Dot“ aus.

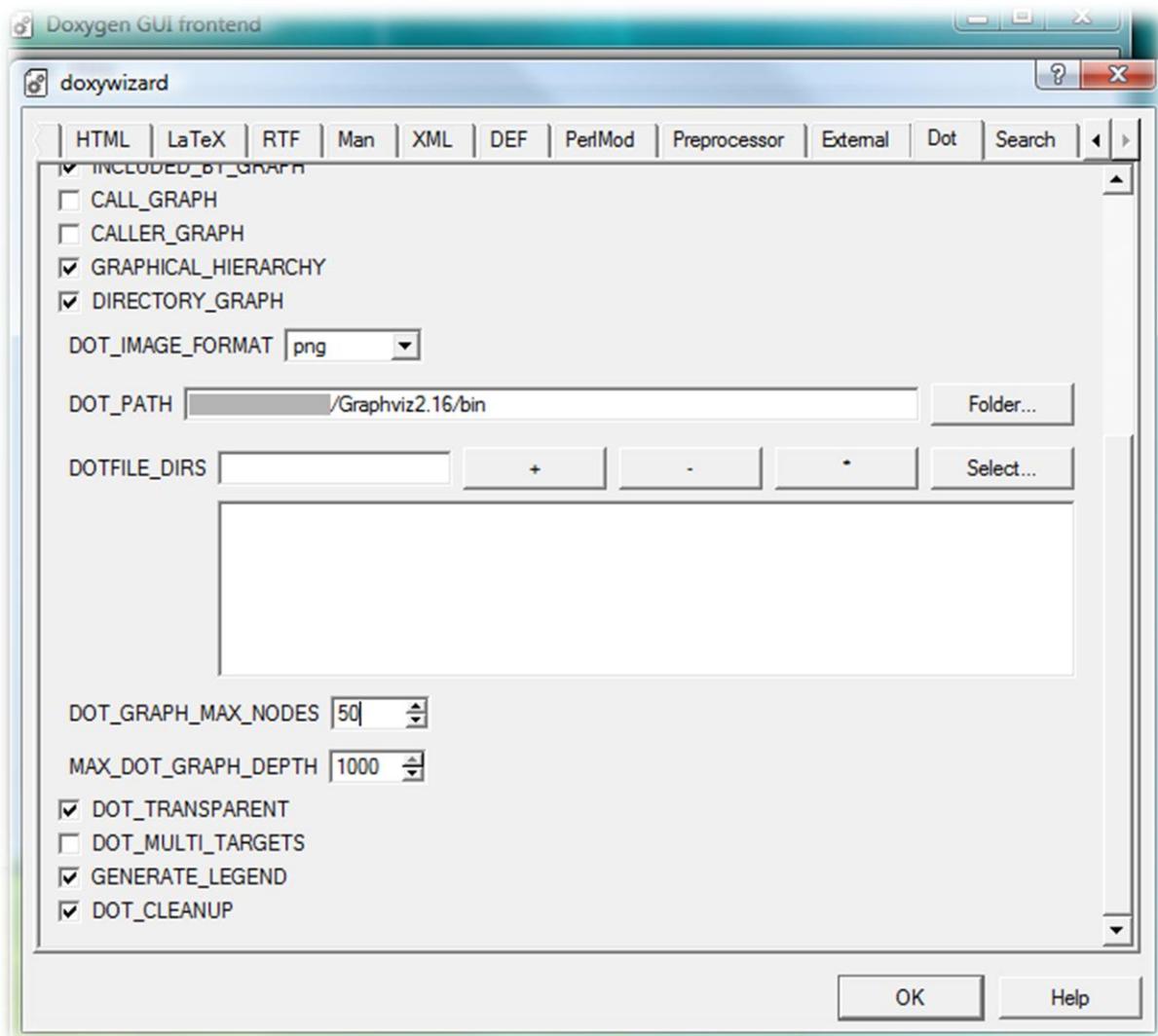


Abbildung 13: Der doxywizard im Expertenmodus, Registerkarte „Dot“ (Screenshot)

Um Graphviz einzubinden, geben Sie unter „DOT\_PATH“ den Pfad zu den ausführbaren Dateien von Graphviz an, in der aktuellen Version heißt dieses Verzeichnis „bin“.

Die Einstellung „DOT\_GRAPH\_MAX\_NODES“ legt fest, aus wie vielen Knotenpunkten der zu erstellende Graph maximal bestehen darf. Je nach Projektgröße muss diese Option entsprechend angepasst werden, da es bei einem zu kleinen Wert zum Programmabsturz kommt.

Nachdem Sie mit „OK“ bestätigt haben, sind alle Einstellungen für die Konfigurationsdatei getätigt. Speichern Sie daraufhin die Konfigurationsdatei, indem Sie in „Step 2“ auf „Save...“ klicken.

