

Übungsblatt 10 (07.01.2020)

Kontrollflussgraph, Codeüberdeckung, Systemtests, Anforderungen

In dieser Übung:

- ✓ Erstellen Sie einen Kontrollflussgraphen.
- ✓ Überprüfen und erhöhen Sie die Testcode-Abdeckung.
- ✓ Definieren Sie Systemtests und führen diese aus.
- ✓ Dokumentieren Sie Anforderungen zu einer neuen Funktionalität des *Movie Managers*.

Code Coverage in Android Studio

Zum Überprüfen der Codeüberdeckung durch Testfälle verwenden Sie in Android Studio die Option „Run with Coverage“, um ihre Testfälle zu starten.

Aufgabe 10.1: Analyse der bestehenden Movie Manager App (Teil 15): White-Box-Test: Kontrollflussgraph & Anweisungsüberdeckung

Präsenz: Nein	Punkte:11	Team: Ja	Testat
---------------	-----------	----------	--------

In dieser Aufgabe erstellen Sie den Kontrollflussgraph zu `calculateOverallRating(Movie m)` und überprüfen Sie die Überdeckung des Codes durch die Komponententests. Nutzen Sie für diese Aufgabe das Android-Studio Projekt aus Aufgabe 9.2.

- 1) Erstellen Sie in einem PDF-Dokument einen Kontrollflussgraph für den Code der Operation `calculateOverallRating(Movie m)` der Klasse `RatingUtils` im Package `de.moviemanager.util` im Ordner `main`. Verwenden Sie dazu den hier angegebenen und etwas angepassten Code. Beschriften Sie die Knoten im Graph mit den relevanten Zeilennummern des Codes.

```
1 public static double calculateOverallRating(Movie m) {
2     List<Performer> performers = storage.getLinkedPerformersOfMovie(m);
3     double movieRating = m.rating();
4     double result = m.rating();
5     if(!performers.isEmpty() && m.isRated()) {
6         DoubleSummaryStatistics stats = new DoubleSummaryStatistics();
7         for (Performer p : performers) {
8             if (p.isRated()) {
9                 double rating = p.rating();
10                stats.accept(rating);
11            }
12        }
13        if(stats.getCount() > 0) {
14            double performersRating = stats.getAverage();
15            result = (movieRating + performersRating) / 2.0;
16        }
17    }
18    return result;
19 }
```

- 2) Führen Sie zunächst nur die bereits vorgegebenen Tests aus Aufgabe 9.2 unter Verwendung der Option „Run with Coverage“ aus. Vergleichen Sie die überdeckten Codezeilen mit Ihrem Kontrollflussgraph und färben Sie alle Knoten, die durch diese JUnit-Tests überdeckt werden, in Ihrem Kontrollflussgraphen ein (Menge KB). Beschreiben Sie in einem PDF-Dokument, inwieweit durch KB Anweisungsüberdeckung erreicht wird. Führen Sie nun auch Ihre selbst implementierten Tests aus Aufgabe 9.2. unter Verwendung der Option „Run with Coverage“ aus. Bestimmen Sie KB neu auf Basis aller 4 Tests und beschreiben Sie, ob sich dadurch etwas an KB ändert.
- 3) Überlegen Sie mithilfe des Kontrollflussgraphen, welche Testfälle nötig sind für eine 100%ige Anweisungsüberdeckung. Beschreiben Sie diese Testfälle durch Angabe geeigneter Pfade, d.h. der Knoten und Kanten in Ihrem PDF-Dokument. Geben Sie geeignete Testdaten an, so dass diese Pfade ausgeführt werden. Überlegen Sie außerdem, welche Testfälle nötig sind für eine 100%ige Zweigüberdeckung. Beschreiben Sie diese Testfälle durch Angabe geeigneter Pfade in Ihrem PDF-Dokument. Geben Sie geeignete Testdaten an, so dass diese Pfade ausgeführt werden. Beschreiben Sie, welchen Black-Box-Testfällen aus Aufgabe 9.1.2 Ihre Pfade entsprechen.

