

Entwerfen und Implementieren einer Verschlüsselungssoftware

Ausarbeitung

Bachelor of Science

Studiengang Informationstechnik

an der

Dualen Hochschule Baden-Württemberg Karlsruhe

von

Nico Schrodtt

Abgabedatum 18. März 2022

Bearbeitungszeitraum	5 + 6 Semester
Kurs	TINF19B3
Dozent	Daniel Lindner

Erklärung

Ich versichere hiermit, dass ich meine Ausarbeitung mit dem Thema:

Entwerfen und Implementieren einer Verschlüsselungssoftware

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt (falls vorhanden).

Karlsruhe, 18. März 2022

Ort, Datum

Unterschrift

Inhaltsverzeichnis

Abbildungsverzeichnis	III
Tabellenverzeichnis	III
Listings	III
Abkürzungsverzeichnis	IV
1 Einführung	1
1.1 Ziel der Arbeit	1
1.2 Repository	1
2 Clean Architecture	2
2.1 Geplante Schichtenarchitektur	2
2.2 Umsetzung	2
2.2.1 Benutzeroberfläche	2
2.2.2 Verschlüsselungsdienst	2
3 Entwurfsmuster	3
4 Programming Principles	4
4.1 SOLID	4
4.2 GRASP	4
4.3 DRY	4
5 Refactoring	5
5.1 Code Smells	5
5.1.1 Code Smells 1	5
5.1.2 Code Smells 2	5
5.1.3 Code Smells 3	5
5.2 Angewendete Refactorings	5
5.2.1 Refactoring 1	5
5.2.2 Refactoring 2	5
6 Unit Tests	6
6.1 Verwendete Unit Tests und getesteter Code	6
6.2 Anwendung der ATRIP-Regeln	6

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

Abkürzungsverzeichnis

1 Einführung

Dieses Kapitel befasst sich vorwiegend mit relevanten Grundlagen der Arbeit. Unter anderem wird das Ziel spezifiziert, elementare Aspekte der Arbeitsweise eines Prozessors werden erläutert und die verschiedenen Werkzeuge mit denen das Ziel realisiert wird werden aufgeführt.

1.1 Ziel der Arbeit

In dieser Arbeit soll ein Simulationsprogramm geschrieben werden, mit dem mehrere unterschiedliche 8-Bit Prozessoren simuliert werden können. Dazu sollen die grundlegenden Eigenschaften in kurzen Lernprogrammen erläutert werden. Ebenfalls soll es eine interaktive Einweisung geben wie der Simulator verwendet werden kann.

1.2 Repository

Der Quellcode kann in folgendem GitHub-Repository abgerufen werden:

<https://github.com/NicoSchrodt/EncryptionService>

2 Clean Architecture

Platzhalter

2.1 Geplante Schichtenarchitektur

Platzhalter

2.2 Umsetzung

Platzhalter

2.2.1 Benutzeroberfläche

Platzhalter

2.2.2 Verschlüsselungsdienst

Platzhalter

3 Entwurfsmuster

Platzhalter

4 Programming Principles

Platzhalter

4.1 SOLID

Platzhalter

4.2 GRASP

Platzhalter

4.3 DRY

Platzhalter

5 Refactoring

Platzhalter

5.1 Code Smells

Platzhalter

5.1.1 Code Smells 1

Platzhalter

5.1.2 Code Smells 2

Platzhalter

5.1.3 Code Smells 3

Platzhalter

5.2 Angewendete Refactorings

Platzhalter

5.2.1 Refactoring 1

Platzhalter

5.2.2 Refactoring 2

Platzhalter

6 Unit Tests

Platzhalter

6.1 Verwendete Unit Tests und getesteter Code

Platzhalter

6.2 Anwendung der ATRIP-Regeln

Platzhalter