

PFLICHTENHEFT

Version: 0.1

Datum: 06.03.2021

DOKUMENTVERSIONEN

Versionsnr.	Datum	Autor	Änderungsgrund / Bemerkungen
0.1	06.03.21	Jano Detzel, Daniel Filler, Christin Bräunling, Maja Pfannendörfer	Ersterstellung
0.2	06.03.21	Jano Detzel, Daniel Filler, Christin Bräunling, Maja Pfannendörfer	Bearbeitung

INHALT

DOKUMENTVERSIONEN.....	1
INHALT	2
1. Einleitung.....	4
1.1 Allgemeines	4
1.1.1 Ziel und Zweck dieses Dokuments.....	4
1.1.2 Projektbezug.....	4
1.1.3 Abkürzungen	4
1.1.4 Ablage, Gültigkeit und Bezüge zu anderen Dokumenten	4
1.2 Verteiler und Freigabe	4
1.2.1 Verteiler für dieses Lastenheft.....	4
1.3 Reviewvermerke und Meeting-Protokolle.....	4
1.3.1 Erstes bis n-tes Review.....	4
2. Konzept und Rahmenbedingungen.....	5
2.1 Benutzer / Zielgruppe	5
2.2 Ziele des Anbieters	5
2.3 Ziele und Nutzen des Anwenders.....	5
2.4 Systemvoraussetzungen.....	5
2.5 Ressourcen	5
2.6 Übersicht der Meilensteine	5
3. Anforderungsbeschreibung	6
3.1 1. Anforderung.....	6
3.1.1 Beschreibung	6
3.1.2 Wechselwirkungen	6
3.1.3 Risiken	6
3.1.4 Schätzung des Aufwands	6
3.2 2. Anforderung.....	6
3.2.1 Beschreibung	6
3.2.2 Wechselwirkungen	6
3.2.3 Risiken	6
3.2.4 Vergleich mit bestehenden Lösungen	7
3.2.5 Schätzung des Aufwands	7
3.2.6 Beschreibung	7
3.2.7 Wechselwirkungen	7
3.2.8 Risiken	7
3.2.9 Vergleich mit bestehenden Lösungen	7
3.2.10 Schätzung des Aufwands	7
3.2.11 Beschreibung	7
3.2.12 Wechselwirkungen	7
3.2.13 Risiken	7
3.2.14 Vergleich mit bestehenden Lösungen	7
3.2.15 Schätzung des Aufwands	8
3.2.16 Beschreibung	8
3.2.17 Wechselwirkungen Abändern des Funktionsaufrufs in frequentRenterPoints().	8

3.2.18	Risiken	8
3.2.19	Vergleich mit bestehenden Lösungen	8
3.2.20	Schätzung des Aufwands	8
3.3	3. Anforderung	8
3.3.1	Beschreibung	8
3.3.2	Wechselwirkungen	8
3.3.3	Vergleich mit bestehenden Lösungen	8
3.3.4	Schätzung des Aufwands	8
3.4	4. Anforderung	9
3.4.1	Beschreibung	9
3.5	5. Anforderung	9
3.5.1	Beschreibung	9
3.5.2	Schätzung des Aufwands	9
3.5.3	Beschreibung	9
3.5.4	Schätzung des Aufwands	9
3.5.5	Beschreibung	9
3.5.6	Schätzung des Aufwands	9
3.5.7	Beschreibung	10
3.5.8	Schätzung des Aufwands	10
3.5.9	Beschreibung	10
3.5.10	Schätzung des Aufwands	10
3.5.11	Beschreibung	10
3.5.12	Schätzung des Aufwands	10
3.5.13	Beschreibung	10
3.5.14	Schätzung des Aufwands	11
4.	Anhang	13

1. EINLEITUNG

1.1 Allgemeines

1.1.1 Ziel und Zweck dieses Dokuments

In diesem Pflichtenheft spezifiziert der Arbeitnehmer konkrete Aufgaben zur Umsetzung zu festgelegten Meilenstein Terminen, die für das Refactoring und die Funktionserweiterung des Quellcodes notwendig sind.

1.1.2 Projektbezug

Das Refactoring-Projekt entspricht dem Code-Beispiel aus dem Buch von Martin Fowler. Refactoring: Improving the Design of Existing Code, Addison-Wesley, 1999, 1. Auflage.

