
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

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 3 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.

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

The screenshot shows the ILIAS user interface for managing groups within a course. The top navigation bar includes 'Open Source eLearning', 'ILIAS', and tabs for 'PERSONLICHER SCHREIBTISCH', 'MAGAZIN', and 'ADMINISTRATION'. The 'ADMINISTRATION' tab is active, and the 'Gruppen verwalten' sub-tab is highlighted with a red arrow. The main content area is divided into two sections: 'INHALT' on the left and a 'Kalender' (Calendar) on the right. The 'INHALT' section lists the following items:

- AdminFolder
- Blatt1
- Gruppe 01 (Freie Plätze: 12)
- Gruppe 02 (Freie Plätze: 11)
- Gruppe 03 (Freie Plätze: 12)
- Gruppe 04 (Freie Plätze: 12)
- Gruppe 05 (Freie Plätze: 12)

The 'Kalender' section shows a calendar for March 2017, with the 21st highlighted. The calendar table is as follows:

KW	Mo	Di	Mi	Do	Fr	Sa	So
09	27	28	1	2	3	4	5
10	6	7	8	9	10	11	12
11	13	14	15	16	17	18	19
12	20	21	22	23	24	25	26
13	27	28	29	30	31	1	2

Abbildung 2.1: Ansicht Tab Gruppen verwalten im jeweiligen Kurs

2.2 Gruppen erstellen

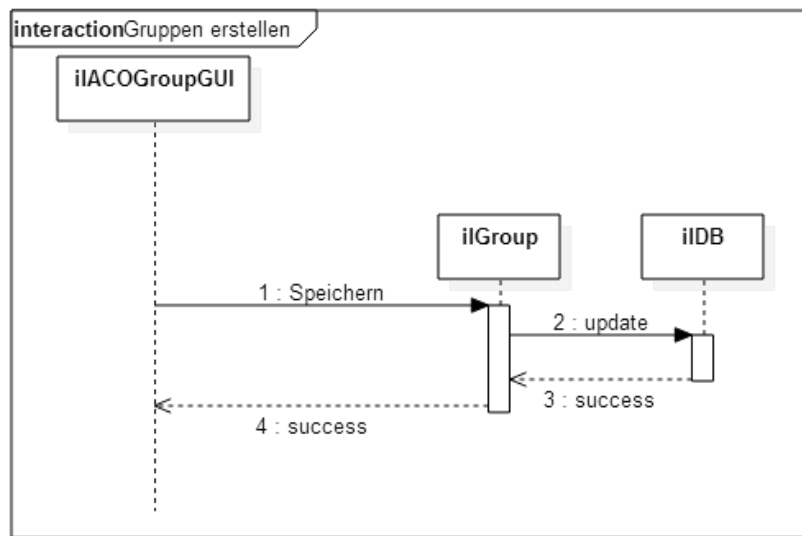


Abbildung 2.2: Sequenzdiagramm Gruppen erstellen

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

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 in dieser Session erzeugten Gruppen dasselbe)

Zeitlich begrenzter Beitritt

- 1 Checkbox setzen
- 2 Zwei Datumseingabefelder tauchen für den Anmelde-Zeitraum auf, sowohl für Beitrittsanfang und 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.

2.3 Kurs bearbeiten

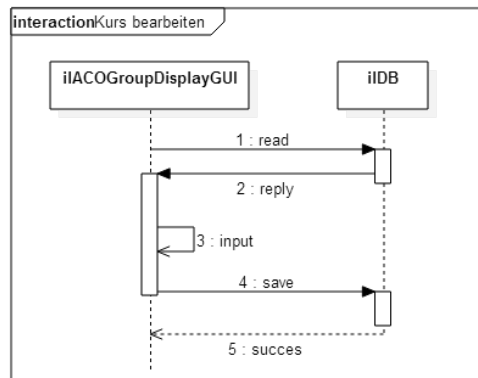


Abbildung 2.4: Sequenzdiagramm Kurs bearbeiten

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. (n aus den natürlichen Zahlen)

Gruppen

Die Felder können einzelnen 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 der aus der Gruppe einem Mitglieder die Tutor bzw. Gruppenadministrationsrechte zuzuweisen bzw. zu entziehen.

hinzufügen Ilias Benutzernamen eingeben. Nach dem dritten Buchstaben funktioniert springt die Auto-Complete Funktion an und hilft beim vervollständigen.

löschen Sollte ein zuvor gefülltes Feld leer hinterlassen werden, werden dem Benutzer hinter dem entfernten Text die Adminrechte entzogen.