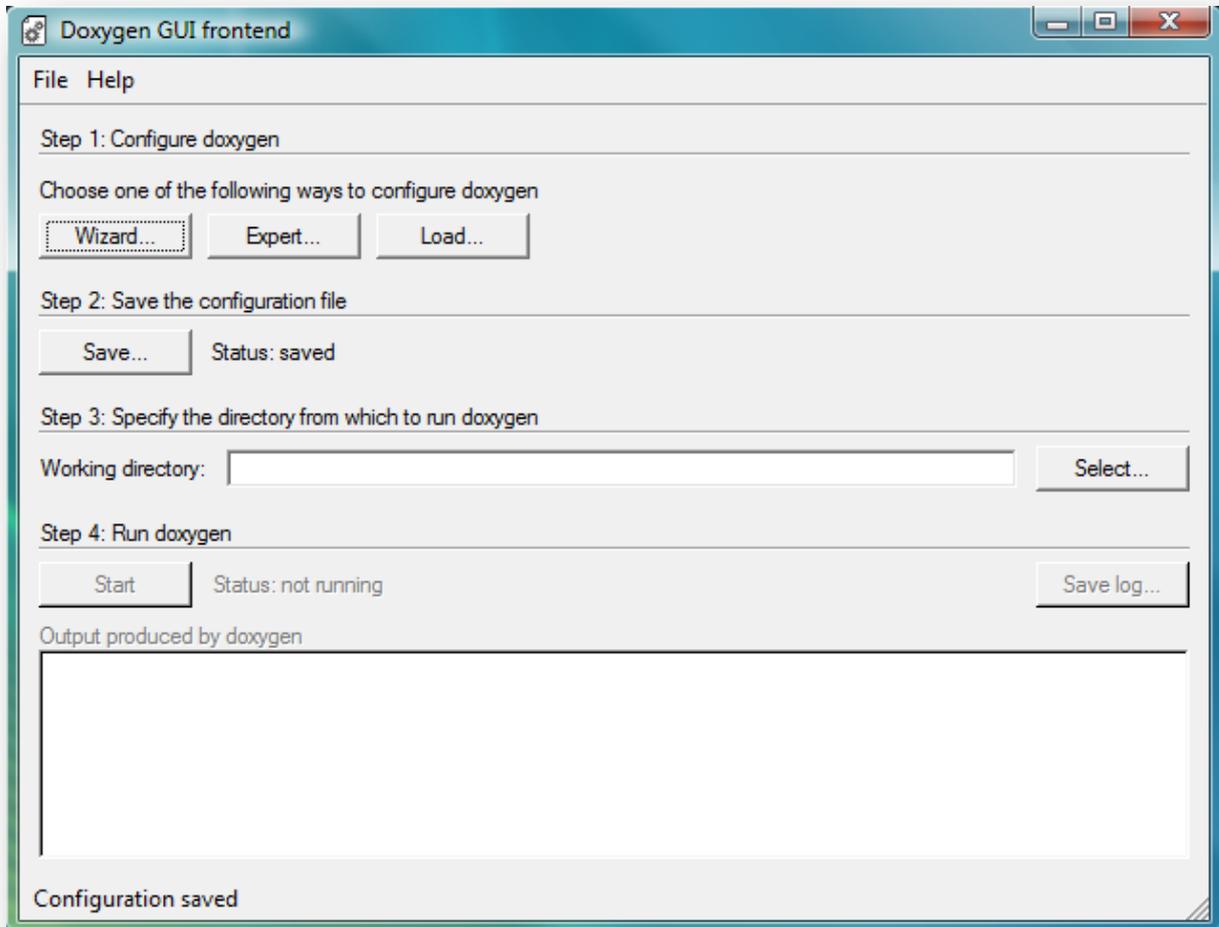


Abbildung 14: Die Benutzeroberfläche von Doxygen nach Speicherung der Konfiguration (Screenshot)

Wählen Sie im folgenden Schritt das Arbeitsverzeichnis für Doxygen aus, in welchem temporäre Dateien zwischengespeichert werden können.