1.1.3 Abkürzungen

- IDE – Integrated Development Environment

1.1.4 Ablage, Gültigkeit und Bezüge zu anderen Dokumenten

Ablage des Projektes befindet sich unter folgendem Link auf Github.com:

<https://github.com/janodetzel/fowler1.git>

1.2 Verteiler und Freigabe

1.2.1 Verteiler für dieses Lastenheft

Rolle	Name	Telefon	E-Mail	Bemerkungen
Projektleiter	Jano Detzel	0000	Jano.detzel@student.dhbw-karlsruhe.de	
Stellv. Projektleiter	Daniel Filler	0000	Daniel.Filler@student.dhbw-karlsruhe.de	
Stellv. Projektleiter	Christin Bräunling	0000	Christin.Braeunling@student.dhbw-karlsruhe.de	
Stellv. Projektleiter	Maja Pfannendörfer	0000	Maja.Pfannendörfer@student.dhbw-karlsruhe.de	

Projektleiter übernehmen auch die Rolle des Entwicklers

1.3 Reviewvermerke und Meeting-Protokolle

1.3.1 Erstes bis n-tes Review

Keine Bemerkungen.

2. KONZEPT UND RAHMENBEDINGUNGEN

2.1 Benutzer / Zielgruppe

Der Benutzer des Systems sind die Angestellte des Videoverleihs.

2.2 Ziele des Anbieters

Durch ein Refactoring des Codes kann die Performance während der Benutzung erhöht werden. Außerdem können dann einfacher die benötigten Erweiterungen angebunden werden.

2.3 Ziele und Nutzen des Anwenders

Der Anwender profitiert davon durch kürzere Ladezeiten und erweiterten Funktionen

2.4 Systemvoraussetzungen

Das System muss die Programmiersprache Java unterstützen und für die Erweiterung dann auch eine Ausgabe und Darstellung von HTML ermöglichen.

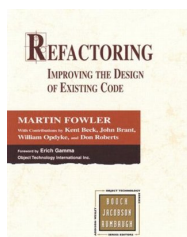
2.5 Ressourcen

Für das Projekt sind 4 Zeitstunden geplant mit 4 Entwicklern.

2.6 Übersicht der Meilensteine

Meilenstein 1: Pflichtenheft erstellt und abgenommen (Anforderung 101)	
Schritt 1: Pflichtenheft fertiggestellt	13:00
Schritt 2: Pflichtenheft präsentiert	13:30
Schritt 2.1: Pflichtenheft nachgearbeitet (opt.)	13:31
Schritt 3: Pflichtenheft abgenommen	13:40
Meilenstein 2: Refactoring durchgeführt	
Schritt 1: Preisbestimmungsmethode in Customer-Klasse ausgelagert	14:26
Schritt 2: Preisbestimmungsmethode für Einzelwerte auslagern	14:27
Schritt 3: Performanceoptimierung durch direkten Methodenaufruf anstelle von Variablen.	14:28
Schritt 4: Auslagern der Methode zur Berechnung der Bonuspunkte	14:29
Meilenstein 3: Alle funktionalitätserweiterungen Implementiert (opt.)	
Schritt 1: Refactoring zur Vorbereitung der Funktionalitätserweiterung durchgeführt	
Schritt 2: HTML-Ausgabe der Statement-Klasse ist implementiert	
Schritt 3: Refactoring mehrerer Methoden zur Vorbereitung auf Änderung der Filmklassifikation durchgeführt.	
Schritt 4: Änderungen der Film-Klassifikation implementiert	

3. ANFORDERUNGSBESCHREIBUNG



Die Anforderungen dieses Refactoring-Projekts betreffen ein Filmverleih-Projekt wie es Martin Fowler in seinem Buch Refactoring: Improving the Design of existing Code aus dem Jahr 1999 als Beispiel einführt.

3.1 1. Anforderung

Nr. / ID	101	Nichttechnischer Titel	Projektplanung mit Meilensteinen und Definition von Artefakten		
Quelle		Verweise		Priorität	muss

3.1.1 Beschreibung

Die Projektplanung wird im Pflichtenheft festgehalten, unter Berücksichtigung der Anforderungen 102, 103, 104, 105. In diesem Zug werden auch die Meilensteine des Projektes definiert.

3.1.2 Wechselwirkungen

Das Projekt wird in einem separaten, privaten Github Repository gehostet und hat keinen Einfluss auf andere bzw. wird nicht von anderen Beeinflusst.

3.1.3 Risiken

Die Projektplanung mit Meilensteinen und Definition von Artefakten ist mit keinen Risiken verbunden.

3.1.4 Schätzung des Aufwands

Die Anforderung wird mit einem Aufwand von 1 bis 1.5 Stunden bewertet.

3.2 2. Anforderung

Nr. / ID	102a	Nichttechnischer Titel	Preisbestimmungsmethode in Customer-Klasse auslagern		
Quelle		Verweise		Priorität	muss

3.2.1 Beschreibung

Code zur Preisbestimmung innerhalb der Methode Statement der Customer-Klasse in eigene Methode mit der Bezeichnung amountFor() mit dem Rückgabewert double und einem Rental-Objekt als Übergabeparameter auslagern. Diese Methode gibt den berechneten Gesamtpreis der Ausleihe zurück.