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 und Beitrittsende einen Zeitpunkt auswählen. Format: Tag/Monat/Jahr/Stunde/Minuten (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

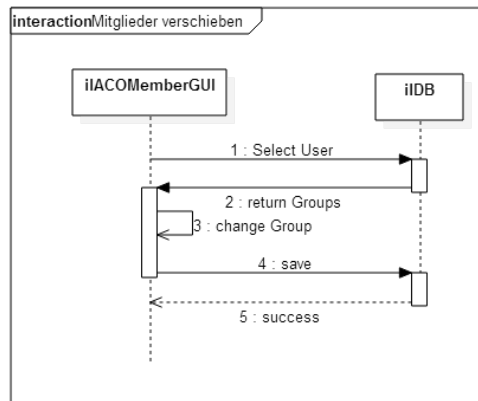


Abbildung 2.6: Sequenzdiagramm Mitglieder verschieben

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 [Benutzer auswählen](#)

* Erforderliche Angabe

(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 [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"
- 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)
- 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

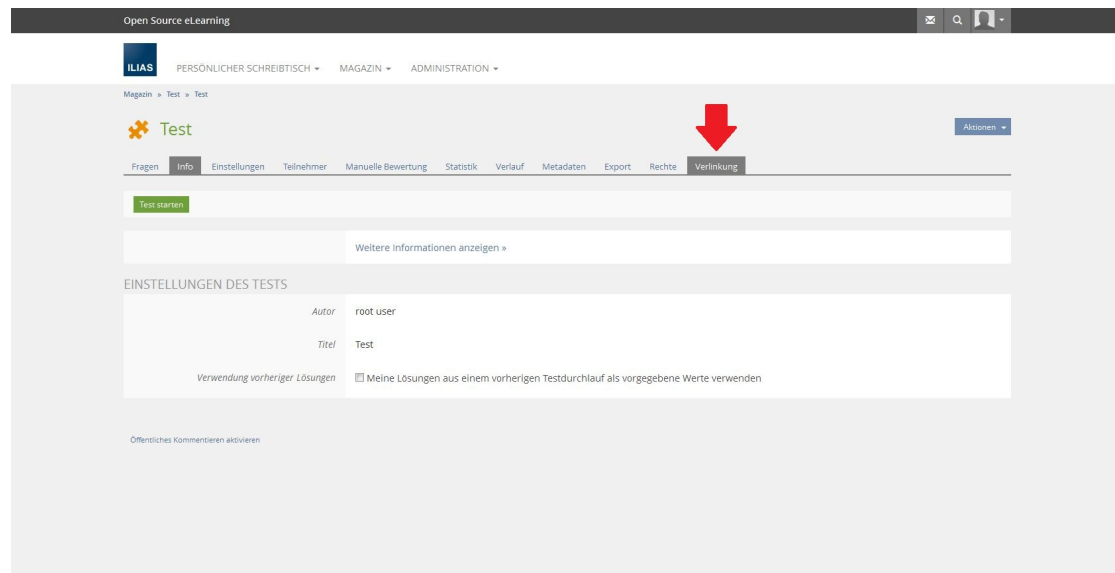


Abbildung 3.1: Ansicht Verlinkung Tab in Test

bisschen Text wo findet man den Tab

3.1 Verlinkung von Tests

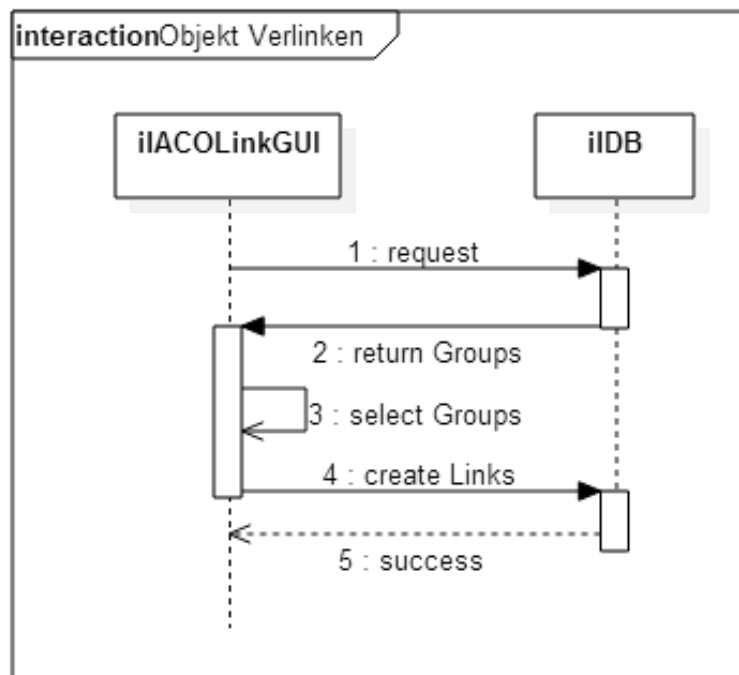


Abbildung 3.2: Sequenzdiagramm Test verlinken

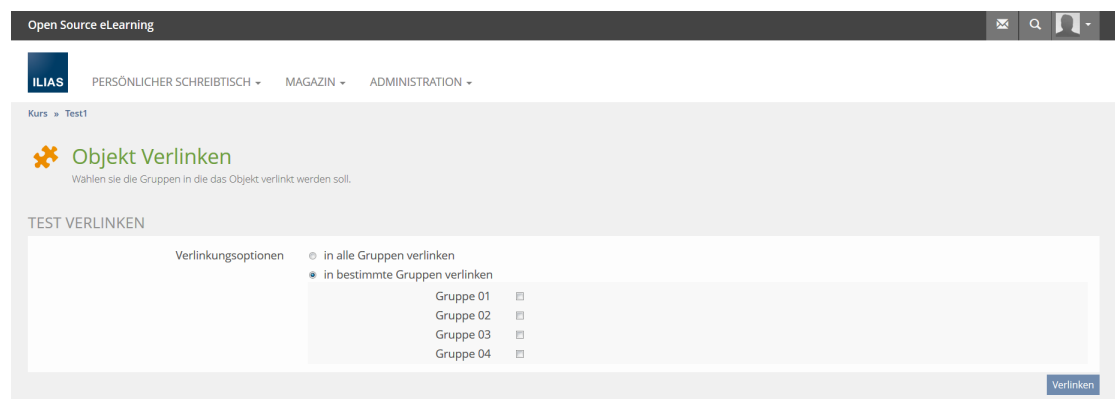


