
IZIS - Ilias Zusatzmodule der
Informatik an der Uni Stuttgart

Administration Container Object (ACO) Dokumentation

Daten zu Studenten und Betreuern:
Manuel Mergl, Kai Durst, Philipp Gruber

Betreut durch das FMI, Abteilung Algorithmik, Uni Stuttgart

SS16-WS16/17

Inhaltsverzeichnis

1	Einleitung	3
2	Gruppen verwalten	4
2.1	Gruppen verwalten	5
2.2	Gruppen erstellen	6
2.3	Kurs bearbeiten	8
2.4	Mitglieder verschieben	10
3	Verlinkung von Tests	12
3.1	Verlinkung von Tests	13
4	Verlinkung von Übungen	15
4.1	Verlinkung von Übungen	16
5	Gruppenfilter	18
5.1	Gruppenfilter	19
6	Klassenbeschreibungen	21
6.1	ilACOEExerciseMemberTableGUI	24
6.2	ilACOGroupDisplayGUI	27
6.3	ilACOGroupGUI	33
6.4	ilACOLinkGUI	38
6.5	ilACOMemberGUI	44
6.6	ilACOPlugin	52
6.7	ilACOTutorGUI	53
6.8	ilACOUIHookGUI	61
7	Tests	62

1 Einleitung

Dieses Dokument dient dem Zweck, die Funktionen und die Bedienung unseres Plugins ACO (Admin Container Objekt) zu dokumentieren und zu beschreiben.

Motivation

Damit sollte es den Benutzern dieses Plugin möglich sein, den Umfang unseres Plugins einschätzen zu können und bei möglichen Fragen bezüglich der Bedienung zu helfen.

Aufbau des Dokumentes

Das Dokument gliedert sich in 5 Abschnitte:

Diese Abschnitte richten sich nach den einzelnen Tabs, die durch unser Plugin dem Ilias hinzugefügt werden.

Im ersten Abschnitt wird der übergeordnete Reiter/Tab *Gruppen verwalten* behandelt.

Der zweite Abschnitt befasst sich mit dem übergeordnete Reiter/Tab *Verlinkung* eines Test bzw. einer Übung.

Im dritten Abschnitt wird der übergeordnete Reiter/Tab *Gruppenfilter* näher beschrieben.

Im vierten Abschnitt beschreiben wir unser System bezüglich seiner Komponenten bzw. Klassen.

Im fünften Abschnitt listen wir tabellarische unsere Testfälle auf.

2 Gruppen verwalten

Contents

2.1	Gruppen verwalten	5
2.2	Gruppen erstellen	6
2.3	Kurs bearbeiten	8
2.4	Mitglieder verschieben	10

2.1 Gruppen verwalten

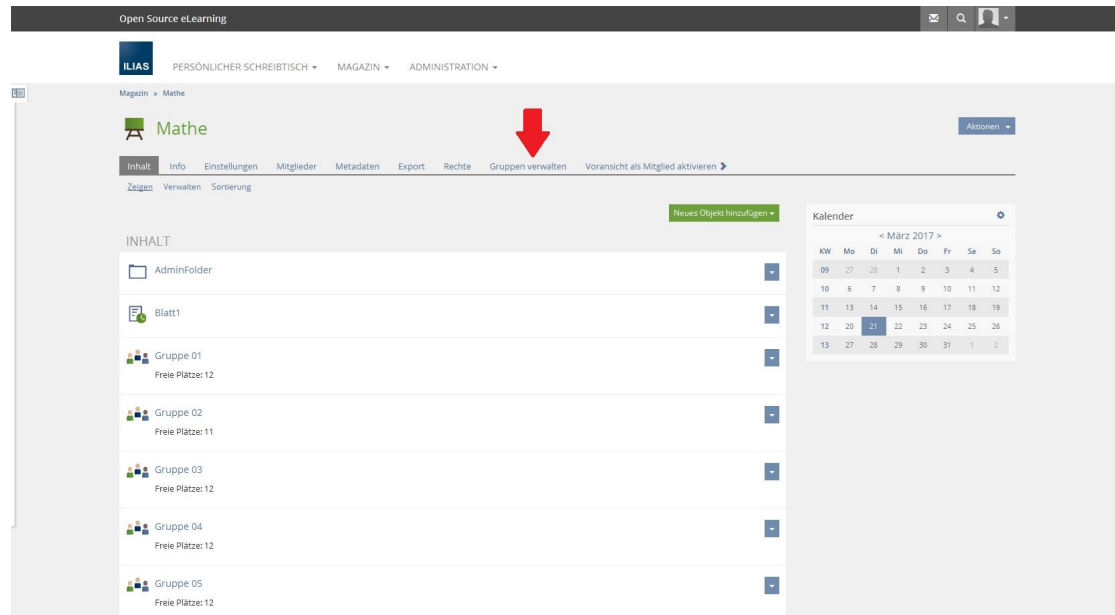


Abbildung 2.1: Ansicht Tab Gruppen verwalten im jeweiligen Kurs

In einem Kurs erscheint für alle Benutzer mit Schreibzugriff der neue Tab *Gruppen verwalten*. In diesem Tab stehen drei Subtabs zur Auswahl.

- *Gruppen erstellen* bietet die Funktionalität, mehrere Gruppen auf einmal zu erstellen. Dabei sind verschiedene Pflichtfelder und optionale Felder verfügbar.
- *Kurs bearbeiten* bietet die Möglichkeit, Einstellungen mehrerer Gruppen auf einer Seite übersichtlich zu bearbeiten.
- *Mitglieder verschieben* bietet die Möglichkeit, Mitglieder zwischen Gruppen hin- und herzuschieben. Voraussetzung dafür ist, dass der Benutzer bereits Mitglied in einer Gruppe ist. Um Tutoren zu verschieben, benutzen Sie bitte den Tab *Kurs bearbeiten*. Tutoren bleiben der bisherigen Gruppe als Administrator erhalten, und werden der Ziel-Gruppe als Mitglied hinzugefügt.

2.2 Gruppen erstellen

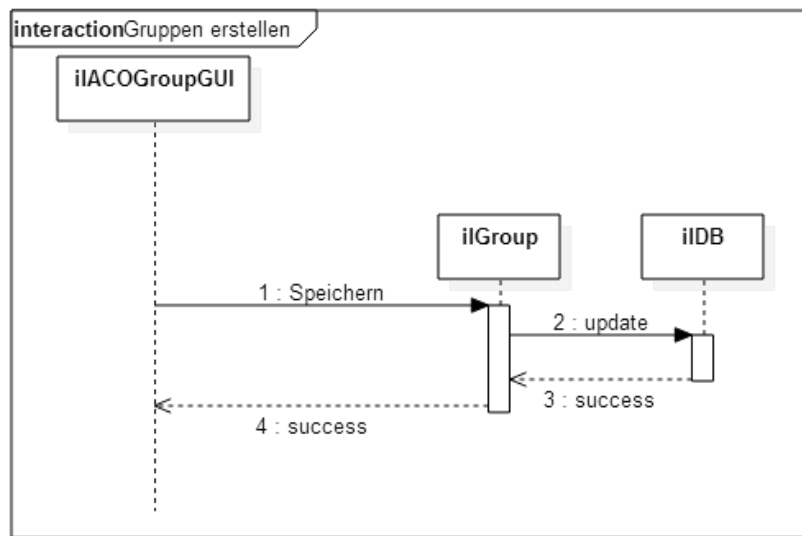


Abbildung 2.2: Sequenzdiagramm Gruppen erstellen

Open Source eLearning

ILIAS PERSÖNLICHER SCHREIBTISCH MAGAZIN ADMINISTRATION

Magazin » Kategorie » Kurs

AdminContainer Plugin
Hier können sie einfach Gruppen erstellen und bearbeiten.

◀ Zurück Gruppen verwalten

Gruppen erstellen Kurs bearbeiten Mitglieder verschieben

GRUPPEN ERSTELLEN Erstellen

Gruppenname *

Anzahl der Gruppen *

Maximale Mitgliederanzahl *

Typ ☒ Direkter Gruppenbeitritt ☐ Gruppenbeitritt mit Passwort

Zeitlich begrenzter Beitritt ☐

Ordnerstruktur für Aufgabenverteilung ☐

* Erforderliche Angabe Erstellen

Abbildung 2.3: Ansicht Gruppen erstellen

In diesem Reiter/Tab lassen sich beliebig viele Gruppen erstellen.

Vorgehen - Pflichtfelder

- Eingeben eines Namenspräfix im Textfeld *Gruppennamen*
- Eingeben einer Zahl größer Null im Textfeld *Anzahl der Gruppen*
- Eingeben einer Zahl im Textfeld *Maximale Mitgliederanzahl* für die maximale Anzahl der Gruppenmitglieder

(Versehentlich eingegebene Buchstaben werden bei Zahleneingaben ignoriert)

Vorgehen - optional

Beitritts - Typ

- 1 Radiobutton auf *Gruppenbeitritt mit Passwort* setzen, um die Gruppen mit einem Passwort zu schützen.
- 2 Ein weiteres Textfeld taucht auf, bei dem das gewünschte Passwort einzugeben ist. (Passwort ist für alle Gruppen dasselbe)

Zeitlich begrenzter Beitritt

- 1 Checkbox setzen
- 2 Zwei Datumseingabefelder tauchen für den Anmelde-Zeitraum auf, sowohl für Beitrittsanfang, als auch für Beitrittsende einen Zeitpunkt auswählen. Format: Tag / Monat / Jahr / Stunde / Minuten (Achtung, Ende kann nicht vor Start liegen)

Ordnerstruktur für Aufgabenverteilung

- 1 Checkbox setzen
- 2 Ein Textfeld taucht auf, bei dem man einen oder mehrere Ordnernamen (durch ";"getrennt) eingeben kann. Dadurch werden im Kurs, als auch in allen zu erstellenden Gruppen, diese Ordner mit den bzw. dem Namen erzeugt. Dies dient der vereinfachten Verteilung von Übungen bzw. Tests. (siehe auch Verlinkung von Übungen und Verlinkung von Tests)

2.3 Kurs bearbeiten

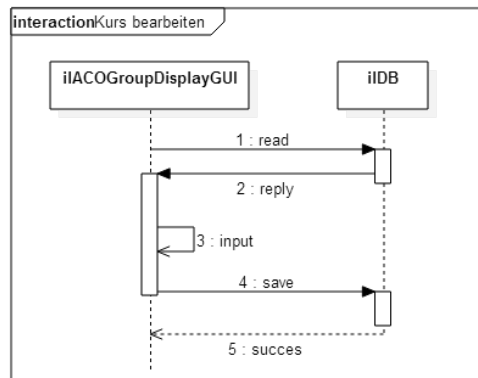


Abbildung 2.4: Sequenzdiagramm Kurs bearbeiten

Open Source eLearning

ILIAS PERSONLICHER SCHREIBTISCH MAGAZIN ADMINISTRATION

Magazin » Kategorie » Kurs

AdminContainer Plugin
Hier können sie einfach Gruppen erstellen und bearbeiten.

← Zurück Gruppen verwalten

Gruppen erstellen Kurs bearbeiten Mitglieder verschieben

KURS BEARBEITEN Änderungen speichern

GRUPPE 01

Gruppenname: Gruppe 01

Raum und Uhrzeit:

Tutor: root

Max. Mitglieder: 2

Zeitlich begrenzter Beitritt: ☒

Anmelde-Zeitraum: Anfang: DD.MM.YYYY HH:mm Ende: DD.MM.YYYY HH:mm

GRUPPE 02

Gruppenname: Gruppe 02

Raum und Uhrzeit:

Tutor: root

Max. Mitglieder: 2

Zeitlich begrenzter Beitritt: ☐

Abbildung 2.5: Ansicht Kurs bearbeiten

In diesem Reiter/Tab erhält man eine Übersicht aller im Kurs vorhandenen Gruppen mit folgenden Feldern.