3.2.2 Wechselwirkungen

Statement-Methode wird um Berechnung des Gesamtpreises reduziert.

3.2.3 Risiken

Nach Überprüfung vorhandener Tests und der Testabdeckung, können diese verwendet werden um sicherzustellen, dass nach dem Refactoringprozess die äußere Funktionalität des Programmes gewährleistet ist.

3.2.4 Vergleich mit bestehenden Lösungen

Verwendung von Extract-Method Methode.

3.2.5 Schätzung des Aufwands

Die Anforderung wird mit einem Aufwand von 0.25 Stunden bewertet.

Nr. / ID	102b	Nichttechnischer Titel	Preisbestimmungsmethode für Einzelwerte auslagern		
Quelle		Verweise		Priorität	muss

3.2.6 Beschreibung

Code zur Preisbestimmung der Einzelwerte durch Move-Method in Rental Klasse auslagern. Hierfür sollten im Vorfeld die Variablenbezeichnungen angepasst werden. Dann wird die Methode getCharge() in Rental Klasse verschoben

3.2.7 Wechselwirkungen

Statement-Methode wird um Berechnung der Einzelwerte reduziert. Der Methodenaufruf getCharge() wird zu each.getCharge() verändert.

3.2.8 Risiken

Nach Überprüfung vorhandener Tests und der Testabdeckung, können diese Verwendet werden um Sicherzustellen, dass nach dem Refactoringprozess die äußere Funktionalität des Programmes gewährleistet ist.

3.2.9 Vergleich mit bestehenden Lösungen

Verwendung von Move-Method Methode.

3.2.10 Schätzung des Aufwands

Die Anforderung wird mit einem Aufwand von 0.25 Stunden bewertet.

Nr. / ID	102c	Nichttechnischer Titel	Performanceoptimierung durch direkten Methodenaufruf anstelle von Variablen.		
Quelle		Verweise		Priorität	muss

3.2.11 Beschreibung

Performanceoptimierung durch direkten Methodenaufruf der each.getCharge() Methode anstelle von Zwischenspeicherung der Ausgabe in Variable thisAmount.

3.2.12 Wechselwirkungen

Variable thisAmount wird eliminiert und durch Methodenaufruf each.getAmount() ersetzt.

3.2.13 Risiken

Nach Überprüfung vorhandener Tests und der Testabdeckung, können diese Verwendet werden um Sicherzustellen, dass nach dem Refactoringprozess die äußere Funktionalität des Programmes gewährleistet ist.

3.2.14 Vergleich mit bestehenden Lösungen

Verwendung von Replace-Temp-With-Query Methode.

3.2.15 Schätzung des Aufwands

Die Anforderung wird mit einem Aufwand von 0.25 Stunden bewertet.

Nr. / ID	102d	Nichttechnischer Titel	Auslagern der Methode zur Berechnung der Bonuspunkte		
Quelle		Verweise		Priorität	muss

3.2.16 Beschreibung

Auslagern der Methode zur Berechnung der Bonuspunkte in Klasse Rental. Abändern des Funktionsaufrufs in frequentRenterPoints(). In diesem Zug werden die Temporären Variablen zur Zwischenspeicherung des Ergebnisses zur Bonuspunktberechnung entfernt und in der Ausgabe der direkte Methodenaufruf verwendet.

3.2.17 Wechselwirkungen

Abändern des Funktionsaufrufs in frequentRenterPoints().

3.2.18 Risiken

Nach Überprüfung vorhandener Tests und der Testabdeckung, können diese verwendet werden um sicherzustellen, dass nach dem Refactoringprozess die äußere Funktionalität des Programmes gewährleistet ist.

3.2.19 Vergleich mit bestehenden Lösungen

Verwendung von Extract-and-Move-Method und Replace-Temp-With-Query Methode.

3.2.20 Schätzung des Aufwands

Die Anforderung wird mit einem Aufwand von 0.25 Stunden bewertet.

3.3 3. Anforderung

Nr. / ID	103	Nichttechnischer Titel	Softwareentwicklung nach Wasserfall		
Quelle		Verweise		Priorität	muss

3.3.1 Beschreibung

Die Entwicklung / Umsetzung des Refactorings soll nach den oben genannten Meilensteinen erfolgen und sich an den Phasen des Wasserfall-Modells orientieren.

3.3.2 Wechselwirkungen

Für die Umsetzung des Wasserfall-Modells werden notwendige Dokumente und Artefakte in Github gehostet.

3.3.3 Vergleich mit bestehenden Lösungen

Pair Programming (Group Programming)

3.3.4 Schätzung des Aufwands

Der Aufwand dieser Anforderung überdauert die vollständige Projektlaufzeit.

