

Auf dieser Seite... (ausblenden)

1. 1. Einleitung
2. 2. Systemvoraussetzungen
 1. 2.1 Ein Webserver, der PHP-Skripte ausführen kann
 2. 2.2 PHP
 3. 2.3 MySQL
3. 3. Installation
 1. 3.1 Stud.IP downloaden
 2. 3.2 Anlegen der Stud.IP-Datenbank
4. 4. Konfiguration des Apache
 1. 4.1 Setzen eines symbolischen Links
 2. 4.2 Anlegen eines `Alias`
 3. 4.3 Verzeichnisse und Pfade
5. 5. Konfiguration von PHP
 1. 5.1 Variante 1: `php.ini`
 2. 5.2 Variante 2: `httpd.conf`
 3. 5.3 Variante 3: `.htaccess`
6. 6. Konfiguration von MySQL
 1. 6.1 `ROW_FORMAT Barracuda`
 2. 6.2 `SQL_MODE`
 3. 6.3 `FULLTEXT-Index`
 4. 6.4 Tuning von InnoDB
7. 7. Konfiguration von Stud.IP
 1. 7.1 Anpassungen in der Datei `config_local.inc.php`
 2. 7.2 `config.inc.php`
 3. 7.3 Optionale Module und Komponenten
 4. 7.4 Konfiguration des E-Mail-Versands
 5. 7.5 Stud.IP-interne Cronjobs
 6. 7.6 Konfiguration der Stud.IP Module und optionalen Features

§1. Einleitung

Die aktuellste Version dieser Anleitung finden Sie jederzeit im Hilfe-Wiki unter folgendem Link: <https://hilfe.studip.de/admin/Admins/Installationsanleitung> . Weitere nützliche Hinweise zur Installation und zum Betrieb finden sie im Dokumentationsverzeichnis des Release-Paketes (`/doc/de/`) in mehreren FAQs zu unterschiedlichen Themenbereichen.

Stud.IP setzt für den produktiven Einsatz ein LAMP-System¹ voraus. Alternativ ist eine Installation auf einem WAMP-System² möglich, wird jedoch von uns nicht empfohlen, da einige Kernfunktionen wie etwa das Versenden von E-Mails oder die Exportfunktion schwierig zu konfigurieren sind.

WICHTIGER HINWEIS: Diese Installationsanleitung setzt `root` Rechte bezüglich des Webserver (inclusive PHP Interpreter) und der MySQL Datenbank voraus.

§2. Systemvoraussetzungen

§2.1 Ein Webserver, der PHP-Skripte ausführen kann

Empfohlen wird der Apache³ in der Version 2.x unter Unix oder Windows.

¹ <http://de.wikipedia.org/wiki/Lamp>

² <http://de.wikipedia.org/wiki/WAMP#Varianten>

³ <http://httpd.apache.org>

§2.2 PHP⁴

Seit Stud.IP 3.4 wird PHP 5.4 zwingend vorausgesetzt. Wir empfehlen aus Sicherheitsgründen immer die neueste Version Ihrer Distribution.

PHP 5.4 ist frühestens ab Stud.IP 2.4 verwendbar, PHP 5.5 frühestens ab Stud.IP 3.1, PHP 5.6 frühestens ab Stud.IP 3.3. PHP 7.0 ist frühestens ab Stud.IP 3.5 verwendbar.

Ab der Stud.IP Version 4.0 wird PHP 5.5 zwingend vorausgesetzt.

Zusätzlich sind folgende Module erforderlich:

- Das PHP-Modul `mysql`⁵ (`pdo_mysql`) erlaubt den Zugriff auf einen MySQL Datenbank-Server.
- Das PHP-Modul `gettext`⁶ implementiert ein NLS (Native Language Support) API, das dazu verwendet wird, Stud.IP zu internationalisieren.
- Das PHP-Modul `session`⁷ bietet die Möglichkeit, bestimmte Daten während einer Folge von Aufrufen Ihrer Website festzuhalten.
- Das PHP-Modul `curl`⁸ für die Kommunikation mit anderen Diensten.
- Das PHP-Modul `gd`⁹
- Das PHP-Modul `mbstring`¹⁰ (ab Stud.IP 4.0)
- Für das Export-Tool werden zusätzlich die Module `xsl`¹¹ und `xml`¹² benötigt. Zudem müssen der Formatting Objects Processor¹³ und ein Java Runtime Environment¹⁴ installiert sein.
- Falls sie Nutzer via LDAP authentifizieren möchten, ist die Einrichtung des `LDAP-Moduls`¹⁵ nötig.
- Für den Import von Literaturlisten wird das Modul `dom`¹⁶ benötigt.
- Für das Öffnen/Erstellen von Zip-Archiven wird das Modul `zip`¹⁷ benötigt (ab Stud.IP 4.0).

Die korrekte Funktion von PHP im Zusammenspiel mit dem Apache und MySQL lässt sich mit einem einfachen Script feststellen, in dem folgender Text enthalten sein muss und anschließend in das Stammverzeichnis des Webservers gelegt wird (etwa unter dem Namen `test.php`):

```
<?
  phpinfo();
?>
```

Dabei werden alle einkompilierten Module angezeigt und es können die Einstellungen der `php.ini` überprüft werden.

4 <http://www.php.net>
5 <http://php.net/mysql>
6 <http://php.net/gettext>
7 <http://php.net/session>
8 <http://php.net/curl>
9 <http://php.net/gd>
10 <http://php.net/mbstring>
11 <http://php.net/xsl>
12 <http://php.net/xml>
13 <http://xml.apache.org/fop/>
14 <http://java.sun.com>
15 <http://php.net/ldap>
16 <http://php.net/dom>
17 <http://php.net/zip>

§2.3 MySQL¹⁸

Seit Stud.IP Version 3.4 werden InnoDB-Tabellen verwendet, diese müssen also aktiviert und konfiguriert sein. Benötigt wird mindestens Version 5.1, empfohlen wird eine MySQL-Version ab 5.5.

Ab Stud.IP Version 4.0 wird mindestens eine MySQL-Version 5.5.3 oder höher benötigt.

§3. Installation

§3.1 Stud.IP downloaden

Sie bekommen die neuesten Stud.IP-Releases unter <http://sourceforge.net/projects/studip>. Das Release ist sowohl als Zip-Datei (Endung: .zip) als auch als Tarball (Endung: .tar.bz2) erhältlich.