Gruppen

Die Felder können einzeln und unabhängig voneinander bearbeitet werden. Felder zeigen den aktuellen Wert/Zustand der Gruppe an. Mit einem Klick auf den *Änderung speichern* Knopf werden die Änderungen übernommen.

Gruppenname

- Auf das Feld klicken, um den Namen verändern, dies beinhaltet den Suffix, dessen Veränderung unter Umständen die Nummerierung durcheinander bringt. (Feld darf nicht leer sein)

Raum und Uhrzeit - als Beschreibung

- In das Textfeld lässt sich eine beliebige Zeichenkette eintragen (z.B. Raum und Uhrzeit), die dann als Beschreibung der Gruppe auftaucht.

Tutor

- Um einem Benutzer die Tutor bzw. Gruppenadministrationsrechte zuzuweisen bzw. zu entziehen.

hinzufügen Um den Gruppenadministrator zu ändern, gibt man hier den Benutzernamen des neuen Administrators ein. Eine Auto-Complete Funktion hilft beim vervollständigen der Benutzernamen. Dem alten eingetragenen Benutzer werden damit die Adminrechte in der Gruppe entzogen, außer es handelt sich um einen Kursadministrator, dann wird der neue Administrator zusätzlich angelegt.

Max Mitglieder

Um den angezeigten Wert für die Gruppe zu verändern

- Eingeben einer Zahl im Textfeld *Maximale Mitgliederanzahl* für die maximale Anzahl der Gruppenmitglieder

Zeitlich begrenzter Beitritt

Wenn Checkbox nicht gesetzt, gibt es keinen zeitlich begrenzten Beitritt bei dieser Gruppe. Um einen zu erzeugen

- 1 Checkbox setzen
- 2 Zwei Datumseingabefelder tauchen für den Anmelde-Zeitraum auf, sowohl für Beitrittsanfang, als auch für Beitrittsende einen Zeitpunkt auswählen. Format: Tag/Monat/Jahr/Stunde/Minute (Achtung, Ende kann nicht vor Start liegen)

Ansonsten Checkbox deaktivieren und begrenzter Beitritt verschwindet. Falls schon ausgewählt und man nur das Datum verändern will, Schritt 1 überspringen.

2.4 Mitglieder verschieben

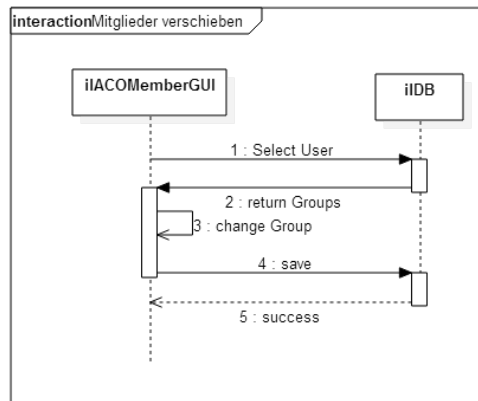


Abbildung 2.6: Sequenzdiagramm Mitglieder verschieben

Open Source eLearning

ILIAS PERSÖNLICHER SCHREIBTISCH MAGAZIN ADMINISTRATION

Magazin > Kategorie > Kurs

AdminContainer Plugin
Hier können sie einfach Gruppen erstellen und bearbeiten.

[Zurück](#) **Gruppen verwalten**

[Gruppen erstellen](#) [Kurs bearbeiten](#) [Mitglieder verschieben](#)

MITGLIEDER VERSCHIEBEN
Geben Sie hier den Benutzernamen des Benutzers sowie den Namen der Gruppe in die der Benutzer verschoben werden soll an.

Benutzer * Student1

* Erforderliche Angabe [Benutzer auswählen](#)

(a) Benutzer auswählen

Open Source eLearning

ILIAS PERSÖNLICHER SCHREIBTISCH MAGAZIN ADMINISTRATION

Magazin > Kategorie > Kurs

AdminContainer Plugin
Hier können sie einfach Gruppen erstellen und bearbeiten.

[Zurück](#) **Gruppen verwalten**

[Gruppen erstellen](#) [Kurs bearbeiten](#) [Mitglieder verschieben](#)

MITGLIEDER VERSCHIEBEN
Geben Sie hier den Benutzernamen des Benutzers sowie den Namen der Gruppe in die der Benutzer verschoben werden soll an.

Benutzername * Student1

bisherige Gruppe * Gruppe 01

neue Gruppe * Gruppe 02

* Erforderliche Angabe [Verschieben](#)

(b) Gruppen auswählen

Abbildung 2.7: Ansicht Mitglieder verschieben

Mitglieder verschieben

Benutzer auswählen

- Benutzernamen eingeben, ab dem dritten Buchstaben hilft die Auto-Complete Funktion beim Vervollständigen. (Pflichtfeld, darf nicht leer sein)
- Auf *Benutzer auswählen* klicken

Benutzer wurde ausgewählt

- Auf der neuen Seite wird in dem Drop-down Menü unter *bisherige Gruppe* angezeigt, in welchen Gruppen sich der Benutzer befindet, darunter unter *neue Gruppe*, die möglichen Gruppen zum verschieben.
- Auswählen, aus welcher Gruppe (Achtung im Drop-down von bisherige Gruppe Menü können mehrere Gruppen angezeigt werden, da sich Benutzer möglicherweise in mehreren Gruppen befindet) der Benutzer verschoben werden soll.
- Auswählen, in welche *neue Gruppe* man den Benutzer verschieben will
- Mit dem Klicken auf *Verschieben* die Eingabe bestätigen und den Benutzer verschieben

3 Verlinkung von Tests

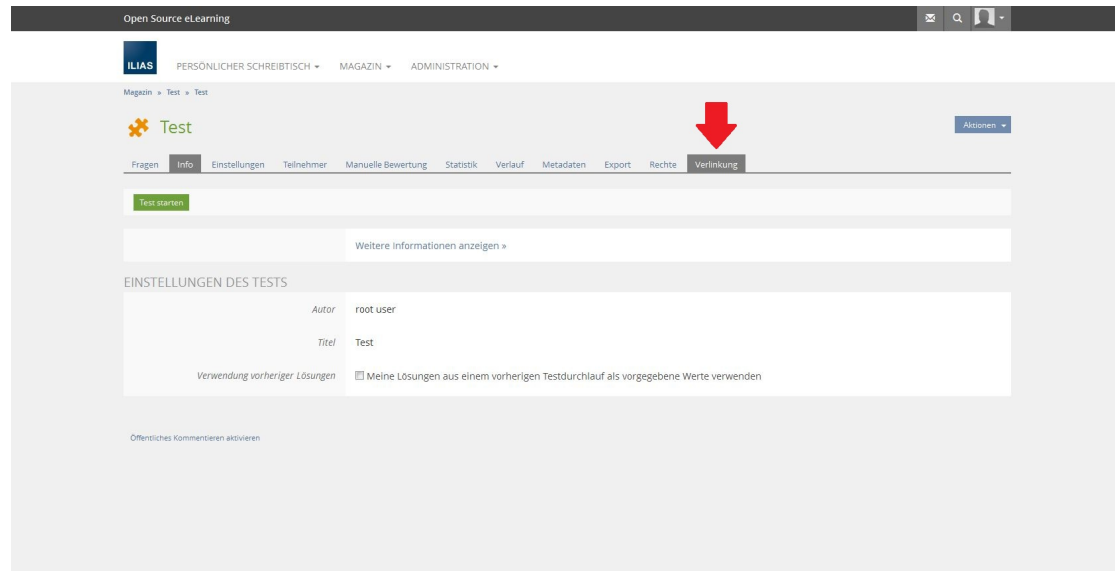


Abbildung 3.1: Ansicht Verlinkung Tab in Test

In einem erstellten Test erscheint für alle Benutzer mit Schreibrechten der neue Tab *Verlinken*. Die Funktionalität ist identisch mit der ILIAS-Standard Funktionalität *Verlinken*, die aber nicht direkt aus einem Testobjekt verfügbar ist. Es bestehen zwei Auswahlmöglichkeiten einen Test zu verlinken. *In alle Gruppen verlinken* verlinkt das Testobjekt in alle dem Kurs enthaltenen Gruppen. Dabei wird versucht, das verlinkte Objekt in einem Ordner in jeder Gruppe zu speichern, der den selben Namen trägt, wie der dem Testobjekt übergeordnete Ordner. Liegt das Objekt nicht in einem Ordner, oder enthalten die Gruppen keinen solchen Ordner, wird das Testobjekt direkt in die Gruppen verlinkt.

3.1 Verlinkung von Tests

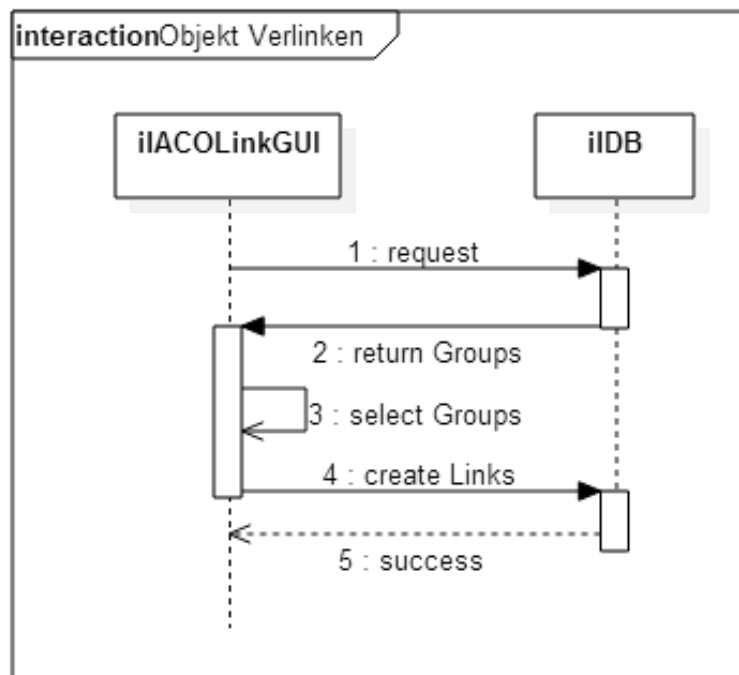


Abbildung 3.2: Sequenzdiagramm Test verlinken

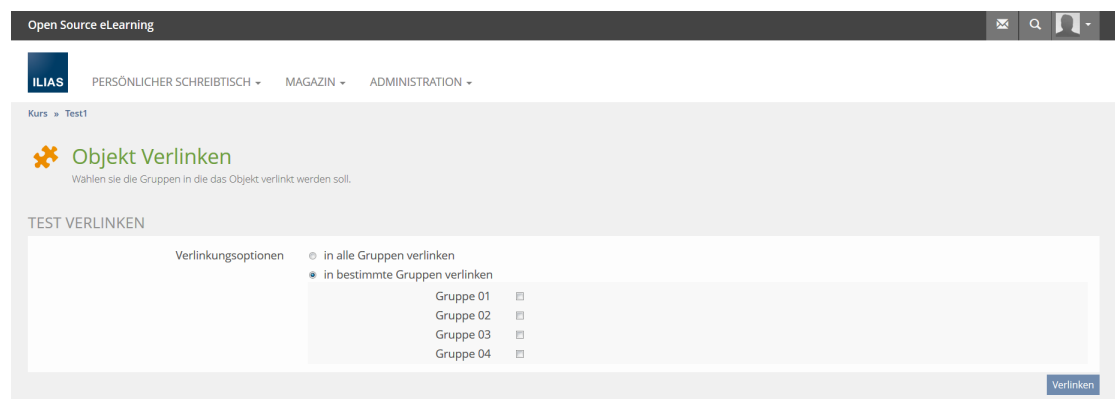


Abbildung 3.3: Ansicht Link Test

Über einen Test gelangen wir über den Tab Verlinkung zu dem dargestellten Bild. Mit der Funktion lässt sich dieser Test in die Gruppen des Kurses, in dem die Übung existiert verlinken/verknüpfen.