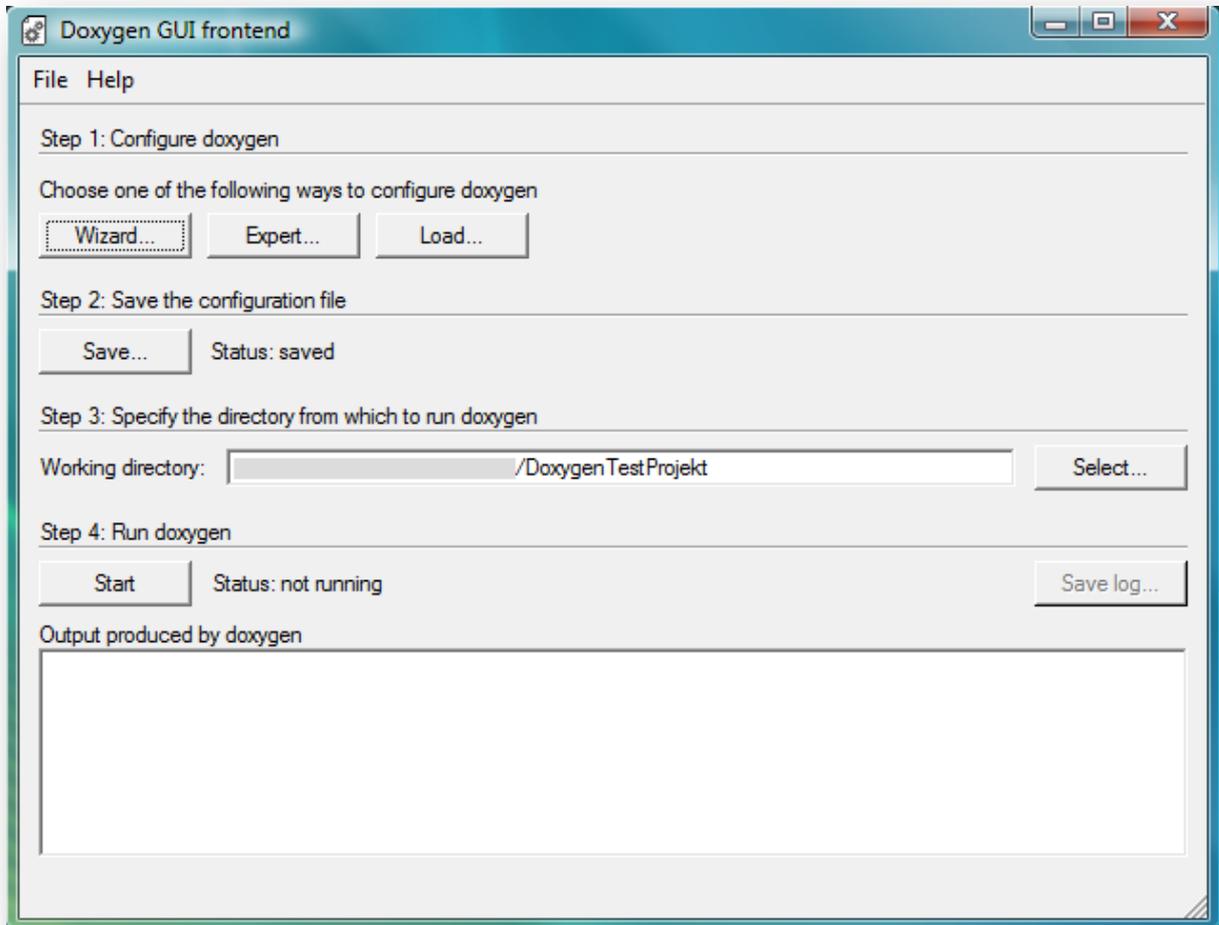


Abbildung 15: Die Benutzeroberfläche von Doxygen nach Angabe des Arbeitsverzeichnisses (Screenshot)

Nun sind alle Einstellungen getätigt und die Dokumentation kann erstellt werden, indem Sie in „Step 4“ auf „Start“ klicken.

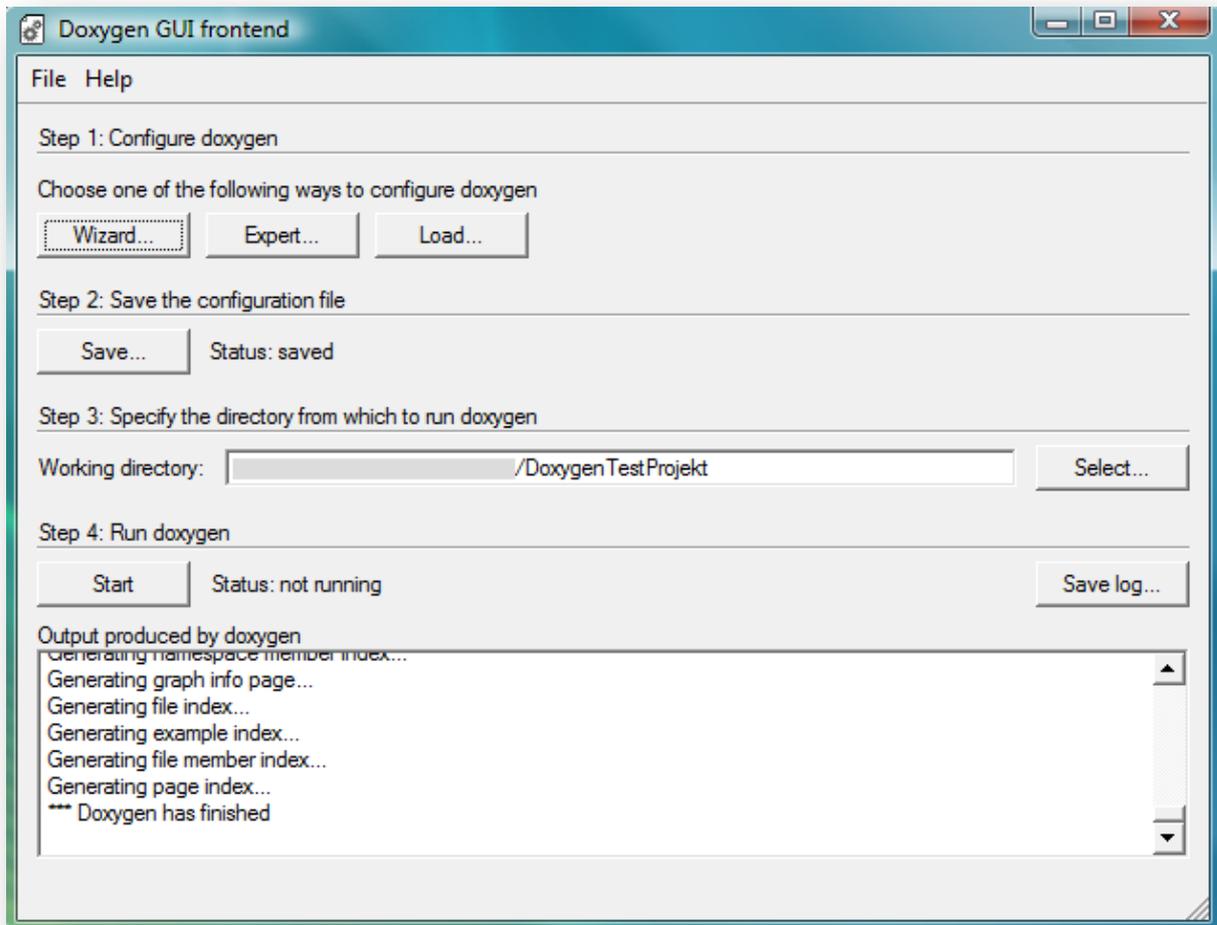


Abbildung 16: Die Benutzeroberfläche von Doxygen nach Fertigstellung der Dokumentation (Screenshot)

Sobald der Vorgang mit der Meldung „\*\*\* Doxygen has finished“ beendet wurde, finden Sie die komplette Dokumentation unter dem zuvor angegebenen Dokumentationspfad.