Abbildung 3.3: Ansicht Link Test

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

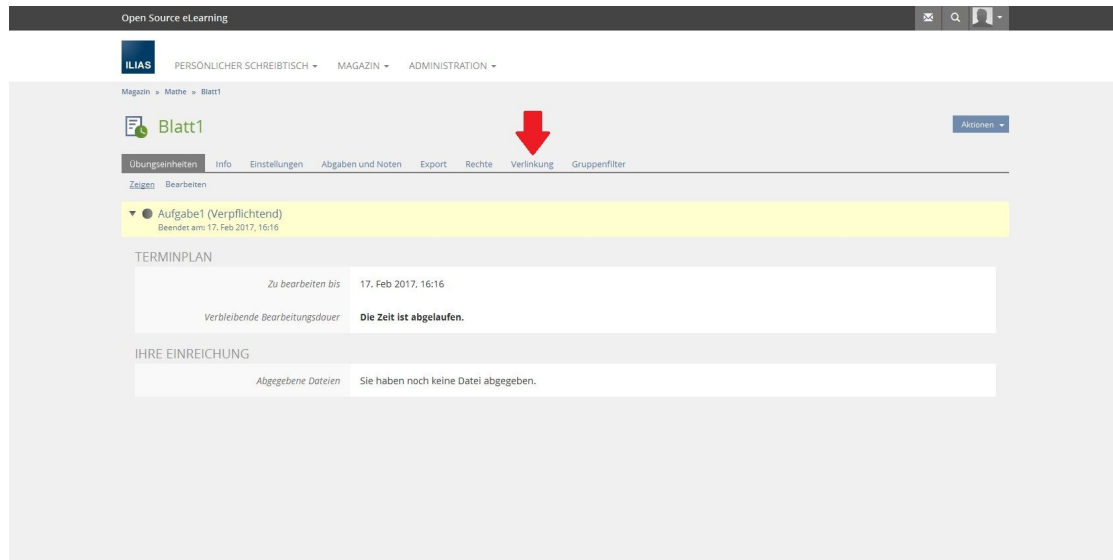


Abbildung 4.1: Ansicht Verlinkung in Exercise/Übung

Bisschen Text

4.1 Verlinkung von Übungen

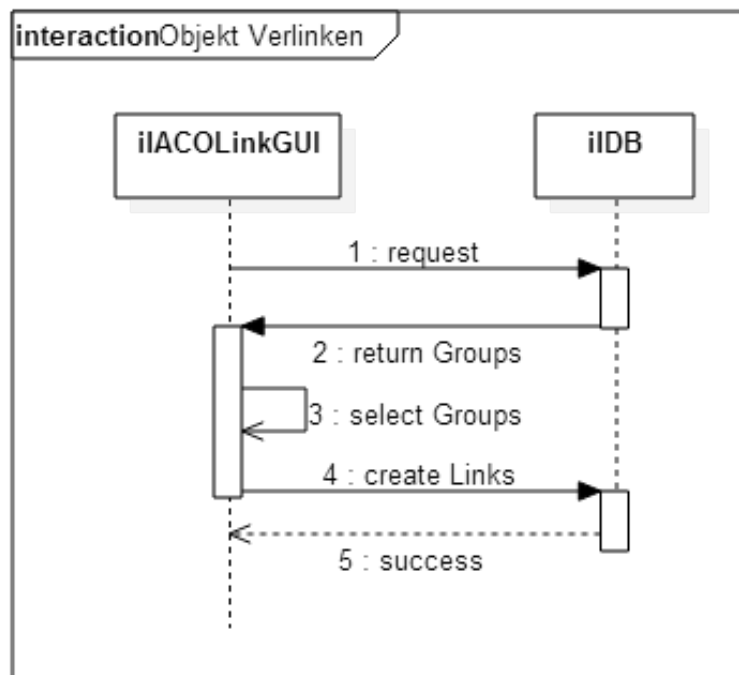


Abbildung 4.2: Sequenzdiagramm Übung verlinken

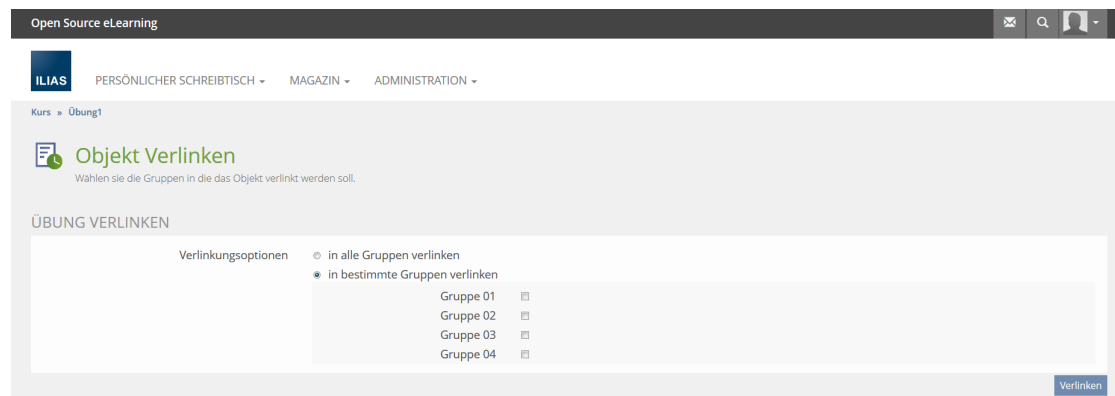


Abbildung 4.3: Ansicht Link Übung

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

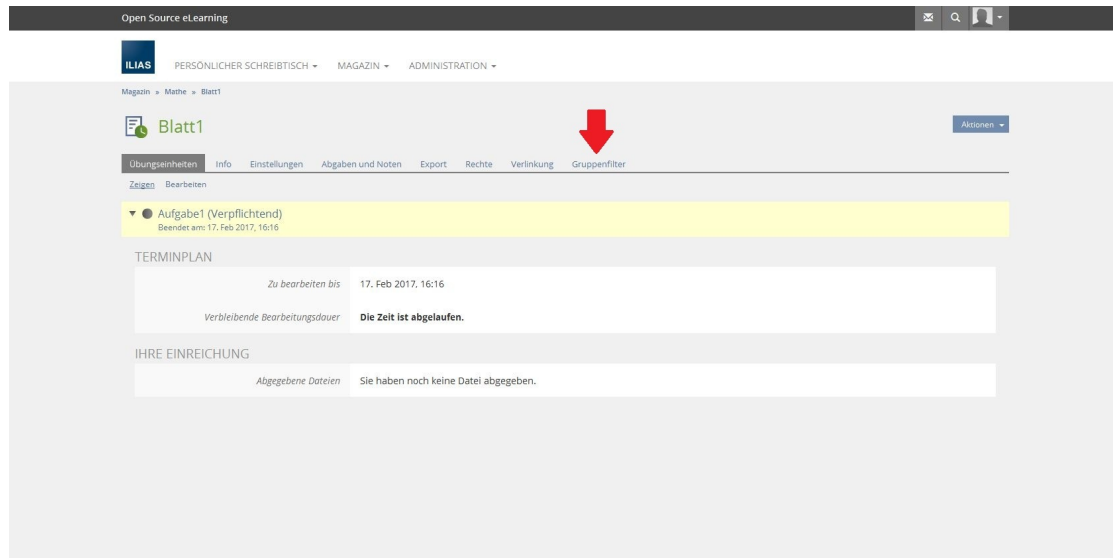


