

DMS



Ari Ayvazyan & Samuel Schmidt

Ein Dokumenten Management System (kurz DMS genannt) erlaubt das zentrale Speichern von beliebigen Dokumenten. Dokumente können somit gezielt an einem Platz gesucht und administriert werden. Eine zentrale Aufgabe eines DMS ist es den Verlauf eines Dokuments aufzuzeichnen und jederzeit abrufen zu können.

4AHIT – TGM – INSY

06.12.2014

Aufgabenstellung

Dokumenten Management System:

Ein Dokumenten Management System (kurz DMS genannt) erlaubt das zentrale Speichern von beliebigen Dokumenten. Dokumente können somit gezielt an einem Platz gesucht und administriert werden. Eine zentrale Aufgabe eines DMS ist es den Verlauf eines Dokuments aufzuzeichnen und jederzeit abrufen zu können.

Suche / Indizierung:

Ein DMS ist nur so gut, wie seine Suchfunktion. Es soll daher möglich sein nach folgenden Parametern zu suchen.

- Autor
- Kategorie
- Kommentar
- Dokumentname
- Dokumenttyp
- Schlüsselwörter

Diese Parameter beschreiben auch die Eigenschaften eines Dokuments in dem DMS.

Authentifikation / Autorisierung:

Bei dem DMS soll es sich um ein rollenbasiertes System handeln. Folgende Rollen sollen im System implementiert werden:

Administrator:

Der Administrator hat alle Rechte und kann auf alle Dokumente zugreifen.

Dokumentbesitzer:

Jener Benutzer der ein Dokument im DMS erstmalig erfasst, wird als Dokument Besitzer vermerkt.

Dokumentnutzer: Der Administrator und der Dokumentbesitzer kann beliebigen andere Benutzer den Zugriff auf das Dokument gewähren.

Zur Vereinfachung wird beim Zugriff nicht zwischen Lese- und Schreibrechten unterschieden, sprich Zugriff auf ein Dokument bedeutet Lese- und Schreibzugriff.

Verlauf:

Für jedes Dokument in dem DMS soll mit einer Versionsnummer versehen und gespeichert werden. Jede Änderung des Dokuments führt dazu, dass die Versionsnummer um eins erhöht wird. Alle Änderungen werden mit folgenden Parameter im DMS gespeichert:

- Versionsnummer
- Benutzer
- Datum / Uhrzeit
- Kommentar
- Upload / Download:

Das DMS soll in dieser Version folgende Aktionen erlauben:

Upload:

Ein neues bzw. eine neue Version eines Dokuments werden im DMS abgelegt und die Versionsnummer wird um eins erhöht. Ebenso wird der Verlauf um diese Aktion erweitert. Wenn das Dokument zuvor von dem Benutzer heruntergeladen wurde, so führt der Upload zu einer Freigabe des Dokuments, wodurch anderen Dokumentennutzer das Dokument bearbeiten können. Es kann immer nur ein Dokument hochgeladen werden. Ein Hochladen mehrerer Dokumente bzw. ganzer Verzeichnisstrukturen sind in der nächsten Ausbaustufe angedacht.

Checkout / Download:

Ein Checkout eines Dokuments führt gleichzeitig dazu, dass das Dokument im DMS als GESPERRT vermerkt wird. Diese Sperre gilt für alle anderen Benutzer und kann nur von dem Dokumentnutzer durch einen UPLOAD einer neuen Version bzw. mit Hilfe der GUI durch den Dokumentbesitzer bzw. Administrator freigegeben werden.

Löschen

Ein bestehendes Dokument kann nur gelöscht werden, wenn es nicht gesperrt ist. Das Löschen des Dokuments erfolgt auch physisch und führt dazu das alle Einträge im DMS (Bsp. Verlauf, Dokumentbenutzer, etc.) gelöscht werden.

Erstelle mit Hilfe der Frameworks JEE oder Play eine Webapplikation, die die Funktionalität dieses Dokumentenmanagementsystems abbildet. Verwende das ORM Framework Hibernate um die Daten des Dokuments in einer Datenbank abzuspeichern. Führe zu Beginn der Arbeit eine ausführliche Analyse & Designphase durch, um die Problem noch vor der Implementierung mit den Projektmitgliedern abzuklären.