in alle Gruppen verlinken

- Sollte dieser Radio Button ausgewählt und auf verlinken geklickt werden, so wird diese Übung in alle im Kurs vorhandenen Gruppen verlinkt.

in bestimmte Gruppen verlinken

- Sollte dieser Radio Button ausgewählt werden, erscheint eine Auswahl aller im Kurs vorhandenen Gruppen mit einer Checkbox
- Man kann jetzt einzelne Gruppen via der Checkbox markieren
- Klickt man auf *Verlinken*, werden diese Gruppen verlinkt

(Achtung: Duplikate werden von diesem Tab ausgeschlossen bzw. es wird nicht doppelt verlinkt, sollte der Test schon in diese Gruppe verlinkt worden sein)

4 Verlinkung von Übungen

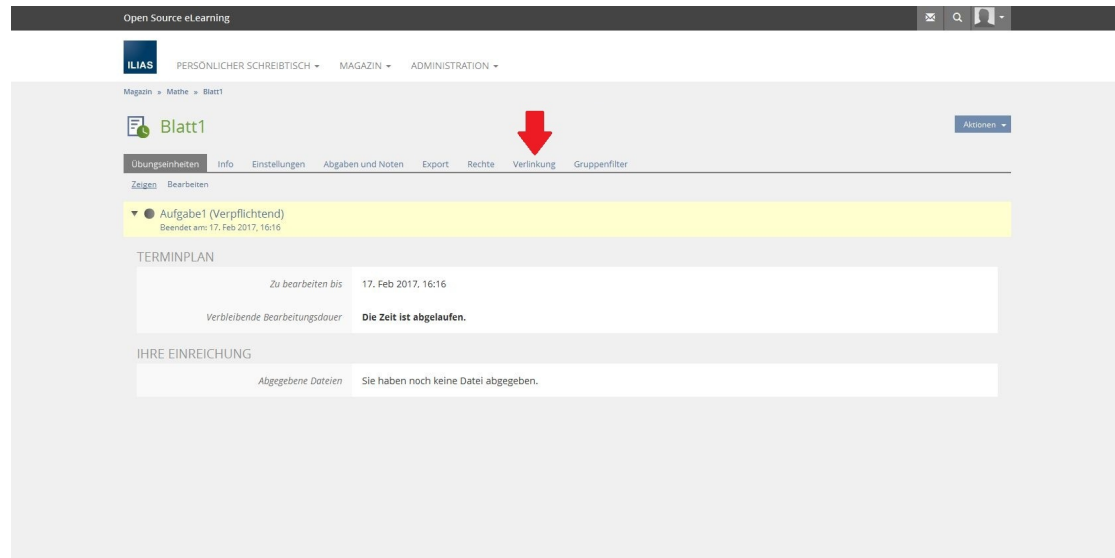


Abbildung 4.1: Ansicht Verlinkung in Exercise/Übung

In einer erstellten Übung erscheint für alle Benutzer mit Schreibrechten der neue Tab *Verlinken*. Die Funktionalität ist identisch mit der ILIAS-Standard Funktionalität *Verlinken*, die aber nicht direkt aus einem Übungsobjekt verfügbar ist. Es bestehen zwei Auswahlmöglichkeiten eine Übung zu verlinken. *In alle Gruppen verlinken* verlinkt das Übungsobjekt in alle dem Kurs enthaltenen Gruppen. Dabei wird versucht, das verlinkte Objekt in einem Ordner in jeder Gruppe zu speichern, der den selben Namen trägt, wie der dem Übungsobjekt übergeordnete Ordner. Liegt das Objekt nicht in einem Ordner, oder enthalten die Gruppen keinen solchen Ordner, wird das Übungsobjekt direkt in die Gruppen verlinkt.

4.1 Verlinkung von Übungen

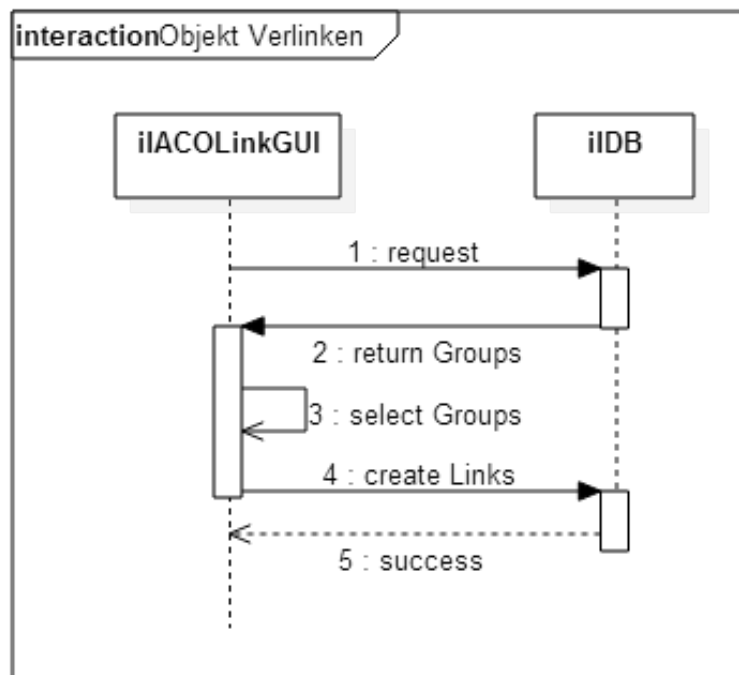


Abbildung 4.2: Sequenzdiagramm Übung verlinken

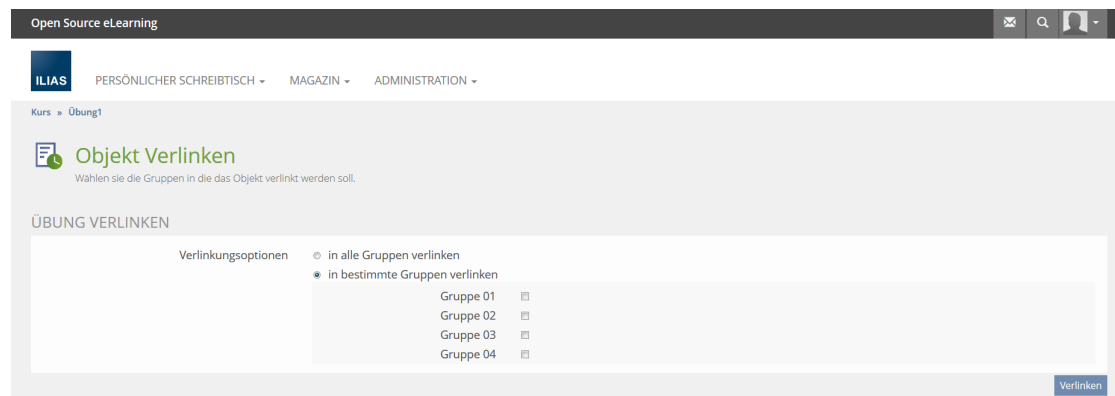


Abbildung 4.3: Ansicht Link Übung

Über eine Übung gelangen wir über den Tab Verlinkung zu dem dargestellten Bild. Mit der Funktion lässt sich diese Übung in die Gruppen des Kurses, in dem die Übung existiert verlinken/verknüpfen.

in alle Gruppen verlinken

- Sollte dieser Radio Button ausgewählt und auf verlinken geklickt werden, so wird diese Übung in alle im Kurs vorhandenen Gruppen verlinkt.

in bestimmte Gruppen verlinken

- Sollte dieser Radio Button ausgewählt werden, erscheint eine Auswahl aller im Kurs vorhandenen Gruppen mit einer Checkbox
- Man kann jetzt einzelne Gruppen via der Checkbox markieren
- Klickt man auf *Verlinken*, werden diese Gruppen verlinkt

(Achtung: Duplikate werden von diesem Tab ausgeschlossen bzw. es wird nicht doppelt verlinkt, sollte die Übung schon in diese Gruppe verlinkt worden sein)

5 Gruppenfilter

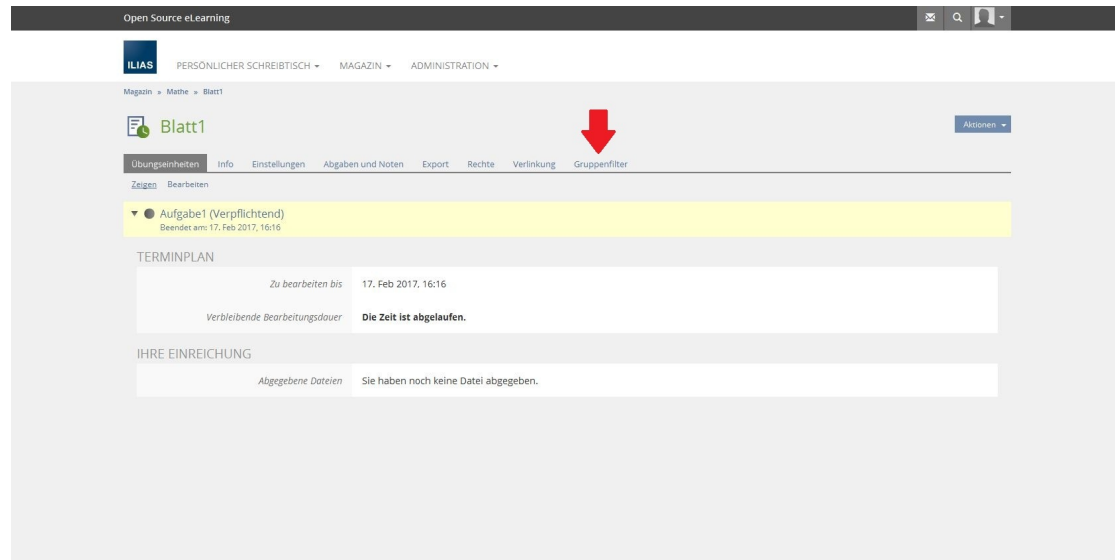


Abbildung 5.1: Ansicht Gruppenfilter Tab

In einer erstellten Übung erscheint für alle Benutzer mit Schreibrechten der neue Tab *Gruppenfilter*. Die Darstellung ist an den ILIAS-Standard Tab *Abgaben und Noten* angelehnt. Allerdings sind etwas weniger Funktionen verfügbar. Das Besondere an diesem Tab ist, dass man alle Benutzer die Abgaben getätigt haben, jetzt nach deren Gruppenzugehörigkeit filtern kann.