## Die fertige Dokumentation

Nachdem die Dokumentation erstellt wurde, kann Sie unter dem Dokumentationspfad gefunden werden. Laden Sie Startseite „index.html“.

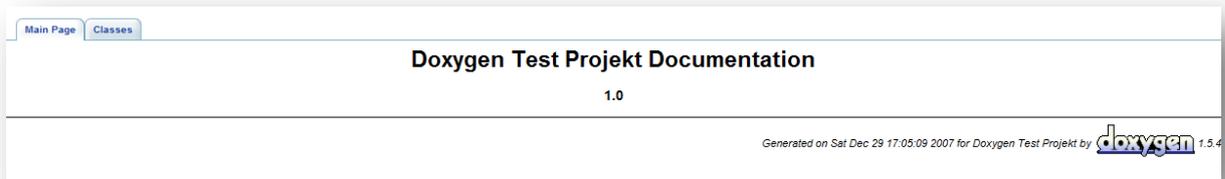


Abbildung 17: Startseite der Dokumentation (Screenshot)

Indem Sie auf „Classes“ klicken, werden alle vorhandenen Klassen aufgelistet. In unserem Beispielprojekt befindet sich nur die Klasse „Test“. Wenn Sie auf die gewünschte Klasse klicken, öffnet sich die dazugehörige Klassenreferenz mit allen öffentlichen Methoden und Variablen.

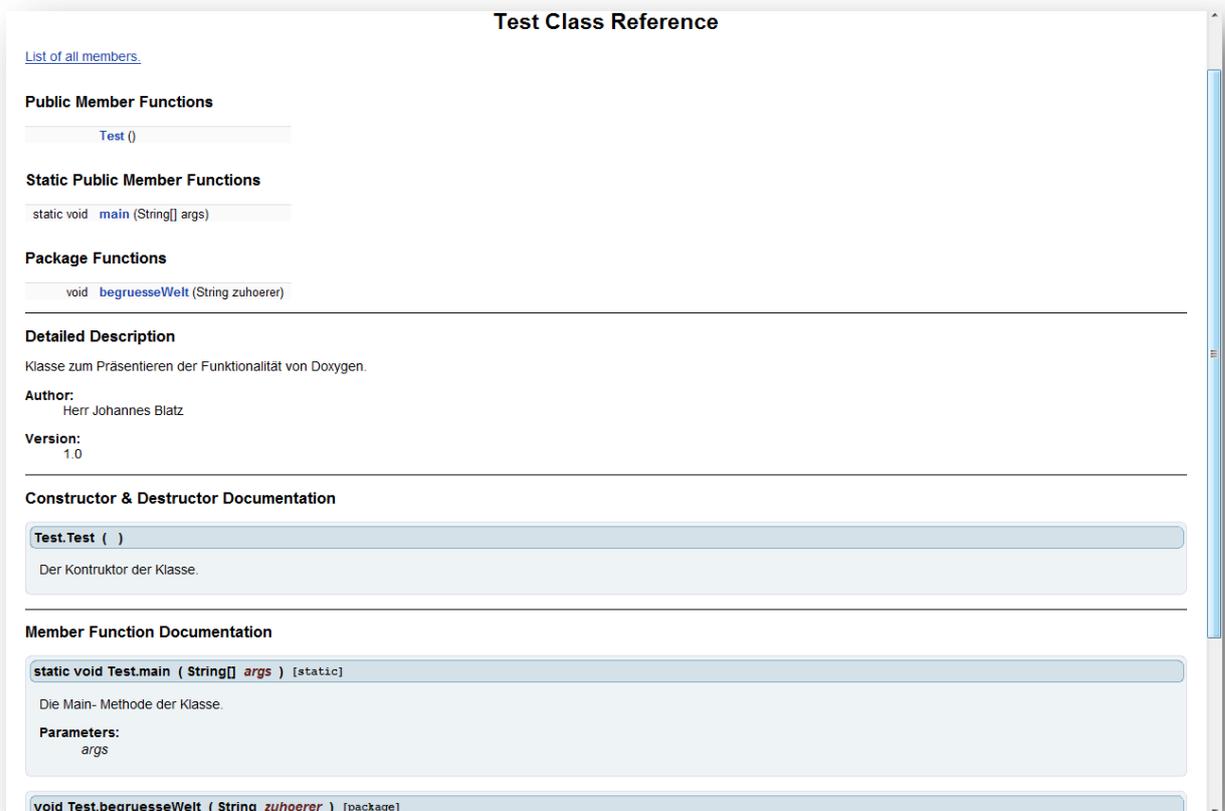


Abbildung 18: Klassenreferenz innerhalb der Dokumentation (Screenshot)

Die Diagramme, die Graphviz erstellt, befassen sich mit der Zusammenarbeit und Abhängigkeit von Klassen. Da hier nur eine Klasse zum Einsatz kommt, wird diese Funktion in einem späteren Kapitel an einem umfangreicheren Projekt vorgeführt.

