C:\Users\haller\Desktop\Logo_HFU.tif

Bachelorarbeit

in

Medieninformatik

Zu Erlangung des Grades Bachelor of Science

Entwicklung eines speziellen DevOps Projekt-Konfigurator Werkzeuges für Projektinitialisierung /– aktualisierung

|  |  |
| --- | --- |
| Referent: | Herr Prof. Dr. Eisenbiegler |
| Korreferent: | Herr Dr.-Ing- Christoph Rathfelder |
| Vorgelegt am: | 31.12.2021 |
| Vorgelegt von: | Nick Stecker |
|  | 25195 |
|  | Haselstiegstr. 5/1  78052 Villingen |
|  | nick.stecker@hs-furtwangen.de |

# Abstract

[Englisch, 100 -120 Worte]

[Deutsch, 100 – 120 Worte]

Inhaltsverzeichnis

[Abstract 2](#_Toc87260749)

[Abbildungsverzeichnis 5](#_Toc87260750)

[Tabellenverzeichnis 6](#_Toc87260751)

[Abkürzungsverzeichnis 7](#_Toc87260752)

[1. Einleitung 8](#_Toc87260753)

[1.1 Nutzen dieser Bachelorarbeit 8](#_Toc87260754)

[1.2 Die Problemstellung 9](#_Toc87260755)

[1.2.1 Die Projekterstellung 9](#_Toc87260756)

[1.2.2 Die Projektaktualisierung 10](#_Toc87260757)

[1.3 Aufgaben des Projekt-Konfigurators 10](#_Toc87260758)

[1.4 Weiterer Aufbau 11](#_Toc87260759)

[2. Grundlagen und Stand der Technik 12](#_Toc87260760)

[2.1 Die Bedeutung von DevOps 12](#_Toc87260761)

[2.2 Build-Management-Tools 12](#_Toc87260762)

[2.3 Frameworks 12](#_Toc87260763)

[2.4 Entwicklungsumgebung (IDE) 12](#_Toc87260764)

[3. Lösungsansatz 13](#_Toc87260765)

[4. Softwareentwurf 14](#_Toc87260766)

[4.1 GUI Sketches 14](#_Toc87260767)

[4.2 Architektur 14](#_Toc87260768)

[4.3 Extra Funktionalitäten 14](#_Toc87260769)

[5. Implementierung und Evaluierung 15](#_Toc87260770)

[6. Zusammenfassung 16](#_Toc87260771)

[6.1 Zusammenfassung 16](#_Toc87260772)

[6.2 Das Problem 16](#_Toc87260773)

[6.3 Mögliche Erweiterungspunkte 16](#_Toc87260774)

[Literaturverzeichnis 17](#_Toc87260775)

[Eidesstattliche Erklärung 18](#_Toc87260776)

[Anhang 19](#_Toc87260777)

# Abbildungsverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

# Tabellenverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

# Abkürzungsverzeichnis

**DevOps** Development und IT Operations

**GUI** Graphical User Interface

**IDE** Integrated Development Environment

**UML** Unified Modeling Language

# Einleitung

*„Gutes Werkzeug, halbe Arbeit“* ~ Ungarisches Sprichwort

Auch wenn man bei diesem Sprichwort direkt an einen Handwerker denkt, die mit einem Hammer Nägel in ein Brett schlägt, trifft dieses Sprichwort auch auf die meisten anderen Berufe zu. Der Chirurg hat sein Skalpell, der Friseur hat seine Schere, der Holzfäller seine Axt und der Bauer sein Traktor. All diese Werkzeuge sind dazu da, den Arbeitern einen gewissen Teil ihrer Arbeit zu erleichtern. Außerdem haben die Werkzeuge erheblichen Einfluss auf die Produktivität der Arbeit und die Qualität der Ergebnisse.

In vielen Berufsfeldern wäre die Arbeit ohne spezielle Werkzeuge undenkbar. Eines dieser Berufsfelder ist die Softwareentwicklung. Für eine professionelle Softwareentwicklung werden heutzutage nämlich viele verschiedene Programmierwerkzeuge (Tools) eingesetzt, die den Software-Entwicklungsprozess optimal unterstützen und beherrschbar machen.

