

ENTWURFSBESCHREIBUNG

KiPMan

Kursverwaltung mit integriertem **P**rüfung**s**managment

Autor: GR-07-3

Leiter: Marvin Frommhold

Home: pcai042.informatik.uni-leipzig.de/~gr-07-3/

Datum: 21. Mai 2007

Version: 1.0

Inhaltsverzeichnis

1	Allgemeines	3
2	Produktübersicht	3
3	Grundsätzliche Struktur- und Entwurfsprinzipien für das Gesamtsystem	4
3.1	Architekturübersicht	4
3.1.1	Der View	5
3.1.2	Die Controller	5
3.1.3	Das Modell	5
3.1.4	Die Datenhaltung	5
3.2	Aufteilung der Systemkomponenten auf die Pakete	5
4	Grundsätzliche Struktur- und Entwurfsprinzipien der einzelnen Pakete	6
4.1	de.kipman.*	6
4.2	de.kipman.controller.*	6
4.3	de.kipman.exams.*	6
4.4	de.kipman.exams.editor.*	7
4.5	de.kipman.exams.io.*	7
4.6	de.kipman.usermanager.*	8
4.7	Ergänzungen	8

1 Allgemeines

KiPMan stellt eine Erweiterung des bereits existierenden und lauffähigen Online-Learning-Management-Systems OLAT (Online Learning and Training) dar. Es ist eine Prüfungsverwaltung, die es Dozenten einer Hochschule ermöglicht, Prüfungen anzulegen, zu denen sich Studenten einschreiben können.

KiPMan ist eine Webanwendung, welche mit Servlets arbeitet. Daher wird ein Servlet-Container (z.B. Tomcat) benötigt, um diese ausführen zu können.

2 Produktübersicht

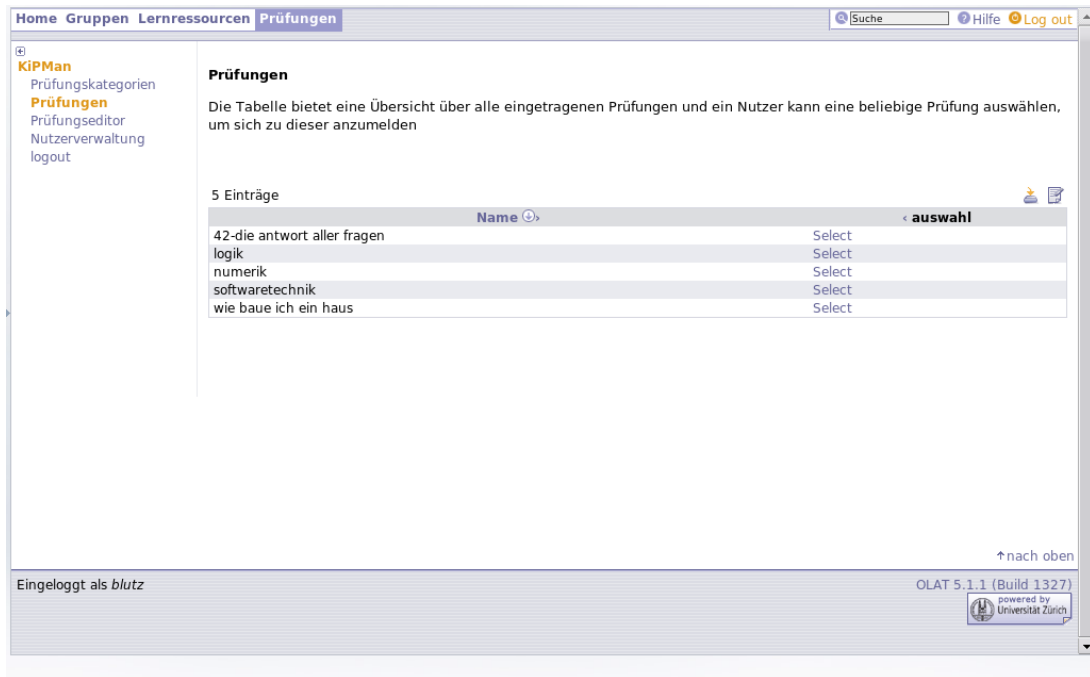
Die Funktionalität differenziert je nach Rolle des Nutzers. Wir unterscheiden zwischen den folgenden Rollen: Student, Prüfer und Prüfungsamt. Dabei ist der Student ein ganz normaler Benutzer, so wie OLAT einen registrierten Nutzer definiert. Die Rolle Prüfungsamt kann neue Benutzer, über die schon vorhandenen Funktionen von OLAT, anlegen oder importieren und ihnen neue Rechte im Sinne der Rollen Prüfer und Prüfungsamt geben. Diese Rollen werden über die Zugehörigkeit zu einer OLAT-SecurityGroup realisiert.