## Erweiterte Funktionen

Doxygen bietet eine interessante Funktionalität gegenüber anderen Systemen. So lässt sich in jede Dokumentation eine auf PHP basierte Suchfunktion einbauen, die die Navigation durch den Pool an Methoden und Variablen deutlich vereinfachen kann.

Natürlich steht diese Funktionalität nur zur Verfügung, wenn die Dokumentation unter einem Webserver ausgeführt wird, der PHP unterstützt. Um diese Funktion lokal auf einem Rechner nutzen zu können, bietet sich das „XAMPP“- Projekt an, das auf einem lokalen Rechner den Apache-Webserver mit PHP- Unterstützung und geringstem Aufwand installiert.

Es handelt sich dabei um ein Open Source- Projekt, das auf der Internetseite (3) des Downloadverzeichnisses heruntergeladen werden kann.

Die Bedienung der Software fällt sehr leicht, es muss lediglich durch einen Klick auf „Start“ der Serverdienst „Apache“ aktiviert werden.

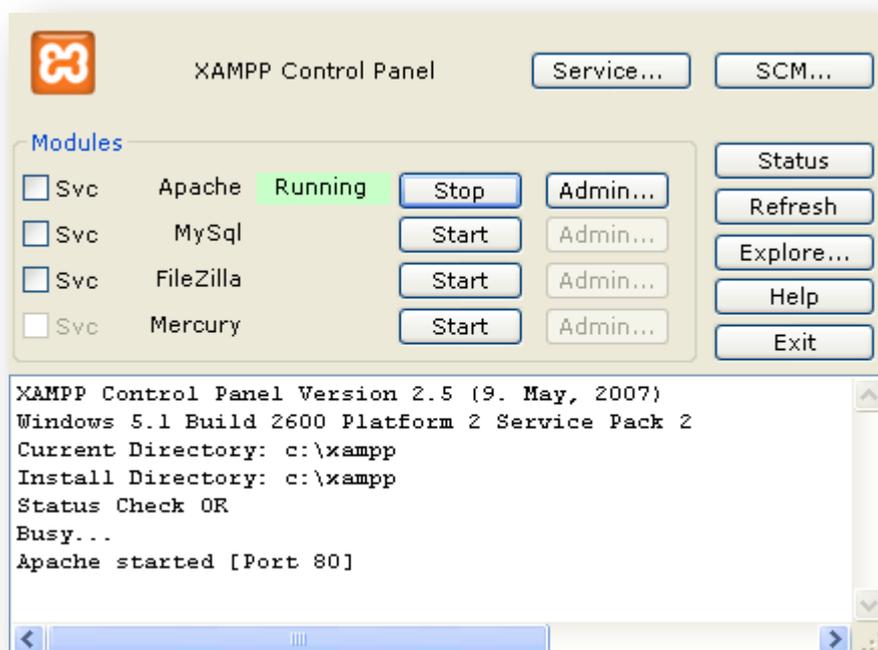


Abbildung 19: XAMPP- Control Panel zur Steuerung des Webservers (Screenshot)

Der Serverdienst beachtet nur die Dateien, die sich in dem dafür vorgesehenen Verzeichnis befinden. Das Verzeichnis liegt im Stammordner, in welchen die Software installiert wurde. In diesem Stammordner öffnen Sie das Verzeichnis „htdocs“ und darin das Unterverzeichnis „xampp“. In dieses Verzeichnis können Sie alle Dateien speichern, die Sie mit PHP- Technologie ausführen möchten. Ihre nun gespeicherten Dateien können Sie mithilfe Ihres Browsers laden, indem Sie die folgende Adresse eingeben: „http://localhost/**Dateiname.Dateiendung**“. Beispiel: „http://localhost/index.php“.

Weitere Informationen bietet die Projektseite, die Sie unter (3) des Downloadverzeichnisses finden.

Um die Suchfunktion standardmäßig in die Dokumentation einzubauen, muss im Doxywizard eine weitere Einstellung getätigt werden. Öffnen Sie hierfür in „Step 1“ wieder den Expertenmodus durch einen Klick auf „Expert...“ und wählen Sie die Registerkarte „Search“ aus.

