



**Westfälische  
Hochschule**

**University of Applied Sciences**  
Gelsenkirchen Bocholt Recklinghausen

# Bachelorarbeit

Titel der Arbeit // Title of Thesis

**Konzeption und Entwicklung einer Continuous-Integration-Strategie  
für Kundenprojekte auf Basis der Shopware-Plattform**

Akademischer Abschlussgrad: Grad, Fachrichtung (Abkürzung) // Degree

**Bachelor of Science (B.Sc.)**

Autorenname, Geburtsort // Name, Place of Birth

**Frederik Bußmann, Coesfeld**

Studiengang // Course of Study

**Informatik.Softwaresysteme**

Fachbereich // Department

**Wirtschaft und Informationstechnik**

Erstprüferin/Erstprüfer // First Examiner

**Prof. Dr.-Ing. Martin Schulten**

Zweitprüferin/Zweitprüfer // Second Examiner

**Martin Knoop**

Abgabedatum // Date of Submission

**01.08.2023**



## Eidesstattliche Versicherung

Bußmann, Frederik

---

Name, Vorname // Name, First Name

Ich versichere hiermit an Eides statt, dass ich die vorliegende Abschlussarbeit mit dem Titel

**Konzeption und Entwicklung einer Continuous-Integration-Strategie für Kundenprojekte auf Basis der Shopware-Plattform**

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Stadtlohn, den 01.08.2023

---

Ort, Datum, Unterschrift // Place, Date, Signature



## **Abstract**

Das Ziel dieser Bachelorarbeit ist die Erarbeitung eines geeigneten Konzepts für das Einbinden von Continuous-Development-Techniken in Shopware-Projekten.

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Unternehmensporträt der best it GmbH . . . . .	1
1.2	Zielsetzung . . . . .	1
1.3	Struktur der Arbeit . . . . .	1
<b>2</b>	<b>Fachlicher Hintergrund</b>	<b>2</b>
2.1	Begrifflichkeiten und Prinzipien von Continuous Integration . . . . .	2
2.2	Übersicht über die Shopware-Plattform . . . . .	3
<b>3</b>	<b>Analyse und Konzept</b>	<b>4</b>
3.1	. . . . .	4
<b>4</b>	<b>Entwicklung der CI/CD-Strategie</b>	<b>5</b>
<b>5</b>	<b>Evaluierung</b>	<b>6</b>
<b>6</b>	<b>Schlussfolgerungen und Ausblick</b>	<b>7</b>
	<b>Literaturverzeichnis</b>	<b>8</b>

**Abkürzungsverzeichnis**

**API** Application Programming Interface - Englisch für „Programmierschnittstelle“, Schnittstelle für die Kommunikation zwischen verschiedenen Softwaresystemen

**CMS** Content Management System - Software-Programm zur Verwaltung von Medien und Text-Inhalten

## **Abbildungsverzeichnis**

## 1 Einleitung

Im Rahmen dieser Arbeit werden verschiedene Aspekte betrachtet, um ein Konzept für die Integration von Continuous-Development-Techniken in Shopware-basierten Projekten zu erarbeiten. Shopware ist eine E-Commerce-Plattform und eine der meistgenutzten Online-Shop-Lösungen in Deutschland<sup>1</sup> welche es Unternehmen ermöglicht, ihre Präsenz im digitalen Markt auszubauen. Das Unternehmen best it GmbH hat für seine Kunden einige Shopware-Projekte im Einsatz, für welche im Nachfolgenden eine Strategie für die automatisierte Prüfung und Auslieferung der entwickelten Software erstellt werden soll.

### 1.1 Unternehmensporträt der best it GmbH

Die best it GmbH ist eine Digitalagentur, die im Jahr 2000 von Manuel Strotmann gegründet wurde und sich auf die Entwicklung von E-Commerce-Lösungen spezialisiert hat. Die Firma ist in verschiedenen Bereichen des Online-Handels tätig und realisiert für ihre Kunden eine Vielzahl von Shopware-Projekten. Neben der Konzeption und Entwicklung von Online-Shops bietet das Unternehmen auch Design- und Beratungsdienstleistungen für Kunden im E-Commerce-Bereich an. Mittlerweile beschäftigt die best it GmbH über 100 Mitarbeiter an drei Standorten in Deutschland und Österreich.

