

Installationsanleitung für die Adaptiven Trainings

AP 1 – Selbstlernkompetenz und Motivation im Selbststudium

Zielgruppe: IT-Administratoren

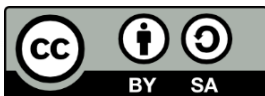


Autoren Gerhard Götz
Sebastian Wankerl

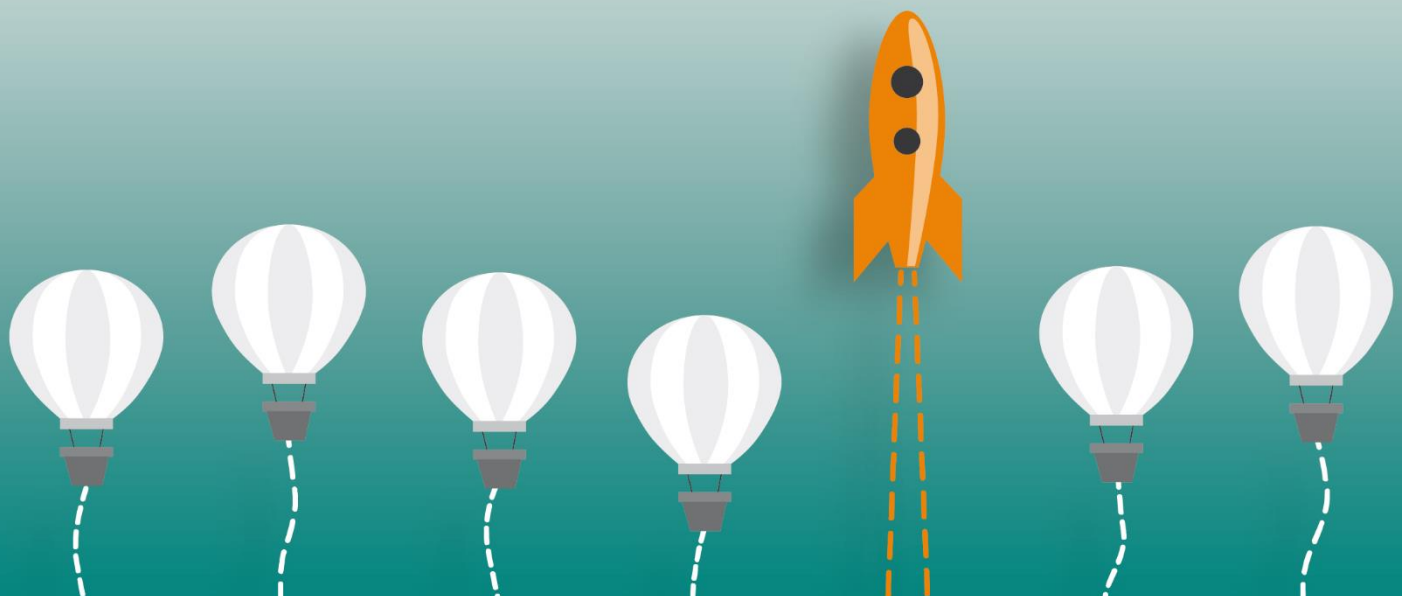
Verbundpartner Duale Hochschule Baden-Württemberg

Veröffentlichung 04.2023

Version 1



Dieses Werk wurde im Rahmen des Projekts DigikoS erstellt und steht unter der Lizenz [Creative Commons „Namensnennung – Weitergabe unter gleichen Bedingungen – 4.0 International“](https://creativecommons.org/licenses/by-sa/4.0/)





Inhalt

Installationsanleitung für die adaptiven Trainings.....	3
1. Technische Umsetzung.....	3
1.1. Allgemeines	3
1.2. Datenschutz	3
2. Schritte zur Installation	5
2.1. Vorbereitungen in der ILIAS-Zielplattform	5
2.1.1. Import der Fragenpools ins ILIAS-System	5
2.1.2. Installation des ILIAS-Plugins	5
2.2. Vorbereitungen auf der Maschine, die den Recommender ausführt.....	6
2.3. Vorbereitung der Python-Umgebung	6
2.4. Datenbank	6
2.4.1. Vorbereitung	6
2.4.2. Kopieren der Datenbank.....	7
2.5. Anpassung/ Konfiguration der Recommender-Software	7
2.6. Start/Test des Recommenders	7
2.7. Anpassen der HTML5 Files	7
2.8. Konfiguration eines Trainingsobjekts	7
3. Überführung in den produktiven Betrieb.....	8
4. Export der Daten.....	8



Installationsanleitung für die adaptiven Trainings

Dieses Dokument beschreibt die grundlegende technische Umsetzung sowie die Installation des im Rahmen von optes (www.optes.de) und DigikoS (www.digikos.de) entwickelten adaptiven Trainings. Der folgende Abschnitt beschreibt zunächst die technische Umsetzung der Software. Im Anschluss daran wird die Installation erklärt.

