**Pflichtenheft**

**Version:** 0.1

**Datum:** 02.04.2022

**Autoren:** Elena Väth, Marco Pelzer, Florian Khan, Ayse Kocak

INHALT

Inhaltsverzeichnis

[DOKUMENTVERSIONEN 1](#_Toc99790615)

[INHALT 2](#_Toc99790616)

[1. Einleitung 4](#_Toc99790617)

[1.1 Allgemeines 4](#_Toc99790618)

[1.1.1 Ziel und Zweck dieses Dokuments 4](#_Toc99790619)

[1.1.2 Projektbezug 4](#_Toc99790620)

[1.2 Verteiler und Freigabe 4](#_Toc99790621)

[1.2.1 Verteiler für dieses Lastenheft 4](#_Toc99790622)

[2. Konzept und Rahmenbedingungen 5](#_Toc99790623)

[2.1 Übersicht der Meilensteine 5](#_Toc99790624)

[3. Anforderungsbeschreibung 6](#_Toc99790625)

[3.1 1. Anforderung 6](#_Toc99790626)

[3.1.1 Beschreibung 6](#_Toc99790627)

[3.1.2 Wechselwirkungen 6](#_Toc99790628)

[3.1.3 Risiken 6](#_Toc99790629)

[3.1.4 Schätzung des Aufwands 6](#_Toc99790630)

[3.2 2. Anforderung 6](#_Toc99790631)

[3.2.1 Beschreibung 6](#_Toc99790632)

[3.2.2 Wechselwirkungen 6](#_Toc99790633)

[3.2.3 Risiken 6](#_Toc99790634)

[3.3 3. Anforderung 7](#_Toc99790635)

[3.3.1 Beschreibung 7](#_Toc99790636)

[3.4 4. Anforderung 7](#_Toc99790637)

[3.4.1 Beschreibung 7](#_Toc99790638)

[3.5 5. Anforderung 7](#_Toc99790639)

[3.5.1 Beschreibung 7](#_Toc99790640)

[3.6 5.Anforderung 7](#_Toc99790641)

[3.6.1 Beschreibung 7](#_Toc99790642)

[4. Genehmigung 8](#_Toc99790643)

# Einleitung

## Allgemeines

### Ziel und Zweck dieses Dokuments

Dieses Pflichtenheft beschreibt die Meilensteine und Anforderungen des geforderten Projektes.

### Projektbezug

Ihr Text

## Verteiler und Freigabe

### Verteiler für dieses Lastenheft

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rolle | Name | Telefon | E-Mail | Bemerkungen |
| Projektleiter | Elena Väth |  |  |  |
| Entwickler | Marco Pelzer, Florian Khan,  Ayse Kocak |  |  |  |
| Auftraggeber | Marc Schanne |  |  |  |

# Konzept und Rahmenbedingungen

## Übersicht der Meilensteine

|  |  |
| --- | --- |
| **Vorbereitungsphase** | |
| 101.1 Projektplanung | 11.00 – 12.00 |
| 103.1 Anwendung des Wasserfallmodells | 11.00 – 16.00 |
| 104.1 Entwicklungsumgebung | 11.00 – 16.00 |
| **Implementierung und Test** | |
| 102.1 Methodenübersicht des Refactorings | 12.00 – 13.00 |
| 102.2 Umsetzung des Refactorings | 13.00 - 14.00 |
| **Erweiterung** | |
| 105.1 Verbesserung in Funktionalität | 15.00 – 16.00 |
| **Voraussichtlicher Verkaufsstart** | 04.04.2022 |

# Anforderungsbeschreibung

## 1. Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 101.1 | **Nichttechnischer Titel** | | Projektplanung | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | muss |

### Beschreibung

Bevor die geforderten Anforderungen umgesetzt und erfüllt werden können, ist zunächst eine Planung des Vorgehens inklusive einer Anordnung von Meilensteinen, die im Projektablauf zu bewältigen sind, erforderlich. Hierfür werden Meilensteine gesetzt, um einen konkreten zeitlichen Ablauf zu gewährleisten.

### Wechselwirkungen

Die Dokumentation und Archivierung der Artefakte des Projektes erfolgen im zugehörigen GitHub-Repository.

### Risiken

Risiken in der Planung des Refactoring-Projektes sind die mögliche zeitliche Fehlschätzung sowie eine ungenaue Anforderungsspezifikation.

### Schätzung des Aufwands

Für die Planung des Projektes steht dem AN eine zeitliche Begrenzung von 1 Zeitstunde pro Meilenstein zur Verfügung. Der Beginn der Planung findet planmäßig um 10.30 Uhr statt und stellt somit den ersten fertigzustellenden Meilenstein dar.