### 1.2 Zielsetzung

Das Ziel dieser Arbeit ist die Entwicklung einer Continuous-Integration-Strategie für auf Shopware basierende Kundenprojekte der Firma best it. Um eine Konzeption zu erstellen, die den Anforderungen des Unternehmens entspricht, werden die Ziele der Strategie nachfolgend definiert:

- **Höhere Entwicklungsgeschwindigkeit:** Die Einführung einer umfangreichen CI-Strategie soll die Effizienz der Softwareentwicklungsteams verbessern und die Zeit bis zum Software-Release senken.
- **Frühe Fehlererkennung:** CI soll dazu beitragen, Fehler frühzeitig im Entwicklungsprozess zu erkennen und zu beheben, was die Qualität des Endprodukts verbessert.
- **Bessere Kommunikation:** Die CI-Strategie und die damit verbundenen Prozesse, die sich für Entwicklerteams ergeben, sollen zu einer verbesserten Kollaboration führen.

Bei der Konzeption sollen diese Ziele verfolgt und die Maßnahmen der zu erarbeiteten CI-Strategie dementsprechend ausgerichtet werden.

### 1.3 Struktur der Arbeit

---

<sup>1</sup> Vgl. eCommerceDB. *The Most Commonly Used Shop Software Among Online Shops in Germany - Shopware and Salesforce Share Rank No. 1.* 2023.



## 2 Fachlicher Hintergrund

Für die Erarbeitung einer geeigneten CI-Strategie werden zunächst die Begrifflichkeiten und Prinzipien von Continuous-Integration definiert. Darüber hinaus wird eine Übersicht über die Funktionen und Mechanismen der Shopware-Plattform gegeben.

### 2.1 Begrifflichkeiten und Prinzipien von Continuous Integration

Das Kernprinzip hinter Continuous Integration (CI) wurde bereits im Jahr 1991 von Grady Booch definiert. Hierbei werden Software-Releases nicht als ein großes Ereignis betrachtet, sondern regelmäßig durchgeführt, wobei die vollständige Software stetig größer wird.<sup>2</sup> Kent Beck popularisierte im Jahr 1998 die Disziplin des „Extreme Programming“, wobei großer Wert auf das frühe und regelmäßige Testen und Integrieren der entwickelten Komponenten einer Software gelegt wird. Beck behauptet hierbei, dass ein Feature, für welches es keine automatisierten Tests gibt, auch nicht funktioniert.<sup>3</sup> Im Jahr 2006 fasste Software-Entwickler Martin Fowler einige Bereiche dieser Methodiken in dem Artikel „Continuous Integration“ unter dem gleichnamigen Begriff zusammen. Fowler beschreibt Continuous Integration als einen Prozess, bei dem Teammitglieder ihre Arbeit regelmäßig integrieren, wobei Integration als der Build-Prozess inklusive automatisierte Tests für die vollständige Software mitsamt der erarbeiteten Änderungen zu verstehen ist.<sup>4</sup> Nachfolgend werden einige wichtige Aspekte von Continuous Integration kurz dargestellt.

#### Regelmäßige Integration

Die namensgebende Methodik der CI ist das regelmäßige Integrieren von Software und damit verbunden ist der Software-Release. In der traditionellen Softwareentwicklung wird der Release als ein einmaliges, großes Ereignis betrachtet. Als Integration wird der Prozess des Einbindens einer einzeln entwickelten Komponente einer Software in die bisher bestehende Gesamtheit der Software bezeichnet, wobei der Release einer Software das Zusammenfinden aller Komponenten und das Ausführen eines Build-Prozesses bis hin zur fertigen, ausführbaren Software beschreibt. In der Vergangenheit wurde bei einem Release jede Einzelkomponente der Software manuell integriert und getestet, wobei dies als eigene Phase im Lebenszyklus eines Programms galt.

#### Automatisierte Tests

In der Regel wird Software vor einem Release getestet, wobei diese Tests in der Vergangenheit manuell durchgeführt wurden.

---

<sup>2</sup> Vgl. Booch. *Object oriented design with applications*. 1991.

<sup>3</sup> Vgl. Beck. „Extreme programming: A humanistic discipline of software development“. 1998, S. 2–4.

<sup>4</sup> Vgl. Fowler. *Continuous Integration*. 2006.

