DHBW
Duale Hochschule
Baden-Württemberg
Karlsruhe

Programmentwurf

SimpsonsGame

von

Dominik Veith

Abgabedatum:	04. April 2023
Bearbeitungszeitraum:	04.10.2022 - 04.04.2023
Matrikelnummer, Kurs:	3352220, TINF20B2

Abstract

- Deutsch -

Dies ist der Beginn des Abstracts. Für die finale Bachelorarbeit musst du ein Abstract in deinem Dokument mit einbauen. So, schreibe es am besten jetzt in Deutsch und Englisch. Das Abstract ist eine kurze Zusammenfassung mit ca. 200 bis 250 Wörtern.

Versuche in das Abstract folgende Punkte aufzunehmen: Fragestellung der Arbeit, methodische Vorgehensweise oder die Hauptergebnisse deiner Arbeit.

Inhaltsverzeichnis

Abkürzungsverzeichnis

JDK Java Development Kit

IDE Integrated Development Environment

Abbildungsverzeichnis

Tabellenverzeichnis

Quellcodeverzeichnis

1 Einführung

1.1 Übersicht über die Applikation

Die Applikation Simpsons-Game verfiziert anhand gezielter Fragen an den User, welchem fiktiven Charakter aus einer Auswahl an Charakteren der TV-Serie 'The Simpsons' er am ehesten entspricht. Danach wird eine Übersicht ausgegeben wo der Charakter wohnen und arbeiten wird.

1.2 Start der Applikation

Zum Start der Applikation sind folgende Vorrausetzungen notwending:

- Das Java Development Kit (JDK) um den Code kompilieren und auszuführen zu können.
- Ein Integrated Development Environment (IDE) um die Ausführung des Codes komfortabler zu gestalten.

Um die Applikation zu starten sollte der Code innerhalb einer IDE der Wahl geöffnet werden. Danach muss die Java Klasse 'SimpsonsTerminal' im Ordner 'SimpsonsGame/src/main/java/de/dhbw/ase/simpsons/plugin' ausgeführt werden. Alle Interaktionen der Applikation erfolgen anhand einer textbasierten Ausgabe über das Terminal der IDE.

1.3 Testen der Applikation

TODO: Nach den UnitTests hier die Ausführung beschreiben.

2 Clean Architecture

Dieses Kapitel befasst sich mit der Clean Architecture.

2.1 Was ist Clean Architecture?

2.2 Analyse der Dependency Rule

2.2.1 Positiv-Beispiel: Dependency Rule

2.2.2 Negativ-Beispiel: Dependency Rule

2.3 Analyse der Schichten

3 SOLID

3.1 Analyse Single-Responsibility-Principle (SRP)

3.1.1 Positiv-Beispiel

3.1.2 Negativ-Beispiel

3.2 Analyse Open-Closed-Principle (OCP)

3.2.1 Positiv-Beispiel

3.2.2 Negativ-Beispiel

3.3 Analyse Liskov-Substitution- (LSP), Interface-Segreggation- (ISP), Dependency-Inversion-Principle (DIP)

3.3.1 Positiv-Beispiel

3.3.2 Negativ-Beispiel

4 Weitere Prinzipien

4.1 Analyse GRASP: Geringe Kopplung

4.1.1 Positiv-Beispiel

4.1.2 Negativ-Beispiel

4.2 Analyse GRASP: Hohe Kohäsion

4.3 Don't Repeat Yourself (DRY)

5 Unit Tests

5.1 10 Unit Tests

5.2 ATRIP: Automatic

5.3 ATRIP: Thorough

5.4 ATRIP: Professional

5.5 Code Coverage

5.6 Fakes und Mocks

6 Domain Driven Design

6.1 Entities

6.2 Value Objects

6.3 Aggregates

7 Refactoring

7.1 Code Smells

7.2 2 Refactorings

Charakter klassen entzerrt mit einführen einer Superklasse und einem Interface