5.1 Gruppenfilter

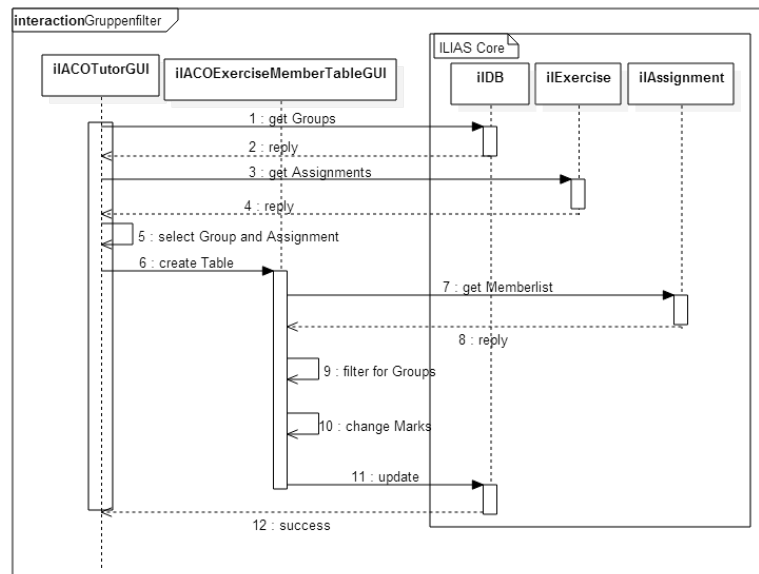


Abbildung 5.2: Sequenzdiagramm Gruppenfilter

ILIAS PERSÖNLICHER SCHREIBTISCH MAGAZIN ADMINISTRATION

Kurs » Exercise1

Abgaben und Noten

Hier können sie Abgaben nach Gruppen und Übung filtern, sowie Punkte eintragen und Abgaben herunterladen.

[Zurück](#)

Group1 Assignment1 -Gruppe und Übungseinheit anzeigen Alle abgegebenen Dateien herunterladen

ÜBUNGSEINHEIT: ASSIGNMENT1 (1 - 4 von 4)

Als Mail versenden Ausführen Alle Speichern und Gesamtnote aktualisieren

Bild	Name ↓	Benutzername	Einreichung	Bewertung	Feedback
<input type="checkbox"/>	user, root	[root]	Datum der letzten Abgabe: 16. Mär 2017, 13:17 Abgegebene Dateien: 1 Abgaben herunterladen Notiz für Tutoren: <input type="text"/>	Nicht bestanden Note: 4 Letzte Änderung: 13. Mär 2017, 15:27	commentss Als Kommentar in Textfeld eingeben Als Mail versenden Versandt am 13. Mär 2017, 12:38
<input type="checkbox"/>	3. User	[user3]	Datum der letzten Abgabe: --- Abgegebene Dateien: 0 Notiz für Tutoren: <input type="text"/>	Nicht bewertet Note: <input type="text"/>	Als Kommentar in Textfeld eingeben Als Mail versenden

Abbildung 5.3: Ansicht Gruppenfilter Tabelle

Innerhalb von Exercises bzw. Übungen erscheint dieser Reiter/Tab, der es ermöglicht Abgaben nach Gruppen zu filtern.

Gruppen filtern

- Man wählt in einer Übung den Reiter *Gruppenfilter* aus.
- In der Toolbar wählt man jetzt die gewünschte Gruppe und die gewünschte Übungseinheit. Bei nur einer Übungseinheit pro Übung wird kein Auswahlménü für die Übungseinheit angezeigt. Jeder Benutzer sieht im Auswahlménü nur die Gruppen, in denen er als Administrator festgelegt ist.
- Bestätigen mit Klick auf den Button *Gruppe und Übungseinheit anzeigen*
- Die Tabelle wird mit den Benutzern der ausgewählten Gruppe gefüllt.
- Es bestehen folgende Möglichkeiten der Bearbeitung:
 - Notiz für Tutoren eingeben
 - Bewertung auswählen
 - Note eingeben
 - Feedback als Kommentar eingeben
- Nach diesen Aktionen bleibt die Tabelle bestehen, bei den folgenden Operationen wird die Tabelle mit nicht gespeicherten Eingaben verworfen. Es wird daher empfohlen immer zuerst die eingegebenen Daten zu speichern.
 - Feedback als Mail versenden
 - Benutzer zu Teams zusammenfügen (nur bei Abgabetypp Team-Upload)
 - Teams auflösen (nur bei Abgabetypp Team-Upload)
 - einzelne Abgabe herunterladen
 - Alle Speichern und Gesamtnote Aktualisieren
- Durch die Bestätigung mit dem Button *Alle Speichern und Gesamtnote Aktualisieren* wird die Gesamtnote der Übung automatisch durch alle enthaltenen Übungseinheiten berechnet.
- Danach kann eine weitere Gruppe ausgewählt werden.

6 Klassenbeschreibungen

Contents

6.1	ilACOExerciseMemberTableGUI	24
6.2	ilACOGroupDisplayGUI	27
6.3	ilACOGroupGUI	33
6.4	ilACOLinkGUI	38
6.5	ilACOMemberGUI	44
6.6	ilACOPlugin	52
6.7	ilACOTutorGUI	53
6.8	ilACOUHookGUI	61

Die Klassen und deren Zusammensetzung als Gesamtsystem wird nachfolgend beschrieben.

Anfangen mit einem Klassendiagramm unseres Plugins sowie dann die Gruppierung der Klassen sowie anderer Plugin Bestandteile in Komponenten.

Tablogik: class.ilACOPlugin, class.ilACOUHookGUI

Gruppenerstellen: class.ilACOGroupGUI

Gruppenfilter: class.ilACOTutorGUI

Verlinkung: class.ilACOLinkGUI, class.ilACOExerciseMemberTableGUI, templates

Gruppen verwalten: class.ilACOGroupDisplayGUI

Mitgliederverschieben: class.ilACOMemberGUI

Lang Files: lang

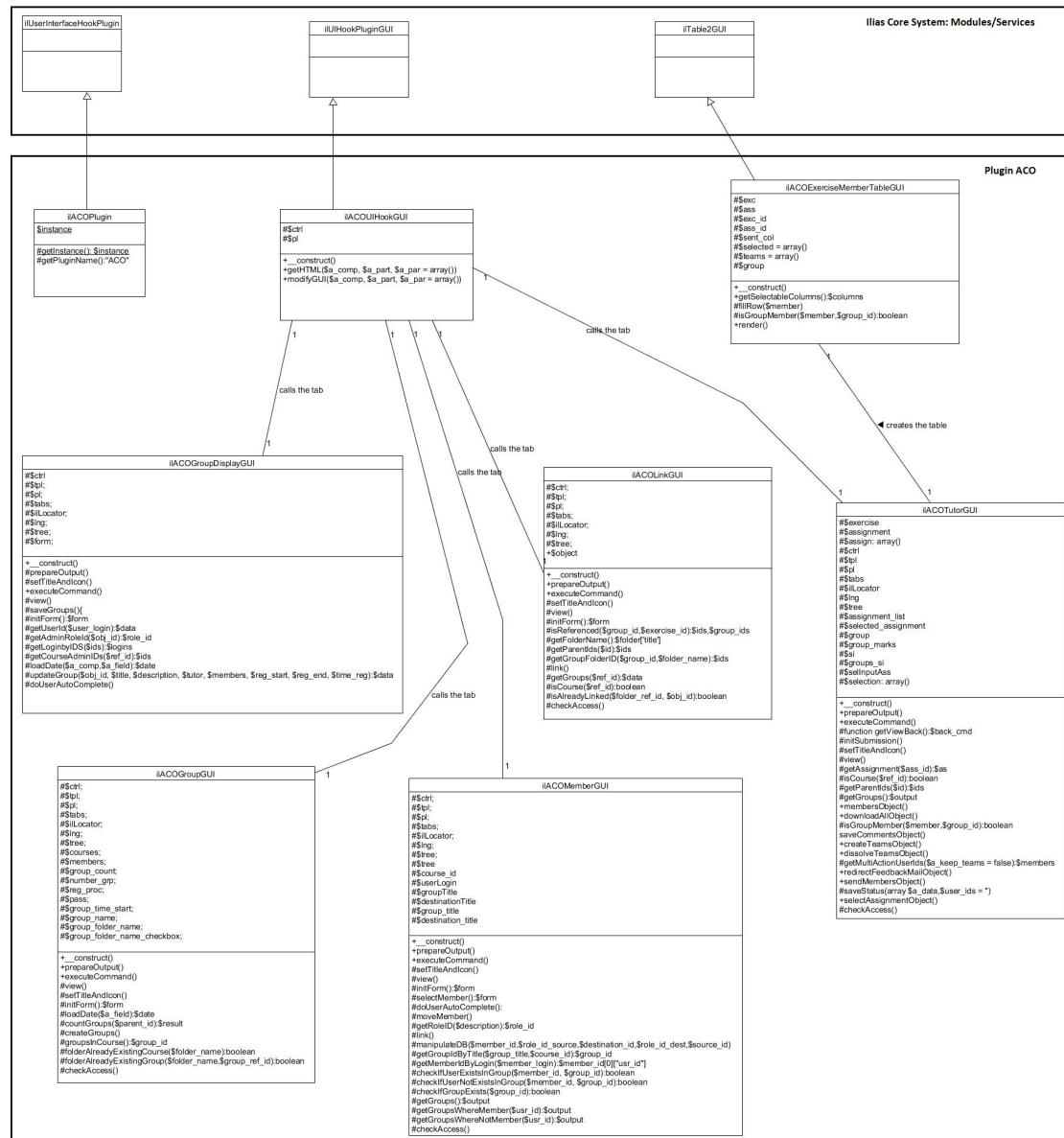


Abbildung 6.1: Klassendiagramm

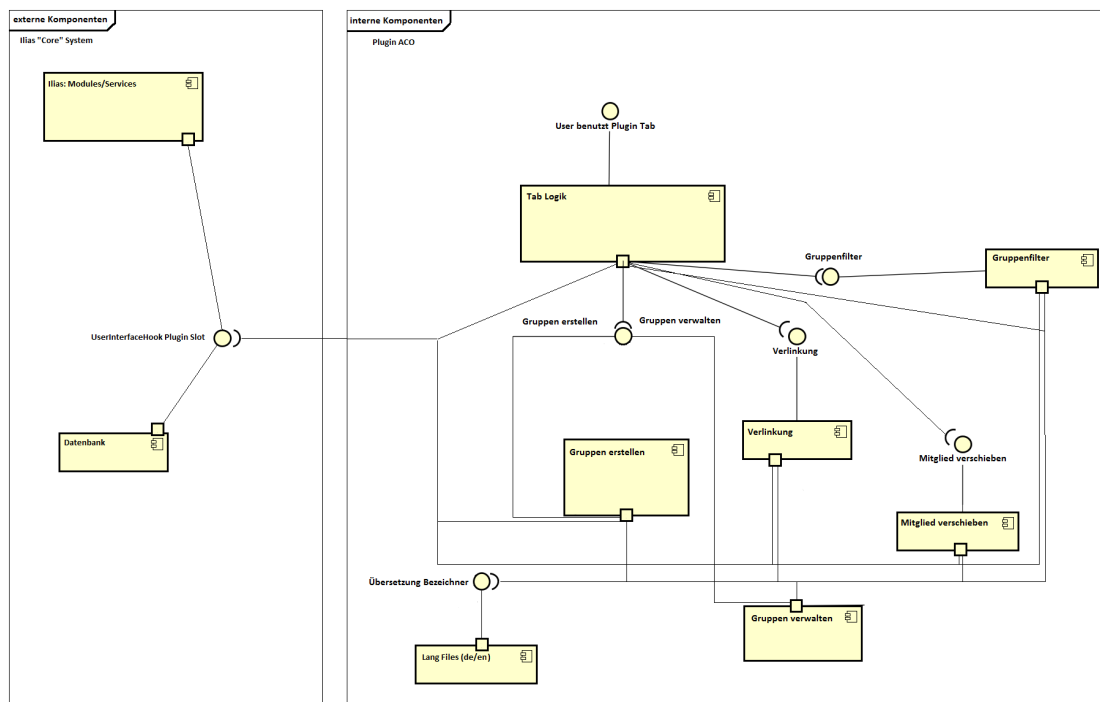


Abbildung 6.2: Komponentendiagramm

6.1 ilACOEExerciseMemberTableGUI

Beschreibung

Diese Klasse implementiert den Aufbau der Tabelle im Tab *Gruppenfilter*. Sie erbt von der ILIAS-Core Klasse ilTable2GUI.