In dieser Bachelorarbeit geht es um die Entwicklung eines solchen Tools. Der Projekt-Konfigurator ist ein spezielles Werkzeug für die Softwareentwicklung. Es soll Softwareentwicklern dabei helfen Softwareprojekte, basierend auf einem Projekt-Template zu erstellen und zu aktualisieren. Das Template-Projekt ist dabei einfach ein Ordner auf dem Computer mit einer gewissen Ordnerstruktur. Diese Ordnerstruktur setzt sich aus verschiedenen Unterordnern und Dateien zusammen.

## Nutzen dieser Bachelorarbeit

Diese Abschlussarbeit löst ein spezifisches Problem der Abteilung Anwendungsentwicklung von der Hahn-Schickard Gesellschaft in Villingen-Schwenningen. Das Problem bezieht sich hierbei auf die Initialisierung bzw. Aktualisierung eines bestehenden Projektes durch ein existierendes Template. Das Problem soll durch den in Abschnitt 1 beschriebene Projekt-Konfigurator gelöst werden.

Im Folgenden wird das genaue Problem dieser Abteilung beschrieben (Abschnitt 1.2) sowie die Aufgaben des Projekt-Konfigurators, die zu erledigen sind, um das Problem der Abteilung Anwendungsentwicklung zu lösen (Abschnitt 1.3).

## Die Problemstellung

Die Abteilung Anwendungsentwicklung arbeitet täglich an vielen verschiedenen Softwareprojekten. Jedes Softwareprojekt, besitzt dabei, je nach Programmiersprache, eine exakte Vorgabe über die Basis der zu verwendenden Verzeichnisstruktur. Das bedeutet, dass jedes Softwareprojekt, einer Programmiersprache, nach der Initialisierung dieselbe Ordnerstruktur aufweist. Inwiefern die Ordnerstruktur jedes Softwareprojekts in ihrer Entwicklungszeit ausgebaut wird (wie viele Dateien also hinzugefügt und bearbeitet werden), ist für jedes Softwareprojekt spezifisch. Allerdings darf sich dabei die Grundstruktur des Projekts nicht ändern.

Wenn ein Mitarbeiter der Abteilung Anwendungsentwicklung nun ein neues Softwareprojekt erstellen will, muss dieser erst einmal wissen wie die Verzeichnisstruktur für die jeweilige Programmiersprache des Projekts aufgebaut ist. Kennt der Mitarbeiter die Struktur, muss er diese auf dem Computer manuell nachbauen. Das heißt, der Mitarbeiter muss manuell alle nötigen Ordern und Dateien in der richtigen Ordnerstruktur erstellen. Kennt der Mitarbeiter die Struktur jedoch nicht, muss der Mitarbeiter diese erst herausfinden.

Um dieses Problem zu umgehen, besitzt die Abteilung für die meisten Softwareprojekte einer Programmiersprache ein vorgefertigtes Template-Projekt. Die Template-Projekte sind in den vorgegebenen Verzeichnisstrukturen aufgebaut und besitzen alle notwenigen Order bzw. Dateien. Die meisten Dateien sind so aufgebaut, dass erstellte Projekte diese ohne Veränderungen benutzen können. Allerdings beinhalten einige Dateien Platzhalter, die bei der Projekterstellung mit den Metadaten eines Softwareprojekts ersetzt werden müssen. Ohne diese Bearbeitungen, haben die Entwickler der Abteilung nur eine exakte Kopie eines Template-Projektes – das erstellte Softwareprojekt würde bis dato also noch kein reales Projekt repräsentieren.

### Die Projekterstellung

Die Projekterstellung mithilfe des beschriebenen Template-Projektes ist das erste große Problem dieser Bachelorarbeit. Denn um ein neues Softwareprojekt zu erstellen, müssen die Mitarbeiter das passende Template-Projekt kopieren und die Platzhalter mit den Metadaten des Projekts bearbeiten. Jedoch müssen die Mitarbeiter vor der Bearbeitung wissen, welche Template-Dateien Platzhalter beinhalten und an welchen Stellen sie sich in der Template-Datei befinden. Wenn die Mitarbeiter dies nicht wissen, müssen sie das, vor der Bearbeitung herausfinden. Wenn sie es allerdings wissen, müssen sie jede zu bearbeitende Datei manuell öffnen und alle Platzhalter eigenständig ersetzen.