Auf einer typischen Unix-Kommandozeile geben sie:

```
tar jxvf studip-x.x.tar.bz2
```

bzw.:

```
unzip studip-x.x.zip
```

ein. Dieses Kommando legt ein neues Verzeichnis `studip-x.x/` an, das alle Stud.IP-Dateien und -Verzeichnisse enthält. Verschieben sie den eben erzeugten Ordner an eine Stelle im Dateisystem, auf die der Webserver **keinen** Zugriff hat:

```
mv studip-x.x /usr/local/studip
```

Die entpackten Dateien dürfen nicht über eine URL erreicht werden, um Sicherheitsprobleme einzuschränken. In den folgenden Schritten gehen wir von `/usr/local/studip` als Installationspfad Ihres Stud.IPs aus.

Kurzübersicht der im Paket vorhandenen Verzeichnisse:

<code>app</code>	Auf Trails (MVC-Paradigma) umgestellte Funktionen.
<code>cli</code>	Kommandozeilenschnittstelle für Stud.IP
<code>config</code>	Konfigurationsbereich inkl. Default-Konfigurationsdateien (<code>config_local.inc.php</code> , <code>config.inc.php</code>)
<code>data</code>	Default-Struktur für Datenverzeichnisse außerhalb des Apache-Document-Root-Verzeichnisses
<code>db</code>	SQL-Dumps für mysql-Datenbankeinrichtung und Dumps mit Demo-Daten und Migrationsskripte für ältere Stud.IP-Versionen (Für Versionsupgrade bitte <code>README</code> lesen!)
<code>doc</code>	Dokumentationen zur Installation
<code>lib</code>	Module/Bibliotheken für Stud.IP
<code>locale</code>	Übersetzungsskripte und Sprachdateien
<code>public</code>	Stud.IP-Kernsystem (später im Document-Root des Webservers)

¹⁸ <http://www.mysql.com>

```

templates
  GUI-Vorlagen (Flexi-Templates für nicht-MVC-Skripte)
test
  Unit-Tests
tools
  Werkzeuge zum Anpassen und Weiterentwickeln von Stud.IP
vendor
  extern entwickelte Bibliotheken

```

§3.2 Anlegen der Stud.IP-Datenbank

Stud.IP benötigt Zugriff auf eine zu erstellende MySQL-Datenbank. Es muss ein Datenbankbenutzer angelegt werden, der entsprechende Rechte erhält. Notieren sie sich bitte den Namen der Datenbank, den Namen und das Passwort des Datenbankbenutzers und den Datenbank-Hostname, da sie diese Informationen später benötigen werden.

Im Folgenden nennen wir die Datenbank **studip**.

Für Stud.IP-Versionen bis Stud.IP 3.5 kann die Datenbank auf derKommandozeile mit folgendem Befehl angelegt werden:

```
mysql -u root -p -e "CREATE DATABASE studip DEFAULT CHARACTER SET latin1 COLLATE latin1_german1_ci"
```

Ab Stud.IP 4.0 muss aufgrund der Umstellung auf die utf-8 Kodierung folgender Befehl verwendet werden:

```
mysql -u root -p -e "CREATE DATABASE studip DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci"
```

Hinweis für Stud.IP bis Version 3.5: Die Collation (`latin1_german1_ci`) und das Character Set (`latin1`) müssen von Ihnen angepasst werden, wenn sie nicht hauptsächlich deutsche Texte in Ihrem Stud.IP speichern wollen. Sollte es bei Ihnen Probleme mit dem Umlauten geben, z.B. weil sie eine aktuelle SuSE benutzen die von utf8 als Default Character Set ausgeht, dann helfen folgende Zeile in der `my.cnf` das Problem zu lösen:

```

skip_character_set_client_handshake
collation_server=latin1_german1_ci
character_set_server=latin1

```

Beim Anlegen des Datenbankbenutzers ist darauf zu achten, dass dieser folgende Rechte erhält: `SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, CREATETEMPORARYTABLES`.

Auf der Kommandozeile können Sie mit diesen Befehlen einen Benutzer 'studip' anlegen, und ihm die notwendigen Rechte erteilen:

```

mysql -u root -p -e "GRANT USAGE ON *.* TO 'studip'@'localhost' IDENTIFIED BY '<password>'"
mysql -u root -p -e "GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, CREATE
TEMPORARY TABLES ON studip.* TO 'studip'@'localhost'"

```

Als einfach zu bedienendes Werkzeug zur Arbeit mit der MySQL-Datenbank haben sich der MySQL Query Browser¹⁹ und phpMyAdmin²⁰ bewährt.

Das Datenbankschema für die von Ihnen erzeugten Datenbank ist in der Datei `/usr/local/studip/db/studip.sql` enthalten. Fügen sie die darin enthaltenen Tabellen Ihrer Datenbank hinzu:

```
mysql -u root -p studip < /usr/local/studip/db/studip.sql
```

19 <http://dev.mysql.com/downloads/query-browser/>
20 <http://www.phpwizard.net/projects/phpMyAdmin/>

Außerdem werden folgende Datenbankdumps mitgeliefert. **Bitte stellen sie die Dumps in der hier vorgegebenen Reihenfolge ein!**

`studip_root_user.sql`

Mit diesem Dump kann ein Stud.IP-Nutzer mit Status `root` (innerhalb von Stud.IP) angelegt werden.

Benutzername: `root@studip`

Passwort: `testing`

ACHTUNG: Das Passwort sollte nach erfolgreicher Installation sofort geändert werden! Dieser Dump sollte installiert werden, wenn sie den `root`-User nicht von Hand in der Datenbank anlegen möchten (nur für erfahrene Nutzer!)

`studip_default_data.sql`

Dieser Dump enthält die Voreinstellungen, die direkt in die Datenbank geschrieben werden.

Dieser Dump muss installiert werden.

`studip_resources_default_data.sql`

Wenn sie die Ressourcenverwaltung nutzen möchten, installieren sie diesen Dump.

`studip_demo_data.sql`, `studip_resources_demo_data.sql`

Wenn sie Stud.IP zunächst testen wollen, spielen sie diese Dumps ein.

Enthalten sind folgenden Testaccounts: `test_admin`, `test_dozent`, `test_tutor` und `test_autor`. Bei allen Accounts ist das Passwort ebenfalls `testing`.

ACHTUNG: Bitte löschen sie diese Accounts, wenn das System in den Produktivbetrieb geht oder die Testphase endet!

§4. Konfiguration des Apache

Dem Webserver müssen nun die Dateien, die sich in `/usr/local/studip/public` befinden, verfügbar gemacht werden. Zu diesem Zweck gibt es verschiedene Möglichkeiten, von denen hier zwei detaillierter erklärt werden.