Klassenvariablen

- exc** protected - type: ilExercise - Die darzustellende Exercise
- ass** protected - type: ilAssignment - Das darzustellende Assignment
- exc_id** protected - type: integer - ID der Exercise
- ass_id** protected - type: integer - ID des Assignment
- sent_col** protected - type: boolean - Ob eine Mail versendet wurde
- selected** protected - type: Array - Selektierte Reihen
- teams** protected - type: Array
- group** protected - type: integer - Gruppen ID der darzustellenden Gruppe

Funktions-Liste

- __construct***
- getSelectableColumns***
- fillRow***
- isGroupMember***

Funktionen

__construct

Beschreibung

<code>__construct(\$a_parent_obj, \$a_parent_cmd, \$a_exc, \$a_ass,\$group_id)</code>

Konstruktor der Tabelle für *Gruppenfilter*. Erstellt eine gefilterte Tabelle nach den übergebenen Parametern

Parameter-Liste

- a_parent_obj** - type: object - Objekt in dem die Tabelle erzeugt wird
- a_parent_cmd** - type: String - Letztes Command der Klasse die den Konstruktor aufruft
- a_exc** - type: ilExercise - Exercise die in der Tabelle dargestellt werden soll
- a_ass** - type: ilAssignment - Assignment nach dem gefiltert wird
- group_id** - type: integer - Gruppen-ID nach der gefiltert wird

getSelectableColumns

Beschreibung

getSelectableColumns()

Gibt ein Array zurück, welches bestimmt welche Spalten in der Tabellenansicht aktiviert bzw. deaktiviert werden können.

Rückgabewerte

Gibt ein mehrdimensionales Array zurück.

fillRow

Beschreibung

fillRow(\$member)

Füllt eine Reihe mit den Daten des übergebenen Parameter. Wird von der Methode `__construct` aufgerufen.

Parameter-Liste

- member** - type: Array - Ein Array mit den Feldern `usr_id`, `team_id`, `team`, und `name`

isGroupMember

Beschreibung

isGroupMember(\$member,\$group_id)

Überprüft auf Gruppenmitgliedschaft

Parameter-Liste

member - type: Array - Ein Array mit minimum dem Feld `usr_id`

group_id - type: integer - Gruppen-ID die überprüft wird

Rückgabewerte

boolean - **TRUE** falls die User-ID in der Gruppe mit der Gruppen-ID vorhanden ist, **FALSE** sonst.

6.2 ilACOGroupDisplayGUI

Beschreibung

Diese Klasse implementiert die Funktion *Kurs bearbeiten* und beinhaltet die Möglichkeit, die einzelnen Gruppen in einem Kurs zentral zu verwalten. Hierzu werden bestimmte Parameter der Gruppen in einer tabellarischen Übersicht dargestellt und können editiert werden.

Klassenvariablen

- ctrl** protected - type: ilCtrl - Steuerung Zugriffsrechte
- tpl** protected - type: ilTemplate - Template für Darstellung und Formatierung
- pl** protected - type: ilACOPugin - Instanz des Plugins
- tabs** protected - type: ilTabsGUI - Verwaltung von Tabs
- ilLocator** protected - type: ilLocatorGUI - Darstellung in der Tree Hierarchie
- lng** protected - type: ilLanguage - Einbindung der Language-File von ILIAS
- tree** protected - type: ilTree - Verwaltung der Tree Hierarchie
- form** protected - type: ilPropertyFormGUI - Instanz eines Formulars
- groupAdmins** protected - type: array - Array mit allen Admins in einer Gruppe

Funktions-Liste

- __construct***
- prepareOutput***
- setTitleAndIcon***
- executeCommand***
- view***
- initForm***

saveGroups

getUserId

getAdminRoleId

getLoginbyIDS

getCourseAdminIDs

loadDate

updateGroup

getTableData

doUserAutoComplete

checkAccess

Funktionen

__construct

Beschreibung

`__construct()`

Konstruktion der Grundinstanz

prepareOutput

Beschreibung

`prepareOutput()`

Grundlegende Eigenschaften werden festgelegt: Tabs, Position in der Tree Hierarchie, BackTarget, Titel und Icon, sowie das verwendete Template

setTitleAndIcon

Beschreibung

`setTitleAndIcon()`

Titel und Symbol des Objekt werden festgelegt

executeCommand

Beschreibung

`executeCommand()`

Regelung der Umsetzung von Befehlen mittels `CommandButtons`

view

Beschreibung

`view()`

Ausgabe der initialen Ansicht als HTML-Dokument

initForm

Beschreibung

`initForm()`

Formular wird generiert und mit initialen Werten gefüllt

Rückgabewerte

form - Formular mit Standardwerten, type: `ilPropertyFormGUI`

saveGroups

Beschreibung

`saveGroups()`

Die Änderungen in der tabellarischen Übersicht der Gruppen werden an die Funktion *updateGroup* übergeben. Anschließend wird die tabellarische Übersicht aktualisiert, um die aktuellen Daten anzuzeigen.

getUserId

Beschreibung

`getUserId($user_login)`

Benutzername als Eingabe liefert die dazugehörige UserID

Parameter-Liste

user_login - Benutzername, type: `String`

Rückgabewerte

`data[0]` - UserID, type: Integer

getAdminRoleId

Beschreibung

getAdminRoleId(\$obj_id)

ObjektID einer Gruppe als Eingabe liefert die ObjektID der Benutzerrolle des Gruppen-Administrators

Parameter-Liste

`obj_id` - ID eines Objekts, type: Integer

Rückgabewerte

`role_id[0]` - Objekt ID der Benutzerrolle Gruppen-Administrator, type: Integer

getLoginbyIDS

Beschreibung

getLoginByIDS(\$ids)

Auf Eingabe von UserIDs werden die dazugehörigen Benutzernamen ausgegeben.

Parameter-Liste

`ids` - UserIDs, type: Integer[]

Rückgabewerte

`logins` - Benutzernamen, type: String[]

getCourseAdminIDs

Beschreibung

getCourseAdminIDs(\$ref_id)

Auf Eingabe der ReferenzID eines Kurses wird die UserID des dazugehörigen Kurs-Administrators ausgegeben.

Parameter-Liste

`ref_id` - ReferenzID eines Kurses, type: Integer

Rückgabewerte

ids - UserID des Kurs-Administrators, type: Integer

loadDate

Beschreibung

loadDate(\$a_field)

Datum im Formular wird angepasst, um es in die Datenbank schreiben zu können.

Parameter-Liste

a_field - Wert eines Datumsfelds im Formular, type: ilDateTime

Rückgabewerte

date - Wert aus dem Feld umgewandelt in Datenbank-kompatibles Format, type: ilDateTime

updateGroup

Beschreibung

updateGroups(\$obj_id, \$title, \$description, \$tutor, \$members, \$reg_start, \$reg_end, \$time_reg)

Die Funktion wird von *saveGroups* aufgerufen und schreibt die geänderten Werte in die Datenbank.

Parameter-Liste

obj_id - ObjektID der Gruppe, type: Integer

title - Titel der Gruppe, type: String

description - Beschreibung der Gruppe, type: String

tutor - Benutername des Gruppentutors, type: String

members - Anzahl maximaler Mitglieder, type: Integer

reg_start - Zeitpunkt Anfang Beitritt, type: ilDateTime

reg_end - Zeitpunkt Ende Beitritt, type: ilDateTime

time_reg - Zeitlich begrenzter Beitritt, type: Boolean

getTableData

Beschreibung

getTableData(\$ref_id)

Liefert auf Eingabe der ReferenzID eines Kurses bestimmte Parameter der Gruppen in diesem Kurs. Anmerkung: Wir müssen in die tree table schauen weil crs item nicht sofort aktualisiert wird.

Parameter-Liste

ref_id - ReferenzID des Kurses, type: Integer

Rückgabewerte

data - Parameter aller Gruppen zu dem Kurs, type: Array

doUserAutoComplete

Beschreibung

doUserAutoComplete()

Werden in einem Eingabefeld mind. 3 Zeichen eingegeben, versucht die Funktion Vorschläge zur Auswahl von Benutzern zu machen

checkAccess

Beschreibung

checkAccess()

Überprüft, ob man Lese- oder Schreibrechte auf das aktuelle Objekt hat.

6.3 ilACOGroupGUI

Beschreibung

In dieser Klasse können über ein Formular mehrere Gruppen auf einmal in einem Kurs angelegt werden. Außerdem können optionale ein oder mehrere Ordner im Kurs und den jeweiligen Gruppen erstellt werden. Folgende Eigenschaften stehen für die Gruppen bereit: Gruppenname (Präfix im Titel, der mit aufsteigenden Ziffern ergänzt wird), Anzahl der zu erstellenden Gruppen, maximale Mitgliederzahl, Beitritt direkt oder mit Passwortschutz, optional: Zeitlich begrenzter Zutritt und anlegen eines oder mehrerer Ordner.

Klassenvariablen

CREATION_SUCCEEDED protected - type: String - Ausgabe Erstellung erfolgt

CREATION_FAILED protected - type: String - Ausgabe fehlgeschlagene Erstellung

ctrl protected - type: ilCtrl - Steuerung Zugriffsrechte

tpl protected - type: ilTemplate - Template für Darstellung und Formatierung

pl protected - type: ilACOPlugin - Instanz des Plugins

tabs protected - type: ilTabsGUI - Verwaltung von Tabs

ilLocator protected - type: ilLocatorGUI - Darstellung in der Tree Hierarchie

lng protected - type: ilLanguage - Einbindung der Language-File von ILIAS

tree protected - type: ilTree - Verwaltung der Tree Hierarchie

courses protected - type: Array - Verwaltung von Kursen

members protected - type: Integer - Max. Mitgliederzahl in einer Gruppe

group_count protected - type: Integer - Anzahl an Gruppen

reg_proc protected - type: ilRadioGroupInputGUI - Auswahl Anmeldeart

pass protected - type: String - Passwort bei Anmeldebeschränkung

group_name protected - type: String - Präfix der Gruppentitel

group_folder_name protected - type: String - Titel des/der Ordners/Ordner

group_folder_name_checkbox protected - type: ilCheckboxInputGUI -
Auswahl zur Erstellung von Ordner(n)

Funktions-Liste

__construct

prepareOutput

setTitleAndIcon

executeCommand

view

loadDate

countGroups

createGroups

groupsInCourse

folderAlreadyExistingCourse

folderAlreadyExistingGroup

checkAccess

Funktionen

__construct

Beschreibung

`__construct()`

Konstruktion der Grundinstanz

prepareOutput

Beschreibung

prepareOutput()

Grundlegende Eigenschaften werden festgelegt: Tabs, Position in der Tree Hierarchie, BackTarget, Titel und Icon, sowie das verwendete Template

setTitleAndIcon

Beschreibung

setTitleAndIcon()

Titel und Symbol des Objekt werden festgelegt

executeCommand

Beschreibung

executeCommand()

Regelung der Umsetzung von Befehlen mittels CommandButtons

view

Beschreibung

view()

Ausgabe der initialen Ansicht als HTML-Dokument

loadDate

Beschreibung

loadDate(\$a_field)

Datum im Formular wird angepasst, um es in die Datenbank schreiben zu können.

Parameter-Liste

a_field - Wert eines Datumsfelds im Formular, type: ilDateTime

Rückgabewerte

date - Wert aus dem Feld umgewandelt in Datenbank-kompatibles Format, type: ilDateTime

countGroups

Beschreibung

```
countGroups($parent_id)
```

Gibt die Anzahl an Gruppen zurück, die in einem Kurs vorhanden sind. Anmerkung: Wir müssen in die tree table schauen weil crs item nicht sofort aktualisiert wird.