## Reproduzierbarkeit

## Feedback

### 2.2 Übersicht über die Shopware-Plattform

Shopware wurde als Online-Shop-Software im Jahr 2000 durch Stefan Hamann ins Leben gerufen<sup>5</sup> und bietet heute in ihrer aktuellen Major-Version 6 eine moderne E-Commerce-Plattform auf Basis des PHP-Frameworks „Symfony“. Das Symfony-Framework wird neben Shopware noch von anderen PHP-Basierten Projekten wie dem CMS „Drupal“, dem Shop-System „Magento“ und einigen weiteren Programmen<sup>6</sup> als Grundlage genutzt und bildet somit ein erprobtes Fundament für die Shopware-Plattform. Shopware selbst ist nach der Installation bereits voll funktionsfähig und kann mit einem Backend und optional mit einem Frontend oder für das Konsumieren der mitgelieferten API eingerichtet werden. Die Software kann auf verschiedenen Plattformen gehostet werden, darunter Linux-Server und containerisierte Umgebungen. Darüber hinaus bietet Shopware als Unternehmen auch eine eigene Hosting-Lösung an, die speziell auf die Anforderungen der Software zugeschnitten ist. Die Plattform kann sowohl im Einzelbetrieb als auch als Cluster genutzt werden, um eine hohe Verfügbarkeit und Skalierbarkeit zu gewährleisten.

---

<sup>5</sup> Vgl. Shopware. *The story behind Shopware AG*. 2023.

<sup>6</sup> Vgl. SensioLabs. *Projects using Symfony - Popular PHP projects using Symfony components or based on the Symfony framework*. 2023.

## 3 Analyse und Konzept

Im folgenden Kapitel werden verschiedene Techniken zur kontinuierlichen Integrierung von Code analysiert und eine Konzeption für Shopware-basierte Projekte erarbeitet.

### 3.1

Um Kundenprojekte mit Shopware umsetzen zu können, muss zunächst definiert sein, welche Bereiche der Plattform anpassbar sind und wie diese Anpassungen zustande kommen. Da das Shop-Backend und Frontend für Kunden erweiterbar sein soll, werden sogenannte „Plugins“ verwendet. Bei Plugins handelt es sich um Erweiterungen in Form von Code, welche einer durch die Plattform vorgegeben Ordner- und Datei-Struktur folgt. Hierbei können sowohl die für verschiedene Backend-Bereiche zuständigen PHP-Dateien als auch die für das Back- und Frontend verwendeten Templates, JavaScript-Logik und Style-Dateien erweitert werden. Für die Analyse der kontinuierlichen Integrierung der Kundenprojekte wird sich somit im folgenden auf die von Shopware vorgegebenen Erweiterungsmöglichkeiten bezogen.

## 4 Entwicklung der CI/CD-Strategie

## 5 **Evaluierung**

## **6 Schlussfolgerungen und Ausblick**

## Literaturverzeichnis

- [1] eCommerceDB. *The Most Commonly Used Shop Software Among Online Shops in Germany - Shopware and Salesforce Share Rank No. 1*. eCommerceDB GmbH. 2023. URL: <https://ecommercedb.com/insights/the-most-commonly-used-shop-softwares-among-online-shops-in-germany-shopware-and-salesforce-share-rank-no-1/4210> (besucht am 22. 06. 2023).
- [2] Grady Booch. *Object oriented design with applications*. 1. Aufl. Benjamin/Cummings Pub. Co, 1991. ISBN: 978-0-805-30091-8.
- [3] Kent Beck. „Extreme programming: A humanistic discipline of software development“. In: *Fundamental Approaches to Software Engineering*. Springer Berlin Heidelberg, 1998. DOI: [10.1007/bfb0053579](https://doi.org/10.1007/978-3-642-53579-9). URL: <https://doi.org/10.1007/978-3-642-53579-9>.
- [4] Martin Fowler. *Continuous Integration*. Martin Fowler. 2006. URL: <https://martinfowler.com/articles/continuousIntegration.html> (besucht am 26. 06. 2023).
- [5] Shopware. *The story behind Shopware AG*. Shopware AG. 2023. URL: <https://www.shopware.com/en/company/story/> (besucht am 25. 06. 2023).
- [6] SensioLabs. *Projects using Symfony - Popular PHP projects using Symfony components or based on the Symfony framework*. SensioLabs. 2023. URL: <https://symfony.com/projects> (besucht am 24. 06. 2023).