Abbildung 5.1: Ansicht Gruppenfilter Tab

Text wo finde ich den Tab

5.1 Gruppenfilter

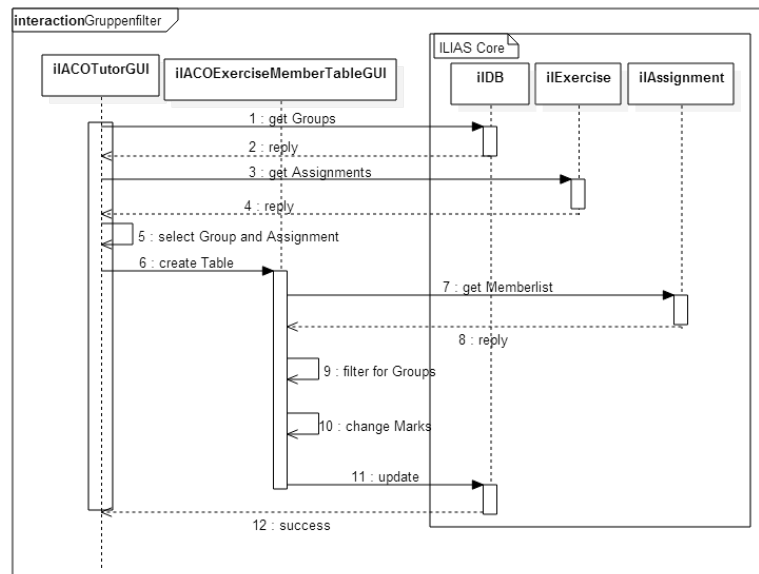


Abbildung 5.2: Sequenzdiagramm Gruppenfilter

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.

Unterpunkt

- Text Unterpunkt

6 Klassenbeschreibungen

Contents

6.1	ilACOEExerciseMemberTableGUI	24
6.2	ilACOGGroupDisplayGUI	27
6.3	ilACOGGroupGUI	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.