Parameter-Liste

parent_id - ReferenzID des Kurses, type: Integer

Rückgabewerte

result - Anzahl an Gruppen im Kurs, type: Integer

createGroups

Beschreibung

```
createGroups()
```

Funktion erstellt die Gruppen und eventuell Ordner in einem Kurs entsprechend den Angaben im Formular.

groupsInCourse

Beschreibung

```
groupsInCourse()
```

Gibt die ReferenzIDs der Gruppen in einem Kurs zurück.

Rückgabewerte

group_id - ReferenzIDs der Gruppen im Kurs, type: Array

folderAlreadyExistingCourse

Beschreibung

```
folderAlreadyExistingCourse($folder_name)
```

Überprüft, ob ein Ordner im Kurs bereits vorhanden ist.

Parameter-Liste

folder_name - Titel des Ordners, type: String

Rückgabewerte

(true,false) - Ordner vorhanden? (ja/nein), type: Boolean

folderAlreadyExistingGroup

Beschreibung

folderAlreadyExistingGroup(\$folder_name,\$group_ref_id)

Überprüft, ob ein Ordner in einer Gruppe bereits vorhanden ist.

Parameter-Liste

folder_name - Titel des Ordners, type: String

ref_id - ReferenzID der Gruppe, type: Integer

Rückgabewerte

(true,false) - Ordner vorhanden? (ja/nein), type: Boolean

checkAccess

Beschreibung

checkAccess()

Überprüft, ob man Lese- oder Schreibrechte auf das aktuelle Objekt hat.

6.4 ilACOLinkGUI

Beschreibung

Diese Klasse implementiert die Funktion *Verlinkung* und beinhaltet die Möglichkeit, Exercise- oder Test-Objekte in einem Kurs in alle oder nur in ausgewählte Gruppen in einem Kurs zu verlinken.

Klassenvariablen

ctrl protected - type: ilCtrl - Steuerung Zugriffsrechte

tpl protected - type: ilTemplate - Template für Darstellung und Formatierung

pl protected - type: ilACOPlugin - Instanz des Plugins

tabs protected - type: ilTabsGUI - Verwaltung von Tabs

ilLocator protected - type: ilLocatorGUI - Darstellung in der Tree Hierarchie

lng protected - type: ilLanguage - Einbindung der Language-File von ILIAS

tree protected - type: ilTree - Verwaltung der Tree Hierarchie

Funktions-Liste

__construct

prepareOutput

setTitleAndIcon

executeCommand

view

initForm

getAdminFolderIds

getFolderName

getParentIds

getGroupFolderID

link

getGroups

isCourse

isAlreadyLinked

checkAccess

Funktionen

__construct

Beschreibung

`__construct()`

Konstruktion der Grundinstanz

prepareOutput

Beschreibung

`prepareOutput()`

Grundlegende Eigenschaften werden festgelegt: Tabs, Position in der Tree Hierarchie, BackTarget, Titel und Icon, sowie das verwendete Template

setTitleAndIcon

Beschreibung

`setTitleAndIcon()`

Titel und Symbol des Objekt werden festgelegt

executeCommand

Beschreibung

`executeCommand()`

Regelung der Umsetzung von Befehlen mittels CommandButtons

view

Beschreibung

view()

Ausgabe der initialen Ansicht als HTML-Dokument

initForm

Beschreibung

initForm()

Formular wird generiert und mit initialen Werten gefüllt

Rückgabewerte

form - Formular mit Standardwerten, type: ilPropertyFormGUI

getAdminFolderIds

Beschreibung

getAdminFolderIds()

Gibt alle ID's der Ordner in den aktuell per Checkbox ausgewählten Gruppen zurück die den selben Namen tragen wie der dem Objekt übergeordnete Ordner.

Rückgabewerte

ids - IDs der Ordner in den in der GUI ausgewählten Gruppen mit dem selben Namen des über dem Objekt liegendem Ordners, type: Array

getFolderName

Beschreibung

getFolderName()

Ausgabe des Titels des übergeordneten Ordners

Rückgabewerte

folder['title'] - Titel des Ordners, type: String

-1 - Falls kein Ordner über dem Objekt existiert, type: Integer

getParentIds

Beschreibung

getParentIds(\$id)

Ausgabe der ID des dem Objekt übergeordneten Objekts

Parameter-Liste

id - ID eines Objekts, type: Integer

Rückgabewerte

ids - IDs zu den übergeordneten Objekten eines Objekts, type: Array

getGroupFolderID

Beschreibung

getGroupFolderID()

Gibt die ID des Ordners in der Gruppe mit einem bestimmten Namen zurück.

Parameter-Liste

group_id - ID der Gruppe, type: Integer

folder_name - Titel des Ordners, type: Integer

Rückgabewerte

ids - IDs der Ordner in einer Gruppe, type: Array

-2 - Falls kein solcher Ordner existiert, type: Integer

link

Beschreibung

link()

Verlinkt das gewählte Objekt in die entsprechenden Gruppen, nach der Überprüfung, ob es dort bereits existiert oder ob es in einen Ordner in der Gruppe oder direkt in die Gruppe verlinkt werden muss.

getGroups

Beschreibung

getGroups()

Funktion liefert alle Gruppen in einem Kurs

Parameter-Liste

ref_id - ReferenzID eines Kurses, type: Integer

Rückgabewerte

data - Titel und ReferenzIDs der Gruppen in einem Kurs, type: Array

isCourse

Beschreibung

isCourse()

Überprüft, ob das Objekt zu einer ReferenzID ein Kurs ist

Parameter-Liste

ref_id - ReferenzID eines Kurses, type: Integer

Rückgabewerte

(true, false) - Ist Objekt ein Kurs? (ja/nein), type: Boolean

isAlreadyLinked

Beschreibung

isAlreadyLinked()

Funktion überprüft, ob das Objekt in eine Gruppe schon verlinkt wurde

Parameter-Liste

folder_ref_id - ReferenzID einer Gruppe, type: Integer

obj_id - ID des Objekts, das verlinkt werden soll, type: Integer

Rückgabewerte

(true, false) - Bereits verlinkt? (ja/nein), type: Boolean

checkAccess

Beschreibung

checkAccess()

Überprüft, ob man Lese- oder Schreibrechte auf das aktuelle Objekt hat.

6.5 ilACOMemberGUI

Beschreibung

Diese Klasse implementiert die Funktion *Mitglieder verschieben* und beinhaltet die Möglichkeit, einzelne Mitglieder einer Gruppe in einem Kurs in eine andere Gruppe zu verschieben. Hierzu wird der zu verschiebende Benutzer über ein AutoComplete-Feld ausgewählt, anschließend werden die Gruppen angezeigt, von denen er verschoben werden kann, sowie die Gruppen, in die er verschoben werden kann.

Klassenvariablen

- ctrl** protected - type: ilCtrl - Steuerung Zugriffsrechte
- tpl** protected - type: ilTemplate - Template für Darstellung und Formatierung
- pl** protected - type: ilACOPugin - Instanz des Plugins
- tabs** protected - type: ilTabsGUI - Verwaltung von Tabs
- ilLocator** protected - type: ilLocatorGUI - Darstellung in der Tree Hierarchie
- lng** protected - type: ilLanguage - Einbindung der Language-File von ILIAS
- tree** protected - type: ilTree - Verwaltung der Tree Hierarchie
- course_id** protected - type: Integer - Kurs ID für Verwendung in der Klasse
- userLogin** protected - type: String - Benutzername für Verwendung in der Klasse
- groupTitle** protected - type: String - Gruppentitel für Gruppe zum Entfernen
- destinationTitle** protected - type: String - Gruppentitel für Gruppe zum Einfügen
- group_title** protected - type: String - Gruppentitel für Gruppe zum Entfernen
- destination_title** protected - type: String - Gruppentitel für Gruppe zum Einfügen

Funktions-Liste

__construct

setTitleAndIcon

prepareOutput

executeCommand

view

initForm

selectMember

doUserAutoComplete

moveMember

getRoleID

manipulateDB

getGroupIdByTitle

getMemberIdByLogin

checkIfUserExistsInGroup

checkIfUserNotExistsInGroup

checkIfGroupExists

getGroups

getGroupsWhereMember

getGroupsWhereNotMember

checkAccess

Funktionen

__construct

Beschreibung

__construct()

Konstruktion der Grundinstanz

prepareOutput

Beschreibung

prepareOutput()

Grundlegende Eigenschaften werden festgelegt: Tabs, Position in der Tree Hierarchie, BackTarget, Titel und Icon, sowie das verwendete Template

setTitleAndIcon

Beschreibung

setTitleAndIcon()

Titel und Symbol des Objekt werden festgelegt

executeCommand

Beschreibung

executeCommand()

Regelung der Umsetzung von Befehlen mittels CommandButtons

view

Beschreibung

view()

Ausgabe der initialen Ansicht als HTML-Dokument

initForm

Beschreibung

initForm()

Formular wird generiert und mit initialen Werten gefüllt

Rückgabewerte

form - Formular mit Standardwerten, type: ilPropertyFormGUI

selectMember

Beschreibung

selectMember()

Formularansicht mit Name des Benutzers, Gruppen, in denen er Mitglied bw. kein Mitglied ist. Ruft nach Klicken auf den CommandButon die Funktion *moveMember* auf

Rückgabewerte

form - Formular gemäß dem angegebenen Benutzer, type: ilPropertyFormGUI

doUserAutoComplete

Beschreibung

doUserAutoComplete()

Werden in einem Eingabefeld mind. 3 Zeichen eingegeben, versucht die Funktion Vorschläge zur Auswahl von Benutzern zu machen

moveMember

Beschreibung

moveMember()

Werte aus dem Formular aus *selectMember* werden ausgewertet und an *manipulateDB* übergeben. Anschließend wieder zurück zur initialen Formularansicht.

getRoleID

Beschreibung

getRoleID(\$description)

Gibt die ObjektID zurück, die dem Eintrag in der Datenbank entspricht, in dem die Rolle als Gruppenmitglied geregelt ist.

Parameter-Liste

description - Datenbankeintrag für Regelung Gruppenmitgliedschaft, type: String

Rückgabewerte

role_id - ObjektID des entsprechenden Datenbankeintrags, type: Integer

manipulateDB

Beschreibung

`manipulateDB($member_id,$role_id_source,$destination_id,$role_id_dest,$source_id)`

Schreibt die Werte aus dem Formular in die Datenbank und verschiebt somit den Benutzer aus einer Gruppe in eine andere und passt die Zugriffsrechte des Benutzers entsprechend an.

Parameter-Liste

member_id - UserID des Benutzers, type: Integer

role_id_source - ID alt zur Festlegung der Rolle des Benutzers, type: Integer

destination_id - ID der neuen Gruppe des Benutzers, type: Integer

role_id_dest - ID neu zur Festlegung der Rolle des Benutzers, type: Integer

source_id - ID der bisherigen Gruppe des Benutzers, type: Integer

getGroupIdByTitle

Beschreibung

`getGroupIdByTitle($group_title,$course_id)`

Gibt bei Eingabe des Gruppentitels die dazugehörige ObjektID zurück.

Parameter-Liste

group_title - Titel der Gruppe, type: String

course_id - ReferenzID des übergeordneten Kurses, type: Integer

Rückgabewerte

group_id - ObjektID der Gruppe, type:

getMemberIdByLogin

Beschreibung

`getMemberIdByLogin($member_login)`

Gibt bei Eingabe des Benutzernamens die dazugehörige UserID zurück.

Parameter-Liste

member_login - Benutzername, type: String

Rückgabewerte

member_id[0][üsr_id] - UserID, type: Integer

checkIfUserExistsInGroup

Beschreibung

checkIfUserExistsInGroup(\$member_id,\$group_id)

Überprüft, ob der Benutzer in einer Gruppe bereits vorhanden ist oder nicht und gibt einen entsprechenden Boolean-Wert zurück.

