

## **Rollenideal »Digital Design«**

Erfolgreiche Digitalisierung und Digitale Transformation erfordern ein Umdenken in der Softwareentwicklung

**bitkom**

### **Herausgeber**

Bitkom e.V.  
Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V.  
Albrechtstraße 10 | 10117 Berlin

### **Ansprechpartner**

Dr. Frank Termer | Bereichsleiter Software  
T 030 27576-232 | f.termer@bitkom.org

### **Verantwortliches Bitkom-Gremium**

Software-Gestalter

### **Gesamtkoordination & Redaktion**

Dr. Kim Lauenroth | adesso AG/IREB e.V.

### **Satz & Layout**

Katrin Krause | Bitkom

### **Copyright**

Bitkom 2017

Diese Publikation stellt eine allgemeine unverbindliche Information dar. Die Inhalte spiegeln die Auffassung im Bitkom zum Zeitpunkt der Veröffentlichung wider. Obwohl die Informationen mit größtmöglicher Sorgfalt erstellt wurden, besteht kein Anspruch auf sachliche Richtigkeit, Vollständigkeit und/oder Aktualität, insbesondere kann diese Publikation nicht den besonderen Umständen des Einzelfalles Rechnung tragen. Eine Verwendung liegt daher in der eigenen Verantwortung des Lesers. Jegliche Haftung wird ausgeschlossen.

# Rollenideal »Digital Design«

Erfolgreiche Digitalisierung und Digitale Transformation erfordern ein Umdenken in der Softwareentwicklung

# Inhaltsverzeichnis

<b>Danksagung</b>	<b>3</b>
<b>1 Einführung und Motivation</b>	<b>5</b>
1.1. Digitale Transformation und Softwareentwicklung	5
1.2. Softwareentwicklung als Realisierung	6
1.3. Das Gestaltungsvakuum in der Softwareentwicklung	7
1.4. Aufbau des vorliegenden Berichts	8
<b>2 Stand der Praxis zur Gestaltung im Software Engineering – Wahrgenommene Ausgangssituation</b>	<b>10</b>
<b>3 Betrachtung der aktuellen Ausbildungslage im Software Engineering mit Blick auf Gestaltungskompetenz</b>	<b>12</b>
<b>4 Design meets IT – eine kleine Bestandsaufnahme zum Zusammenspiel von IT und Design</b>	<b>15</b>
4.1. Impulsvorträge	15
4.2. Podiumsdiskussion	16
<b>5 Fazit – Der »Digital Designer« als Gestaltungsrolle</b>	<b>18</b>
5.1. Herleitung des Begriffs »Digital Designer«	18
5.2. Digital Designer als idealisiertes Bild und Brückenkopf	19
5.3. Abgrenzung zu anderen idealisierten Rollenbildern	19
5.4. Kompetenzprofil »Digital Designer«	21
<b>6 Zusammenfassung und nächste Schritte</b>	<b>24</b>
<b>7 Literatur</b>	<b>26</b>

# Danksagung

Software ist überall: Sie steckt in Autos, Flugzeugen, Mobiltelefonen, Haushaltsgeräten, Filmen und Spielen. Damit ist Software auch einer der wesentlichen Treiber der Digitalen Transformation. Die Entwicklung qualitativ hochwertiger und sicherer Software spielt in allen wirtschaftlichen Bereichen, bspw. in der Industrie 4.0, im Gesundheitswesen, der Energieversorgung, bei der Mobilität, im Bereich der Dienstleistungen, aber auch in der öffentlichen Verwaltung eine zentrale Rolle. Entsprechend ist die Fähigkeit zur Entwicklung hochwertiger Software eine Kernkompetenz, die über die Zukunftsfähigkeit des Standorts Deutschland entscheidet.

