

Andreas Yalcin

**Software Engineering Konzepte in PHP:
Eine Untersuchung**

Diplomarbeit

Bibliografische Information der Deutschen Nationalbibliothek:

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2006 Diplomica Verlag GmbH
ISBN: 9783836606554

Andreas Yalcin

Software Engineering Konzepte in PHP: Eine Untersuchung

Andreas Yalcin

Software Engineering Konzepte in PHP: Eine Untersuchung

Diplomarbeit
Hochschule für Angewandte Wissenschaften Hamburg
Fachbereich Technik und Informatik
Oktober 2006



Diplom.de

Diplomica Verlag GmbH
Hermannstal 119k
22119 Hamburg

Fon: 040 / 655 99 20
Fax: 040 / 655 99 222

agentur@diplom.de
www.diplom.de

Andreas Yalcin

Software Engineering Konzepte in PHP: Eine Untersuchung

ISBN: 978-3-8366-0655-4

Druck Diplomica® Verlag GmbH, Hamburg, 2008

Zugl. Hochschule für Angewandte Wissenschaften Hamburg, Hamburg, Deutschland,
Diplomarbeit, 2006

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Die Informationen in diesem Werk wurden mit Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden, und die Diplomarbeiten Agentur, die Autoren oder Übersetzer übernehmen keine juristische Verantwortung oder irgendeine Haftung für evtl. verbliebene fehlerhafte Angaben und deren Folgen.

© Diplomica Verlag GmbH

<http://www.diplom.de>, Hamburg 2008

Printed in Germany

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ziel und Aufbau der Arbeit	2
2	Grundlagen.....	4
2.1	PHP.....	4
2.1.1	PHP Code in HTML einbetten	5
2.1.2	Variablen.....	6
2.1.3	Datentypen	7
2.1.4	Kontrollstrukturen.....	12
2.2	Software Engineering.....	13
2.2.1	Software Prozess Modelle	14
2.2.2	Analyse und Anforderungen	14
2.2.3	Entwurf.....	16
2.2.4	Implementierung	18
2.2.5	Validierung und Test.....	19
2.2.6	Inbetriebnahme und Wartung	20
3	Software Prozess Modelle	22
3.1	Das Wasserfallmodell	22
3.2	Evolutionäre Entwicklung.....	23
3.3	Wiederverwendungsorientierte Entwicklung.....	24
3.4	Das V-Modell.....	25
3.5	Software Prozess Modelle bezogen auf PHP Projekte.....	26
4	CASE-Tools	28
4.1	Versionskontrolle.....	28
4.1.1	Das Versionskontrollsystem CVS.....	29
4.1.2	Das Versionskontrollsystem Subversion	29
4.2	Testumgebung	30
4.2.1	Unit-Testing	31
4.2.2	Debugging	33
4.2.3	Lasttests	36
4.2.4	Profiling.....	37

5	IDEs für PHP Anwendungen.....	39
5.1	Eclipse.....	39
5.2	Zend Studio.....	40
5.3	Untersuchung von Eclipse und Zend Studio.....	41
5.3.1	Untersuchung von Eclipse.....	43
5.3.2	Untersuchung von Zend Studio.....	46
5.3.3	Vergleich von Eclipse und Zend Studio.....	48
6	Das neue Objektmodell von PHP 5.....	53
6.1	Vererbung, abstrakte Klassen und Interfaces.....	53
6.2	Kapselung.....	54
6.3	Polymorphismus.....	55
6.4	Hinweise auf Klassentypen.....	55
6.5	Kopier- und Referenzsemantik.....	56
6.6	Ausnahmebehandlung.....	56
6.7	Überladen mit __get, __set und __call.....	57
6.8	Übersicht.....	58
7	Beispielanwendung.....	60
7.1	Untersuchung eines Frameworks auf Nutzung von Software Engineering Konzepten..	60
7.1.1	Prinzipien der Implementierung.....	61
7.1.2	Design Patterns.....	62
7.1.3	Komponenten.....	63
7.1.4	Schichtenarchitektur.....	65
7.2	Erstellung einer Beispielanwendung.....	66
7.2.1	Beschreibung der Anwendung.....	66
7.2.2	Software Prozess Modell.....	66
7.2.3	Anforderungen.....	66
7.2.4	Entwurf.....	68
7.2.5	Implementierung.....	70
7.2.6	Test.....	74
7.2.7	Inbetriebnahme.....	78
7.3	Bewertung.....	80
7.3.1	Anforderungen.....	80
7.3.2	Entwurf.....	81
7.3.3	Implementierung.....	82

