Entwicklung eines Eclipse-Plugins für SpotBugs um eine Warnung zu

erzeugen, wenn für eine Variable nicht der allgemeinste Typ verwendet wird

**Studienarbeit T3\_3100**

des Studienganges Informatik

an der Dualen Hochschule Baden-Württemberg Horb

von

Tim Röthel

Abgabedatum: 25.05.2020

Bearbeitungszeitraum 7 Wochen

Matrikelnummer, Kurs 3080093, HOR-TINF2017

Ausbildungsfirma DATEV eG, Nürnberg

Betreuer der Ausbildungsfirma Wolfgang Pflaumer

Erklärung

gemäß § 5 (3) der „Studien- und Prüfungsordnung DHBW Technik“

vom 29. September 2015.

Ich versichere hiermit, dass ich meine Projektarbeit mit dem Thema

**Protokollierung von Wertänderungen in der DATEV-Lösung "Standard-Planung"**

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.



Schwabach, 21.02.2020 Unterschrift

**Zusammenfassung:**

In dieser Arbeit wird der Prozess der Erstellung eines Plugins für das statische Analysetool SpotBugs beschrieben. Das erstellte Plugin soll den Entwickler dabei unterstützen seinen Code besser wiederzuverwenden. Dies wird durch das Suchen von möglichen Generalisierungen erreicht. Eine Generalisierung ist, wenn ein Parameter oder eine lokale Variable des Typs X auf einen allgemeineren Typ Y generalisiert werden kann, ohne, dass das Programm nicht mehr kompilierbar ist.

Das erste Kapitel bildet die Motivation. In dieser wird genauer darauf eingegangen, wieso das Plugin eine sinnvolle Erweiterung bietet. Des Weiteren wird die Problemstellung grob erklärt.

Im zweiten Kapitel wird erklärt, was Design Patterns sind, welche Gruppierungen es hierbei gibt und einzelne Beispiele für Design Patterns genauer erklärt.

Im dritten Kapitel werden die statische und die dynamische Codeanalyse erläutert. Im Anschluss daran werden sie miteinander verglichen.

Im vierten Kapitel wird die Java Virtual Machine und ihre Struktur erläutert.

Im fünften Kapitel wird das Tool SpotBugs vorgestellt. Hierbei wird auf die Geschichte, auf die Funktionsweise, auf die Sicht eines Anwenders von SpotBugs und auf die Sicht eines Plugin Entwicklers eingegangen.

Im sechsten Kapitel wird genauer auf die Problemstellung eingegangen. Dies geschieht in Vorbereitung für die praktische Umsetzung.

Im siebten und letzten Hauptkapitel wird die praktische Umsetzung der Lösung zum Problem erläutert.

Im achten Kapitel wird überprüft, ob das Ziel erreicht wurde.