§4.1 Setzen eines symbolischen Links

Damit dieser Weg funktioniert, muss der Apache symbolischen Links folgen dürfen. Die relevanten Optionen müssen daher `FollowSymLinks` oder `SymLinksIfOwnerMatch` enthalten.

Für das Verständnis der folgenden Erläuterungen, müssen sie den `DocumentRoot` Ihres Apache kennen. Für diese Installationsanleitung gehen wir von `/srv/www/` aus.

Geben sie auf der Kommandozeile als `root` folgende Kommandos ein:

```
cd /srv/www
```

```
ln -s /usr/local/studip/public studip
```

Nach erfolgreichem Absetzen wurde ein symbolischer Link im Verzeichnis `/srv/www/` angelegt, so dass nun im Browser unter der URL `http://www.example.com/studip/` Stud.IP zur Verfügung steht.

§4.2 Anlegen eines Alias

Mit der `Alias`-Direktive können sie einen beliebigen Teil des Dateisystems in den "web space" abbilden. Mit der folgenden Zeile in Ihrer Apache-Konfiguration:

```
Alias /studip /usr/local/studip/public
```

wird die URL `http://www.example.com/studip/xyz.php` auf den Pfad `/usr/local/studip/public/xyz.php` abgebildet.

§4.3 Verzeichnisse und Pfade

In den folgenden Verzeichnissen, die in der `config/config_local.inc.php` angegeben werden, muss der Webserver-User (zumeist ist das `wwwrun`, zur Sicherheit am besten in der Datei `httpd.conf` nachsehen) Schreibrechte haben:

```
$UPLOAD_PATH
    Ablage für hochgeladene Dokumente
    (Standardwert: <Installationspfad>/data/upload_doc/)
$USER_DOC_PATH
    Ablage für persönliche Dokumente
    (Standardwert: <Installationspfad>/data/user_doc/)
$PLUGIN_ASSETS_PATH
    Hier werden kompilierte less-dateien (also css-dateien) von Plugins hin geschrieben.
    (Standardwert: <Installationspfad>/data/assets_cache)
$ARCHIV_PATH
    Ablage für archivierte Veranstaltungen
    (Standardwert: <Installationspfad>/data/archiv/)
$EXTERN_CONFIG_FILE_PATH
    wenn die SRI-Schnittstelle benutzt wird
    (Standardwert: <Installationspfad>/data/extern_config/)
$MEDIA_CACHE_PATH
    wenn der Media-Proxy zur Einbindung externer Medien (Bilder, Audio/Video-Dateien) benutzt wird
    (Standardwert: <Installationspfad>/data/media_cache)
$DYNAMIC_CONTENT_PATH
    vom System erzeugten Bilder (Nutzerbilder, Smileys, usw.)
    (Standardwert: <Installationspfad>/public/pictures/)
$tmp_PATH
    Ablage für temporäre Dateien
    (Standardwert: /tmp/)
```

Die ersten sechs Verzeichnisse dürfen **NICHT** von außen über den Webserver erreichbar sein.

Der Zugriff auf das Temp-Verzeichnis (Standardwert: `/tmp`) muss ebenfalls gewährleistet sein.

Die einfachste (aber im Produktivbetrieb nicht empfohlene) Möglichkeit ist, das komplette Stud.IP Verzeichnis dem Webserver Nutzer zu geben, z.B.:

```
chown -R wwwrun /usr/local/studip
```

§5. Konfiguration von PHP

Für den Betrieb von Stud.IP müssen bestimmte Parameter von PHP konfiguriert werden:

Folgende Werte müssen angepasst werden:

```
upload_max_filesize = 8M
```

Dieser Wert beeinflusst direkt die maximale Größe von Datei-Uploads. Wenn sie größere Dateien zulassen wollen, müssen sie diesen Wert entsprechend hoch einstellen.

```
memory_limit = 128M
post_max_size = 9M
```

Diese Werte bestimmen das Speicherlimit eines PHP-Skripts bzw. die maximale Größe von HTTP-POST-Requests. Letzterer Wert muss geringfügig über `upload_max_filesize` liegen.

```
max_execution_time = 300
```

Dieser Wert beeinflusst die maximale Ausführungsdauer eines PHP-Skriptes. Ein höherer Wert hat sich hier bewährt.

```
allow_url_fopen On
```

Wenn sie die Dateiverlinkung benutzen wollen, muss diese Option eingeschaltet sein.

```
error_reporting E_ALL & ~E_NOTICE & ~E_STRICT & ~E_DEPRECATED
```

Die Fehlerausgabe sollte auf diese Optionen reduziert werden.

Für die Konfiguration sind drei unterschiedliche Varianten möglich:

- Setzen sie direkt die entsprechenden Werte in der PHP-Konfigurationsdatei `php.ini`. Wenn sie auf Ihrem Server Stud.IP als einzige PHP-Software einsetzen, ist dies eine gute Wahl.
- Sie können die Werte auch direkt in der Apache-Konfigurationsdatei `httpd.conf` per `<Directory>`-Anweisung setzen. Das hat den Vorteil, dass verschiedene PHP-Programme nebeneinander installiert werden können und verschiedene Einstellungen der Optionen je nach Verzeichnis gelten.
- Schließlich können sie auch die Werte direkt in einer `.htaccess` Datei ablegen und damit direkt im Verzeichnis der Software die Einstellungen setzen. Diese Variante hat den Vorteil, dass sie keinen Zugriff auf den Webserver benötigen (etwa bei einer Installation auf dem Rechner eines externen Providers).

§5.1 Variante 1: `php.ini`

Sie sollten diese Variante der Konfiguration nur wählen, wenn sie nur eine PHP-Webapplikation installiert haben. Wir empfehlen für einen reibungslosen Betrieb die beiden letzten Varianten.

Die PHP-Konfigurationsdatei `php.ini` sollte sich nach der Installation von PHP in `/etc/` befinden. Bei aktuellen Distributionen finden sie die Datei in `/etc/php4/apache2` respektive in `/etc/php5/apache2`. Wenn sie das oben erwähnte Skript ausführen, finden sie unter dem Eintrag "Configuration File (`php.ini`) Path" den für Ihre Installation gültigen Pfad.