Ein Beispiel: Wenn ein Entwickler den Namen eines Projekts bearbeiten/ändern möchte, muss er den Projektnamen manuell in dem Readme-Titel, in den Buildscript-Variablen und der Dokumentierung des Projekts ändern. Das sind drei Dateien die der Entwickler nur für den Projektnamen öffnen und bearbeiten muss.

Dieses Vorgehen ist im gleichen Maße zeitaufwändig wie fehleranfällig.

### Die Projektaktualisierung

Das zweite große Problem dieser Bachelorarbeit ist, dass ein Template-Projekte jederzeit aktualisiert werden kann. Die Entwickler können jederzeit bestimmte Order/Dateien des Template-Projektes überarbeiten, oder ihm neue Order/Dateien hinzufügen.

Die Problematik hierbei ist, dass diese Änderungen im Anschluss auf alle bestehenden Softwareprojekte, die aus dem veränderten Template-Projekt entstanden sind, übertragen werden müssen. Dazu müssen die Entwickler jedes dieser Softwareprojekte manuell auf dem Computer öffnen und in diesen die gleichen Änderungen vornehmen, die sie davor im Template-Projekt durchgeführt haben.

Dieses Vorgehen ist wie im vorherigen Abschnitt extrem zeitaufwändig und fehleranfällig.

## Ziel dieser Bachelorarbeit

Das Ziel dieser Bachelorarbeit ist die Konzeption und Implementierung eines Projekt-Konfigurators, der die beschriebenen Probleme in Abschnitt 1.2.1 und Abschnitt 1.2.2 löst.

Der Projekt-Konfigurator ist ein spezielles Programmierwerkzeug und besteht aus einer Software sowie einer grafischen Benutzeroberfläche (in Englisch, Graphical User Interface; abgekürzt GUI), die von den Entwicklern der Abteilung Anwendungsentwicklung ohne weitere Vorkenntnisse benutzt werden kann. Mithilfe der GUI können die Probleme wie folgt gelöst werden:

Das erste Problem löst der Projekt-Konfigurator, indem er den Entwicklern in der GUI ein Formular zu Verfügung stellt. In dieses Formular müssen die Mitarbeiter alle Metadaten des zu erstellenden Softwareprojekts eintragen. Auf Basis der eingegebenen Daten, erstellt der Projekt-Konfigurator ein neues konfiguriertes Projekt.

Das bedeutet, dass die Entwickler die Daten des neuen Softwareprojektes nicht mehr manuell ändern müssen, sondern alles automatisiert erstellt und angepasst wird. Alle notwendigen Änderungen/Anpassungen werden durch das Werkzeug gemacht und müssen nicht umständlich per Hand erfolgen. Dadurch werden die Projekte effizient und fehlerfrei erstellt.

Das zweite Problem löst der Projekt-Konfigurator, indem er die Template-version eines ausgewählten Softwareprojekts überprüft und daraufhin selbst entscheidet, ob das Projekt auf den neusten Template-Stand beinhaltet oder nicht. Wenn das Softwareprojekt nicht auf dem neusten Template-Stand ist, gibt der Projekt-Konfigurator den Entwicklern Möglichkeiten vor, um das Projekt zu aktualisieren. Dieser Mechanismus ist der Hauptteil dieser Abschlussarbeit.

## Weiterer Aufbau

In Kapitel zwei wird der aktuelle Stand der Technik bezüglich aktuellen Möglichkeiten für die Projekterstellung /-aktualisierung beleuchtet. Bereits vorhandene Werkzeuge/Software und deren Eigenschaften werden mittels Recherche ermittelt.

Das dritte Kapitel behandelt den Lösungsansatz des beschriebenen Problems (in Abschnitt 1.2).

In Kapitel vier geht es um den Softwareentwurf des Projekts. Hier wird das geplante Aussehen der GUI anhand von Skizzen gezeigt und erklärt. Darüber hinaus werden an dieser Stelle die geplante Architektur, sowie alle Funktionalitäten mithilfe von UML Diagrammen erklärt und begründet. Ein wesentlicher Fokus liegt hier auf der Modularität des Quellcodes

Im fünften Kapitel geht es um die Implementierung und Evaluierung des Projekts. Zum einen wird hier auf die Umsetzung des Projekt-Konfigurators eingegangen. Es werden einzelne Teile des Quellcodes und verschiedene Konfigurationsdateien näher beschrieben. Zum anderen geht es in diesem Kapitel um die Evaluierung aller Funktionalitäten. Jedes Anwendungsbeispiel wird aus Anwendersicht mithilfe von GUI-Screenshots beschrieben. Implementierte Extrafunktionalitäten die über die Aufgabenstellung hinausgehen werden hier besonders erwähnt.

