



Hochschule für angewandte Wissenschaften Augsburg
Fakultät für Informatik

Bachelorarbeit

Vergleich von Python Test-Tools für Test-driven development

zur Erlangung des akademischen Grades
Bachelor of Science

Thema:	Vergleich von Python Test-Tools für Test-driven development
Autor:	Maximilian Konter maximilian.konter@hs-augsburg.de MatNr. 951004
Version vom:	14. März 2019
1. Betreuer:	Dipl.-Inf. (FH), Dipl.-De Erich Seifert, MA
2. BetreuerIn:	Prof. Dr. X

Abstract kommt hier hin

Inhaltsverzeichnis

Abbildungsverzeichnis	5
Tabellenverzeichnis	6
Listingverzeichnis	7
Abkürzungsverzeichnis	8
1 Einleitung	9
1.1 Was ist Python	9
1.2 Was ist Test-driven development	9
2 Python Test-Tools	9
2.1 Welche Tools bietet die Standard Bibliothek von Python?	9
2.2 Welche Tools gibt es abseits der Standard Bibliothek?	9
2.2.1 Python Test-Tools	9
2.2.2 Python-Framework Test-Tools	9
2.2.3 Weitere Tools	9
2.3 unittest	9
3 Zusammenfassung	10
3.1 unittest	10
4 Vergleich der Tools	10
5 Kombinierung von Tools	10
6 Diskussion: Lohnt sich Test-driven development	10
6.1 Welche Stärken hat Test-driven development?	10
6.2 Welche Schwächen hat Test-driven development?	10
6.2.1 Welche Möglichkeiten gibt es diese Schwächen zu umgehen?	10
6.2.2 Welche Schwächen müssen akzeptiert werden?	10
6.3 Welche wirtschaftlichen Aspekte müssen bei Test-driven development be- achtet werden?	11
6.4 Zusammenfassung	11
7 Fazit	11
8 Nachwort	11
Literaturverzeichnis	12
Eidesstattliche Erklärung	12

Abbildungsverzeichnis

Tabellenverzeichnis

Listingverzeichnis

Abkürzungsverzeichnis

GNU	GNU is not Unix
GPL	GNU General Public License
GUI	Graphical User Interface
LGPL	GNU Lesser General Public License
TDD	Test-driven development

1 Einleitung

Das ist meine Einleitung

1.1 Was ist Python

Blubb

1.2 Was ist Test-driven development

Blubb

2 Python Test-Tools

Das ist meine Einleitung

2.1 Welche Tools bietet die Standard Bibliothek von Python?

Die Standard Bibliothek von Python bietet zwei verschiedene Test-Tools Baiju et al., 2011. Zum einen ist dies unittest und zum anderen doctest2

2.2 Welche Tools gibt es abseits der Standard Bibliothek?

Alle tools abseits der stdlib

2.2.1 Python Test-Tools

Blubb

2.2.2 Python-Framework Test-Tools

Blubb

2.2.3 Weitere Tools

Blubb

2.3 unittest

Blubb

3 Zusammenfassung

Alle Tools hier

3.1 unittest

Blubb

4 Vergleich der Tools

Blubb

5 Kombinierung von Tools

Blubb

6 Diskussion: Lohnt sich Test-driven development

Blubb

6.1 Welche Stärken hat Test-driven development?

Blubb

6.2 Welche Schwächen hat Test-driven development?

Blubb

6.2.1 Welche Möglichkeiten gibt es diese Schwächen zu umgehen?

Blubb

6.2.2 Welche Schwächen müssen akzeptiert werden?

Blubb

6.3 Welche wirtschaftlichen Aspekte müssen bei Test-driven development beachtet werden?

Blubb

6.4 Zusammenfassung

Blubb

7 Fazit

FAZIT

8 Nachwort

Nachwort

Literaturverzeichnis

Baiju, M., jtatum, CPE0014bf07fd2-CM001ac30d4aca, 71-35-143-156, gjb1002, little-black-box, ... SteveBarnes. (2011). PythonTestingToolsTaxonomy. Website. Online erhältlich unter <https://wiki.python.org/moin/PythonTestingToolsTaxonomy>; abgerufen am 14. März 2019.

Eidesstattliche Erklärung

Eidesstattliche Erklärung zur Abschlussarbeit

Hiermit versichere ich, die eingereichte Abschlussarbeit selbständig verfasst und keine andere als die von mir angegebenen Quellen und Hilfsmittel benutzt zu haben. Wörtlich oder inhaltlich verwendete Quellen wurden entsprechend den anerkannten Regeln wissenschaftlichen Arbeitens zitiert. Ich erkläre weiterhin, dass die vorliegende Arbeit noch nicht anderweitig als Abschlussarbeit eingereicht wurde.

Das Merkblatt zum Täuschungsverbot im Prüfungsverfahren der Hochschule Augsburg habe ich gelesen und zur Kenntnis genommen. Ich versichere, dass die von mir abgegebene Arbeit keinerlei Plagiate, Texte oder Bilder umfasst, die durch von mir beauftragte Dritte erstellt wurden.

Unterschrift :

Ort, Datum :