§5.2 Variante 2: `httpd.conf`

Nach Möglichkeit sollte die Anpassungen der PHP-Konfigurationseinstellungen über eine `<Directory>`-Anweisung in der Apache Konfigurationsdatei vorgenommen werden. Auf diese Weise werden andere laufende PHP-Applikationen nicht beeinflusst. Im Installationsarchiv unter `config` befindet sich eine Datei `studip-httpd.conf.dist`, die alle notwendigen Einstellungen enthält. Mit einer aktuellen SuSE Distribution wäre die Vorgehensweise folgende:

- Kopieren von `studip-httpd.conf.dist` nach zB. `/etc/apache2/conf.d/studip-httpd.conf`
- Anpassen der Pfadangaben in `studip-httpd.conf`
- Webserver neustarten

In anderen Distributionen oder mit älteren Apache Versionen kann man den Inhalt von `studip-httpd.conf.dist` einfach in die `httpd.conf` kopieren, oder eine `include` Anweisung verwenden, um `studip-httpd.conf` einzubinden.

§5.3 Variante 3: `.htaccess`

Ähnlich verhält es sich, wenn die Werte über eine `.htaccess` Datei verändert werden. Diese Variante hat den Vorteil, dass man keinen Zugriff auf den Webserver bzw. die Konfigurationsdatei des Webserver haben muss. Dazu muss in der `httpd.conf` lediglich der Eintrag:

```
AllowOverride All
```

entweder global für alle Verzeichnisse oder das verwendete Verzeichnis als `<Directory>`-Anweisung eingetragen worden sein.

Anschließend legt man in das `public` Verzeichnis Ihrer Stud.IP-Installation (also `/usr/local/studip/public`) eine `.htaccess`-Datei an, die die entsprechenden Werte der `php.ini` enthält.

Im Verzeichnis `config` befindet sich eine solche `.htaccess.dist`-Datei, die alle notwendigen Einstellungen enthält.

§6. Konfiguration von MySQL

Durch die Umstellung der Stud.IP-Datenbank von MyISAM auf InnoDB ab Version Stud.IP-3.4 gibt es Einstellungen, die beachtet werden sollten:

§6.1 ROW_FORMAT Barracuda

MySQL ab der Version 5.5 (wie auch MariaDB und Percona Server) unterstützen neben dem schon länger bestehenden Format `Antelope` nun das neue Zeilenformat `Barracuda`. Dies ist auch das empfohlene Zeilenformat für InnoDB-Tabellen, und zwar in der Variante `DYNAMIC` (nicht `COMPRESSED`). Wenn Sie noch das `Antelope`-Format verwenden, haben Sie entweder `COMPACT` oder `REDUNDANT` als `ROW_FORMAT` Ihrer Tabellen.

Um `Barracuda` verwenden zu können, müssen zwei Variablen Ihres Datenbankservers auf bestimmte Werte gesetzt sein:

```
innodb_file_per_table = 1
innodb_file_format = Barracuda
innodb_large_prefix = 1
```

Mit diesen beiden Werten konfigurieren Sie Ihren Server so, dass pro Datenbanktabelle eine eigene Datei verwendet wird und das Format dieser Datei im `Barracuda`-Format sein kann.

Ab MySQL 5.7.6 gilt obige Einschränkung nicht mehr (<https://dev.mysql.com/doc/refman/5.7/en/innodb-row-format-dynamic.html>) und es kann ab Version 5.7.9 auch über die Systemvariable `innodb_default_row_format` vorgegeben werden, dass für neue Tabellen `Barracuda` (`DYNAMIC`) verwendet werden soll. Ältere Versionen müssen dies beim Anlegen der Tabelle explizit im `CREATE TABLE`-Query angeben (`ROW_FORMAT=DYNAMIC`).

Für nachträgliche Änderungen an den Datenbanktabellen gibt es im `cli`-Ordner die beiden `php`-Skripten

```
myisam_to_innodb.php
antelope_to_barracuda.php
```


Diese können Sie jederzeit ausführen und so die Änderungen auch nach Updates des Datenbankservers oder Einstellung der entsprechenden Variablen nachziehen. Bereits konvertierte Datenbanktabellen bleiben völlig unberührt.

§6.2 SQL_MODE

In neueren Mysql Versionen ist die Einstellung STRICT_TRANS_TABLES aktiviert, dies führt ab Stud.IP Version 3.4 zu Problemen. Diese Einstellung sollte dann in die Konfiguration der Datenbank mit aufgenommen werden:

```
sql_mode = NO_ENGINE_SUBSTITUTION
```

§6.3 FULLTEXT-Index

FULLTEXT-Indizes werden für InnoDB aber erst ab MySQL 5.6 unterstützt. Bei der Migration der Stud.IP-Tabellen werden daher für alle älteren MySQL-Versionen Tabelle, die einen solchen Index benutzen im MyISAM-Format belassen. Bei einem späteren Upgrade des Datenbankservers können Sie die Tabelle z.B. über das mitgelieferte Kommandozeilenskript `cli/myisam_to_innodb.php` nachträglich konvertieren.

§6.4 Tuning von InnoDB

Um die bestmögliche Leistung aus der Datenbank herauszuholen, müssen für InnoDB andere Variablen des Datenbankservers gesetzt werden als es bei MyISAM der Fall war. Hier seien besonders folgende Links empfohlen:

- <https://www.percona.com/blog/2013/09/20/innodb-performance-optimization-basics-updated/>
- <https://www.percona.com/blog/2014/01/28/10-mysql-settings-to-tune-after-installation/>
- <https://www.percona.com/live/london-2014/sessions/tuning-mysql-its-about-performance>
- <http://www.tocker.ca/2013/09/17/what-to-tune-in-mysql-56-after-installation.html>

Für eine Stud.IP-Test-Installation ist keinerlei Konfiguration von MySQL notwendig. Für ein Produktivsystem vor allem bei großen Nutzerzahlen ist aber eine Optimierung der Standardeinstellungen erforderlich:

```
innodb_buffer_pool_size = Größe der DB
innodb_buffer_pool_instances = 1 / GB Größe
innodb_log_file_size = 50MB
innodb_flush_log_at_trx_commit = 2
```

§7. Konfiguration von Stud.IP

Für die Stud.IP-Konfiguration müssen die Dateien `config/config_local.inc.php` und `config/config.inc.php` angepaßt werden. Außerdem müssen einige Verzeichnisse für den Apache Webserver schreibbar gemacht werden. Die Konfiguration der optionalen Module und Komponenten erfolgt ebenfalls in der Datei `config/config_local.inc.php`:

- Export-Modul
- E-Learning Schnittstelle (Ilias 4 + 5)
- Literaturverwaltung
- Plugin Schnittstelle
- Authentifizierungsplugins (LDAP)

§7.1 Anpassungen in der Datei `config_local.inc.php`

Die Datei `config_local.inc.php` im Verzeichnis `config` ist eine der beiden zentralen Konfigurationsdateien des Stud.IP-Systems. Hier werden vor allem grundlegende technische Einstellungen festgelegt.

