

Recherchebericht

Inhaltsverzeichnis

1. Glossar	2
1.1. Exportieren.....	2
1.2. Backup.....	2
1.3. Importieren.....	2
1.4. Wiki	2
1.5. Forum.....	2
1.6. Nutzerdaten.....	2
1.7. XML	2
1.8. OLAT-Core.....	3
1.9. Servlet.....	3
1.10. Erweiterungspunkt.....	3
2. Konzepte	3
2.1. Open Source	3
2.2. Modularer Aufbau.....	3
2.3. Schichtenkonzept.....	3
2.4. MVC.....	4
2.5. Java	4
2.6. Servlets	4
2.7. Rechtekonzept	5
2.8. Client-Server-Modell.....	5
3. Beschreibung der Applikation	5
3.1. Verzeichnisstruktur.....	5
3.2. Funktionalitäten.....	6
3.2.1. Privater (Nutzer) Bereich:	7
3.2.2. Kollaborativer Bereich und kooperative Aktivitäten:	7
3.2.3. Öffentlicher Bereich:	7
3.3. Erweiterbarkeit	7
3.4. Funktionseinschränkung	8
3.5. Rechtekonzept	9
4. Quellen	10

1. Glossar

1.1. Exportieren

Das Speichern von Daten in eine externe Datei. Gründe hierfür sind zum Beispiel: das Sichern von Daten (=> Backup) und/oder das Konvertieren und Einspeisen in ein anderes Dateisystem. Ein oft verwendetes Format für die Exportdatei ist hierbei XML (=> XML).

1.2. Backup

Eine Präventivmaßnahme um Datenverlust zu vermeiden. Hierbei wird eine Kopie der bestehenden Daten erstellt (=> Exportieren), welche so angelegt wird, dass sie im Falle des Verlustes des Originals wieder zurück gespielt werden kann (=> Importieren).

1.3. Importieren

Die Aufnahme von Daten einer Datei in ein System. Hierbei muss das Format der Datei kompatibel mit dem des Systems sein, d.h. das System muss die Datei lesen können.

1.4. Wiki

Eine Sammlung von Information im Internet, die durch die Benutzer selbst erweitert werden kann. Bei einer erfolglosen Suche in einem Wiki wird dem Benutzer angeboten selbst einen Artikel über den gesuchten Begriff zu verfassen. Das wohl bekannteste Wiki im WorldWideWeb ist Wikipedia. Im OLAT kann jede Rechte-, Lern- und Arbeitsgruppe über ein eigenes Wiki verfügen, welches von den Teilnehmern editiert werden kann.

1.5. Forum

Ein Forum ist eine Diskussionsrunde im Internet. Hier bei gibt es mehrere Themen (Threads) in denen jeder, der die Rechte dazu besitzt, eine Nachricht (Post) schreiben kann. Oft muss man sich, um in einem Forum etwas schreiben zu können, registrieren. Im OLAT kann jede Rechte-, Lern- und Arbeitsgruppe über ein eigenes Forum verfügen, welches von den Teilnehmern benutzt werden kann.

1.6. Nutzerdaten

Ein Nutzer ist die Darstellung einer realen Person (im OLAT). Dieser hat Informationen beim Registrieren oder im Nachhinein in sein Profil eingegeben, dies sind die Nutzerdaten. Der Umgang mit Nutzerdaten unterliegt, aufgrund des Datenschutzes, besonderer Vorsicht und Diskretion. Jedem OLAT-Benutzer stehen ein Kalender, ein persönlicher Ordner, Notizblätter und Bookmarks zur Verfügung. Zudem werden für jeden Nutzer Leistungsnachweise, seine erhaltenen Benachrichtigungen und seine persönlichen Einstellungen gespeichert.

1.7. XML

Extensible Markup Language ist eine plattformunabhängige Auszeichnungssprache. Sie dient zum Austausch und zur Darstellung strukturierter Daten. Im Gegensatz zum Beispiel zu HTML ist XML eine strikte Sprache, wird ein Zeichen („ ; “ ; < ; >) weggelassen so ist die gesamte Datei unbrauchbar.

1.8. OLAT-Core

Ziel ist es mit dem **Online Learning And Training** eine Organisations- und Lernplattform für Universitätsveranstaltungen wie Vorlesungen, Übungen, Seminare und Prüfungen bereit zu stellen. OLAT existiert bereits in der Version 6.1 und läuft auf verschiedenen Systemen, wie Linux und Windows.