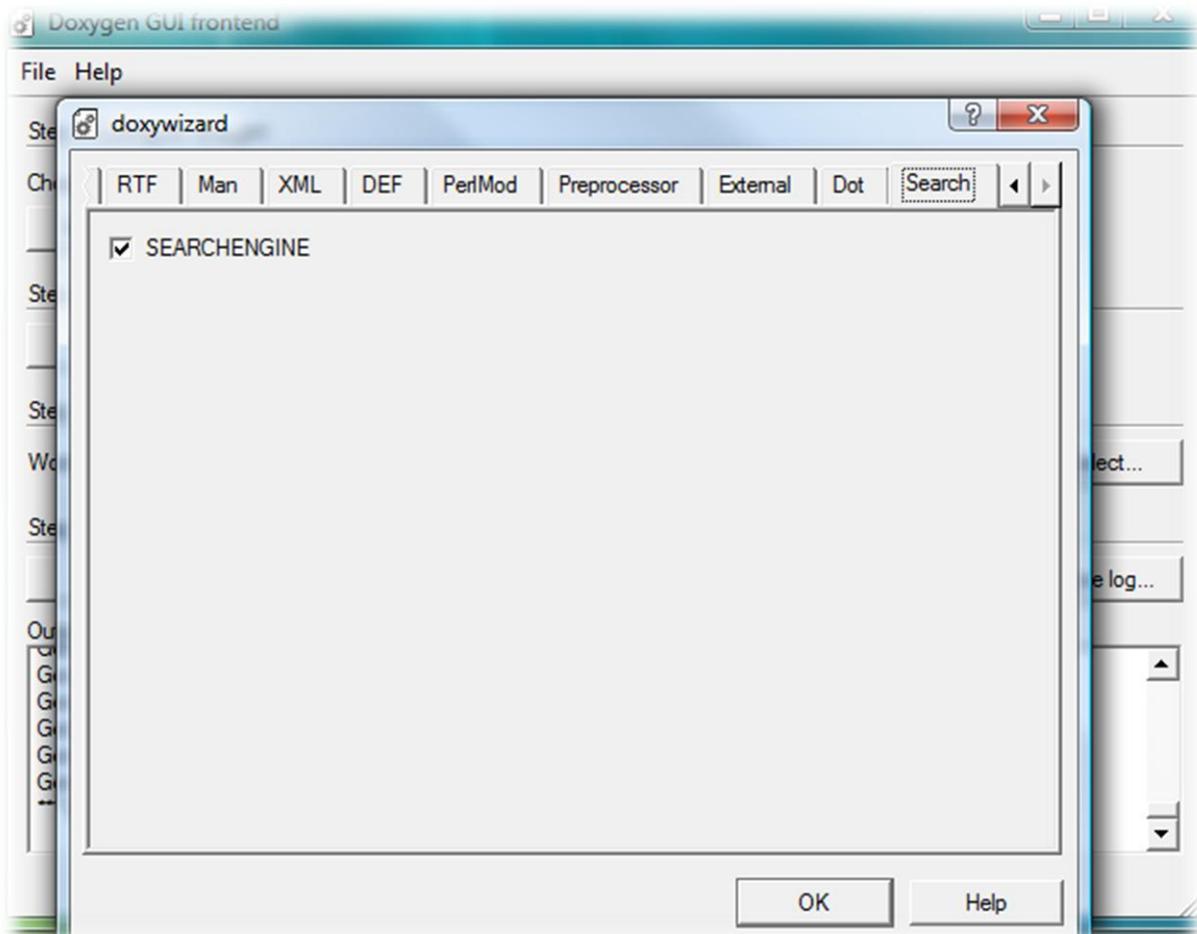


Abbildung 20: Der doxywizard im Expertenmodus, Registerkarte „Search“ (Screenshot)

Setzen Sie den Haken vor „SEARCHENGINE“ und die Suchfunktion wird in die Dokumentation eingebaut.

Nachdem die Dokumentation neu erstellt wurde, finden Sie das Eingabefeld für die Suchfunktion am oberen rechten Fensterrand.

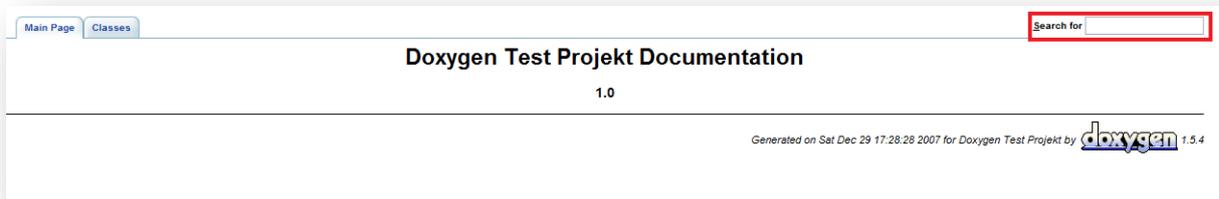


Abbildung 21: Startseite der Dokumentation mit integrierter Suchfunktion (Screenshot)

## Präsentation der Suchfunktion und von Graphviz

Unser Beispielprojekt ist zu klein, um die Mächtigkeit der Suchfunktion von Doxygen oder die Graphenerstellung von Graphviz demonstrieren zu können. Aus diesem Grund wird zur Demonstration auf ein komplexeres, bereits bestehendes, Projekt zurückgegriffen, das hier nur kurz erläutert wird.

Es handelt sich um ein Projekt, das mit einer Datenbank und einem Baum arbeitet. Graphviz erstellt im Stil der UML ein übersichtliches Diagramm.