Weitere Einstellungen können in der Datei `config.inc.php` vorgenommen werden (dort sind vor allem inhaltliche Einstellungen wie diverse Bezeichnungen und andere Voreinstellungen abgelegt) oder im System im Konfigurationseditor bearbeitet werden. Letzter befindet sich noch im Testbetrieb, so dass bisher nur ein sehr kleiner Teil der Einstellungen darüber bearbeitet werden kann.

Kopieren sie die Dateien `config_defaults.inc.php` und `config.inc.php.dist` nach `config_local.inc.php` und `config.inc.php`. z.B.:

```
cd /usr/local/studip/config
cp config_defaults.inc.php.dist config_local.inc.php
cp config.inc.php.dist config.inc.php
```

Danach gehen Sie die Einstellungen in der neuen `config_local.inc.php` einzeln durch. Bei jeder Variable entscheiden Sie sich nach Möglichkeit entweder dafür, diese umzuändern oder zu löschen. Ihre `config_local.inc.php` sollte am Ende nur Variablen besitzen, die auch wirklich von den Standardwerten abweichen. Zu den einzelnen Werten:

Die im Abschnitt "Anlegen der Stud.IP-Datenbank" notierten Werte werden in den folgenden Variablen benötigt:

```
$DB_STUDIP_HOST
    Datenbankhostname
$DB_STUDIP_USER
    Datenbanknutzer
$DB_STUDIP_PASSWORD
    Datenbankpassword
$DB_STUDIP_DATABASE
    Datenbankname
$MAIL_HOST_NAME
    Anpassung des Mailhostes, wenn Mails nicht über einen lokalen MTA, sondern einen externen Rechner
    verschickt werden sollen
$UPLOAD_PATH
    Anpassung des Pfades zum Upload-Verzeichnis, muss nicht für den Webserver sichtbar sein

ACHTUNG: Für dieses Verzeichnis benötigt der Webserver Schreibrechte!
$ARCHIV_PATH
    Absoluter Pfad zum Archiv-Verzeichnis, muss nicht für den Webserver sichtbar sein

ACHTUNG: Für dieses Verzeichnis benötigt der Webserver Schreibrechte!
$TMP_PATH
    Anpassung des Temp-Verzeichnisses, falls nötig
$ABSOLUTE_URI_STUDIP
    muss ggf. angegeben werden, wenn der Server aus irgendwelchen Gründen die korrekte URL nicht zur
    Laufzeit bestimmt werden kann (z.B. weil er nur über einem Proxy mit dem Internet verbunden ist).
```

Außerdem:

- Anpassung der Pfade zu allen benötigten Tools für die Bildbearbeitung
- Kann die Stud.IP Installation über mehrere Adressen (etwa der reale Servername und ein virtueller Webserver) erreicht werden, sollte sie im Abschnitt "domain and path translation" alle verfügbare Servernamen bzw. Pfadangaben ablegen. Stud.IP "übersetzt" dann evtl. im System erstellte interne Links in die jeweils passende Form (ansonsten gibt es Probleme mit der Cookie-Verwaltung durch den Browser).
- Falls Sie ein Update von einer 3.x-Version auf eine 4.x Version von Stud.IP durchführen, werden Sie sich wundern, dass die `config_local.inc.php` in der 4.x nur noch ganz wenige Einträge enthält und in der 3.x sehr viele Einträge. Ganz wichtig dazu: behalten Sie Ihre alte `config_local.inc.php` an Ort und

Stelle, führen Sie dann die Datenbankmigrationen aus. Erst danach sollten Sie Ihre alte `config_local.inc.php` aufräumen und alles raus schmeißen, was nicht in der `config_defaults.inc.php` genauso auch drin steht oder in der `config_defaults.inc.php` nicht mehr erwähnt wird (diese Einstellungen werden in 4.x in der Datenbank gepflegt). Am Ende sollten nur die Einträge in der `config_local.inc.php` stehen, die von den Einträgen in der `config_defaults.inc.php` abweichen. Aber entschlacken Sie Ihre `config_local.inc.php` erst **nach der Migration**, ansonsten fehlt Ihrem Stud.IP etwas.

§7.2 config.inc.php

Zum Betrieb von Stud.IP muss im `conf`-Verzeichnis der Stud.IP Installation die Datei `config.inc.php` angelegt werden. Sie finden dort bereits eine Distributionsversion mit dem Namen `config.inc.php.dist`. Benennen Sie die Datei in `config.inc.php` um.

In der Datei `config.inc.php` sollten Sie zunächst nur die allgemeinen Daten wie `$UNI_URL`, `$UNI_CONTACT` und `$UNI_INFO` setzen. Alle weiteren Einstellungen sollten als Standardwerte gut funktionieren - verändern Sie diese Optionen bitte erst, wenn Sie mit dem System bereits vertraut sind.

ACHTUNG: Wenn Sie das System im Regelbetrieb einsetzen, sollten nachträglich außer den obengenannten Optionen möglichst keine Einstellungen mehr verändert werden, da sonst Inkonsistenzen in der Datenbank auftreten können.

§7.3 Optionale Module und Komponenten

Unter Administration -> Globale Einstellungen -> Konfiguration können Sie auch die Aktivierung der Inhaltelemente (Module) vornehmen. Für jedes Modul gibt es einen Schalter `$<MODULNAME>_ENABLE`, über den Sie das Modul ein- und ausschalten können. Diese befinden sich in der Gruppe "modules".

Einige Module verlangen weitere Angaben in der `config_local.inc.php`, wie Pfadangaben oder andere Einstellungsoptionen, die Sie im Normalfall einfach so lassen, wie im Auslieferungszustand eingestellt.

Wichtig sind nur folgende Optionen:

- Falls Sie das Modul "Export" benutzen wollen, so müssen die Werte `$FOP_SH_CALL` und evtl. `$JAVA_ENV_CALL` entsprechend gesetzt werden.