## 2. Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 102.1 | **Nichttechnischer Titel** | | Methodenübersicht des Refactorings | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | muss |

### Beschreibung

Zu den gewünschten Anforderungen für das Projekt gehören vor allem die Strukturierung der Inhalte des bestehenden Projekts. Hierfür werden Methoden der OO-Programmierung und Software-Entwurfsmuster verwendet. Des Weiteren werden Methoden des Refactorings aus Quellen von Martin Fowler genutzt.

Insbesondere sind diese Methoden folgende:

* Extract Method
* Move Method
* Replace Temp with Query
* Replace Type Code with State/Strategy
* Replace Switch with Polymorphism
* Form Template Method

### Wechselwirkungen

Für die Beseitigung von bestehenden Code-Smells dienen die von Martin Fowler vorgestellten Refactorings als Basis.

### Risiken

Mögliche auftretbare Risiken hinsichtlich der Umsetzung von Rafectoring-Methoden sind die funktionale Änderung des bestehenden Codes. Hierfür spielt das Durchführen geeigneter Tests eine wesentliche Rolle.

## 3. Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 102.2 | **Nichttechnischer Titel** | | Umsetzung des Refactorings | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | muss |

### Beschreibung

Die zuvor genannten Methoden müssen auf das Projekt angewandt werden. Hierfür müssen für jede Methode die folgenden Schritte angewandt werden.

* Extract Method
  + Erstelle die Methode benannt nach ihrem Zweck
  + Kopiere den ausgelagerten Code
  + Deklariere lokale Variablen und Parameter innerhalb der Methode
  + Kompiliere
  + Ersetze Code-Fragment durch Aufruf der neu erstellten Methode
  + Kompiliere und Teste
* Move Method
  + Deklariere die Methode in der Zielklasse
  + Kopiere und Setze Code ein
  + Referenziere im Anfangsobjekt auf das Zielobjekt
  + Wandle originale Methode um und achte auf überschriebene Methoden
  + Kompiliere und Teste
  + Passe Code, in dem Methode genutzt wird, enstsprechend an
  + Kompiliere und Teste
* Replace Temp with Query
  + Finde temporäre Hilfsvariable, die einmal deklariert wurde
  + Lese zugewiesenen Wert der Variable aus
  + Ersetze alle Referenzen der temporären Variable mit der neu erstellten Methode
  + Entferne Variable und die Zuweisung der temporären Hilfsvariable
  + Kompiliere und Teste
* Replace Type Code with State/Strategy
  + Erstelle eine neue Klasse für den Type Code
  + Füge Subklassen des Statusobjekts hinzu
  + Erstelle eine abstrakte Anfrage in der Superklasse und überschreibe die Subklassen
  + Kompiliere
  + Erstelle Feld in der alten Klasse für das Statusobjekt
  + Änder die Type Code Anfrage um an das Statusobjekt zu gelangen
  + Weise eine Instanz der Subklasse zu
* Replace Switch with Polymorphism
  + Bewege Methode in Superklasse
  + Kopiere
  + Kompiliere und teste
  + Wiederhole
  + Ersetze durch abstrakte Methode
* Form Template Method
  + Suche zwei Methoden mit ähnlicher Struktur aber unterschiedlichen Komponenten
  + Entnehme für jede Variante Methoden aus jeder Klasse mit dem selben
  + Deklariere die Methode in der Superklasse und setze die variierenden Rümpfe in Subklassen
  + Sind alle Varianten in der Superklasse entfernt, bewege Methodenkopf in die Superklasse

## 4. Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 103.1 | **Nichttechnischer Titel** | | Anwendung des Wasserfallmodells | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | muss |

### Beschreibung

Die Umsetzung des Projektes muss innerhalb des Wasserfall-Modells erfolgen.

## 5. Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 104.1 | **Nichttechnischer Titel** | | Entwicklungsumgebung | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | soll |

### Beschreibung

Für das Projekt genutzte Werkzeuge sollen die Entwicklungsumgebung Visual Studio Code oder weitere Entwicklungsumgebungen wie IntelliJ und Eclipse sein. Des Weiteren muss die Versionsverwaltung über Git erfolgen.

## 5.Anforderung

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr. / ID** | 105.1 | **Nichttechnischer Titel** | | Verbesserung in Funktionalität | | |
| **Quellen** |  | | **Verweise** |  | **Priorität** | kann |

### Beschreibung

Durch Einsetzen der Refactoring-Methoden ist es möglich, konkrete Aspekte in der Funktionalität des Projektes zu erweitern. Insbesondere die Komponente Movie des Projektes kann ohne Veränderung ihrer Logik um spezielle Unterkomponenten ergänzt werden.

(Siehe nachfolgende Seite, UML-Diagramm)

# 4. Genehmigung

Die Genehmigung erfolgt...

|  |  |
| --- | --- |
| Datum: |  |
| Unterschrift Auftraggeber: |  |
| Unterschrift Projektleiter: | Elena Väth |
| Weitere Unterschriften: | Marco Pelzer, Florian Khan, Ayse Kocak |