Der OLAT-Core beschreibt nun das Grundgerüst des OLATs. Im Laufe der Entwicklung wurden nach und nach immer mehr Features hinzugefügt und das OLAT somit erweitert.

1.9. Servlet

Im Gegensatz zu einem Applet ist ein Servlet ein Java Programm, das nicht auf der Client- sondern auf der Serverseite läuft. Sie nehmen Anfragen von Clients entgegen und verarbeiten diese entsprechend. Hierbei ist hervorzuheben, dass Servlets gegenüber den alternativen CGI-Skripten oft schneller sind, da keine externen Programme gestartet werden müssen.

1.10. Erweiterungspunkt

Durch die Erweiterungspunkte, auch Extension-Points genannt, ist es möglich, dem OLAT zusätzliche Funktionen einzupflegen, ohne dabei den OLAT-Core (=> OLAT-Core) verändern zu müssen. Das OLAT kann so beliebig mit neuen Funktionalitäten erweitert werden, welche zum Beispiel an der Universität Leipzig entwickelt werden. Zum OLAT-Core gehören zum Beispiel auch das Wiki und das Forum.

2. Konzepte

2.1. Open Source

OLAT ist eine Open Source Software und unterliegt damit den folgenden Standards:

Der Quelltext liegt offen vor und ist lesbar.

Die Software darf kostenlos benutzt, kopiert und verbreitet werden.

Die Software darf verändert werden und in modifizierter Form verbreitet werden.

Somit entfallen die, im Gegensatz zur kommerziellen Softwarewelt, üblichen Lizenzgebühren. Berühmte Vertreter des Open Source Bereiches sind Linux, Mozilla und Apache, um nur einige zu nennen.

2.2. Modularer Aufbau

Ein Aufbau einer Software in verschiedene Module birgt viele Vorteile, nachfolgend werden nun einige genannt. Bereits geschriebene Module lassen sich wiederverwenden und ersparen somit Zeit und Arbeitsaufwand. Ebenso lässt sich die Arbeit nun einfacher in mehrere Gruppen aufteilen, da der Austausch der Module als Ganzes möglich ist. Zusätzliches kommt es auch zu Vereinfachungen in Planungs-, Definitions- und Entwicklungsphase, sowie auch in Tests der Module.

2.3. Schichtenkonzept

Im OLAT gibt es drei Schichten (Tiers).

Der User Tier wird mit JavaScript realisiert und ermöglicht das Benutzen des OLAT mit dem Browser. Diese Schicht läuft auf dem Client.

Der Business Tier steuert die Logik der Prozesse, verwaltet die Benutzerdaten und Benutzersitzungen und sorgt für die sichere Verwahrung der Datenbank und des Dateisystems.

Der Data Tier enthält die Datenbank und das Dateisystem. Hier finden Zugriffe via JDBC auf die, mit Hibernate abstrahierte, Datenbank statt, während das virtuelle Dateisystem den Zugriff auf das Dateisystem ermöglicht.

Data und Business Tier laufen auf einem oder mehreren separaten Servern.

2.4. MVC

Model-View-Controller – kurz MVC – ist ein Konzept der Programmierung bei der es um strikte Trennung von Anzeige (View), Eingabe (Controller) und Verarbeitung (Model) geht. Der Controller erhält aus dem View (meist eine Oberfläche) die Eingaben und leitet sie an das Model weiter. Hier finden die eigentlichen Berechnungen statt. Die Ergebnisse sind dann wiederum im View zu sehen. Dieses Dreiecks-konzept basiert auf Modularem Aufbau und ermöglicht so leicht Änderungen an den einzelnen Bausteinen.

2.5. Java

Eine objektorientierte Programmiersprache die von Sun Microsystems entwickelt wird und mittlerweile in Version 6 frei erhältlich ist. Die Portabilität ist eine der großen Vorteile der Sprache, so läuft Java auf Linux, Mac OS X, Windows und Solaris. Eine Besonderheit ist die Java Virtual Machine, welche den Bytecode interpretiert und bei Bedarf kompiliert und das System ebenfalls schützt, z.B.: da die Virtual Machine einen festen Arbeitsspeicher zugewiesen hat und dieser somit nicht voll laufen kann. Der Modulare Aufbau und die Erweiterbarkeit des OLAT basieren darauf, dass Java objektorientiert ist.