Ergebnis:

Speichern Sie bitte eine .zip-Datei bis **Montag 13.01.2020 um 10.00 Uhr** in Moodle bestehend aus:

- PDF-Dokument mit Kontrollflussgraph
- PDF-Dokument mit den Testfallmengen und der Unterschiedsbeschreibung

**Aufgabe 10.2: Analyse der bestehenden Movie Manager App (Teil 16):
Systemtests**

Präsenz: Nein	Punkte: 5	Team: Nein	Testat
----------------------	------------------	-------------------	---------------

Im Jira-Projekt zum *Movie Manager* finden Sie die Beschreibung der Systemfunktion *Unlink Performer*. Überlegen Sie, welche wichtigen Systemtestfälle für diese Funktion benötigt werden. Überlegen Sie dabei entsprechend des Black-Box-Tests, welche wichtigen Unterschiede es bzgl. des Ergebnisses geben kann und beschreiben Sie die entsprechenden Äquivalenzklassen in einem PDF-Dokument. Betrachten Sie dabei insbesondere auch die Ausnahmen, in denen aufgrund einer NutzerInnenaktion die Funktion nicht erfolgreich ausgeführt wird. Beschreiben Sie dann geeignete logische und konkrete Systemtestfälle und tragen Sie diese in die Excel-Tabelle *10-MovieManager-Systemtests.xlsx* (Datei zu finden in Moodle) ein. In der Datei ist bereits ein Systemtest zu einer **anderen** Systemfunktion als Beispiel vorgegeben.

Starten Sie nun den *Movie Manager*. Führen Sie anschließend Ihre spezifizierten Systemtests aus und protokollieren Sie die konkret getesteten Daten, sowie das Ergebnis in der Excel-Tabelle.

Ergebnis:

Speichern Sie bitte ihre Ergebnisse als .zip-Datei bis **Montag 13.01.2020 um 10.00 Uhr** in Moodle, bestehend aus

- Excel-Datei *10-MovieManager-Systemtests.xlsx* mit Ihren Testfällen (logisch und konkret) und Ihrem Testprotokoll
- PDF-Dokument mit den Äquivalenzklassen.

Erweiterung der *Movie Manager* App

Auf diesem und den nachfolgenden Übungsblättern werden Sie die *Movie Manager* App um eine Verwaltung von **Filmverleihdaten** erweitern. Die nachfolgende Aufgabe auf diesem Blatt und die Aufgaben auf den nächsten Blättern hängen daher sehr stark zusammen.

Beachten Sie, dass alle Teammitglieder alle Schritte durchführen. Zum einen sollten Sie sich gemeinsam auf die Anforderungen und die Umsetzungen einigen. Zum anderen sollte jeder die Beschreibungstechniken und Vorgehensweisen üben, da Sie diese im Blockprojekt und in der Klausur anwenden.

In der nachfolgenden Aufgabe 10.3 und in Aufgabe 11.1 werden Sie zunächst die Anforderungen für die Umsetzung der neuen Funktionalität dokumentieren.

Szenario und Anforderungen zur Verwaltung von Filmverleihdaten

Die Firma *Movie Experience* hat durch eine öffentlich zugängliche Befragung auf der Produktwebseite des *Movie Managers* die Funktionalität zur Verwaltung von Filmverleihdaten als wichtiges neues Feature identifiziert. Um die NutzerInnen des *Movie Managers* bei der Pflege besser zu unterstützen, werden Sie im Rahmen der Übungsblätter 10 bis 13 die nachfolgenden **Anforderungen an die Verwaltung von Filmverleihdaten** umsetzen:

1. Als NutzerIn kann ich im *Movie Manager* vermerken, dass ein Film verliehen ist. Dabei gilt, dass
 - a. Filme bis zu einem selbstgewählten Datum verliehen werden.
 - b. Das gewählte Datum der Filmrückgabe nicht in der Vergangenheit liegen darf.
 - c. Ein bereits verliehener Film erst wieder verliehen werden kann, wenn der Film zuvor zurückgegeben wurde.
2. Als NutzerIn kann ich im *Movie Manager* vermerken, dass ein Film zurückgegeben wurde.
3. Als NutzerIn kann ich im *Movie Manager* das Rückgabedatum ändern, aber nicht vor das aktuelle Datum zurücksetzen.
4. Als NutzerIn kann ich im *Movie Manager* durch Sortierung in einer Übersicht schnell sehen, welche Filme verliehen sind. Dabei gilt, dass
 - a. Verliehene Filme nach dem Rückgabedatum sowohl aufsteigend als auch absteigend sortiert werden können
5. Als NutzerIn kann ich im *Movie Manager* durch Sortierung in einer Übersicht schnell sehen, welche und wie viele ausgeliehene Filme überfällig sind.
6. Als NutzerIn kann ich im *Movie Manager* in der Filmliste und bei jedem einzelnen Film sehen, ob ein Film ausgeliehen ist und ob er überfällig ist.
7. Als NutzerIn kann ich im *Movie Manager* nur Filme als gesehen vermerken, die nicht verliehen sind

