



## Improved search engine capabilities

**Software Requirements Specification**

**for**

**Google Sommer of Code 2008**

Updated on: 20.03.2008

# Inhaltsverzeichnis

<b>1. Introduction.....</b>	<b>3</b>
<b>1.1. About this document.....</b>	<b>3</b>
<b>1.2. Technicalities.....</b>	<b>3</b>
<b>2. Definition of objectives.....</b>	<b>4</b>
<b>2.1. Actual state and motivation.....</b>	<b>4</b>
<b>2.2. Must have criteria.....</b>	<b>4</b>
<b>2.3. Should have criteria.....</b>	<b>4</b>
<b>2.4. Functional constraints.....</b>	<b>4</b>
<b>3. Design goals.....</b>	<b>5</b>
<b>3.1. Maintainability.....</b>	<b>5</b>
<b>3.2. Extensibility.....</b>	<b>5</b>
<b>3.3. Reusability.....</b>	<b>5</b>
<b>3.4. Usability.....</b>	<b>5</b>
<b>3.5. Performance.....</b>	<b>5</b>
<b>4. Development aspects.....</b>	<b>6</b>
<b>4.1. Used tools.....</b>	<b>6</b>
<b>4.2. Communication with the mentor.....</b>	<b>6</b>

# 1 Introduction

## 1.1 About this document

This is a requirements specification for the project „Improved search engine capabilities“ within the participation of the PostNuke Application Framework in Google Summer of Code 2008.

This document does not contain specific details like use cases, service conditions or implementation aspects. It lists up the most important aspects, but it can be mostly seen as a rough guideline based on our own ideas. These are not obligatory, we are also looking forward to getting innovative input from the students.

## 1.2 Technicalities

- Project: Improved search engine capabilities
- Difficulty: hard
- Required experience: PHP, MySQL

This project will be mentored by {MENTOR NAME}. Please find more information about communication between mentors and students at chapter 4.2 on page 6.

## 2 Definition of objectives

### 2.1 Actual state and motivation

In these days people have to cope with more content than ever. Thus powerful means to find it become more important. While the search module of PostNuke got some improvements in the latest core versions, there is still a lack of extended features advancing the user experience.

### 2.2 Must have criteria

In order to be able to extend the search module's abilities this project aims on developing a powerful search engine bearing up under the requirements in times of information overload and social networks.

It should be possible to perform unsharp searches finding also terms being similar to the search term. With this functionality users get probably expected results in spite of typing errors.

Another important feature is supporting semantic queries. An easy example is finding all content of a particular person. This can be enhanced for several directions, e.g. groups, relations, and so on.

### 2.3 Should have criteria

A possible third function is indexing office and pdf documents for providing extended search capabilities. To keep the time-frame this could start only with open formats like open office documents.

The social thingy could be enhanced to let modules decorate their object types with a special technology like OWL, RDF or something else.

### 2.4 Functional constraints

None.

## 3 Design goals

### 3.1 Maintainability

The whole source code should follow the PostNuke coding standards and contain some comments for improved understandability.

Ideally some pattern are used during the design phase for reducing the amount of physical maintenance points.

### 3.2 Extensibility

Ideally the whole search type functionality will be implemented in a class structure with special interfaces taking care for encapsulation and easy addition of extensions. For example including a new search type searching and fetching external resources should not affect the internal components.

### 3.3 Reusability

If this class structure can be kept quite independent of the data fields it processes, this would result in a greater reusability for other modules. Using interface types in operative code ensures exchangeability through polymorphism.

### 3.4 Usability

An easy and intuitive user guidance is necessary for acceptance on the part of the community.

If possible within the given time frame a small documentation could be written describing functionality and usage.

### 3.5 Performance

For using this project in productive areas caching mechanism might be needed to reduce the server load caused by the search operations being more complex and producing heavy server load.

## 4 Development aspects

### 4.1 Used tools

There are no particular requirements. In general the following software components are needed for this project:

- Webserver (e.g. Apache)
- Database (e.g. MySQL, PostgreSQL, MS SQL or Oracle)
- PHP 5
- Latest version of PostNuke .8
- Programming editor (either a comprehensive IDE like Eclipse with PDT or a text editor with syntax highlighting)

### 4.2 Communication with the mentor

We created a mailing list where general things can be discussed between all mentors and students. The mentors are looking forward to some exciting projects.

If you have any further questions please contact the PostNuke Steering Committee at [sc@postnuke.com](mailto:sc@postnuke.com).