



Developer Guide for W5Base

Dieses Dokument beschreibt die Rahmenbedingungen bei der Entwicklung von W5Base Applikationen und Modulen. Es ist nicht zulässig, Patches ins offizielle Repository einzuspielen, die diesem Regelwerk nicht entsprechen.

Inhaltsverzeichnis

Developer Guide für W5Base.....	1
1. Allgemeine Rahmenbedingungen.....	2
2. Verzeichnisstruktur.....	3
3. Signifikante Dateien und Verzeichnisse.....	4
4. Genereller Ablauf eines Web-Requests.....	5
5. Programmcode Style.....	6
6. Notwendige Skills für Entwickler.....	7
Glosar.....	8

Version 15 - 13.01.2012



1. Allgemeine Rahmenbedingungen

- W5Base Applikationen müssen so entwickelt und getestet werden, dass sie mit ALLEN gängigen Browsern bedient werden können.
- Alle Komponenten der W5Base müssen unter GPLv2 lizenziert werden.
- Das zentrale Entwicklungs-Repository befindet sich unter ...
<http://w5base.svn.sourceforge.net/viewvc/w5base/>
- Der Commit von „offiziellen Patches“ ist nur Projekt-Managern auf ...
<https://sourceforge.net/projects/w5base/>
... gestattet.
- Als offizielles Logo darf nur ...
... verwendet werden.
-





2. Verzeichnisstruktur

<code>bin/</code>	Einsprungspunkt für den Apache Web-Server.
<code>sbin/</code>	Management-Tools, die auf einem W5Base System durch die Anwendungsbetreuer und Entwickler direkt von der Unix-Systemshell aus verwendet werden können.
<code>contrib/</code>	Beispielprogramme und anderes, was u.U. hilfreich beim Umgang mit einem W5Base System sein könnte.
<code>mod/</code>	Programmmodule In diesem Verzeichnis findet die eigentlich Entwicklung von Datenobjekten und Programm-Modulen statt. Alles was Programm-Code ist und NICHT zum Kernsystem gehört, ist hier zu finden.
<code>skin/</code>	Alles, was das grundsätzliche Aussehen der Oberfläche erzeugt. Dazu zählen auch die Language-Files.
<code>sql/</code>	SQL-Skripte die die Datenbankstruktur unter Kontrolle halten.
<code>etc/</code>	Konfigurationsdateien
<code>etc/w5base/</code>	Default-Konfiguration der W5Base, die direkt mit dem Programmcode ausgeliefert wird.
<code>lib/</code>	Zentrale Bibliotheken, die alle Module gemeinsam nutzen.
<code>lib/kernel/</code>	Enthält den eigentlich W5Base-Kern also das Framework
<code>dependence/</code>	Quellen/Sourcecodes für Module und Bibliotheken, die auf einem „normalen“ W5Base-System installiert werden müssen.



`static/`

Statische Web-Seiten, die auf einem W5Base-Server direkt im Web-Server Eingebunden sein müssen.



3. Signifikante Dateien und Verzeichnisse



4. Genereller Ablauf eines Web-Requests



bin/app.pl

Dieses Perl-Script ist der Einsprungspunkt, über den ALLE Apache Request an die W5Base Applikation weitergegeben werden. Über Rewrite Regeln im Apache werden die jeweiligen Web-Pfadnamen dann als Variablen in dieses Perl-Script „eingeschläußt“. In app.pl werden grundsätzliche Umgebungseinstellungen getroffen und der Aufruf der WebAppRun Methode vorbereitet. Mit dem Aufruf von `kernel::App::Web::RunWebApp()` findet der Wechsel in das eigentliche W5Base-Kernsystem statt. Das Programm app.pl ist also ein Adapter, der die Anbindung an u.U. unterschiedliche Web-Server Umgebungen darstellen kann.

lib/kernel/App/Web.pm

Die Methode `RunWebApp()` bereitet die Erzeugung des eigentlichen Programmobjektes vor.

Von hier wird auch die Methode `InitRequest()` aufgerufen, die das W5Base-Cache System steuert.

Befindet sich die W5Base im Maintenance Mode, so wird in `RunWebApp` direkt in die Darstellung der Wartungsseite verzweigt und somit keinerlei Operationen an die W5Base-Module weitergegeben.

Es wird versucht, das ausgewählte Programm-Objekt (gleichzusetzen mit `DataObj`) erzeugt. Die Auswahl welches Programm-Objekt erzeugt werden soll, erfolgt direkt über die aufgerufene URL im Web-Browser.

Konnte ein Programm-Objekt erfolgreich erzeugt werden, so wird in die Methode `Run()` des Programm-Objektes verzweigt. In dieser wird dann in Abhängigkeit zur aufgerufenen WebFunktion in die jeweilige Perl-Methode (nach Prüfung auf Zulässigkeit) verzweigt.

DataObj

Ein `DataObj` ist i.d.R. auch immer gleichzeitig auch ein Programm-Objekt. In dieses wird per Default über den Aufruf durch einen Web-Browser in die Methode `Run()` eingesprungen.

Je nach Eltern-Klassen des Datenobjektes kann dann das Verhalten unterschiedlich ausgeprägt sein.