Die folgenden Abbildungen sollen einen Überblick über die Einbindung von KiPMan in OLAT geben:

The screenshot displays the OLAT web application interface. At the top, there is a navigation bar with 'Home', 'Gruppen', 'Lernressourcen', and 'Prüfungen'. Below this, a search bar and 'Hilfe' and 'Log out' links are visible. The main content area is titled 'Willkommen bei OLAT' and includes a 'Zugang zum KiPMan(Prüfungsverwaltung)' link. The interface is divided into several sections:

- OLAT Schnellstart Links:** A list of links for 'Kurse', 'Katalog', 'Hilfe', 'Einstellungen', and 'Arbeitsgruppe' with brief descriptions.
- Wussten Sie...?:** A section titled '... dass OLAT Open Source ist?' providing information about OLAT's open-source nature and where to find more details.
- Macartney Comic: Die Loge:** A section featuring comic strips from 'Macartney Comic'.
- Right-side Widgets:**
 - Heute: 19.05.2007:** A calendar widget showing 'Keine Termine im angegebenen Zeitraum'.
 - Meine Bookmarks:** A section stating 'Sie haben keine Bookmarks erstellt'.
 - Meine Gruppen:** A section stating 'Sie sind in keiner Gruppe'.
 - Meine Benachrichtigungen:** A section stating 'Sie haben keine Benachrichtigungen'.

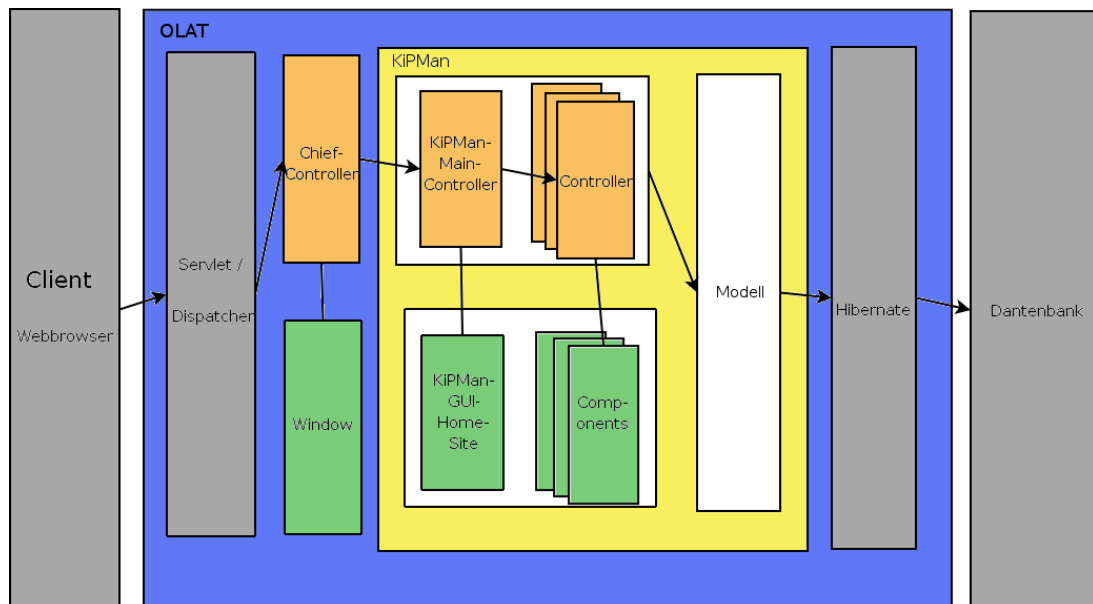
The URL at the bottom of the browser window is `http://localhost:8080/olat/auth/1%3A8%3A167%3A0%3A0%3AAcid%3At4/`.



Das Menü wird je nach Rolle des Nutzers an die jeweiligen zur Verfügung stehenden Funktionen angepasst.

3 Grundsätzliche Struktur- und Entwurfsprinzipien für das Gesamtsystem

3.1 Architekturübersicht



KIPMan wird als neue Seite (mit Hilfe des Paketes *org.olat.core.gui.control.navigation.SiteInstance*)

in OLAT eingebunden. Dies wird durch einbinden einer neuen OLAT-Extension über einen Eintrag in der Datei *olat_extensions.xml* realisiert.