2.6. Servlets

Der Begriff Servlet ergibt sich aus der Wortkombination von Server und Applet und wird benutzt, um ein serverseitiges Applet zu beschreiben. Genauer gesagt bezeichnet man als Servlet, Java-Klassen, deren Instanzen innerhalb eines Java-Webserver Anfragen von Clients entgegennehmen und beantworten. Servlet-Anwendungen bieten die Möglichkeit zur Ausführung von Java-Programmen über einen WebServer. Diese Java-Programme (die Servlets) werden auf der Server-Seite ausgeführt, im Gegensatz zu Applets, die auf der Client-Seite, also auf dem Rechner des Endbenutzers ausgeführt werden. Der Inhalt der Antworten kann dabei dynamisch, also im Moment der Anfrage, erstellt werden und muss nicht bereits statisch für den Webserver verfügbar sein. Weiterhin sind Servlets fester Bestandteil aller Java-EE-Anwendungsserver. Sie stellen das Java-Pendant zu CGI-Skripten oder anderen Konzepten dar, mit denen dynamisch Web-Inhalte erstellt werden können. Zur Ausführung des JavaCodes stellen Systeme wie Tomcat / Apache eine entsprechende Umgebung bereit. Dabei handelt es sich um einen in Java geschriebenen Servlet-Container, der mithilfe des JSP-Compilers Jasper auch Java Server Pages in Servlets übersetzen und ausführen kann.

Um ein Servlet aufzurufen, fordert der Endbenutzer über seinen Browser eine URL an. Der WebServer ist so konfiguriert, dass er alle Aufrufe an ein bestimmtes Unterverzeichnis (/WEBINF) in einen Aufruf an die sog. Servlet-Engine umwandelt, die als ApplicationServer fungiert. Diese leitet den Aufruf an das Servlet weiter. Das Servlet interpretiert den Schlussteil der URL als Aufrufparameter und setzt sie in die Aktivierung eines bestimmten Dienstes um.

2.7. Rechtekonzept

OLAT stellt jedem Benutzer eine Anzahl von Rechten/Funktionen, abhängig von der Rolle in der sich der Benutzer befindet, zur Verfügung. Diese Rechte/Funktionen folgen einem Hierarchiekonzept, das vom einfachen Benutzer bis hin zum Administrator reicht. Einfachen Benutzern steht nur ein eingeschränkter Umfang der Funktionen, welche OLAT bereitstellt, zur Verfügung, während Administratoren den vollen Zugriff auf die angebotenen Möglichkeiten haben.

2.8. Client-Server-Modell

Besteht aus einem passiven Server der auf einen aktiven Client wartet, der sich mit ihm verbindet. Einmal verbunden können Daten innerhalb eines Netzwerkes ausgetauscht werden. Client und Server müssen nicht auf zwei verschiedenen Rechnern laufen, es ist auch möglich beide auf derselben Maschine laufen zu lassen. Beispiele für Anwendungen sind: Datenbankserver, Webserver, Mailserver und Applikationsserver.

3. Beschreibung der Applikation

OLAT (Online Learning And Training) ist ein webbasiertes Learning Management System (LMS), das seit 1999 an der Züricher Universität, in Form eines Open Source Projektes, entwickelt wird. Es bietet die Möglichkeit verschiedene Kontexte in Gruppen zusammenzufassen, zu organisieren und zu administrieren. Verschiedene Werkzeuge, wie Kalender, Wiki, Forum, usw. ermöglichen einen zentralen und effektiven Informationsaustausch. OLAT basiert größtenteils auf der Programmiersprache Java wodurch der Betrieb auf verschiedenen Betriebssystemen keiner größeren Anpassung bedarf. Die generierten Daten können persistent über ein Datenbanksystem, wie MySQL, Orakel, usw. gespeichert werden. Durch die Erweiterungspunkte bietet sich für Entwickler eine einfache Möglichkeit OLAT mit Funktionen und Komponenten zu erweitern, ohne dabei den OLAT-Core verändern zu müssen.

3.1. Verzeichnisstruktur

Die nachfolgende Auflistung umfasst alle Ordner die im Quellverzeichnis `PATH_TO_OLAT` (e.g. `/usr/local/olat`) des OLAT vorzufinden sind.