Neben den oben beschriebenen Optionen gibt es noch weitere Parameter, die in der `config_local.inc.php` optional festgelegt werden können, wie z.B. `$LATEXRENDER_ENABLE`

- `$ALLOW_GROUPING_SEMINARS`, `$ALLOW_SELFASSIGN_STUDYCOURSE`, `$SHOW_TERMS_ON_FIRST_LOGIN` sind einige erweiterte Einstellungen, die Sie unter Umständen umstellen können, etwa wenn Sie mit einer externen Authentifizierung (LDAP o.ä.) arbeiten
- Einstellungen für die Internationalisierung von Stud.IP (braucht im Normalfall nicht angepasst zu werden)
- Einstellungen für Authentifizierungs-Plugins (brauchen nur angepasst werden, wenn Sie eine externe Authentifizierung einsetzen).
- Weitere Optionen zur Authentifizierung können in Verbindung mit der Standard-Authentifizierung (also dem Auslieferungszustand) angepasst werden:
 - `$ALLOW_CHANGE_USERNAME`
 - `$ALLOW_CHANGE_EMAIL`
 - `$ENABLE_SELF_REGISTRATION`

Weitere Parameter können seit der Version 1.2 direkt im Konfigurationseditor im Bereich "globale Einstellungen", "Konfiguration" durch `root`-Benutzer im Webfrontend eingegeben bzw. umgestellt werden.

§7.4 Konfiguration des E-Mail-Versands

Damit das Registrierungsmodul von Stud.IP funktioniert, muss der E-Mailversand konfiguriert sein, d.h. der Webserver muss in der Lage sein E-Mails über den in `$MAIL_HOST_NAME` angegebenen E-Mailserver zu verschicken. In den verschickten E-Mails ist Reply-To auf `abuse@<SERVER_NAME>` gesetzt, E-Mails an diese Adresse sollten an einen der Administratoren weitergeleitet werden.

ACHTUNG: Wenn der E-Mailversand nicht ordnungsgemäß konfiguriert wurde, funktioniert weder die Selbstregistrierung von Nutzern noch das Eintragen von neuen Nutzern, da die Registrierungs-E-Mail bzw. das Passwort auf diese Weise an die Nutzer zugestellt werden.

Die beiden Parameter `$MESSAGING_FORWARD_AS_EMAIL` und `$MESSAGING_FORWARD_DEFAULT` erlauben es, systeminterne Nachrichten die gespeicherte E-Mail-Adresse eines Nutzers weiterzuleiten und dafür einen Standardwert festzulegen.

E-Mailbenachrichtigungen einrichten

Der Versand von Mailbenachrichtigungen über neue Inhalte kann über den Cronjob "Versendet tägliche Mailbenachrichtigungen" eingeschaltet werden. Hierzu wählt man als root-Nutzer die Cronjob-Einstellungen unter Admin -> System -> Cronjobs aus und setzt das Aktiv-Häkchen bei diesem Cronjob.

§7.5 Stud.IP-interne Cronjobs

Um die Stud.IP-internen Cronjobs nutzen zu können, müssen folgende Schritte durchgeführt werden.

- Die Datei `cli/cronjob-worker.php` muss als globaler Cronjob in Ihrem System eingerichtet. Dieser Cronjob sollte minütlich laufen, die Steuerung der Ausführung der einzelnen Stud.IP-Cronjobs wird dann intern verwaltet. Wenn das PHP-Kommando in `/usr/bin/php` zu finden ist (`which php` auf der Kommandozeile gibt darüber Auskunft), und die Datei ausführbar gemacht ist, kann man das Skript direkt in einen cronjob einbauen. Ein entsprechender Eintrag in `crontab` würde beispielsweise folgendermassen aussehen:


```
* * * root php /usr/local/studip/cli/cronjob-worker.php >>
/var/log/studip-cronjobs 2>&1
```
- In der globalen Konfiguration unter *Admin / Globale Einstellungen / Konfiguration* muss der Wert für `CRONJOBS_ENABLE` auf `true` gestellt werden, um die Cronjobs tatsächlich zu aktivieren.

Die Konfiguration für die einzelnen Cronjobs finden sich unter *Admin / Globale Einstellungen / Cronjobs*, sofern der oben genannte Wert `CRONJOBS_ENABLE` gesetzt ist.

§7.6 Konfiguration der Stud.IP Module und optionalen Features

Neben dem Basissystem existieren einige Stud.IP Module, die besondere Voraussetzungen zum Betrieb benötigen oder speziell konfiguriert werden müssen. In Auslieferungszustand werden diese Module mit installiert, daher sollten sie im entsprechenden Abschnitt zur Datei `config_local.inc.php` einen Blick darauf werden, welche Module existieren und welche überhaupt benötigt werden.

Export-Modul

Für den Export von Daten als HTML, PDF etc. wird die XSLT-Unterstützung für PHP benötigt. Falls die in der verwendeten PHP-Installation nicht vorhanden ist, sind die Sablotron und expat Bibliotheken zu installieren und PHP entsprechend neu zu konfigurieren (`configure --enable-xslt --with-xslt-sablot ...`). Für Debian erledigt beides das Paket `php4-xslt`.

Für den Export als PDF ist zusätzlich FOP (Formatting Objects Processor) aus dem Apache XML Projekt notwendig +²¹. FOP ist in Java realisiert, es wird also auch ein lauffähiges JRE (Java Runtime Environment) +²² benötigt.

\$FOP_SH_CALL muss den kompletten Pfad zum FOP Startskript enthalten.

In \$JAVA_ENV_CALL muss evtl. der Pfad zu einem Skript zum setzen der Java Umgebungsvariablen eingetragen werden (/etc/profile.d/alljava.sh bei einer SuSE Distribution).

Grafische Evaluationsauswertung

Die Evaluationsauswertung kann ebenfalls als PDF exportiert werden, und benötigt dazu den oben erwähnten FOP.

Elearning Schnittstelle (Ilias 4 + 5)

Eine ausführlichere Anleitung für dieses Modul finden sie in der Datei doc/de/verbindung_studip_ilias3.rtf oder doc/de/verbindung_studip_ilias3.pdf in diesem Verzeichnis. Die wichtigsten Schritte in einer Kurzfassung.

- Installieren sie Ilias in einer Version ab 3.7.1 nach Anleitung. Die Installation kann auch auf einem anderen Server liegen.
- Aktivieren sie in Ilias unter Administration/Einstellungen den Punkt "Externe Benutzerverwaltung (SOAP)".
- Legen sie in Ilias einen Administrationsaccount an, der zur Kommunikation benutzt werden soll. **Wichtig:** Melden sie sich einmal mit diesem Account in Ilias an!
- Legen sie im "Magazin" eine Kategorie an, unter der die Daten von Stud.IP Nutzern gespeichert werden sollen.
- In der Konfiguration von Stud.IP (config_local.inc.php) setzen sie \$SOAP_ENABLE auf TRUE.
- In der Variable \$ELEARNING_INTERFACE_MODULES['ilias3'] tragen sie die Daten ihrer Ilias Installation ein (name, ABSOLUTE_PATH_ELEARNINGMODULES, ABSOLUTE_PATH_SOAP, soap_data (username und password für den gerade angelegten Administratoraccount, client_id (Name des Ilias Mandanten) der Ilias Installation)).