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

Tablogik: class.ilACOPlugin, class.ilACOUHookGUI

Gruppenerstellen: class.ilACOGGroupGUI

Gruppenfilter: class.ilACOTutorGUI

Verlinkung: class.ilACOLinkGUI, class.ilACOEExerciseMemberTableGUI, templates

Gruppen verwalten: class.ilACOGGroupDisplayGUI

Mitgliederverschieben: class.ilACOMemberGUI

Lang Files: lang

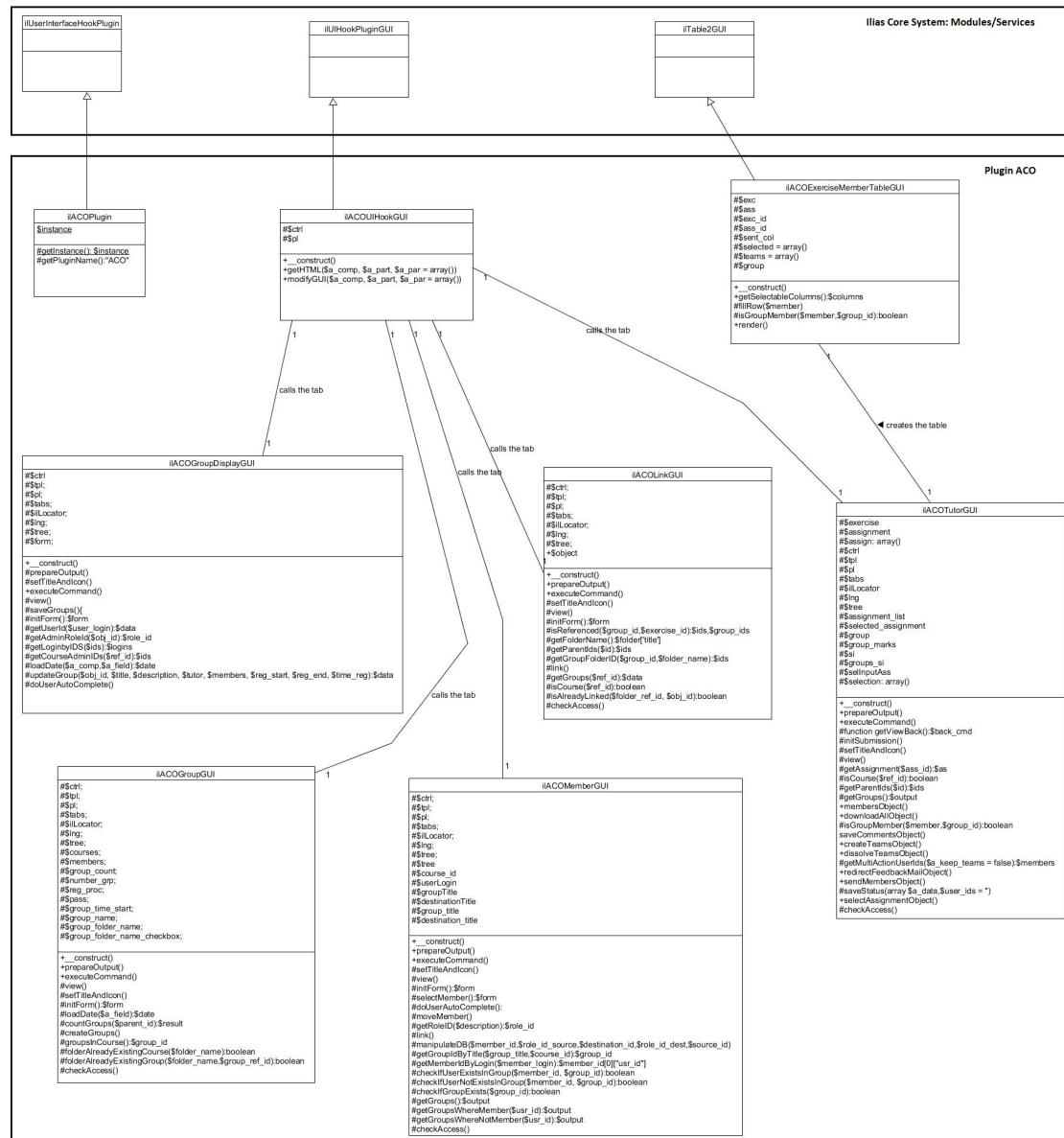


Abbildung 6.1: Klassendiagramm

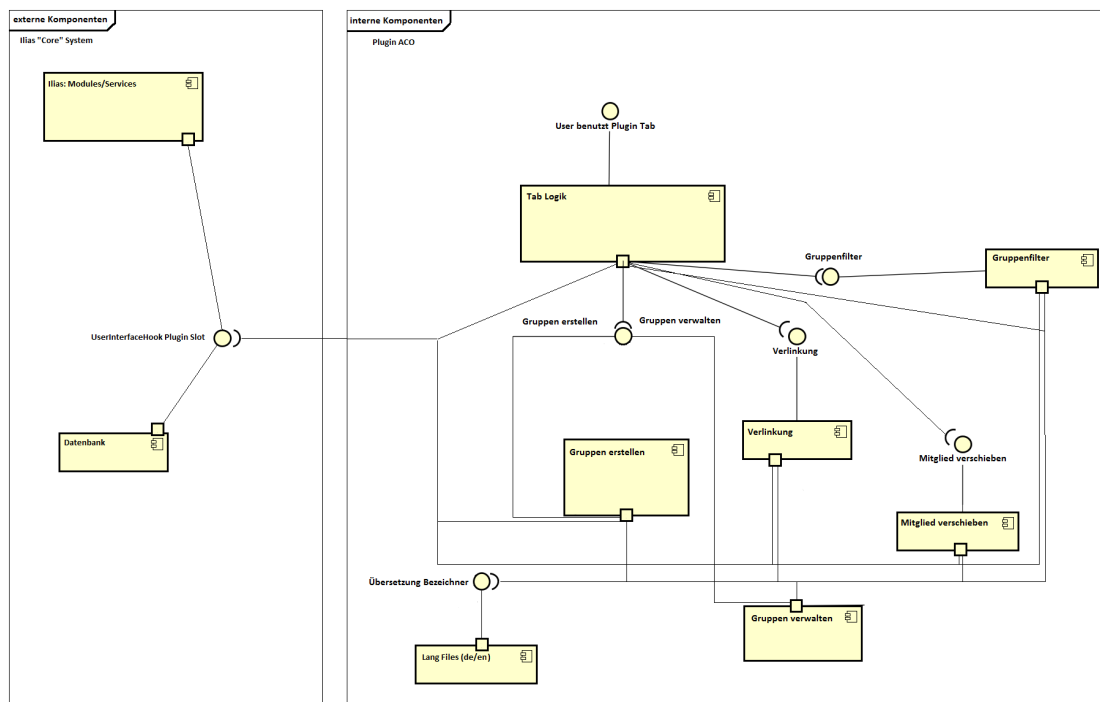


Abbildung 6.2: Komponentendiagramm

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: 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
- 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.

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.

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]["usr_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 -

VIEW_PARTICIPANT protected - type: Integer -

VIEW_GRADES protected - type: Integer -

exercise protected - type: ilObjExercise -

assignment protected - type: ilExAssignment -

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

assign public - type: Array -

assignment_list protected - type: -

selected_assignment protected - type: -

group protected - type: -

group_marks protected - type: -

si protected - type: -

`group_si` protected - type: -

`sellInputAss` protected - type: -

`selection` protected - type: Array -

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()

Rückgabewerte

back_cmd - , type: String

initSubmission

Beschreibung

initSubmission()

Rückgabewerte

new ilExSubmission - , type: ilExSubmission

view

Beschreibung

view()

Ausgabe der initialen Ansicht als HTML-Dokument