Parameter-Liste

member_id - UserID des Benutzers, type:

group_id - ObjektID der Gruppe, type:

Rückgabewerte

(true,false) - Benutzer bereits Mitglied? (ja, nein), type: Boolean

checkIfUserNotExistsInGroup

Beschreibung

checkIfUserNotExistsInGroup(\$member_id,\$group_id)

Überprüft, ob der Benutzer in einer Gruppe bereits vorhanden ist oder nicht und gibt einen entsprechenden Boolean-Wert zurück.

Parameter-Liste

member_id - UserID des Benutzers, type: Integer

group_id - ObjektID der Gruppe, type: Integer

Rückgabewerte

(true,false) - Benutzer kein Mitglied? (ja, nein), type:

checkIfGroupExists

Beschreibung

checkIfGroupExists(\$group_id)

Überprüft, ob die angegebene Gruppe tatsächlich existiert.

Parameter-Liste

\$group_id - ObjektID der Gruppe, type: Integer

Rückgabewerte

(true,false) - Gruppe vorhanden? (ja/nein), type: Boolean

getGroups

Beschreibung

getGroups()

Funktion gibt die Titel aller Gruppen zurück, die in einem Kurs existent sind.

Rückgabewerte

output - Titel der Gruppen, type: Array

getGroupsWhereMember

Beschreibung

getGroupsWhereMember(\$usr_id)

Gibt die Titel der Gruppen zurück, in denen ein bestimmter Benutzer ein Mitglied ist.

Parameter-Liste

usr_id - UserID des Benutzers, type: Integer

Rückgabewerte

output - Titel der Gruppen, type: Array

getGroupsWhereNotMember

Beschreibung

`getGroupsWhereNotMember($usr_id)`

Gibt die Titel der Gruppen zurück, in denen ein bestimmter Benutzer kein Mitglied ist.

Parameter-Liste

usr_id - UserID des Benutzers, type: Integer

Rückgabewerte

output - Titel der Gruppen, type: Array

checkAccess

Beschreibung

`checkAccess()`

Überprüft, ob man Lese- oder Schreibrechte auf das aktuelle Objekt hat.

6.6 ilACOPlugin

Beschreibung

Standard Klasse für ILIAS-Plugins erbt von *ilUserInterfaceHookPlugin*

Klassenvariablen

instance - type: ilACOPlugin - Instanz des Plugin

Funktions-Liste

getInstance

getPluginName

Funktionen

getInstance

Beschreibung

getInstance()

Gibt die Instanz des Plugins zurück

Rückgabewerte

ilACOPlugin - aktuelle Instanz des Plugins

getPluginName

Beschreibung

getPluginName()

Gibt den Name des Plugins zurück

Rückgabewerte

String: ACO

6.7 ilACOTutorGUI

Beschreibung

blabla

Klassenvariablen

VIEW_ASSIGNMENT protected - type: Integer - Konstante

VIEW_PARTICIPANT protected - type: Integer - Konstante

VIEW_GRADES protected - type: Integer - Konstante

exercise protected - type: ilObjExercise - Exercise Objekt

assignment protected - type: ilExAssignment - Ausgewähltes Assignment

ctrl protected - type: ilCtrl - Steuerung Zugriffsrechte

tpl protected - type: ilTemplate - Template für Darstellung und Formatierung

pl protected - type: ilACOPlugin - Instanz des Plugins

tabs protected - type: ilTabsGUI - Verwaltung von Tabs

ilLocator protected - type: ilLocatorGUI - Darstellung in der Tree Hierarchie

lng protected - type: ilLanguage - Einbindung der Language-File von ILIAS

tree protected - type: ilTree - Verwaltung der Tree Hierarchie

group protected - type: - Gruppen-ID der selektierten Gruppe

Funktions-Liste

__construct

prepareOutput

setTitleAndIcon

executeCommand

getViewBack

initSubmission

view

getAssignment

isCourse

getParentIds

getGroups

membersObject

downloadAllObject

isGroupMember

saveCommentsObject

saveCommentForLearnersObject

createTeamsObject

dissolveTeamsObject

getMultiActionUserIds

redirectFeedbackMailObject

sendMembersObject

saveStatusAllObject

saveStatus

selectAssignmentObject

checkAccess

Funktionen

__construct

Beschreibung

`__construct()`

Konstruktion der Grundinstanz

prepareOutput

Beschreibung

`prepareOutput()`

Grundlegende Eigenschaften werden festgelegt: Tabs, Position in der Tree Hierarchie, BackTarget, Titel und Icon, sowie das verwendete Template

setTitleAndIcon

Beschreibung

`setTitleAndIcon()`

Titel und Symbol des Objekt werden festgelegt

executeCommand

Beschreibung

`executeCommand()`

Regelung der Umsetzung von Befehlen mittels CommandButtons

getViewBack

Beschreibung

`getViewBack()`

Gibt das letztes Command für eine Ansicht zurück

Rückgabewerte

`back_cmd` - Command , type: String

initSubmission

Beschreibung

initSubmission()

Erstellt eine leere neue Abgabe für einen Benutzer

Rückgabewerte

new ilExSubmission - Neue Abgabe für den User , type: ilExSubmission

view

Beschreibung

view()

Ausgabe der initialen Ansicht als HTML-Dokument

getAssignment

Beschreibung

getAssignment(\$ass_id)

Gibt das Assignment mit der übergebenen ID der aktuellen Exercise zurück

Parameter-Liste

as_id - Assignment ID , type: Integer

Rückgabewerte

as - Assignment mit übergebener ID, type: ilExAssignment

isCourse

Beschreibung

isCourse(\$ref_id)

Überprüft ob das Object mit der ID \$ref_id vom Typ Kurs ist

Parameter-Liste

ref_id - ID des zu überprüfenden Objekts, type: Integer

Rückgabewerte

TRUE, falls das Objekt ein Kurs ist, **FALSE** sonst, type: Boolean

getParentIds

Beschreibung

getParentIds(\$id)

Gibt alle IDs der Objekte zurück in denen das Objekt mit der ID \$id liegt.

Parameter-Liste

id - ID des zu überprüfenden Objekts , type: Integer

Rückgabewerte

ids - IDs aller Übergeordneter Objekte , type: Array

getGroups

Beschreibung

getGroups()

Gibt alle Gruppen des Kurses zurück

Rückgabewerte

output - Alle Gruppen des Kurses als mehrdimensionales Array , type: Array

membersObject

Beschreibung

membersObject()

Erstellt die Standardansicht mit Toolbar und Buttons, ruft die Tabelle auf und übergibt die nötigen Parameter

downloadAllObject

Beschreibung

downloadAllObject()

Lädt alle Abgegebenen Dateien des Aktuellen Gruppenfilter herunter

isGroupMember

Beschreibung

isGroupMember(\$member,\$group_id)

Überprüft ob ein Benutzer Mitglied in einer Gruppe ist

Parameter-Liste

member - Benutzer ID des zu überprüfenden Benutzers , type: Integer

group_id - Gruppen ID der zu überprüfenden Gruppe , type: Integer

Rückgabewerte

TRUE, falls der Benutzer Mitglied in der Gruppe ist, **FALSE** sonst - , type: Boolean

saveCommentsObject

Beschreibung

saveCommentsObject()

Speichert die eingegebene Notiz für Tutoren

saveCommentForLearnersObject

Beschreibung

saveCommentForLearnersObject()

Speichert den eingegebenen Kommentar für den Benutzer

createTeamsObject

Beschreibung

createTeamsObject()

Erstellt ein Team aus allen selektierten Benutzern

dissolveTeamsObject

Beschreibung

dissolveTeamsObject()

löst alle ausgewählten Teams auf

getMultiActionUserIds

Beschreibung

```
getMultiActionUserIds($a_keep_teams)
```

Gibt die Benutzer-IDs aller selektierten Benutzer zurück

Parameter-Liste

\$a_keep_teams - , type: Boolean

Rückgabewerte

members - Array mit Benutzer IDs , type: Array

redirectFeedbackMailObject

Beschreibung

```
redirectFeedbackMailObject()
```

Setzt den Feedback Status des Benutzers und leitet zum Email Fenster weiter

sendMembersObject

Beschreibung

```
sendMembersObject()
```

Sendet das Assignment per Mail an alle ausgewählten Benutzer

saveStatusAllObject

Beschreibung

```
saveStatusAllObject()
```

Lieft alle Daten aus der Tabelle aus

saveStatus

Beschreibung

```
saveStatus(array $a_data,$user_ids)
```

Speichert alle veränderten Daten aus der Tabelle in die Datenbank. Berechnet die Gesamtpunkte eines Arbeitsblatts neu

Parameter-Liste

a_data - Daten die gespeichert werden , type: Array

user_id - Benutzer IDs aller veränderten Benutzer , type: Array

selectAssignmentObject

Beschreibung

`selectAssignmentObject()`

Generiert die Tabellenansicht für die ausgewählte Gruppe und das ausgewählte Assignment

checkAccess

Beschreibung

`checkAccess()`

Überprüft, ob man Lese- oder Schreibrechte auf das aktuelle Objekt hat.

6.8 ilACOUHookGUI

Beschreibung

Grundklasse zur Konstruktion einer Instanz des Plugins in ILIAS und Festlegung des Namens.

Klassenvariablen

instance protected - type: ilACOPlugin - Instanz des Plugins

Funktions-Liste

getInstance

getPluginName

Funktionen

getInstance

Beschreibung

getInstance()

Erstellt eine Instanz des Plugins

Rückgabewerte

instance - Instanz des Plugins, type: ilACOPlugin

getPluginName

Beschreibung

getPluginName()

Gibt den Namen des Plugins aus

Rückgabewerte

ÄCO - Name des Plugins, type: String

7 Tests

Hier werden unsere Testfälle tabellarisch aufgelistet, mit denen wir unser Plugin unter Ilias 5.2.2 getestet haben.

Die Tests sind nach den Tabs und Subtabs aufgeteilt. Voraussetzung, um diese Test durchführen zu können, ist die erfolgreiche Installation und Aktivierung des Plugins. Der Haken bedeutet, dass das Ist-Resultat dem Soll-Resultat entsprochen hat.

