

Softwaretechnikpraktikum 2011

SWP 11-2 Projektleiter: Matthias Haeßner | Dokument erstellt von Axel Fischer

Entwurfsbeschreibung des Vorprojekts**Gruppe SWP 11-2****1. Allgemeines**

Das Produkt ist eine in Java realisierte Erweiterung des OLAT-Learning Management System in Version 7.0. Es erweitert OLAT um die Funktionalität, dass während des Authentifizierungsprozesses geprüft wird, ob der sich einloggende Nutzer in seinem Profil eine Matrikelnummer eingegeben hat und fordert diesen ggf. über einen modalen Dialog auf, diese nachzutragen.

2. Produktübersicht

Das Produkt überprüft während des Authentifizierungsprozesses ob der sich einloggende Nutzer in seinem Profil eine Matrikelnummer eingetragen hat. Es ergeben sich somit die folgenden zwei Fälle:

- 1.) Der Nutzer hat eine Matrikelnummer eingetragen: dann erfolgt keine weitere Aktion.
- 2.) Der Nutzer hat keine Matrikelnummer eingetragen: der Nutzer wird über einen modalen Dialog aufgefordert diese nachzutragen.

3. Grundsätzliche Struktur- und Entwurfsprinzipien für das Gesamtsystem

Ein Klassendiagramm unseres Produkts befindet sich auf der folgenden Seite.

Das Produkt wurde so entworfen, dass die Koppelung zwischen OLAT-LMS und der Erweiterung möglichst schwach ist.

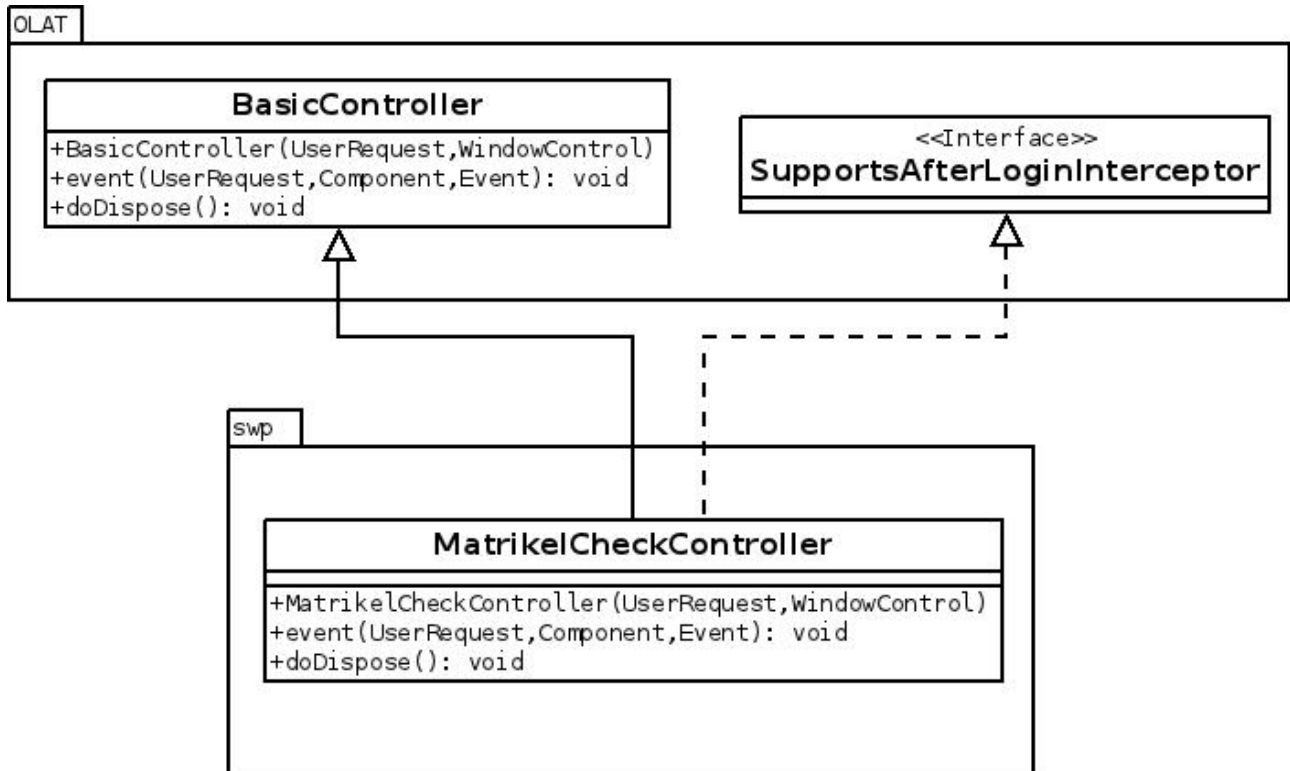
Die Zielsetzung stellt einen Workflow direkt nach dem Login dar. Somit bot sich die Benutzung des Erweiterungspunktes `org.olat.login.AfterLoginInterceptorController` an. Durch die Implementierung des Produkts als OLAT-Extension wird zudem die Trennung des Produkts vom OLAT-Code erreicht.

Hierbei wird dem Erweiterungspunkt ein Controller hinzugefügt, welcher dem Nutzer direkt nach dem Login präsentiert wird. Dies erfolgt über einen Eintrag in der Datei `UserContext.xml`, welche die Konfigurationen der Controller hält, die dem Nutzer direkt nach dem Login präsentiert werden.

OLAT stellt für den Controller diverse Controller-Klassen zur Verfügung.

Softwaretechnikpraktikum 2011

SWP 11-2 Projektleiter: Matthias Haeßner | Dokument erstellt von Axel Fischer



4. Grundsätzliche Struktur- und Entwurfsprinzipien der Pakete

Unser Paket swp stellt den gewünschten workflow nach dem Login dar.

Durch einen entsprechenden Eintrag in `UserContext.xml` an der Stelle `afterLoginConfig`, wird dem `AfterLoginInterceptionManager` eine `AfterLoginConfig` zu `MatrielCheckController` hinzugefügt.

`org.olat.core.gui.control.creator.AutoCreator` erzeugt dann bei Bedarf via Reflection eine Instanz von `MatrielCheckController`.

Der Konstruktor von `MatrielCheckController` erhält hierbei Referenzen auf `UserRequest` und `WindowControl`.

Über `UserRequest` ist es dann möglich die Properties des Nutzers auszulesen, welche in eine `HashMap` hinterlegt sind. Der Controller zur Darstellung der Dialogbox wird mit Hilfe der entsprechenden Funktion in `BasicController` erzeugt.

Diese Implementierung erzeugt in dieser Form zwei Fenster, zum einen den gewünschten Dialog wie auch einen zweiten Dialog welcher im Konstruktor von `AfterLoginInterceptionController` erzeugt wird. Da es keine Möglichkeit gibt diesen zweiten Dialog zu modifizieren müssen die entsprechenden Zeilen

```

cmc = new CloseableModalController(getWindowControl(), translate("close"), vC, true, translate("runonce.title"));
cmc.activate();
listenTo(cmc);
  
```

auskommentiert werden, um sein Erscheinen zu unterbinden.