Inhalt und Funktion der Verzeichnisse, welche für den Betrieb einer OLAT-Instanz benötigt werden:

<code>/PATH_TO_OLAT/</code>	
<code>/bin</code>	Hilfsskripte(Shellscripte für Mac & Linux)
<code>/cluster</code>	Beinhaltet Scripte zum Clustering des OLAT ab Version 6.1 zur besseren Skalierbarkeit
<code>/conf</code>	Konfigurationsvorlagen z.B. für den Tomcat-Server oder für das Instant-Messaging
<code>/database</code>	Datenbank SQL Vorlagen zur Konfiguration der OLAT-Datenbank
<code>/monitoring</code>	Vorlagen für die Traffic-Überwachung(MRTG)
<code>/olatdata</code>	Enthält die OLAT-Laufzeitdaten
<code>/bcroot</code>	
<code> /course</code>	Beinhaltet Daten aller Kurse
<code> /cts</code>	Beinhaltet alle Gruppenordner und -werkzeuge
<code> /homepages</code>	Beinhalten alle Webseiten der Nutzer

/homes	Beinhaltet den persönlichen Ordner aller Nutzer
/repository	Beinhaltet alle angelegten Lernressourcen
/tmp	Temporäre Daten
/calendars	Beinhaltet alle Kalenderdaten
/customizing	Anpassungen z.B. Der Sprache
/logs	Log-Dateien
/monitoring	Überwachungsdaten werden hier gesammelt
/system	Installierte Upgrades sind hier zu finden
/tmp	Temporäre Daten
/interfaces	Vorlagen für den grafischen OLAT-Aufbau
/temp	Temporäre Daten
/webapp	Beinhaltet alles Wichtige an Webapplikationen
/examples	Beispiele für Kurse; OLAT-Demo
/help	OLAT-Hilfe
/static	Hier liegen wichtige Layoutdateien (Themes, CSS, Icons)
/WEB-INF	
/classes	Hier liegen die Java-Klassendateien
/lib	Speicherort einiger verwendeter Java-Bibliotheken

Inhalt und Funktion der Verzeichnisse, welche für die Entwicklung am OLAT wichtig sind:

/PATH_TO_OLAT/	
/webapp	
/WEB-INF	
/patchesSrc	Quelldateien für Patches
/src	Quelldateien des OLAT selbst

Sonstige Ordner:

/PATH_TO_OLAT/	
/doc	OLAT Dokumentation
/htdocs	Anpassbare Fehlerberichtseiten

Im Allgemeinen besteht eine einfache Servlet-Anwendung aus einer XML-Konfigurationsdatei, einem Ordner in dem sich die Java-Klassen befinden und einem Ordner mit benötigten Bibliotheken.

OLAT ist damit eine sehr umfangreiche und komplexe Servlet-Anwendung. Dies äußert sich zum einen darin, dass die Ordnerstruktur als solche umfangreicher ist als dies bei einer Standard-Servlet-Anwendung der Fall ist und zum anderen, dass auch eine Reihe von XML-Dateien für die Konfiguration und den Betrieb notwendig sind.

3.2. Funktionalitäten

OLAT bietet verschiedene Funktionalitäten, die in die drei verschiedenen Kategorien eingeteilt werden können. Der Private (Nutzer) Bereich stellt Werkzeuge und Managementkonzepte für die Daten dieser Sparte bereit. Im Kollaborativen Bereich ist es möglich Räume zu erzeugen und Daten auszutauschen um einen ungehinderten Informationsfluss für kooperative Aktivitäten zu erreichen. Im öffentlichen Bereich werden Information und Erfahrungen verfügbar gemacht, gespeichert und zur Diskussion freigegeben.