- Rufen sie in Stud.IP als root in den Administrationsbereich unter Tools/Lernmodul-Schnittstelle und wählen sie ihre Ilias Installation aus. Geben sie weiter unten auf der Seite unter "Kategorie" den Namen der Kategorie ein, die sie im Ilias "Magazin" erstellt haben, und klicken sie auf "aktivieren".

Literaturverwaltung

Zunächst ist sicherzustellen, dass PHP mit der Yaz-Erweiterung kompiliert wurde. Eine kurze Erklärung finden sie unter [1]²³. Diese dient dazu, externe Literaturkataloge über die z39.50-Schnittstelle abzufragen. Sie können neben den mitgelieferten Plugins weitere Plugins im Verzeichnis lib/classes/lit_search_plugins für ihre Bibliothek(en) anlegen. Bitte nutzen sie die mitgelieferten Plugins für den GVK (Gemeinsamer Verbundkatalog des GBV) und die SUB Göttingen (Niedersächsische Staats- und Universitätsbibliothek) als Vorlage für weitere Plugins und erkundigen sie sich bei Ihrer Bibliothek nach Einzelheiten für die Anbindung. Falls ihre Bibliothek Mitglied im GBV ist, können sie einfach eine neue Klasse von StudipLitSearchPluginGvk.class.php ableiten, und müssen nur die Felder z_host und z_profile anpassen. Der Name ihres Plugins muss mit StudipLitSearchPlugin beginnen.

Dann können sie das Plugin in der Datei config/config_local.inc.php unterhalb von "literature search_plugins" aktivieren, indem sie es in das Array \$_lit_search_plugins eintragen. Hier können sie zusätzlich einen Link eintragen, der in den Web Opac ihrer Bibliothek verweist.

²¹ <http://xml.apache.org/fop/download.html>

²² <http://java.sun.com/>

²³ <http://www.php.net/yaz>

Downloadlinks

In Stud.IP wird an vielen Stellen zum Übertragen von Dateien das PHP-Skript `sendfile.php` verwendet. Seit der Stud.IP-Version 1.3 besteht die Möglichkeit das Aussehen der Downloadlinks über den Parameter `SENDFILE_LINK_MODE` in der Systemkonfiguration zu ändern. Es sind die Werte `old`, `normal` und `rewrite` möglich.

`old`

```
http://www.server.de/studip/sendfile.php?force_download=1&type=0&file_id=43dce79ba3ab55b906c59317b5e6cc03&file_name=test.txt
```

`normal`

```
http://www.server.de/studip/sendfile.php?force_download=1&type=0&file_id=43dce79ba3ab55b906c59317b5e6cc03&file_name=test.txt
```

Der Unterschied zwischen `old` und `normal` besteht lediglich in einem zusätzlichen "/" hinter dem Fragezeichen das Probleme bei der Anzeige von PDF-Dateien bei einigen Browsern umgehen soll.

`rewrite`

```
http://www.server.de/studip/download/force_download/0/43dce79ba3ab55b906c59317b5e6cc03/test.txt
```

Der `rewrite`-Modus liefert eine URL die mit jedem Browser problemlos nutzbar ist. Benötigt wird dazu das Apache-Rewrite-Modul. Es wird mit einem "RewriteEngine on" in der Datei `studip-httpd.conf` eingeschaltet. Die weitere Konfiguration in der `studip-httpd.conf` unterscheidet sich geringfügig je nachdem ob das Stud.IP-Verzeichnis direkt im <Document-Root> des Apache liegt oder ob es per Alias-Anweisung eingebunden wird.

Innerhalb des <Document-Root>:

```
<Directory "/srv/www/htdocs/studip">
# für rewrite wird die Option FollowSymLinks oder SymLinksIfOwnerMatch benötigt ...

#Options SymLinksIfOwnerMatch
Options FollowSymLinks

RewriteEngine on
RewriteRule ^download/(normal|force_download|zip)/([0-46])/([^/]+)/(.+)$
sendfile.php?$1=1&type=$2&file_id=$3&file_name=$4 [L]
RewriteRule ^download/(normal|force_download|zip)/5/([^/]+)/([^/]+)/(.+)$
sendfile.php?$1=1&type=5&range_id=$2&list_id=$3&file_name=$4 [L]

...
</Directory>
```

per Alias:

```
Alias /abcd /srv/studip/public

<Directory "/srv/studip/public">
# für rewrite wird die Option FollowSymLinks oder SymLinksIfOwnerMatch benötigt ...

#Options SymLinksIfOwnerMatch
Options FollowSymLinks

RewriteEngine on
RewriteBase /srv/studip/public
RewriteRule ^download/(normal|force_download|zip)/([0-46])/([0-9a-f]+)/(.+)$
/abcd/sendfile.php?$1=1&type=$2&file_id=$3&file_name=$4 [L]
RewriteRule ^download/(normal|force_download|zip)/5/([^/]+)/([^/]+)/(.+)$
/abcd/sendfile.php?$1=1&type=5&range_id=$2&list_id=$3&file_name=$4 [L]
```

```
...
</Directory>
```

Die `rewrite`-Regeln befinden sich bereits in der mitgelieferten `studip-httpd.conf`, es müssen nur die Kommentarzeichen vor der gewünschten Variante entfernt werden.

Plugin Schnittstelle

Die PluginEngine lässt sich über verschiedene Konfigurationsvariablen in der Datei `config_local.inc.php` anpassen. Zur Aktivierung der PluginEngine muss die Variable `$PLUGINS_ENABLE` auf `TRUE` gesetzt werden. Danach erscheint auf der Startseite des `root`-Administrators der Menüeintrag Verwaltung von Plugins. In der Standardkonfiguration können hier nur Pluginpakete aus einem definierten Verzeichnis auf dem Webserver in Stud.IP importiert werden. Dieses Verzeichnis lässt sich über die Variable `$NEW_PLUGINS_PATH` ändern. Hier muss ein absoluter Pfad angegeben werden. Dieses Verzeichnis muss vom Webserver-Benutzer lesbar sein.

Über die Variable `$PLUGINS_UPLOAD_ENABLE` lässt sich ein Upload-Formular freischalten, so dass der `root`-Administrator Plugin-Pakete direkt per Upload in das System laden kann. Sollte der `root`-Administratoren-Account jedoch gehackt werden, so stellt dies eine nicht zu unterschätzende Sicherheitslücke dar, denn einem Hacker werden hier weitreichende Möglichkeiten eröffnet, indem nahezu beliebiger Sourcecode in das System geladen werden kann.