7.3.4	Test.....	83
7.3.5	Inbetriebnahme.....	84
7.3.6	Architektur des Frameworks.....	84
8	Zusammenfassung und Ausblick.....	85
8.1	Zusammenfassung.....	85
8.2	Ausblick.....	86
	Anhang.....	87
	Abkürzungsverzeichnis.....	87
	Glossar.....	88
	Literaturverzeichnis.....	89

1 Einleitung

Das Internet ermöglicht Unternehmen und Institutionen, ihre Dienste und Informationen einer großen Internetnutzergemeinschaft online zur Verfügung zu stellen. Um ihre Dienste und Angebote besser verwalten zu können, benötigen sie eine auf einem Webserver laufende Anwendung, eine so genannte Webanwendung. Beispiele für Webanwendungen sind Online Shops, virtuelle Auktionshäuser oder Content Management Systeme. Es gibt Programmiersprachen, die sich für das Entwickeln von Webanwendungen bewährt haben und daher besonders eignen. Neben Java und ASP.net hat sich vor allem auch die Skriptsprache PHP in den letzten Jahren hervor getan.

Gründe für die hohe Beliebtheit von PHP sind unter anderem, dass es einfach zu Erlernen ist (auch für Nicht-Informatiker). Desweiteren ist es einfach ein PHP Programm auszuführen, ohne hohen Konfigurationsaufwand, so dass schnell Erfolge erzielt werden können. Darüber hinaus sind Schnittstellen zu allen gängigen DBMS in die Programmiersprache integriert. Viele Webhosting Unternehmen bieten die nötige Umgebung (LAMP) zu erschwinglichen Preisen an, was PHP zu einer kostengünstigen Alternative zu anderen Technologien macht, vor allem für kleine und mittelgroße Unternehmen. Gründe, sich nicht für PHP zu entscheiden, sind ein nur rudimentäres Objektmodell und prozedurale Schnittstellen, die Veränderungen oder Erweiterungen an einer bestehenden Anwendung erschweren. Desweiteren haben sich einige Eigenschaften von PHP, die das Programmieren vereinfachen sollten, als sicherheitskritisch und portabilitätsmindernd herausgestellt. Diese Punkte wirken negativ auf die Qualität der Anwendung.

Um qualitativ hochwertige Software herzustellen, ist es während des Herstellungsprozesses notwendig, Prinzipien und Konzepte des Software Engineering umzusetzen. Das Software Engineering setzt sich mit allen Aspekten der Softwareherstellung auseinander. Es werden

Theorien, Methoden und Werkzeuge verwendet, um Probleme innerhalb organisatorischer und finanzieller Beschränkungen zu lösen.

In der Version 5 verfügt PHP über ein überarbeitetes Objektmodell, mit dem objektorientierte Techniken und Prinzipien besser unterstützt werden. Desweiteren werden immer mehr Software Engineering Werkzeuge (CASE-Tools) angeboten, die den Software Herstellungsprozess unterstützen, ebenso wie Frameworks für Webanwendungen, die auf bewährten Konzepten anderer objektorientierter Programmiersprachen basieren.

1.1 Ziel und Aufbau der Arbeit

Diese Arbeit soll den Software Herstellungsprozess einer PHP Webanwendung, unter der Verwendung von Software Engineering Konzepten, untersuchen, dokumentieren und bewerten.

In Kapitel 2 wird zunächst die Programmiersprache PHP vorgestellt und anhand ihrer speziellen Eigenschaften, sowie anhand des Umgangs mit Variablen, Datentypen und Kontrollstrukturen erklärt. Desweiteren werden grundlegende Konzepte des Software Engineering erklärt.

In Kapitel 3 werden vier grundlegende Software Prozess Modelle beschrieben, die jeweils Repräsentanten einer spezifischen Vorgehensweise sind. Diese behandeln die wesentlichen Prozessabläufe Analyse, Entwurf, Implementierung, Test und Inbetriebnahme. Anschließend werden die beschriebenen Prozess Modelle auf PHP bezogen.

In Kapitel 4 werden CASE-Tools für die Versionskontrolle und für den Testprozess vorgestellt. Die CASE-Tools für den Testprozess beziehen sich auf Unit-Testing, Debugging, Lasttests und Profiling-Tests.

In Kapitel 5 werden die IDEs EasyEclipse for PHP und Zend Studio vorgestellt. Anhand einer kleinen Beispielanwendung wird untersucht, inwieweit sie sich für PHP Anwendungen eignen. Anschließend wird bei beiden die Eignung bewertet und ein Fazit gezogen.

In Kapitel 6 wird das neue Objektmodell von PHP 5 vorgestellt. Anhand von Beispielen zu Polymorphismus, Kapselung, Vererbung, abstrakten Klassen und Interfaces, werden objektorientierte Techniken und Prinzipien erklärt. Es werden weitere Neuerungen vorgestellt