

SE2, Aufgabenblatt 6 (2 Termine)

Modul: Softwareentwicklung II – Sommersemester 2016

Entwurf im Kleinen, Fachliche Aufgabenverteilung, Klassenschnittstellen erweitern

CommSy-Projektraum SE2 CommSy SoSe 2016

Ausgabedatum 13. Mai 2016

Vormerken-Funktion in die Mediathek einbauen

Die Nutzer der Mediathek wünschen eine Möglichkeit zum Vormerken von Medien. Ein Interview hat folgende **Anforderungen** ergeben:

- a) **Jedes Medium kann von maximal drei unterschiedlichen Kunden vorgemerkt werden.**
- b) **Medien können unabhängig von ihrem Verleihstatus vorgemerkt werden.**
- c) **Ein Ausleiher kann ein von ihm ausgeliehenes Medium nicht vormerken.**
- d) **Ein vorgemerktes Medium kann nur an den ersten Vormerker ausgeliehen werden.**
- e) **Wird ein vorgemerktes Medium ausgeliehen, wandern weitere Vormerker auf der zweiten und dritten Positionen um je einen Platz nach vorne.**

(Termin 1) Aufgabe 6.1 Lösungsvorschlag erarbeiten (1-1,5 Stunden)

Erarbeitet euch zuerst den Ist-Zustand des Ausgangssystems Mediathek_Vorlage_Blatt06. Hier gibt es bereits eine Benutzungsschnittstelle, die die gewünschte Funktionalität repräsentiert: das *Vormerkwerkzeug*. Dieses Werkzeug enthält noch einige ToDos, da die benötigten, fachlichen Methoden noch nicht existieren.

Entwickelt danach einen Lösungsvorschlag für die von den Nutzern gestellten Anforderungen, den ihr in eurem Übungsraum in Aufgabe 6.2 präsentiert.

Euer Lösungsvorschlag soll die folgenden Fragen beantworten:

- Welche Informationen gehören zu einer Vormerkung?
- Wie werden diese Informationen intern abgelegt? Welche Objekte bilden eine Vormerkung ab?
- Welche Klassen sind an dem Vorgang „Vormerken“ beteiligt?
- Welche Aufgaben und welche Schnittstellen haben diese Klassen?
 - Nur die relevanten Teile der Schnittstellen sollen dargestellt werden.
 - Welche Parameter und Rückgabetypen haben die benötigten Operationen?
- Welche Programmschritte werden ausgeführt, wenn
 - ein Medium vorgemerkt wird.
 - ein vorgemerktes Medium ausgeliehen wird.
- Welche Vorbedingungen und Nachbedingungen gelten hierbei?

Haltet eure Ideen in einem UML-Klassendiagramm fest. Dabei sollten die für diese Aufgabe relevanten Operationen auf jeden Fall in das Klassendiagramm aufgenommen werden. Hierfür empfehlen wir euch die Software *VioletUML*, die bereits auf den Rechnern installiert ist.

(Termin 1) Aufgabe 6.2 Lösungsvorschlag präsentieren (1 Stunde)

Präsentiert euren Lösungsvorschlag im Übungsraum. Hierfür stehen einer ausgewählten Gruppe etwa 20 Minuten zur Verfügung. Die von euch erstellten Entwürfe müssen dabei sinnvoll aufbereitet sein. Bereitet die Präsentation so vor, dass ihr die Fragestellungen aus 6.1. erläutern könnt.

Die Präsentation hat das Ziel, Feedback zu euren Ideen einzuholen. Es ist also wichtig, dass eure Zuhörer euren Entwurf nachvollziehen können. Außerdem sollten Zwischenergebnisse von Diskussionen durch euch explizit festgehalten werden; beispielsweise direkt an der Tafel oder durch das Verändern von euren Klassendiagrammen.

(Termin 1) Aufgabe 6.3 Lösungsvorschlag überarbeiten (0,5-1 Stunde)

Überarbeitet euren eigenen Entwurf auf Basis der Präsentationen aller Gruppen. Das Ergebnis dieser Aufgabe muss so ausführlich sein, dass ihr euch in der nächsten Woche an die Umsetzung machen könnt. Nutzt die unbetreute Übungszeit um mit der Implementierung zu beginnen.

(Termin 2) Aufgabe 6.4 Lösung fertig implementieren (1,5 Stunden)

Implementiert euren Lösungsvorschlag vom Termin 1. Haltet dabei die vorgegebenen softwaretechnischen Qualitätsmerkmale ein:

- ☐ Test
- ☐ Vertragsmodell
- ☐ Schnittstellenkommentare
- ☐ Quelltextkonventionen

Hinweis: Um den Überblick zu behalten, kann es sinnvoll sein, die Anforderungspunkte nacheinander umzusetzen.

Zusatzaufgabe: Baut die Möglichkeit zum Stornieren einer Vormerkung in die Mediathek ein.

(Termin 2) Aufgabe 6.5 Lösung vorstellen/Code Review (1 Stunde)

Stellt eure Lösung in eurem Übungsraum vor. Hierfür stehen einer ausgewählten Gruppe etwa 20 Minuten zur Verfügung. Es eignen sich zuerst eine Präsentation des laufenden Programms und danach ein Blick auf den Quelltext am Beamer. Eventuell sollte auch euer anfänglicher Lösungsvorschlag wieder für alle Zuhörer sichtbar sein. Beantwortet dabei folgende Fragen:

- Welche Klassen habt ihr neu implementiert oder erweitert?
- Warum stehen Klassen in der von euch gewählten Beziehung zueinander?
- Konntet ihr euren Lösungsvorschlag 1 zu 1 umsetzen, oder gab es Änderungen aufgrund von unvorhergesehenen Abhängigkeiten? Hatte euer Entwurf auf weitere Klassen, die nicht direkt an dieser neuen Funktionalität beteiligt sind, Auswirkungen?

Es ist das Ziel dieser Aufgabe, die Qualität des von euch geschriebenen Quelltexts zu verbessern.