Tabelle 7.1: Testfälle

Testfälle Plugin ACO**Tab Kurs verwalten****Subtab Gruppen erstellen**

Testfall	Sollresultat	Istresultat
Klicken auf Kurs verwalten	Weiterführung auf neue Seite mit den Subtabs Gruppen erstellen/ verwalten/ Mitglieder verschieben	✓
Klicken auf den Knopf zurück	Führt einen zurück zum Kurs, in dem man den Tab aufgerufen hat	✓
Erstelle in neuen Kurs 12 Gruppen (Name: Gruppe) mit max. Mitglieder 15 und direktem Beitritt	12 Gruppen <i>Gruppe</i> fortlaufend nummeriert mit max. Mitgliedern 15 und direktem Beitritt	✓
Erstelle in Kurs, in dem schon 12 Gruppen existieren, 5 neue mit denselben Einstellungen wie oben	Erstellt 5 neue Gruppen mit fortlaufender Nummer, max. Mitglieder 15 und direktem Beitritt	✓
Erstelle eine Gruppe mit zeitlich begrenzten Beitritt Start: 27.03.17 15:30 Ende: 09.12.17 00:01	Erstellt diese Gruppe mit fortlaufender Nummer und diesem Registrierungszeitraum	✓
Erstelle zwei Gruppen mit Passwortgeschütztem Beitritt, pw: passwort"	Erstellt zwei Gruppen mit diesem Passwort	✓
Erstelle eine neue Gruppe mit Ordnerstruktur: Admin Folder	Erstellt Gruppe mit fortlaufender Nummer und AdminFolder im Kurs und in jeder existierenden Gruppe	✓
Lösche eine Gruppe im Kurs und erstelle danach im Plugin eine neue	Neue Gruppe wird mit derselben Nummer erstellt wie die gelöschte	✓
Erstelle Gruppe mit Buchstaben in Gruppenanzahl	Fehlermeldung: Erstellen der Gruppen fehlgeschlagen, überprüfen Sie ihre Eingaben.	✓
Erstelle Gruppe mit Zahl gefolgt von Buchstaben im max. Mitglieder Feld	Ignoriere Buchstaben hinter der Zahl und erstelle Gruppe mit dieser Anzahl max. Mitglieder	✓
Erstelle Gruppe mit Buchstaben in Anzahl Gruppen	Fehlermeldung: <i>Erstellen der Gruppen fehlgeschlagen, überprüfen Sie ihre Eingaben.</i> wird angezeigt und keine Gruppe erstellt	✓
Erstelle Gruppe mit Buchstaben hinter erster Zahl in Anzahl Gruppen	System ignoriert Eingabe hinter dem ersten Buchstaben	✓
Erstelle Gruppe mit Buchstaben in max. Mitglieder	Interpretiere Buchstaben als 0 und erzeuge Gruppe ohne Beschränkung	✓
Erzeuge Gruppe mit Ordnerstruktur (Admin Folder)	Gruppe und der Ordner im Kurs, als auch in der Gruppe wurden erzeugt	✓

Tabelle 7.2: Testfälle

Erzeuge Gruppe mit Ordnerstruktur (AdminFolder und Blätter), obwohl Admin Folder schon im Kurs existiert, aber noch in keiner Gruppe	Gruppe wurde erzeugt und in jeder Gruppe befinden sich beide Ordner 2 mal, im Kurs selbst auch, d.h. es gibt kein AdminFolder Duplikat	✓
Erstelle Gruppe mit Anmeldezeitraumende vor Anmeldezeitraumstart	Gruppe wird mit Anmeldezeitraumstart = Anmeldezeitraumende erstellt	✓
Teste mit einem User, ob dieser Zugriff auf die Plugin Tabs hat	User hat keinen Zugriff auf die Plugin Tabs	✓
Subtab Kurs bearbeiten und Mitglieder verschieben aufrufen	Weiterführung auf die entsprechenden Subtabs	✓

Subtab Kurs bearbeiten

Testfall	Sollresultat	Istresultat
Subtab Kurs bearbeiten aufrufen		
Klicken in den drei Subtabs auf den Knopf zurück	Führt einen zurück zum Kurs in dem man den Tab aufgerufen hat	✓
In Kurs mit bestehenden Gruppen nichts verändern und auf speichern drücken	Einstellungen bleiben gleich	✓
Gruppenname einer bestehenden Gruppe ändern inkl. Nummer und Sonderzeichen	übernimmt neuen Namen für die Gruppe (unabhängig, ob man nur Buchstaben eingibt oder gar nur Sonderzeichen)	✓
ändere leeres Feld Raum und Uhrzeit auf Raum 0.125 8 Uhr	übernimmt die Eingabe und zeigt sie in der Gruppenbeschreibung an	✓
Ändere max. Mitglieder auf 8	ändere max. Mitglieder genau dieser Gruppe auf 8	✓
Besuche Sub Tab bei Kurs ohne Gruppen und mit genau 22 Gruppen, lösche dann eine Gruppe bei den 22 und füge bei dem ohne eine hinzu	Zeigt 22 Gruppen, keine, eine und 21 und deren Einstellungen an	✓
Eingabe von Buchstaben, Sonderzeichen, 0 oder leerer Eingabe in max. Mitglieder	Interpretiert all diese Eingaben als 0 und übernimmt diesen Wert	✓
Bearbeite Gruppe mit Buchstaben hinter ersten Zahl in Anzahl Gruppen	System ignoriert Eingabe hinter dem ersten Buchstaben	✓
Bearbeite Gruppe mit Buchstaben in max. Mitglieder	Interpretiere Buchstaben als 0 und erzeuge Gruppe ohne Beschränkung	✓
Füge zeitlich begrenzten Beitritt bei einer Gruppe hinzu	übernimmt den festgelegten Anmeldezeitraumstart und -ende	✓

Tabelle 7.3: Testfälle

Bearbeite Gruppe mit Anmeldezeitraumende vor Anmeldezeitraumstart	Gruppe wird mit Anmeldezeitraumstart und Anmeldezeitraumende + 2 Tage übernommen	✓
Entferne Checkboxhaken bei einer Gruppe mit zeitlich begrenztem Beitritt	Gruppe hat keinen begrenzten Beitritt mehr	✓
Trage einen Benutzer, der nicht in der entsprechenden Gruppe ist unter Tutor ein	Füge Benutzer in der Gruppe hinzu und gebe ihm Adminrechte (Autocomplete hat funktioniert)	✓
Entferne Eintrag im Tutorfeld bzw. den Tutor	Fehlermeldung und übernimmt wieder alten Tutor	
Trage einen Benutzer, welcher in der entsprechenden Gruppe ist unter Tutor ein	Benutzer hat jetzt Tutorrechte bzw Gruppenadminrechte (Autocomplete hat funktioniert)	✓
Führe obere Tests für mehrere Gruppen, die existieren, gleichzeitig aus	Führt die Tests genauso aus, nur für alle Gruppen die geändert wurden	✓
Subtab Gruppen erstellen und Mitglieder verschieben aufrufen	Weiterführung auf die entsprechenden Sub tabs	✓

Subtab Mitglieder verschieben

Testfall	Sollresultat	Istresultat
Klicken in den drei Subtabs auf den Knopf zurück	Führt einen zurück zum Kurs, in dem man den Tab aufgerufen hat	✓
Gebe Nutzernamen ein, der in Gruppe 1 und 2 ist und verschiebe ihn aus Gruppe 1 in Gruppe 3	Zeige in neuer Seite Benutzernamen und bisherige Gruppen, sowie im drop down darunter alle ex. Gruppen an (Autocomplete hat funktioniert), Nutzer wurde verschoben	✓
Testfall 1 und Anmeldungsbeginn in der Zukunft	Verschiebt Nutzer trotzdem	✓
Testfall 1 und Anmeldungsende schon vorbei	Verschiebt den Nutzer trotzdem	✓
Testfall 1 und max. Mitglieder überschritten (keine Plätze mehr verfügbar)	Verschiebt Nutzer trotzdem	✓
Nutzer noch in keiner Gruppe, soll verschoben werden	Dropdown bisherige Gruppe leer, Fehlermeldung "Gruppe existiert nicht"	✓
Nutzer nicht im Kurs, will ihn trotzdem verschieben	Dropdown bisherige Gruppe leer, Fehlermeldung "Gruppe existiert nicht"	✓
Es gibt nur eine Gruppe und zu verschiebender Nutzer in dieser	Dropdown neue Gruppe leer, Fehlermeldung "Gruppe existiert nicht"	✓
Subtab Gruppen erstellen und Kurs bearbeiten aufrufen	Weiterführung auf die entsprechenden Sub tabs	✓

Tabelle 7.4: Testfälle

Verschiebe User, der Gruppenadmin ist, in eine andere Gruppe	bleibt Gruppenadmin in Ausgangsgruppe und ist neues Gruppenmitglied in neuer Gruppe	✓
--	---	---

Tab Verlinkung

Verlinke Übung in eine Gruppe	Übung wurde in diese Gruppe verlinkt	✓
Locator bzw. Hierarchie im Verlinkungstab benutzen	Hierarchie führt einen an den beschrieben Ort	✓
Verlinke Übung in alle 20 Gruppen, obwohl in drei schon der Test verlinkt wurde	Verlinke in alle Gruppen, in denen noch kein Test existiert und nur in den Admin Folder, falls er existiert, sonst einfach in die Gruppe	✓
Verlinke Test in alle 20 Gruppen, obwohl in drei schon der Test verlinkt wurde	Verlinke in alle Gruppen, in denen noch kein Test existiert und nur in den Admin Folder, falls er existiert, sonst einfach in die Gruppe	✓
In Kurs existieren keine Gruppen, versuche Übung zu verlinken	Zeige keine Gruppen in bestimmten Gruppe verlinken an und verlinkt auch nicht	✓
Erzeuge Test	Verlinkung Tab wird sofort nach der Testerstellung angezeigt	✓
Verlinke Test in 4 von 21 Gruppen aus Admin Folder, obwohl in einer Gruppe kein Admin Folder existiert	Verlinke in alle ausgewählten Gruppen den Test	✓
Verlinke Test in alle 21 Gruppen, obwohl in vier schon der Test verlinkt wurde	Verlinke in alle Gruppen, in denen noch kein Test existiert und nur in den Admin Folder, falls er existiert, sonst einfach in die Gruppe	✓
Backtarget aus Verlinkungstab Test benutzen	Führt zurück auf Testinfo	✓
Locator bzw. Hierarchie im Verlinkstab Test benutzen	Führt zurück auf erwartetes Ziel	✓
In Kurs existieren keine Gruppen versuche Test zu verlinken	Zeige keine Gruppen in bestimmten Gruppe verlinken an und verlinkt auch nicht	✓
Verlinke Test in eine Gruppe	Test wurde in diese Gruppe verlinkt	✓

Tabelle 7.5: Testfälle

Gruppenfilter

Öffne Tab Gruppenfilter	Eine leere Tabelle wird angezeigt, in der Toolbar können alle Gruppen, in denen man Schreibrechte besitzt, ausgewählt werden, alle Übungseinheiten der Übung können ausgewählt werden	✓
Wähle eine Gruppe und eine Übungseinheit aus	Es werden in der Tabelle nur die Benutzer der ausgewählten Gruppe angezeigt. Alle abgegebenen Dateien werden angezeigt, die eingestellten Noten, Bewertungen, Notizen und Kommentare werden in der Tabelle angezeigt	✓
Speichere Eingaben	Alle Eingaben werden in die Datenbank geschrieben, die Gesamtnote der Übung wird neu berechnet. Es wird wieder die leere Tabelle ohne Auswahl angezeigt	✓
Markiere Benutzer und wähle <i>Als Mail versenden</i>	Weiterleitung zum Mail Fenster, in dem als Empfänger alle ausgewählten Benutzer eingetragen sind	✓
Mail versenden	Die Mail wird an alle Empfänger verschickt, es wird wieder die leere Tabelle angezeigt.	✓
Einzelne Abgabe herunterladen	Nach dem Klick auf <i>Abgaben herunterladen</i> wird die Abgabe heruntergeladen	✓
Klick auf <i>Zurück</i>	Der Benutzer wird zurück auf die Info Seite der Übung geleitet	✓
Klick auf die übergeordneten Elemente, die am oberen Rand der Seite angezeigt werden	Der Benutzer wird zu dem jeweiligen Objekt weitergeleitet	✓
Auswählen einer Gruppe ohne Abgaben	Die Tabelle bleibt leer	✓
Auswählen einer Teamabgabe	Teams werden angezeigt, Auswahlmöglichkeiten für Team erstellen und Team auflösen erscheinen	✓
Benutzer ist in mehreren Gruppen Mitglied	Benutzer wird bei beiden Gruppen angezeigt	✓
Benutzer hat nichts abgegeben	Keine Abgabe erscheint in der Tabelle	✓
Benutzer besitzt keine Schreibrechte	Tab <i>Gruppenfilter</i> wird nicht angezeigt	✓
Benutzer besitzt Schreibrechte, ist aber kein Gruppenadministrator	Der Benutzer bekommt keine Gruppen als Auswahlmöglichkeit angezeigt, es wird eine leere Tabelle angezeigt	✓