3.2.1. Privater (Nutzer) Bereich:

- Profil, Einstellungen und Passwort anpassen
- Portal planen und organisieren
- Kalender verwalten und Notizen führen
- Kurse bookmarken, Persönliche Ordner
- Persönliche Leistungen im besuchten Kurs anschauen
- Kurse starten, Einschreibungen in Veranstaltungen
- Administration von Gruppen und Benutzern
- Benachrichtigungen versenden über Ereignisse
- Systeminformationen abfragen
- Leistungsnachweise
- Notizen

3.2.2. Kollaborativer Bereich und kooperative Aktivitäten:

- Lerngruppen
- Rechtegruppen
- Arbeitsgruppen
- Gruppenkalender
- Mitgliederlisten anschauen
- Benachrichtigungsservice via E-Mail oder RSS, Chat, Wiki, Forum
- Ordner zum Austausch von Dateien

3.2.3. Öffentlicher Bereich:

- Administration
- Durch Gruppenmanagement Kurse verwalten, Kurskonfiguration Kurse einstellen
- Durch Rechtemanagement den Zugang zu Werkzeugen verteilen
- Datenarchivierung
- Durch Bewertungswerkzeug Punkte von Kursteilnehmern anschauen und editieren
- Visitenkarte
- Benutzersuche
- Chat

3.3. Erweiterbarkeit

Erweiterungen werden in OLAT über Extension-Points integriert, dies ermöglicht die Software kundenspezifisch anzupassen ohne dabei das Grundsystem verändern zu müssen. Dies ermöglicht eine einfache Einbindung von Updates. Die Erweiterungen werden als .jar Paket in den Ordner WEB-INF/lib abgelegt und in der `olat_extensions.xml` definiert. Geladen werden sie vom OLAT-Extension-Manager (`org.olat.core.extensions.ExtManager`). Identifiziert werden die Erweiterungen über eine ID sowie über ihre Klasse bzw. über ihren PackageNamen.

Folgende Extension-Points sind definiert:

Interface:	<code>org.olat.extensions.globalmapper</code>
Erweiterungspunkt:	<code>org.olat.dispatcher.DispatcherAction</code>
Klassen:	<code>org.olat.dispatcher.DispatcherAction</code>
Beschreibung:	Die Erweiterung kann als Mapper bezeichnet werden und kennt den Pfad mit welchem er assoziiert wurde.

Interface:	org.olat.extensions.action.ActionExtensionorg. olat.extensions.globalmapper
Erweiterungspunkt:	org.olat.home.HomeMainController
Klassen:	org.olat.dispatcher.DispatcherAction
Beschreibung:	Die Erweiterung bietet einen Link mit Text an und eine Beschreibung und definiert was beim anklicken des Links geschehen soll.
Interface:	org.olat.extensions.css.CSSIncluder
Erweiterungspunkt:	org.olat.gui.components.Window
Klassen:	org.olat.gui.css.CSSGenerator
Beschreibung:	Neue CSS-Stylesheets
Interface:	org.olat.extensions.hibernate.HibernateConfigurator
Erweiterungspunkt:	org.olat.persistence.DB
Klassen:	org.olat.persistence.DB
Beschreibung:	Die Erweiterung kann zusätzlich hibernate mappings enthalten. Es sei bemerkt, dass sich das Erstellen von Tabellen als schwierig erweist.
Interface:	org.olat.extensions.sitescreator.SitesCreator
Erweiterungspunkt:	org.olat.gui.control.generic.dtabs.DTabs
Klassen:	org.olat.FullChiefController
Beschreibung:	Neue Seiten; muss SitesCreator enthalten, der Liste von SiteDefinition Objekten enthält

3.4. Funktionseinschränkung

Das Erzeugen einer OLAT-Instanz ohne den Tab „Lernressourcen“ kann folgendermaßen bewerkstelligt werden:

- Öffnen der Datei „/webapp/WEB-INF/olat_extensions.xml“
- Entfernen bzw. Auskommentieren der Zeile:

```
<bean id="olatsites_repository"
      class="org.olat.repository.site.RepositorySiteDef"
      singleton="false" />
```
- Neustart von OLAT bzw. Apache Tomcat

Das Erzeugen einer initialen OLAT-Instanz ohne die Demokurse vom Autor „sroth“ kann folgendermaßen bewerkstelligt werden:

- Öffnen der Datei „/webapp/WEB-INF/olat_config.xml.in“
- Entfernen bzw. Auskommentieren der Zeilen:

```
<CourseExportFile access="4">
  examples/Course_template_big.zip</CourseExportFile>
<CourseExportFile access="4">
  examples/Course_template_small.zip</CourseExportFile>
<CourseExportFile access="4">
  examples/Demo_course_with_questionnaire.zip</CourseExportFile>
<CourseExportFile access="4">
  examples/Kursvorlage_gross.zip</CourseExportFile>
<CourseExportFile access="4">
  examples/Kursvorlage_klein.zip</CourseExportFile>
<CourseExportFile access="4">
```



```
examples/Demo_course_enrolment.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demo_course_enrol_to_topics.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demo_course_wiki.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demokurs_Einschreibung.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demokurs_Wiki.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demokurs_mit_Fragebogen.zip</CourseExportFile>  
<CourseExportFile access="4">  
examples/Demokurs_Themeneinschreibung.zip</CourseExportFile>
```

- OLAT normal (weiter) installieren

3.5. Rechtekonzept

Das OLAT-System stellt drei verschiedenen Arten von Gruppen zur Verfügung.