1. Technische Umsetzung

1.1. Allgemeines

Als Basis für das den Trainings zugrundeliegende Empfehlungssystem dient ein Server mit moderner Linux-Distribution (z.B. SLES, Ubuntu) und einer Installation von ILIAS 6.x oder höher. Die Software gliedert sich technisch in zwei Komponenten: ein ILIAS-Plugin (Plugin) sowie eine serverseitige Anwendung (Recommender).

Der Recommender entscheidet hierbei, welche Aufgaben einem Studierenden gezeigt werden sollen, indem er individuell auf Basis der bisher bearbeiteten Aufgaben sowie gegebenen Antworten eine neue, passende Aufgabe bestimmt.

Das Plugin hingegen fungiert als Interface, d.h. sein Zweck besteht darin, die vom Recommender vorgegebene Übungsaufgabe anzuzeigen und nach der Bearbeitung die eingegebene Antwort an den Recommender zurückzusenden. Die Auswertung, ob die Aufgabe korrekt bearbeitet wurde, erfolgt im Plugin. Im Anschluss daran zeigt das Plugin die Musterlösung der Aufgabe an und fragt ihren subjektiven Schwierigkeitsgrad ab, welcher wiederum an den Recommender zurückgegeben wird.

Diese beiden Schritte können beliebig oft wiederholt werden. Es steht den Studierenden frei, das Training jederzeit zu unterbrechen und zu einem späteren Zeitpunkt fortzusetzen. Nach einer gewissen Anzahl an bearbeiteten Aufgaben - diese hängt neben einer Festlegung zu Beginn (ca. 25 Aufgaben) von der durchschnittlichen Korrektheit der Bearbeitungen ab - erhält der Nutzer eine ausführliche Rückmeldung über die erreichten Ziele. Im Anschluss daran kann nach Bedarf ein weiteres Training begonnen werden.

Der Recommender ist in Python geschrieben und benötigt eine eigene Datenbank, da er sich die Historie der bearbeiteten Aufgaben merken muss. Die Kommunikation mit dem Plugin erfolgt webbasiert über eine REST-Schnittstelle, allerdings ausschließlich vom ILIAS-Server aus. Direkte Datentransfers zwischen dem Browser des Users und dem Recommender finden daher nicht statt. Der Recommender und die ILIAS-Installation müssen folglich nicht auf derselben physikalischen Maschine ausgeführt werden.

1.2. Datenschutz

Bei der Entwicklung der Software wurde großer Wert auf den Datenschutz gelegt. Es werden keine Informationen übertragen, die nicht notwendig sind. Die nachfolgende Tabelle zeigt alle bearbeitungsspezifischen Attribute, die zwischen dem Recommender und dem Plugin übertragen werden können.



Die Daten werden bereits auf Seite des Plugins vollständig pseudonymisiert, sodass auf Seite des Recommenders keine Möglichkeit mehr besteht, den ursprünglichen Nutzer zu identifizieren. Um dies zu gewährleisten, wird auf die ILIAS-interne Nutzer-ID zunächst ein Salt (das ist eine beliebige, vom ILIAS-Administrator festgelegte Ganzzahl) addiert und anschließend der Hash dieser Summe berechnet. Dieser Prozess ist aufgrund der Hashfunktion nicht umkehrbar. Auch können die IDs nicht mit Hilfe von bruteforce berechnet werden, da sie durch das Salt, welches nur dem Administrator der ILIAS-Installation bekannte ist, bereits verändert wurden. Es kann ferner davon ausgegangen werden, dass diejenigen, die die Daten des Recommenders auswerten, keinerlei Kenntnisse über die interne Nutzerdatenbank der ILIAS-Installation haben, auf der die Daten gesammelt wurden.