getAssignment

Beschreibung

getAssignment(\$ass_id)

Parameter-Liste

`as_id` - , type:

Rückgabewerte

`as` - , type:

isCourse

Beschreibung

isCourse(\$ref_id)

Parameter-Liste

`ref_id` - , type: Integer

Rückgabewerte

`(true,false)` - , type: Boolean

getParentIds

Beschreibung

getParentIds(\$id)

Parameter-Liste

`id` - , type:

Rückgabewerte

`ids` - , type:

getGroups

Beschreibung

getGroups()

Rückgabewerte

output - , type:

membersObject

Beschreibung

membersObject()

Rückgabewerte

- , type:

downloadAllObject

Beschreibung

downloadAllObject()

isGroupMember

Beschreibung

isGroupMember(\$member,\$group_id)

Parameter-Liste

member - , type:

group_id - , type:

Rückgabewerte

(true,false) - , type: Boolean

saveCommentsObject

Beschreibung

saveCommentsObject()

Rückgabewerte

- , type:

saveCommentForLearnersObject

Beschreibung

saveCommentForLearnersObject()

createTeamsObject

Beschreibung

createTeamsObject()

dissolveTeamsObject

Beschreibung

dissolveTeamsObject()

getMultiActionUserIds

Beschreibung

getMultiActionUserIds(\$a_keep_teams)

Parameter-Liste

\$a_keep_teams - , type: Boolean

Rückgabewerte

members - , type:

redirectFeedbackMailObject

Beschreibung

redirectFeedbackMailObject()

sendMembersObject

Beschreibung

```
sendMembersObject()
```

saveStatusAllObject

Beschreibung

```
saveStatusAllObject()
```

saveStatus

Beschreibung

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

Parameter-Liste

a_data - , type:

user_id - , type:

selectAssignmentObject

Beschreibung

```
selectAssignmentObject()
```

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

"ACO" - Name des Plugins, type: String