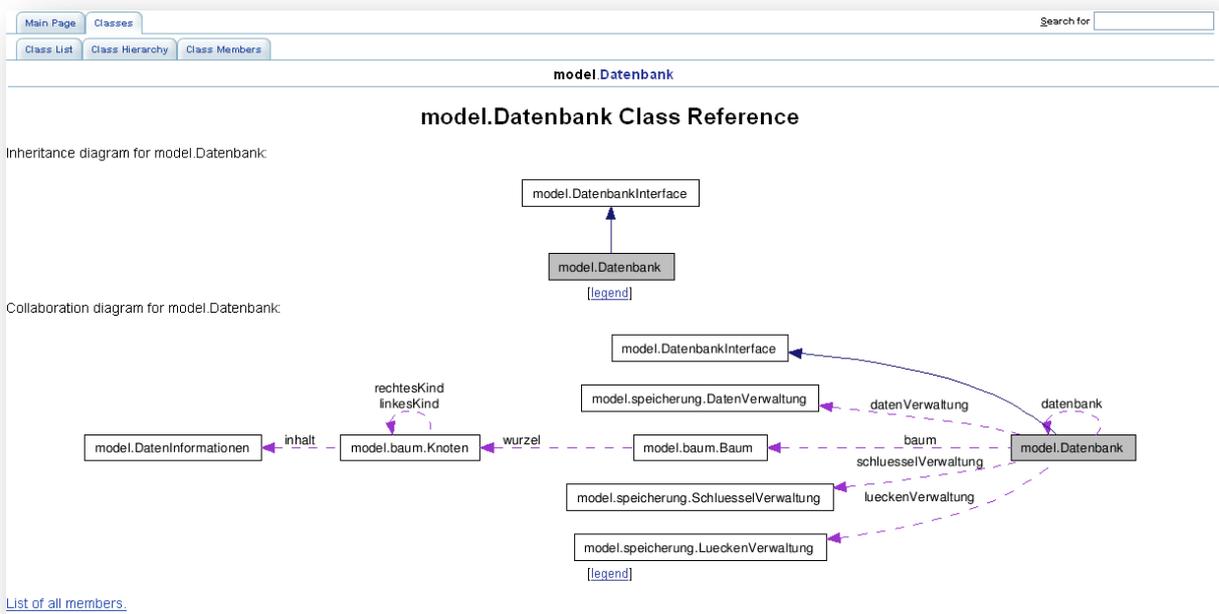


Abbildung 22: Klassenreferenz innerhalb der Dokumentation mit graphischer Unterstützung von Graphviz (Screenshot)

Gerade in größeren Projekten fällt die Navigation durch viele Klassen sehr schwer. Für diesen Fall stellt die Suchfunktion ein sehr gutes Hilfsmittel dar. Im folgenden Beispiel handelt es sich um dasselbe Projekt, das zur Demonstration von Graphviz diente. In diesem Beispiel wird nach der Main-Methode gesucht, was zu folgenden Suchergebnissen führt.



Abbildung 23: Suchergebnisse innerhalb der Dokumentation (Screenshot)

Durch einen Klick auf das Suchergebnis, landet man direkt in der Dokumentation zu dieser Methode.

## Das Dokumentieren

Doxygen unterstützt im Wesentlichen zwei große Dokumentationsstile, den Qt- Stil der Firma Trolltech und den JavaDoc- Stil. Die beiden Stile unterscheiden sich nur sehr wenig voneinander. Während bei JavaDoc gerne mit dem Doppelstern „\*\*“ gearbeitet wird, verwendet Qt einen Stern und ein Ausrufezeichen „\*!“.

Beispiel für JavaDoc- Stil:

```
/**
 * Diese Methode sendet eine Begrüßung an die Welt der Studenten.
 * @param zuhoerer Gibt an, wer begrüßt werden soll
 */
void begruesseWelt(String zuhoerer)
{
    System.out.println("Hallo Ihr lieben " + zuhoerer);
}
```

Abbildung 24: Quellcode- Auszug mit Dokumentation im JavaDoc- Stil (Screenshot)

Dasselbe Beispiel im Qt- Stil:

```
/*!  
 * Diese Methode sendet eine Begrüßung an die Welt der Studenten.  
 * @param zuhoerer Gibt an, wer begrüßt werden soll  
 */  
void begruesseWelt(String zuhoerer)  
{  
    System.out.println("Hallo Ihr lieben " + zuhoerer);  
}
```

Abbildung 25: Quellcode- Auszug mit Dokumentation im Qt- Stil (Screenshot)

Die spätere Ausgabe in der Dokumentation unterscheidet sich nicht.

Die Sterne zwischen „/\*\*“ bzw. „/\*!“ und „\*/“ dienen alleine der Übersichtlichkeit und sind nicht zwingend notwendig. Dies kann hilfreich sein, denn nicht jeder Editor setzt, wie „Eclipse“, die Sterne automatisch.

```
/**  
 *  
 *  
 */
```

## Doxygen- Tags

Eine wichtige Rolle beim Dokumentieren spielen die „Tags“ , die die beschriebene Klasse, Methode, Struktur oder auch die gesamte Datei näher spezifizieren. Dabei folgt vor jedem Tag das „@“- oder „\“- Zeilen und direkt dahinter der Tagname. In dieser Ausarbeitung wird immer das „@“- Zeichen verwendet.

Die wichtigsten Doxygen- Tags sind:

- @author
  - Gibt Auskunft über den Autor
- @brief
  - Eine Kurzbeschreibung des nachfolgenden Elements (bsp. Klasse, Methode, etc.)
- @bug
  - Erläutert bekannte Fehler
- @class
  - Gibt der nachfolgenden Klasse einen Namen
- @date
  - Gibt das Erstellungs- bzw. Änderungsdatum an
- @file
  - Beschreibt den Inhalt einer Datei
- @param
  - Beschreibt einen Parameter einer Methode
- @return
  - Beschreibt den Rückgabewert einer Methode
- @see
  - Stellt eine Referenz auf eine andere Dokumentation bzw. einen anderen Quelltext dar
- @struct
  - Erläutert die Struktur genauer
- @todo
  - Beschreibt, welche Arbeitsgänge noch zu tätigen sind
- @version
  - Gibt die Versionsnummer an
- @warning
  - Beschreibt Warnungen und enthält Hinweise

## HTML- Tags

Ferner unterstützt Doxygen HTML- Tags, wodurch die Dokumentation besser und übersichtlicher strukturiert werden kann.