Attribut	Beschreibung	Beispielwert
User ID	Siehe Text oben: md5sum(UserID+Salt)	13f9896df61279c928f19721878fac41
Question ID	ID der bearbeiteten Frage	Aufgabe 3000
Korrektheit	Gibt an, ob die Frage korrekt (1) oder falsch (0) bearbeitet wurde	1
Antworttext	Text, der als Antwort angegeben wurde, bzw. Text der gewählten Antwort bei Single-Choice-Fragen	y=x+1
Bearbeitungszeit	Zeit in Sekunden, die für die Bearbeitung der Frage benötigt wurde	155
Run	Gibt an, um den wievielten Durchlauf des Trainings es sich handelt	1
Rating	Subjektiver Schwierigkeitsgrad der bearbeiteten Aufgabe (1=sehr einfach, 4=sehr schwierig)	2
Feedbackzeit	Zeit in Sekunden, die mit dem Durchsehen der Musterlösung sowie dem Auswählen des Schwierigkeitsgrads verbracht wurden	35



Trainingsabschluss	Falls das Training abgeschlossen wurde, gibt dieses Attribut an, ob es erfolgreich (1) oder nicht erfolgreich (2) bearbeitet wurde	1
---------------------------	--	---

Ferner werden statische Attribute der Aufgabe übertragen, welche jedoch nicht von der Antwort der Nutzer abhängen. Dabei handelt es sich um den zugehörigen Lernzielorientierten Kurs (LoK) sowie die Zugehörigkeit innerhalb des Referenzmodells (fachdidaktische Modelle, die die Aufgaben verschiedenen Aspekten des Wissens und Könnens zuordnen) und der Lernziele.

2. Schritte zur Installation

Die Installation des Recommenders und die Integration in die Zielplattform erfordern mehrere Schritte. Die Reihenfolge ist zwingend einzuhalten.

Alle benötigten Downloaddateien finden Sie unter https://www.digikos.de/goto_digikos_cat_312.html (kostenloser Benutzerzugang nötig).

2.1. Vorbereitungen in der ILIAS-Zielplattform

2.1.1. Import der Fragenpools ins ILIAS-System

Das Plugin funktioniert nur mit einem Fragenpool, in dem die Beschreibung der enthaltenen Fragen auf „_[[ID]]“ endet. Bei dieser ID handelt es sich um eine 20-stellige Hexadezimalzahl. Die entsprechend vorbereiteten Fragenpools können ebenfalls dem Downloadpaket entnommen werden und müssen wie gewohnt auf der ILIAS-Plattform importiert werden. Es gibt insgesamt sechs Fragenpools, die zur Verfügung stehen. Es können beliebig viele importiert werden, um die gewünschten Themengebiete der Mathematik abzudecken. Die verwendeten Pools sollten nicht für User sichtbar sein und können offline geschaltet bleiben (Diese Fragenpools sind nicht dafür gedacht, die Fragen in Tests bereitzustellen, sondern dienen einzig der Bereitstellung der Fragen für das Plugin Adaptive Trainings. Daher können sie offline bleiben. Eine Nutzung der Fragen in Tests ist jedoch nach einer online-Schaltung der Fragenpools selbstverständlich ebenfalls möglich).

2.1.2. Installation des ILIAS-Plugins

Ausgehend vom root-Verzeichnis der ILIAS-Installation müssen die folgenden Befehle ausgeführt werden:

```
mkdir -p Customizing/global/plugins/Services/Repository/RepositoryObject
```

```
cd Customizing/global/plugins/Services/Repository/RepositoryObject
```

```
git clone https://github.com/studer-raimann/DhbwTraining.git
```

Als ILIAS-Administrator muss unter *Administration* → *ILIAS erweitern* → *Plugins* das Plugin installiert und aktiviert werden.



2.2. Vorbereitungen auf der Maschine, die den Recommender ausführt

- a) Auf dieser Maschine (zukünftig Server genannt) muss Folgendes installiert sein:
 - Python3
 - Pip3 (das Paket heißt manchmal auch python3-pip)
 - gcc (zypper install gcc und zypper install gcc-devel)
 - Python3-developer- Libraries (zypper install python3-devel)
 - Mysql (oder MariaDB) als Server
- b) Für das Kommando mysql_config müssen (distributionsabhängig) die mysql-developer libraries (SLES12: zypper install libmysqlclient-devel) installiert werden
- c) Für den Recommender muss ein User mit eigenem Homedirectory und Passwort angelegt werden (SLES12: useradd -m recommender und anschließend passwd recommender)
- d) Im Homeverzeichnis dieses Users (oder einem Unterverzeichnis davon) muss der zum Betrieb des Recommenders erforderliche Datenbank-Dump und das Archiv mit den Recommender-Quellen bereitgestellt werden.

