

Konzeption und Implementierung einer modularen Testsuite zur Automatisierung von Softwaretests

Test

Fachbereich Informatik

B-FMP02XX

Autor	Marvin Böck
Matrikelnummer	1892597
Anschrift	Solweg 5, 78647 Trossingen
Abgabedatum	07. Juli 2022

Erklärung an Eidesstatt

Hiermit erkläre ich, dass ich die vorliegende Abschlussarbeit mit dem Titel:
"Konzeption und Implementierung einer modularen Testsuite zur Automatisierung von Softwaretests"

eigenständig und ohne fremde Hilfe angefertigt habe. Textpassagen, die wörtlich oder dem Sinn nach auf Publikationen oder Vorträgen anderer Autoren beruhen, sind als solche kenntlich gemacht. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

.....

Ort, Datum

.....

Unterschrift

Sperrvermerk

Der Inhalt der Arbeit darf weder als Ganzes noch in Auszügen Personen außerhalb des Prüfungsprozesses und des Evaluationsverfahrens zugänglich gemacht werden, sofern keine anderslautende Genehmigung der SICK STEGMANN GmbH vorliegt

Zusammenfassung

Die vorliegende Arbeit befasst sich mit der Kozeptionierung sowie Proof of Concept Implementierung einer modularen Testsuite für Softwaretests. Grund für die Entwicklung der Suite ist die Portierung eines Bestandsprodukts auf eine neue Mikrocontroller Generation und der damit verbundener erhöhter Testaufwand. Bei der Firma SICK STEGAMNN GmbH ist seit kurzem ein Framework zum automatischen Testen von Software im Einsatz welches die Möglichkeit bietet, entsprechende Softwaretests komfortabel abzudecken. Um die Generierung der Testfälle für das Portierungsprojekt zu erleichtern, soll im Rahmen dieser Arbeit eine Testsuite konzeptioniert, sowie prototypisch implementiert werden. Ziel hierbei ist es, den späteren Prozess der Testentwicklung zu beschleunigen, sowie die Testfiles übersichtlicher und robuster zu gestalten. Zu Beginn der Arbeit wird sich ein Überblick über den Bereich Automated Testing @ GBC07 sowie Software Tests im Embedded Bereich geschaffen. Im Anschluss daran, werden die Testfälle eines vergleichbaren Produkts analysiert und geeignete Testfälle identifiziert. Danach wird eine Softwarearchitektur für die Suite konzeptioniert und diese prototypisch implementiert. Zum Schluss werden die Ergebnisse der Arbeit evaluiert.

Abstract

This thesis deals with the conceptual design and proof of concept implementation of a modular test suite for software testing. The reason for the development of the suite is the porting of an existing product to a new microcontroller generation and the associated increased test effort. The company SICK STEGAMNN GmbH has recently implemented a framework for automatic software testing which offers the possibility to comfortably cover the corresponding software tests. In order to facilitate the generation of test cases for the porting project, a test suite is to be conceptualized and prototypically implemented within the scope of this work. The goal is to accelerate the later process of test development and to make the test files clearer and more robust. At the beginning of the work, an overview of the area of Automated Testing @ GBC07 and software tests in the embedded area is created. Subsequently, the test cases of a comparable product are analyzed and suitable test cases are identified. Afterwards a software architecture for the suite is conceptualized and prototypically implemented. Finally, the results of the work are evaluated.

Inhaltsverzeichnis

1	Einleitung	1
1.1	SICK	1
2	Related Work	2
3	Grundlagen Softwareentwicklung und Verifikation	4
4	Konzept	5
5	Implementierung	6
6	Evaluation	7
7	Fazit / Ausblick	8
8	Literatur	I
9	Anhang	II

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

Abkürzungsverzeichnis

1 Einleitung

1.1 SICK

2 Related Work

In diesem Kapitel werden verschiedene Literaturquellen diskutiert, welche im Zusammenhang mit dem Thema der Arbeit stehen. Hierbei wird der Inhalt kurz zusammengefasst und erläutert, in welchem Zusammenhang die Arbeiten stehen.

Optimized test suites for automated testing using different optimization techniques

In diesem Artikel, welcher von Manju Khari, Prabhat Kumar, Daniel Burgos und Rubén González Crespo geschrieben wurde, befassen sich die Autoren mit der Optimierung von automatisierten Testsuites unter Betrachtung verschiedener Optimierungstechniken. Hierbei werden Tools zur automatischen Generierung von Testsuites verglichen. Die Ergebnisse werden dann im Kontext des Automated Testing betrachtet. Die Arbeit zeigt auf, welche Möglichkeiten bei der Optimierung und automatischen Generierung von Softwaretests bestehen. Im Zuge des konkreten Projekts wird ebenfalls eine Testsuite erstellt. Die Betrachtung von eventuellen Optimierungsmöglichkeiten ist für den späteren Verlauf des Projekts durchaus von Interesse.[1]

Eine Technologie für das durchgängige und automatisierte testen eingebetteter Software

Die Arbeit, welche durch Dipl.-Inform. Till Fischer im Rahmen seiner Dissertation an der Fakultät für Elektrotechnik und Informationstechnik des Karlsruher Instituts für Technologie (KIT) durchgeführt wurde, werden neben den Grundlagen des Testens von eingebetteten Systemen auch die verschiedenen Testebenen diskutiert. Ziel der Arbeit ist die Verbesserung der Durchgängigkeit des Testprozesses für eingebettete Systeme. Als Lösungsstrategie wird durch Fischer eine Testlösung versiert, welche den Quelltext der ausgeführten Software, die Netzwerkkommunikation, sowie das physikalische Verhalten an elektrischen Schnittstellen und der simulierten Umgebung abdeckt. Hier ist

der Bezug zum in dieser Arbeit behandelten Automated Testing @ GBC07 Projekt zu erkennen. [2]

3 Grundlagen Softwareentwicklung und Verifikation

4 Konzept

5 Implementierung

6 Evaluation

7 Fazit / Ausblick

8 Literatur

Literatur

- [1] Manju Khari, “Optimized test suites for automated testing using different optimization techniques,” 2017, zugegriffen am 26.08.2021 14:00.
- [2] Dipl.-Inform. Till Fischer, “Eine technologie für das durchgängige und automatisierte testen eingebetteter software,” 2016, zugegriffen am 26.08.2021 14:00.

9 Anhang