Die wichtigsten HTML- Tags für diese Zwecke sind:

- `<a href="Hyperlink">Hyperlinkbeschreibung</a>`
  - Setzt einen Hyperlink zu einer anderen Internetdatei
- `<b>Text</b>`
  - Stellt den eingeschlossenen Text fett dar
- `<br>`
  - Erzwingt einen Zeilenumbruch
- `<center>Text</center>`
  - Zentriert den eingeschlossenen Beschreibungstext
- `<code>Codetext</code>`
  - Gibt den eingeschlossenen Text in der typischen Quellcode- Schriftart wieder
- `<em>Text</em>`
  - Stellt den eingeschlossenen Text kursiv dar
- `<p>Text</p>`
  - Legt für den eingeschlossenen Text einen eigenen Paragraphen an
- `<small>Text</small>`
  - Stellt den eingeschlossenen Text verkleinert dar
- `<table>Tabellenzeilen</table>`
  - Erstellt eine Tabelle mit spezifischen Tabellenzeilen
  - `<tr>Tabellenelemente</tr>`
    - Erstellt eine neue Tabellenzeile innerhalb der Tabelle
    - `<td>Elementtext</td>`
      - Erstellt ein neues Tabellenelement innerhalb der Tabellenzeile mit dem eingeschlossenen Elementtext

## Alternativen

Die Liste der Alternativen ist sehr lange, da es zu nahezu jeder Programmiersprache ein eigenes Dokumentationssystem gibt, beispielsweise Javadoc. Es handelt sich dabei um eine Liste von vielen freien Projekten und auch einigen kommerziellen Produkten. Kaum eine Alternative unterstützt jedoch so viele Programmiersprachen wie Doxygen, und oftmals ist die Dokumentation sehr unübersichtlich und schlecht aufbereitet.

Unter den freien Projekten, die sich mit mehr als einer Programmiersprache beschäftigen, sind Folgende hervorzuheben, die sich durch gute Dokumentationsqualität auszeichnen:

- ROBODoc (<http://www.xs4all.nl/~rfsber/Robo/robodoc.html>)
- Synopsis (<http://synopsis.fresco.org/index.html>)
- DOC++ (<http://docpp.sourceforge.net/>)

Unter den kommerziellen Produkten sind Folgende verbreitet:

- Borland Together (<http://www.borland.com/us/products/together/index.html>)
- Doc-O-Matic (<http://www.doc-o-matic.com>)

## Diskussion

Doxygen bietet gegenüber anderen Dokumentationssystemen einige Vorteile, aber es finden sich einige Kritikpunkte.

### Vorteile

- Einheitlicher Dokumentierstil bei unterschiedlichen Projekten und unter Verwendung verschiedener Programmiersprachen
- Erweiterbar durch das Plug-In GraphViz, das die Zusammenarbeit von Quellcodeelementen veranschaulicht.
- Gute Suchfunktion, die direkt in die Dokumentation integriert ist.
- Unterstützung vieler Programmiersprachen
- Wird von einer aktiven Gemeinschaft weiterentwickelt (Open Source)
- Kostenlos

### Nachteile

- Erstellte Dokumentationen sind teilweise unübersichtlich, besonders bei großem Einsatz von Variablen, Klassen und Methoden
- Mehrere Programmiersprachen lassen sich nur sehr unübersichtlich und nicht deutlich voneinander abgegrenzt in einer einzigen Dokumentation zusammenführen
- Je nach Einsatzgebiet und Umfang sind einige Zusatzkenntnisse erforderlich (PHP, Webserver, HTML)

Im Allgemeinen erweist sich die Arbeit mit Doxygen als sehr hilfreich und sinnvoll. Sowohl Doxygen selbst als auch die für manche Einsatzgebiete notwendigen Zusatztechnologien (Webserver, Plug-In GraphViz) stehen jedem Nutzer kostenlos zur Verfügung.

Für eine intensivere Arbeit mit diesem Dokumentationstool ist es ratsam, das Handbuch zu diesem Projekt zu lesen oder in entsprechenden Fällen zu Rate zu ziehen. Viele Doxygen- und HTML- Tags sowie weitere Funktionsmerkmale, die den Umfang dieser Ausarbeitung überschreiten, sind dort ausführlich beschrieben.

## Bekannte Beispiele, in denen Doxygen eingesetzt wird

K Desktop Environment



Abbildung 26: KDE- Logo (<http://www.kde.org>)

MySQL Datenbank



Abbildung 27: MySQL- Logo (<http://www.mysql.org>)

## Quellenverzeichnis

- <http://www.doxygen.org>
- Doxygen- Benutzerhandbuch (Download über Projekthomepage, in Englisch)
- <http://www.graphviz.org>
- <http://www.in.tu-clausthal.de/abteilungen/winf/lehre/ss05/unterstuetzungsprozesse-fuer-entwicklungsprojekte>
- <http://www.selflinux.org/selflinux/pdf/doxygen.pdf>
- <http://www.clug.de/vortraege/doxygen>

Es wurde keine weitere Literatur für die Ausarbeitung verwendet.

## Downloadverzeichnis

- (1) <http://www.doxygen.org>
- (2) <http://www.graphviz.org>
- (3) <http://www.apachefriends.org/de/xampp.html>

Präsentiert von [www.Netzwerk-des-Wissens.de](http://www.Netzwerk-des-Wissens.de)