Um auch bei aktiviertem Upload-Formular dennoch eine gewisse Sicherheit zu erreichen, ist der Einsatz des PHP-Safemodes zu empfehlen. Ebenso sollte der WebServer niemals mit dem Benutzer `root` gestartet werden. Der Einsatz einer `chroot`-Umgebung für den Webserver ist ratsam. Plugins werden in der Standardkonfiguration in ein Verzeichnis `plugins_packages` abgelegt, welches sich unterhalb des `public`-Verzeichnisses befindet. Dieses Verzeichnis muss für den Webserver-Nutzer schreibbar sein. Dem Verzeichnis sollten die Rechte `755` eingeräumt werden.

LDAP Authentifizierung

Zur Authentifizierung gegen einen LDAP Server stehen im Release drei Plugins zur Verfügung (in der `config/config_local.inc` als `Ldap`, `LdapReader` und `LdapReadAndBind` bezeichnet. Die entsprechenden Klassen befinden sich im Verzeichnis `lib/classes/auth_plugins` und heißen dort `StudipAuthLdap.class.php`, `StudipAuthLdapReader.class.php` und `StudipAuthLdapReadAndBind.class.php`. Die Konfiguration dieser Plugins sollte in der `config_local.inc` vorgenommen werden. Hierzu muss der Name des Plugins in die Konfigurationsvariable `$STUDIP_AUTH_PLUGIN[]` eingetragen werden, normalerweise reicht das Entfernen der Kommentarzeichen. Es kann mehr als ein Plugin aktiviert sein, die Reihenfolge, in der die Plugins bei einem Authentifizierungsvorgang abgearbeitet werden, kann hier auch vorgenommen werden. Unterhalb von `$STUDIP_AUTH_PLUGIN[]` müssen dann die Optionen für das ausgewählte Plugin konfiguriert werden (`$STUDIP_AUTH_CONFIG_xxx`). Für `Ldap` sind folgende Einstellungen nötig:

- `host`: Hostname des LDAP Servers, oder eine URL der Form `ldap[s]://[hostname]:[port]`
- `protocol_version`: LDAPv2 oder LDAPv3, für OpenLDAP 2.x.x besser 3
- `base_dn`: base DN, die Suchbasis unterhalb derer die Einträge zu finden sein müssen
- `username_attribute`: LDAP-Attribut, welches den Nutzernamen enthält. Entspricht dem LDAP-Suchfilter `username_attribute=%u`.
- `ldap_filter`: LDAP-Suchfilter²⁴, der verwendet wird, um den Nutzer im LDAP zu finden. Die Einstellung für `ldap_filter` hat Vorrang vor dem `username_attribute`, d.h. `username_attribute` wird nicht verwendet, wenn es einen `ldap_filter` gibt. Im Filter-Ausdruck werden bestimmte Platzhalter ersetzt: `%u` durch den Nutzernamen, `%U` durch den lokalen Namensbestandteil (vor dem `@`) und `%d` durch den Domänenteil (nach dem `@`), `%%` durch ein einfaches `%`.
- `anonymous_bind`: lässt der LDAP Server ein anonymes anmelden zu, so kann vor der eigentlichen Authentifizierung das Verzeichnis nach dem eingegebenen Nutzernamen durchsucht werden, um den korrekten Nutzer DN zu finden. Ist das anonyme Binden ausgeschaltet, muss die Suchbasis dem Zweig entsprechen, indem die Nutzerdaten zu finden sind, da der Nutzer DN in diesem Fall direkt aus der Suchbasis

²⁴ <http://www.ietf.org/rfc/rfc2254.txt>

gebildet wird. Außerdem ist dann keine Möglichkeit gegeben, eine Meldung über ein falsches Passwort zu geben.

- `error_head`: Präfix der Fehlermeldung bei fehlgeschlagener Anmeldung mit diesem Plugin
- `user_data_mapping`: in diesem Array werden die Zuordnungen von LDAP Attributen zu Stud.IP Datenbankeinträgen vorgenommen. Diese Zuordnungen werden bei jeder erfolgreichen Anmeldung in der Stud.IP Datenbank aktualisiert. Der Schlüssel eines Eintrages ist das Stud.IP Datenbankfeld in der Form `<tabelle>.<feld>`. Zu jedem Schlüssel sind zwei weitere Einträge nötig. In "callback" wird der Name der Methode im Plugin angegeben, die die Zuordnung vornehmen soll, "map_args" enthält die Argumente, die an die Methode übergeben werden. Der Eintrag dummy als callback nimmt keine Zuordnung vor, verhindert aber, dass die entsprechenden Felder im Stud.IP vom Nutzer verändert werden können, man sollte die default Einträge für username und password besser nicht verändern. Im LDAP Plugin existieren zwei simple mapping Methoden: `doLdapMap` und `doLdapMapVorname`. Erstere erwartet als Argument den Namen des LDAP Attributes (z.B. `sn` für den Nachnamen) und gibt einfach den Inhalt zurück. Letztere nimmt zwei Argumente entgegen und versucht aus dem `cn` den Vornamen zu extrahieren (nur ein Beispiel). Benötigt man komplexere Mappings, so empfiehlt es sich ein eigenes Plugin von `AuthLdap` abzuleiten und die entsprechenden Methoden dort zu realisieren.

`LdapReader` authentifiziert nicht den Nutzer direkt gegen den LDAP Server, sondern benutzt einen konfigurierten LDAP Account mit Leserechten, um das Passwort des Nutzers aus dem Verzeichnis zu lesen und mit dem eingegebenen zu vergleichen. Anonymes Binden wird dazu nicht benötigt. Dieses Plugin ist als Beispiel gedacht, und sollte als Vorlage für eigene Plugins dienen.

`LdapReadAndBind` benutzt ebenfalls einen konfigurierten Account, um den user dn des Nutzers zu finden. Es wird aber nicht versucht das Passwort auszulesen, sondern zur Authentifizierung des Nutzers wird ein weiterer Anmeldevorgang mit dem ermittelten user dn und dem eingegebenen Passwort des Nutzers versucht.

Einstellungen:

- `user_password_attribute`: das Attribut des Nutzerpasswortes (Passwort muss als MD5 Hash vorliegen!)
- `reader_dn`: der Nutzer DN des Accounts, der zum Auslesen benutzt wird
- `reader_password`: das Passwort des Accounts, der zum Auslesen benutzt wird

Weitere Module

Weitere Module wie der Kalender, die Ressourcenverwaltung, das Wiki- oder die Vote- und Evaluations-Module benötigen keine spezielle Konfiguration oder Zusatzsoftware.