2.3. Vorbereitung der Python-Umgebung

Unter der Kennung dieses Users (siehe Punkt c) müssen weitere Python-Module mithilfe des folgenden Befehls installiert werden:

```
pip3 install --user modulname
```

Für *modulname* muss dann nacheinander Folgendes eingesetzt werden:

- SQLAlchemy
- Pymysql
- Flask
- Flask-RESTful
- Flask-SQLAlchemy
- gunicorn
- numpy

2.4. Datenbank

2.4.1. Vorbereitung

Mit dem mysql root User müssen nun eine Datenbank und ein Datenbankuser angelegt werden. Die Namen der Datenbank und des Datenbankusers sind derzeit in der Recommender-Software kodiert und müssen exakt so verwendet werden. Es können alternativ auch die Python-Quellen angepasst werden.

```
create user 'recomm_user'@'localhost' identified by '<Datenbankpasswort für
recomm_user>';
```

```
create database recommender_db_v2 character set utf8mb4 collate utf8mb4_unicode_ci;
```

```
grant all on recommender_db_v2.* to 'recomm_user'@'localhost';
```

```
flush privileges;
```



2.4.2. Kopieren der Datenbank

Mit dem Datenbankuser `recomm_user` werden die nötigen Tabellen importiert:

```
mysql -u recomm_user -p recommender_db_v2 < dateiname_der_import_db
```

Bitte überprüfen Sie anschließend, ob sich in der Tabelle `questions` genauso viele Eintragungen befinden, wie es Fragen im Pool gibt (`select count(*) from questions;`).

2.5. Anpassung/ Konfiguration der Recommender-Software

Das Recommender-Archiv muss entpackt werden. Anschließend muss in der Datei `recommender@.service` folgendes angepasst werden:

- `DB_PASSWORD`=Passwort des Datenbanknutzers
- `WorkingDirectory`=das Verzeichnis, in dem der Code des Recommenders liegt

2.6. Start/Test des Recommenders

Die Instanzen des Recommenders werden im Anschluss wie folgt gestartet:

```
systemctl start recommender@{1..6}.service
```

```
systemctl enable recommender@{1..6}.service
```

Hierbei werden die Ports 5001-5006 verwendet. Ob der Recommender läuft, kann z.B. auf folgende Weise überprüft werden:

```
wget http://ip_des_recommenders:PORT
```

Ein Status 404 zeigt hier bereits, dass der Recommender erreichbar ist. Falls die Verbindung nicht zustande kommt, prüfen Sie bitte sowohl die IP-Adressen als auch die Firewall.

2.7. Anpassen der HTML5 Files

Den Aufruf von `gstatic` in den folgenden Dateien anpassen:

- `geo_kre_gen_110.html`
- `geo_fla_pyt_104.html`
- `geo_kre_gen_112.html.sec`
- `geo_ste_pyr_102.html`
- `gle_ung_gen_107_fahrrad.html`
- `0402_linse.html`
- `geo_fla_dre_rec_102.html`
- `geo_fla_tra_104.html`

2.8. Konfiguration eines Trainingsobjekts

Erstellen Sie zunächst in der ILIAS-Plattform im Inhaltsreiter eines LoKs ein neues Trainingsobjekt.

Dort können in der Konfiguration für `apikey` und `secret` beliebige Werte eingetragen werden.



Als URL muss `http://ip_oder_hostname_des_Recommenders:PORT_LOK/` eingetragen werden, wobei `PORT_LOK` dem Port entspricht, der bei obiger Konfiguration (Kapitel 2.6) für den jeweiligen LoK gesetzt wurde (5001 für Arithmetik, usw.). Die URL muss hierbei mit einem `..//` enden.

3. Überführung in den produktiven Betrieb

Nach Abschluss der internen Tests sollten die im Installationsprozess gesammelten Nutzerinteraktionen wieder aus der Datenbank gelöscht werden. Das bedeutet, dass alle Daten aus den folgenden Tabellen der `recommender_db_v2` gelöscht werden müssen:

- `delete from user_interactions;`
- `delete from user_ratings;`
- `delete from user_finished;`
- `delete from user_skills;`

Um im Anschluss ein Verfälschen der Daten zu vermeiden, sollte der Recommender nur noch von Studierenden genutzt werden, da die gesammelten Daten nicht von denen der Administrator:innen unterschieden werden können.

4. Export der Daten

Um die Daten zu exportieren, müssen die Tabellen `user_interactions`, `user_ratings`, `user_finished` und `user_skills` aus der Recommender-Datenbank exportiert werden. Sie enthalten die nutzerspezifischen und pseudonymisierten Informationen, die Sie der Tabelle auf Seite 4 entnehmen können.