Es wird strikt zwischen Darstellungs-, Ablaufs- und Datenerhaltungslogik getrennt. Dabei wird soweit wie möglich auf schon vorhandene Pakete von OLAT zurückgegriffen.

3.1.1 Der View

Besonders die Realisierung des View wird vorrangig mit durch OLAT definierten Elementen geschehen, um eine einheitliche Oberfläche zu ermöglichen. So benutzen wir die Velocity-Schnittstellen von OLAT (*org.olat.core.gui.components.velocity.VelocityContainer*) und stellen die Oberfläche über Velocity-Schablonen und mit Hilfe von Controllern dar, die den Velocity-Containern die Schablonen und darzustellenden Elemente übergeben.

3.1.2 Die Controller

Die Klasse *de.kipman.MainController* ist für die Initialisierung von KiPMan zuständig. Es baut die Eröffnungsseite auf und übergibt den einzelnen Komponenten weitere Controller, die für die Ablauflogik verantwortlich sind.

3.1.3 Das Modell

Enthält die Klassen, die für die einzelnen Geschäftsprozesse zuständig sind. Dabei handelt es sich um das Prüfungs- und Nutzermanagement, sowie das Erstellen von Prüfungen.

3.1.4 Die Datenhaltung

Die Nutzerdaten werden in der schon vorhandenen OLAT-Datenbank gespeichert. Für die Prüfungsdaten und die Daten der Prüfungskategorien werden neue Tabellen angelegt. Dazu wird auf das Hibernate-Framework zurückgegriffen.

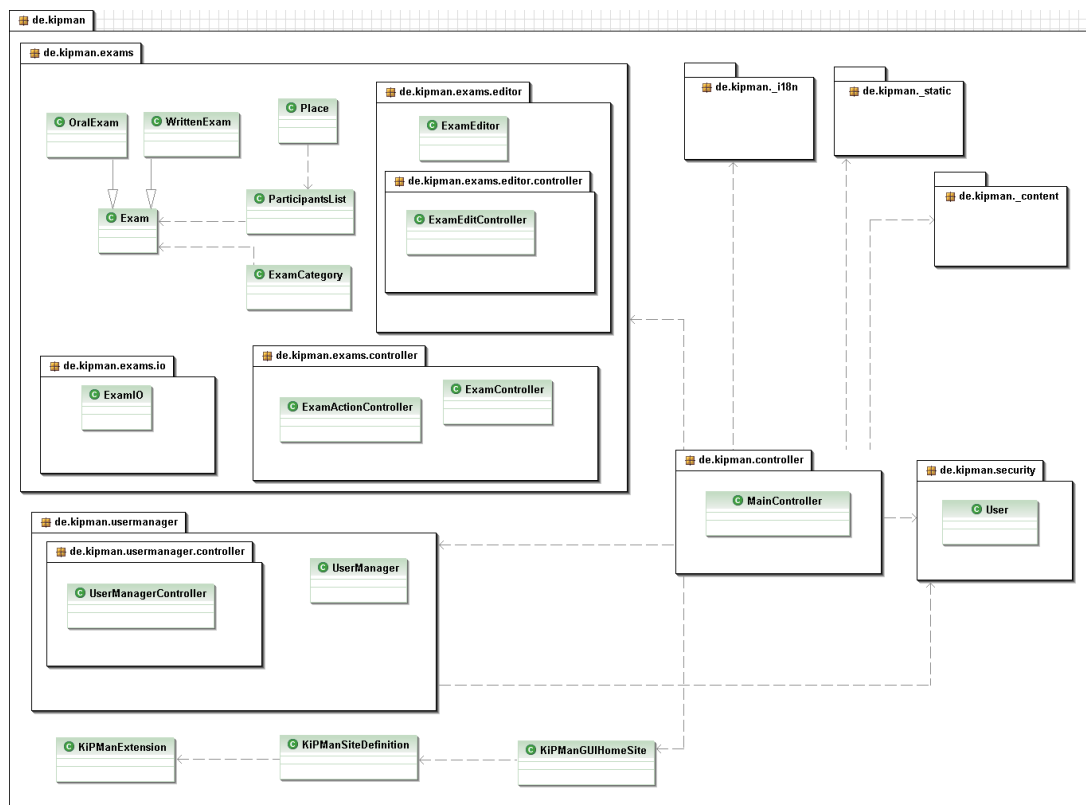
3.2 Aufteilung der Systemkomponenten auf die Pakete

Die einzelnen Systemkomponenten wurden folgendermaßen auf die Pakete aufgeteilt:

- View: *de.kipman._content*
- Controller: *de.kipman.controller*, *de.kipman.exams.controller*, *de.kipman.usermanage.controller*
- Modell: *de.kipman*, *de.kipman.exams*, *de.kipman.security*, *de.kipman.usermanage*, *de.kipman.exams.editor*
- Datenhaltung: *de.kipman.exam.io*

Vorerst werden alle Velocity-Schablonen im Paket *de.kipman._content* sein. Es wird aber wahrscheinlich während der Implementierungsphase auch für alle anderen Pakete die Unterordner **._content* geben, wohin die Velocity-Schablonen abgelegt werden. Auch die Unterordner **._i18n* und **._static* können noch eingefügt werden um sich an die Struktur und Namensgebung der OLAT-Vorgaben zu halten.

Folgend eine Übersicht über die Pakete:



4 Grundsätzliche Struktur- und Entwurfsprinzipien der einzelnen Pakete

4.1 de.kipman.*

Die Klassen im Hauptpaket bilden die Integration der Prüfungsverwaltung in das OLAT-System. *KiPManExtension.java* ist von *org.olat.core.extensions.Extension* abgeleitet und wird von OLAT aufgerufen. Die Klassen *KiPManSiteDefinition.java* und *KiPManGuiHomeSite.java* erzeugen den Reiter bzw. die Menüstruktur (siehe 2. Produktübersicht) und sind von *org.olat.core.gui.control.navigation.SiteDefinition/SiteInstance* abgeleitet.

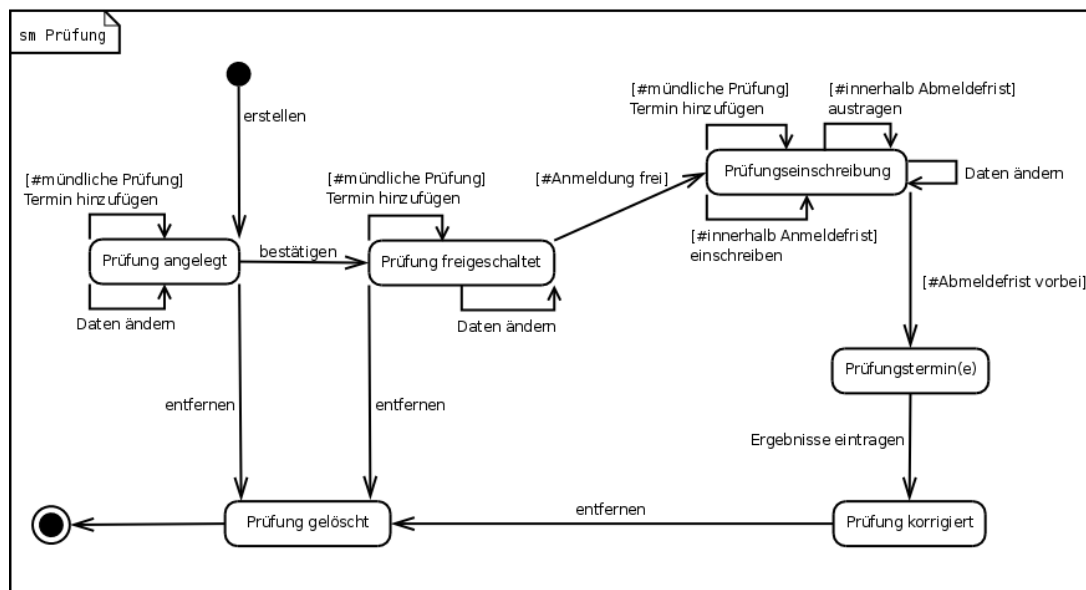
4.2 de.kipman.controller.*

Die Klasse *MainController.java* stellt den Haupt-Controller dar, der die Startseite und Menüsteuerung der Prüfungsverwaltung steuert. Er initialisiert ebenfalls bei Bedarf die weiteren Controller der einzelnen Unterpakete. Dieser ist abgeleitet von dem OLAT-Interface *org.olat.gui.control.DefaultController*.

4.3 de.kipman.exams.*

Dieses Paket stellt alle Klassen, die zur Verwaltung von Prüfungen benötigt werden, zur Verfügung. So bietet zum Beispiel die Klasse *Exam.java* die Methoden zum Einschreiben (*addParticipant()*) und Austragen (*removeParticipant()*) eines Teilnehmers an. Von dieser werden die beiden

Klassen *OralExam.java* und *WrittenExam.java* abgeleitet, welche weitere spezielle Methoden und Attribute besitzen.