Lerngruppen werden in Kursen verwendet um Personen aus administrativen oder pädagogischen Gründen zu gruppieren, z.B. um Ihnen Lernmaterialien in einem geschützten Bereich zugänglich zu machen.

Rechtegruppen werden in Kursen verwendet um Personen gezielt spezielle Rechte innerhalb eines Kurses zuzuteilen, z.B. um das Bewertungswerkzeug zu bedienen.

Arbeitsgruppen können von allen Benutzern selbst erstellt werden um z.B. gemeinsam an einem Projekt zu arbeiten oder Dokumente auszutauschen. Sie haben keinen Zusammenhang zu einem Kurs.

Diese Gruppentypen sind auf einem Businessgroup-Konzept aufgebaut. Die Businessgroup bildet in diesem Zusammenhang eine Klasse, von der die drei Gruppentypen abgeleitet und mit unterschiedlichen Konfigurationen ausgestattet werden.

Alle Gruppen verfügen über Werkzeuge die gemeinsam benutzt werden können, um Diskussionen zu führen oder um an Dokumenten zu arbeiten. Gruppen die in einem Kurs zusammengefasst werden, bilden einen Gruppenkontext der immer eine Beziehung zu einem oder mehreren Kursen hat. Es ist möglich kursinterne wie auch kursübergreifende Gruppen zu modellieren oder mit einem Gruppenkontext von einem Kurs zum nächsten zu ziehen (Abbildung von Klassen).

SecurityGroup:

Mit der Erzeugung von einer Lernressource, z.B. einem Kurs, wird automatisch eine SecurityGroup für diese Ressource erzeugt, der Erzeuger wird in die Gruppe hinzugefügt. Jeder in der Gruppe kann dann die Ressource verwalten.

BusinessGroup:

ist ein Konzept für eine Gruppe, die etwas Gemeinsames macht und kollaborativ aktiv ist. BusinessGroup hat einen Namen und Beschreibung und implementiert eine Anzahl von Funktionen, die nützlich für eine Zusammenarbeit sein können.

Dem Benutzer eines OLAT-Systems wird eine Systemrolle zugewiesen. Diese Rollen gliedern sich in verschiedene Hierarchiestufen, die jeweils verschiedene Funktionen/Rechte zur Verfügung stellen.

Folgende Systemrollen sind in einem OLAT-System vorhanden.

Gäste sind anonyme Benutzer mit eingeschränkten Rechten. Gäste können ihre Benutzeroberfläche nicht anpassen und weder Tests absolvieren noch Forenbeiträge verfassen.

Benutzer können ihre Benutzeroberfläche selber gestalten und einen Kurs als Teilnehmer starten.

Autoren können zusätzlich zu den Berechtigungen eines Benutzers Lernressourcen erstellen, kopieren, archivieren, löschen etc.

Gruppenverwalter können kursübergreifende Lern- und Rechtegruppen verwalten.

Benutzerverwalter können neue Benutzer erstellen oder importieren und ihnen Rollen zuweisen.

Administratoren sind zu administrativen Tätigkeiten im gesamten OLAT-System berechtigt. Dies beinhaltet ebenfalls die Rechte aller anderen Systemrollen.

4. Quellen

Kundengespräch vom 09.04.2009

Lokale OLAT-Installation und OLAT-Dokumentation

<http://www.frentix.com/de/olat/docu/OLAT-Systemdokumentation.pdf>

http://www.boscheri.ch/computer/java/javainse15/javainse17_002.htm

<http://www.wikipedia.org> (Glossarbegriffe und Konzepte)

<http://www.w3c.de/Misc/XML-in-10-points.html>

<http://pcai042.informatik.uni-leipzig.de/swp/SWP-08/index.html>