Die Softwareentwicklung hat sich in den vergangenen Jahren dramatisch beschleunigt. Innovation und Weiterentwicklung finden in immer kürzer werdenden Zyklen statt. Software wird nicht mehr nur mittels klassischer Vorgehensmodelle sondern auch mit Hilfe agiler Ansätze und Methoden entwickelt, um dieser gestiegenen Geschwindigkeit bei angemessener Qualität und Sicherheit Rechnung zu tragen. Damit einher geht auch die gemeinsame Anwendung von Agilität und ingenieurmäßigem Vorgehen bei der Softwareentwicklung. Dies hat zur Folge, dass Softwareentwicklung zunehmend in cross-funktionalen Teams gemeinsam entwickelt wird. Die existierende und durch die aktuelle Gestaltung der IT-Ausbildung weiter konstruierte Zersplitterung der Softwareentwicklung in einzelne Teildisziplinen und damit auch Rollen wird zunehmend zum Hemmnis bei der Bewältigung der geschilderten Herausforderungen in der Softwareentwicklung.

Die Taskforce »Software-Gestalter« hat sich diese Situation zum Anlass und als Ausgangspunkt genommen, um über neue Wege und Notwendigkeiten nachzudenken, wie die Softwareentwicklung im Zuge der Digitalen Transformation ebenfalls verändert werden muss. Das vorliegende Dokument beschreibt das Ergebnis dieser Überlegungen, welche im Diskurs mit zahlreichen Unternehmen, Einrichtungen der Wissenschaft und Forschung sowie der interessierten Öffentlichkeit entstanden sind. Für die Mitarbeit am Ergebnisdokument danken wir den folgenden Personen, die mit Herzblut und hohem persönlichen Engagement dessen Erstellung möglich gemacht haben:

- Dr. Kim Lauenroth (adesso AG/IREB e.V.)
- Prof. Dr. Karsten Lehn (Hochschule Hamm-Lippstadt)
- Ulf Schubert (Datev eG)
- Dr. Marcus Trapp (Fraunhofer Institut für Experimentelles Software Engineering – IESE)

Weiterhin danken wir folgenden Personen, die sich aktiv in die Arbeit der Taskforce eingebracht und wertvolle Hinweise bei der Erstellung des Dokuments geliefert haben:

- Kai Bünseler (Wirtschaftsförderung Dortmund)
- Ralf Dehnhof (iSAQB e.V.)
- Thomas Geis (ProContext Consulting GmbH/UXQB e.V.)
- David Gilbert (DB Systel GmbH)
- Dr. Ronald Hartwig (untrouble GmbH)
- Michael Mahlberg (iSAQB e.V.)
- Prof. Dr. Sabine Radomski (Hochschule für Telekommunikation Leipzig)
- Dr. Frank Simon (Zurich Versicherung/GTB e.V.)

# 1 Einführung und Motivation

# 1 Einführung und Motivation

## 1.1. Digitale Transformation und Softwareentwicklung

Die Schlagworte »Digitalisierung« und »Digitale Transformation« umschreiben die Veränderung der Gesellschaft und der Wirtschaft durch die sukzessive Einführung digitaler Technologien (Internet, Computer, Smartphones etc.). Die Beispiele für diesen Wandel reichen vom Einkaufen im Internet (bspw. Amazon, Ebay) über die Digitalisierung von Medien (Musik, Filme, Bücher, Zeitungen, usw.) sowie die Integration sozialer Netzwerke (bspw. Facebook, Twitter, XING) in unser tägliches Leben bis hin zu vollkommen neuen Geschäftsmodellen auf digitaler Basis (bspw. Airbnb, Uber, Spotify). In der ganzen Welt werden große Konferenzen ausgelobt, um sich dem Thema zu widmen. Die G20 hielten unter Führung Deutschlands in 2017 sogar eine eigene G20-Konferenz zum Thema »Digitalisierung« ab.

Betrachtet man die Softwareentwicklung als Disziplin, kommt ihr eine zentrale Rolle in der Digitalisierung zu: Ohne Software kann keine Digitalisierung stattfinden. Erstaunlich ist jedoch, dass das Thema »Digitalisierung« oder »Digitale Transformation« in der Softwarebranche kaum eine Rolle zu spielen scheint und eher als Hype gesehen wird. Auf einschlägigen Software-Konferenzen (Industrie und Wissenschaft) dominieren Themen wie »Cloud«, »Big Data«, »Microservices«, »Internet of Things« oder »Agile Entwicklung«. Überspitzt formuliert kümmert man sich um sich selbst, d.h. man befasst sich mit neuen Technologien, Werkzeugen oder mit Vorgehensweisen zur Entwicklung. Spricht man mit verschiedensten Experten, so scheint die einhellige Meinung zu sein, dass das Thema »Digitalisierung« doch schon immer Teil der Softwarebranche war, da man ja auch schon immer Software entwickelt hat. Diese Perspektive ist stark technologie-zentriert gedacht. Es mangelt vor allem an der Auseinandersetzung mit der Wirkung von Digitalisierung für Menschen, Unternehmen und Gesellschaft. Zugegebenermaßen ist diese Sicht extrem verkürzt und vereinfacht, dennoch entsteht das Bild, dass sich in den letzten Jahren eine größer werdende Kluft zwischen der Welt der Nutzer der Digitalisierung und der Welt der Realisierer der Digitalisierung entwickelt hat.