Aufgabe 10.3: Anforderungen neue Funktionalität (1)

Präsenz: Nein	Punkte: 12	Team: Ja	Testat
----------------------	-------------------	-----------------	---------------

In dieser Aufgabe dokumentieren Sie die funktionalen Anforderungen für die Verwaltung von Filmverleihdaten des *Movie Managers* in Form von User-Tasks und Domänendaten. Zudem legen Sie die Systemfunktionen, die durch das neue Feature entstehen, fest und ordnen sie den Workspaces in der UI-Struktur zu. Führen Sie die nachfolgenden Teilaufgaben aufeinander abgestimmt aus.

- 1) Lesen Sie die Anforderungsbeschreibung im blauen Kasten und überlegen Sie sich, welchen Teilaufgaben (Sub-Task) eines/einer NutzerIn das entspricht und welche Systemfunktionen zur

Umsetzung nötig sind. Erweitern Sie die Beschreibung der User-Task in der Datei *10-MovieManager-UserTask.docx* (Datei zu finden in Moodle) durch entsprechende neue Sub-Tasks, sodass die neu unterstützten Teilaufgaben und dazu mögliche unterstützende Funktionen deutlich werden.

- 2) Erweitern Sie das vorhandene Domänendatenmodell in der Datei *10-MovieManager-DomainData.png* (Datei zu finden in Moodle) um die in den Sub-Tasks neu genannten Daten. Begründen Sie in einem PDF-Dokument, aus welchem Text der sub-task-Beschreibung Sie neue Daten übernehmen. Sie können das Diagramm entweder per Hand ergänzen und dann einscannen oder Sie verwenden ein Zeichenwerkzeug ihrer Wahl, z.B. Paint, Visio, draw.io.
- 3) Schauen Sie sich die vorhandenen Systemfunktionen in Jira und das UI-Strukturdiagramm in der Datei *10-MovieManager-UI-Structure.png* (Datei zu finden in Moodle) an. Beschreiben Sie anschließend im gleichen Detailgrad Ihre Systemfunktionen für das neue Feature näher. Benutzen Sie dazu die bereitgestellte Tabelle *10-MovieManager-Systemfunctions.xlsx* (Datei zu finden in Moodle). Geben Sie bei der Beschreibung der Systemfunktion an, welche Anforderung (identifiziert durch die Nummer) durch diese Systemfunktion umgesetzt wird. Beachten Sie, dass aufgrund der neuen Anforderungen auch Änderungen an der Beschreibung existierender Systemfunktionen nötig sind. Übernehmen Sie die Beschreibung dieser Systemfunktion aus Jira in die Tabelle und ergänzen Sie diese entsprechend.
- 4) Orientieren Sie sich bei der Spezifikation der Systemfunktionen an dem Beispiel aus der Vorlesung (vgl. Foliensatz 3) und gestalten Sie gleichzeitig die Workspaces im UI-Struktur-Diagramm. Ergänzen Sie die Workspaces entsprechend mit den Funktionen und Daten bzw. definieren Sie ggf. neue Workspaces. Beachten Sie dabei, dass im UI-Struktur-Diagramm nur Funktionen angezeigt werden, die nicht automatisch ablaufen. Begründen Sie bei den Daten, wie diese mit den neuen Daten aus dem Domänendatendiagramm zusammenhängen. Halten Sie dies in Ihrem PDF-Dokument fest. Sie können das Diagramm entweder per Hand ergänzen und dann einscannen oder Sie verwenden ein Zeichenwerkzeug ihrer Wahl, z.B. Paint, Visio, draw.io.

Ergebnis:

Speichern Sie bitte Ihre Ergebnisse als .zip-Datei bis **Montag 13.01.2020 um 10.00 Uhr** in Moodle bestehend aus:

- Word-Datei *10-MovieManager-UserTask.docx*
- PNG-Datei *10-MovieManager-DomainData.png*
- PNG-Datei *10-MovieManager-UI-Structure.png*
- Excel-Datei *10-MovieManager-Systemfunctions.xlsx*
- PDF-Datei mit Ihren Begründungen