5. Programmcode Style

1. Zeichensatz

Alle Programmcodes und Controll-Dateien werden im Zeichensatz „ISO-Latin1“ verfasst. Zeilenumbrüche in allen Dateien werden nach Unix-Style, d.h. nur durch das Zeichen ASCII 13 repräsentiert.

2. Einrückungen

Einrückungen sind immer als 3 Leerzeichen Blöcke aufzubauen. Einrückungen mit TAB Zeichen sind nicht erlaubt (da diese Editor-spezifisch dargestellt werden).

3. Zeilenlänge

Soweit möglich, ist eine Programmcode Zeile auf 80 Zeichen (Spalten) zu begrenzen. Davon abgewichen werden kann z.B. bei komplexen Reguären Ausdrücken, in die nur schwer ein Umbruch eingebaut werden kann.

4. Sprache

Alle Variablen, Dateinamen und Funktionen sind von englischen Ausdrücken abzuleiten. Soweit möglich, sind alle Bemerkungen im Programmcode in englisch einzufügen.

6. Notwendige Skills für Entwickler

Für die Entwicklung von Modulen innerhalb des W5Base-Frameworks sind die folgenden Skills notwendig:

- **Perl (Fortgeschrittene Kenntnisse !)**

OO Programmierung in Perl

Modul-Programmierung

Perl in ModPerl2 Umgebungen

- **SQL**

Für die reine Entwicklung von Modulen sind nur grundlegende Kenntnisse in SQL

notwendig. Für die Anpassung von Kern-Klassen sind fortgeschrittene SQL

Kenntnisse in allen gängigen SQL Dialekten notwendig (MySQL, Oracle, ODBC ...)

- **Betriebssystem Unix (Linux)**

Der Umgang mit einem Unix-System sollte den Entwicklern kein Problem bereiten.

- **SubVersion (SVN)**

Versierter Umgang mit dem Code-Revisionverwaltungssystem SubVersion ist eine

zwingende Voraussetzung, da alle Code-Anpassung über dieses System getrackt

werden.

Glosar

Configuration Management System ⁽¹⁾	System, dass für die Verwaltung der Configuration-Daten verwendet wird. Es beinhaltet Informationen zu Mitarbeiter, Kunden, Standorten, etc. und dient zur Übersicht über Changes, Problems, Incidents, ...
QuickFind ⁽²⁾	Funktion von W5Base / Darwin um Config-Daten, Nutzer, FAQs und anderes zu durchsuchen
Oracle ⁽³⁾	Großer Software Hersteller, der vor allem durch das Datenbankmanagementsystem „Oracle Database“ bekannt ist
Perl ⁽⁴⁾	Eine plattformunabhängige Skriptsprache, die mit Hilfe eines Interpreters ausgeführt wird. Die Sprache eignet sich besonders um schnell und einfach zu programmieren. Unter diesem Aspekt leidet jedoch teilweise die Übersichtlichkeit des Quellcodes.
Patch ⁽⁵⁾	Änderung am Programmcode, die Fehler behebt oder Funktionen hinzufügt
Entwicklungsumgebung ⁽⁶⁾	Umgebung, in der Neuerungen getestet werden können und auf den Betrieb vorbereitet werden können.
Wirkumgebung ⁽⁷⁾	Umgebung, in der die Software im tatsächlichen Wirkbetrieb läuft
GNU Lesser General Public License (LGPL) ⁽⁸⁾	Eine Lizenz für freie Software, in der unter anderem festgehalten ist, dass Programme, die unter dieser Lizenz veröffentlicht werden, beliebig weitergegeben, nach eigenen Bedürfnissen angepasst und die veränderten



	Versionen weitergegeben werden dürfen. Verändert man ein unter der LGPL stehendes Programm und möchte dieses veröffentlichen, so muss man dies auch unter der LGPL oder wahlweise der GPL (GNU General Public License) tun.
SQLRunner ⁽⁹⁾	Eine freie Software, mit der man sich zu Datenbanken verbinden, sich deren Datenstrukturen anzeigen lassen und Abfragen erstellen kann.
View ⁽¹⁰⁾	Eine View (dt. Sicht) ist eine Abfrage, die in einem Datenbanksystem gespeichert ist und dann vom Benutzer wie eine normale Tabelle genutzt werden kann.
SDK (Software Development Kit) ⁽¹¹⁾	Werkzeug um es Entwicklern zu ermöglichen, Software zu schreiben, die auf eine bestimmte Anwendung aufbauen
Paketmanager ⁽¹²⁾	Software, mit der Programme installiert und deinstalliert werden können; logt installierte Pakete / Programme mit um Deinstallation zu vereinfachen; wird häufig in Linux-Distributionen verwendet
CPAN (Comprehensive Perl archive Network) ⁽¹³⁾	Repository für Perl-Module und Dokumentationen
Repository ⁽¹⁴⁾	Veraltetes Verzeichnis für Daten; wird häufig genutzt um verschiedene Versionen von Programmen zu verwalten und zu archivieren
Umgebungsvariablen ⁽¹⁵⁾	Systemvariablen, die die Verzeichnisse für bestimmte Programme festlegen