Bewertungskriterien:

Domänenmodell (PFLICHT!): 4 Punkte

Authentifikation & Autorisierung (PFLICHT!): 4 Punkte

DMS Funktionen:

Suche: 2 Punkte

Upload: 2 Punkte

Checkout: 2 Punkte

Löschen: 1 Punkte

Anzeige und Speicherung des Verlaufs: 1 Punkte

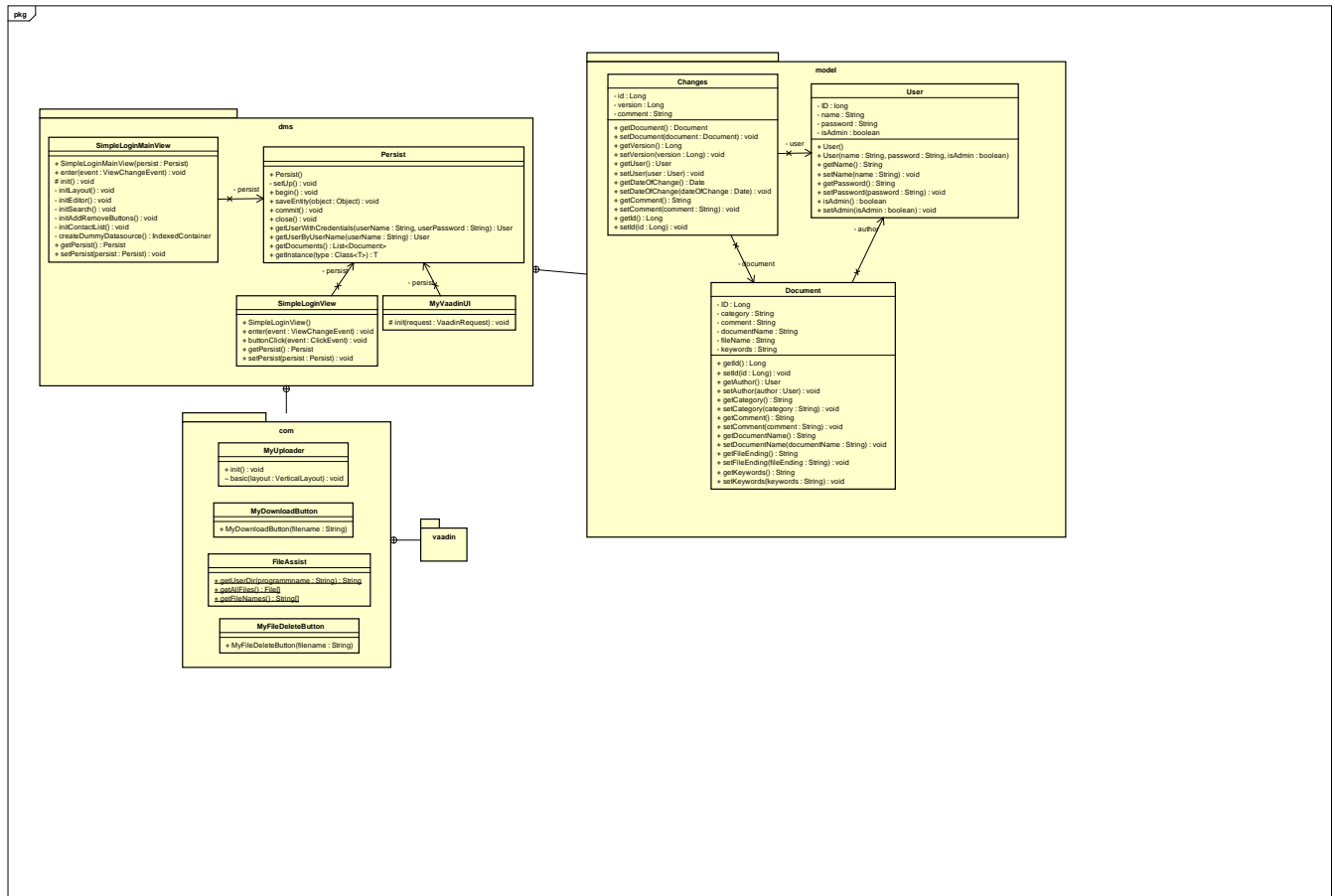
Gruppengröße: 2 Personen

Aufwandschätzung + Endzeitaufteilung

<i>Aufgabe</i>	<i>Beschreibung</i>	<i>Aufwandschätzung</i>	<i>Endzeitaufteilung</i>
Domainmodell	Konzept von DMS erstellen, (neu mit Vaadin)	1h	2h
GUI	Bis jetzt noch nie mit Vaadin gearbeitet aber leichter als erwartet.	3h	2h
Hibernate User/Document	Modelklassen erstellt	1h	1h
Authentifikation/Autorisierung	Mit Mapping gut funktioniert	1h	1h
Upload/Download	Am Server, nicht in Hibernate	2h	3h
Summe		8h	9h

Arbeitsdurchführung

Konzept erstellt:



Mapping von User und Document in Hibernate gemacht und ausprobiert und gedebuggt.
Mit unserem Java EE Web Framework Vaadin vertraut gemacht und den Login implementiert.
Eine Table ohne Anbindung zu Hibernate (erstellen von Dokumenten usw.) implementiert
Den Upload/Download hinzugefügt.
GUI mit auslesen von Hibernate DB verbunden.