3.4 4. Anforderung

Nr. / ID	104	Nichttechnischer Titel	Werkzeugumgebung		
Quelle		Verweise		Priorität	soll

3.4.1 Beschreibung

Für die moderne Softwareentwicklung wird die bekannte IDE IntelliJ Idea der Firma JetBrains verwendet. Zur Kollaboration und für die Versionsverwaltung wird Github.com verwendet.

3.5 5. Anforderung

Nr. / ID	105a	Nichttechnischer Titel	Vorbereitung des Quellcodes an mögliche Änderungen an der Bezahlung und der Berechnung der Rabatte		
Quelle		Verweise		Priorität	kann

3.5.1 Beschreibung

Zwei neue Methoden erstellen. Die erste Methode berechnet den Gesamtpreis der Ausleihe, die zweite Methode berechnet die Bonuspunkte für die Ausleihe. Ersetzen des vorhandenen Codes durch Methodenaufzuruf der neuen Methoden innerhalb der Statement-Methode.

3.5.2 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105b	Nichttechnischer Titel	Statement um HTML Ausgabe erweitern		
Quelle		Verweise		Priorität	kann

3.5.3 Beschreibung

Die Abrechnungsklasse Statement soll um eine HTML-Ausgabe erweitert werden. Hierfür wird die neue Methode StatementHTML() mit einem Text als Rückgabewert in der Statement Klasse implementiert.

3.5.4 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105c	Nichttechnischer Titel	Refactoring auf Vorbereitung der Änderungen der Filmklassifikation		
Quelle		Verweise		Priorität	kann

3.5.5 Beschreibung

Methoden getCharge() und frequentRenterPoints() müssen von der Klasse Rental zur Klasse Movie umgezogen werden. Damit wird ermöglicht, dass andere Klassen von Filmen von der Klasse Movie vererbt werden können.

3.5.6 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105d	Nichttechnischer Titel	Implementierung des State-Pattern zur Preiskategorisierung		
Quelle		Verweise		Priorität	kann

3.5.7 Beschreibung

Es wird eine neue abstrakte Klasse Price erstellt. Diese implementiert eine abstrakte Methode getPriceCode(). Nachfolgend werden drei Unterklassen erstellt, welche von der Klasse Price erben und die Methode getPriceCode beinhalten. Anschließend werden die Zugriffe in der Klasse Movie in der Methode setPriceCode auf die jeweiligen Unterklassen der Klasse Price abgeändert.

3.5.8 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105e	Nichttechnischer Titel	Implementierung von Vielgestaltigkeit für die Unterklassen der Price Klassen		
Quelle		Verweise		Priorität	kann

3.5.9 Beschreibung

Die Methode getCharge() wird aus der Klasse Movie gelöscht und wird als abstrakte Methode in der Oberklasse Price angelegt. Die Klassen, welche von Price erben, implementieren die getCharge() Methoden und geben ihren spezifischen Wert zurück.

3.5.10 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105f	Nichttechnischer Titel	Zusammenführen der Statement-Aufrufe zur Ausgabe der Statement-Ausgaben		
Quelle		Verweise		Priorität	kann

3.5.11 Beschreibung

Es werden zwei neue Klassen Textstatement und HTMLStatement angelegt. Diese beinhalten jeweils die Methoden value(customer), headerString(customer), eachRentalString(rental), footerString(customer) zur Rückgabe der einzelnen Ausgabeteile. Innerhalb der Klasse Customer, wird die Methode statement() angepasst, damit diese nun noch einen der beiden Ausgabetypen zurückgibt. Unterschiedliche Codeteile zwischen Text- und HTML-Ausgabe werden herausgenommen.

3.5.12 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Nr. / ID	105g	Nichttechnischer Titel	Extrahieren der Methode value() in Oberklasse Statement		
Quelle		Verweise		Priorität	kann

3.5.13 Beschreibung

Aufgrund der Differenzen zwischen Text- und HTML-Ausgabe in den Methoden headerString(customer), eachRentalString(rental), footerString(customer) kann die value() Methode in der Oberklasse Statement aufgerufen werden.

3.5.14 Schätzung des Aufwands

Der Aufwand dieser Anforderung wird mit 0.5 Stunden bewertet.

Genehmigung
Die Genehmigung erfolgt...

Datum:

Unterschrift Auftraggeber:

Unterschrift Projektleiter:

Weitere Unterschriften:

4. ANHANG

Ihr Text

Diese Vorlage wurde erstellt von:
vorlage-kostenlos.de



Mehr Informationen auf www.sevdesk.de



sevDesk
Buchhaltung stressfrei selber machen

KOSTENLOS TESTEN