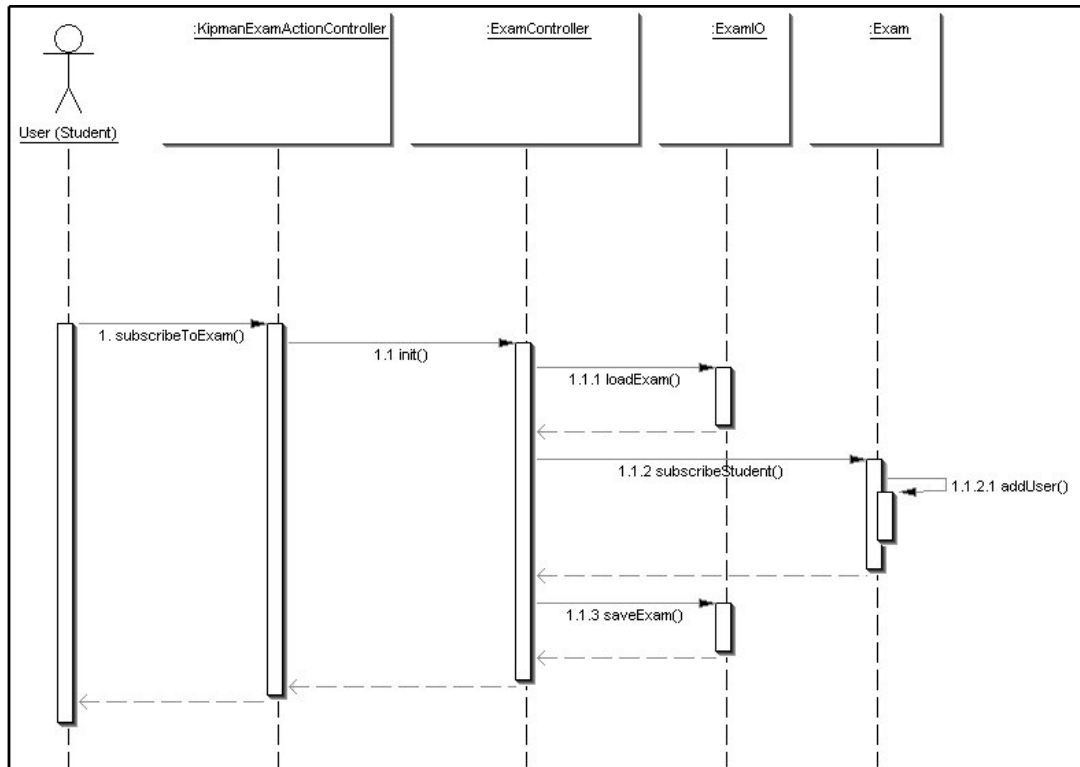
Es werden in dem Diagramm alle grundsätzlichen Funktionen angezeigt, die auf ein Objekt Prüfung zugreifen. Auf eine Unterscheidung der einzelnen Rollen und deren nutzbare Funktionen wurde nicht genauer eingegangen.

4.4 de.kipman.exams.editor.*

Die Klasse *ExamEditor.java* dient zum Verwalten der Prüfungen (erzeugen/löschen/bearbeiten).

4.5 de.kipman.exams.io.*

In diesem Paket werden die Klassen zur persistenten Datenhaltung bereitgestellt. Prüfungen können dauerhaft gespeichert werden. Die Klasse *ExamIO.java* kommuniziert über das Hibernate-Framework, welches OLAT bereitstellt, mit der Datenbank. Sie liest die benötigten Informationen aus und stellt diese in geeigneten Formaten/Objekten zur Verfügung bzw. speichert relevante Daten.



Dieses Sequenzdiagramm veranschaulicht die Interaktion und den Austausch von Daten zwischen den einzelnen Klassen, die bei einer Einschreibung genutzt werden.

4.6 de.kipman.usermanager.*

Das Paket dient zur Verwaltung der Nutzer. Mit Hilfe der Klasse *UserManager.java* können die Nutzer verschiedenen SecurityGroups hinzugefügt bzw. entfernt werden. Des Weiteren können neue Nutzeraccounts angelegt werden.

4.7 Ergänzungen

Alle Pakete werden wie schon erwähnt noch weitere Unterpakete und Klassen enthalten, um die Einbindung in OLAT zu gewährleisten. Dazu gehören die Controller, welche die Aktionen der Nutzer entgegen nehmen und weiterleiten. Sowie die Pakete für die Übersetzungen (*.i18n) und für die Velocity-Schablonen (*.content und *.static).