Im sechsten und letzten Kapitel geht es um eine Zusammenfassung der geleisteten Arbeit und um die Beantwortung der Frage inwiefern der Projekt-Konfigurator das beschriebene Problem (in Abschnitt 1.3) löst. Außerdem gibt dieses Kapitel noch einen Ausblick auf künftige Erweiterungspunkte.

# Grundlagen und Stand der Technik

## Die Bedeutung von DevOps

Um dieses recht abstrakte Thema verstehen zu können, muss man zuerst einmal verstehen was der Begriff DevOps bedeutet:

DevOps ist ein sogenanntes Kofferwort, dass sich aus den Begriffen „Development (Entwicklung)“ und „IT Operations (IT-Betrieb)“ zusammensetzt. Die Entwicklung kümmert sich, um die Entwicklung der Softwareprojekte. IT Operations hingegen sind die Prozesse und Dienste, die von der IT-Abteilung eines Unternehmens administriert werden. Unter diesen Bereich fällt zusätzlich die Bereitstellung und Wartung von Hardware und Software für die Mitarbeiter. Normalerweise sind diese Bereiche grundverschieden. Mit dem DevOps-Ansatz sollen diese Bereiche nun zusammenrücken. Das Ziel dabei ist, dass die Softwareentwicklung und der IT-Betrieb effizienter und effektiver zusammenarbeiten. Dadurch kommt es nicht nur zu einer beschleunigten Entwicklung, sondern zusätzlich zu einer deutlichen Qualitätssteigerung des Produktes.

## Continuous Integration, Continuous Delivery (CI/CD)

Damit der DevOps-Ansatz erfolgreich umgesetzt werden kann, werden hierzu viele einzelne Werkzeuge (Tools) für bestimmte Aufgaben eingesetzt. Diese Tools müssen allerdings intelligent miteinander kombiniert werden, damit der DevOps-Prozess fehlerfrei funktioniert. Der Projekt-Konfigurator, ist eines dieser speziellen Werkzeuge.

## Build-Management-Tools

## Frameworks

## Entwicklungsumgebung (IDE)

Eclipse IDE

Visual Studio Code IDE

CLion IDE

# Lösungsansatz

Bei der Projekterstellung, führt der Projekt-Konfigurator vorgegebene Konfigurationen an dem Projekt-Template aus. Die Konfigurationsmaßnahmen werden von dem Softwareentwickler selbst bestimmt. Durch die Konfigurationen entsteht am Ende ein neues einzigartiges Projekt.

Bei der Projektaktualisierung prüft der Projekt-Konfigurator die Version eines vorgegebenen Softwareprojekts verglichen mit der Version des Projekt-Templates. Wenn die Version des Projekt-Templates neuer ist, teilt der Projekt-Konfigurator dem bedienenden Softwareentwickler mit welche Ordner und Dateien des Projekts aktualisierbar sind.

# Softwareentwurf

## GUI Sketches

## Architektur

Projekt Erstellung

Projekt Aktualisierung

Erweiterung der Projekt Module

Projekt Template Aktualisierung

## Extra Funktionalitäten

Suchungsmechanismus von bestehenden C++ Projekten

Run utilities

# Implementierung und Evaluierung

Internes Projekt Template

GUI Einstellungen

Projekt Erstellung

Suchungsmechanismus von bestehenden C++ Projekten

Erweiterung der Projekt Module

Projekt Template Aktualiserung

Run utilities

# Zusammenfassung

## Zusammenfassung

## Das Problem

## Mögliche Erweiterungspunkte

# Literaturverzeichnis

**Im aktuellen Dokument sind keine Quellen vorhanden.**

# Eidesstattliche Erklärung

Ich versichere, dass ich die vorstehende Arbeit selbständig verfasst und hierzu keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Stellen der Arbeit die wörtlich oder sinngemäß aus fremden Quellen entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt oder an anderer Stelle veröffentlicht.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[Ort, Datum Name]

# Anhang