Diese Kluft führt dazu, dass sich ein Teil der Softwareentwicklung – insbesondere in größeren, technologiefernen Unternehmen – immer stärker in der Rolle des passiven Realisierers eingerichtet hat und sich dieses Rollenbild auch immer stärker in den Köpfen der handelnden Personen auf beiden Seiten (Kunden/Auftraggeber/Nutzer bzw. Entwicklung) verfestigt hat. Diese recht harte Diagnose ist vermutlich in der Historie der Softwareentwicklung begründet. Von professioneller Softwareentwicklung kann man seit den 1950er/60er Jahren sprechen. Ein wesentlicher Treiber für die Softwarebranche war sicherlich die Geschäftswelt. Banken und Versicherungen haben in Computern und in Software ein Werkzeug gesehen, das ihre bisherigen Geschäftsprozesse effizienter machen kann. Konkret wurden verstandene Prozesse (bisher auf Basis von Papier und Menschen realisiert) auf den Rechner übertragen. Schon hier könnte man von einer frühen Form der Digitalisierung sprechen.

Die Herausforderung für die Softwareentwicklung bestand in dieser Situation darin, die bestehenden Prozesse und Sachverhalte zu verstehen und geeignet in qualitativ hochwertige Software zu überführen. Zwei Herausforderungen dominierten für eine lange Zeit die Softwareentwicklung (Vgl. Boehm 2006):

1. Die geringe aber wachsende Leistungsfähigkeit der vorhandenen Hardware: Unter anderem begrenzte Speicherkapazität, Rechengeschwindigkeit, Übertragungskapazität von Netzwerken und Verfügbarkeit von leistungsfähigen Endgeräten.
2. Die Entwicklung von qualitativ hochwertiger Software, d.h. möglichst fehlerfreie und den Vorgaben der Kunden entsprechende Software.

Die erste Herausforderung lag zwar außerhalb der Softwareentwicklung, hatte aber einen massiven Einfluss auf die Anwendbarkeit und Verbreitung von Software. Ohne akzeptable Hardwarevoraussetzungen konnte auch die beste Software keinen wirklichen Nutzen erzielen oder neue Nutzer gewinnen. Durch den technologischen Fortschritt wurde diese Situation jedoch kontinuierlich besser. Die zweite Herausforderung führte dazu, dass die Softwareentwicklung zunehmend als Ingenieursdisziplin aufgefasst wurde und man versuchte, durch ingenieurmäßiges Vorgehen bessere Qualität zu erzielen. So wurde der Begriff Software Engineering geprägt. Bereits in den 70er Jahren (vgl. Royce 1970) kam das oft missverstandene Wasserfallmodell auf, auf dessen Basis viele Ansätze fußen, die Entwicklungsarbeit so zu strukturieren wie einen Umsetzungsprozess. Seit Mitte der 1990er Jahre haben die sogenannte Agile Entwicklung und das Human-Centered Design zunehmend an Bedeutung gewonnen und sich als alternative Modelle zur Entwicklung von Software etabliert.

## 1.2. Softwareentwicklung als Realisierung

Unterm Strich kann man festhalten, dass die Softwareentwicklung in einer Welt gewachsen ist, in der verstandene Prozesse und Abläufe in Software übertragen wurden und in der die verfügbare Technologie immer den Erwartungen der Kunden/Nutzer hinterhergelaufen sind. Ihren Höhepunkt hatte diese Entwicklung in der sogenannten Dotcom-Blase: die Erwartungen an IT und an Software schossen in ungeahnte Höhen. Die Blase platze, als allen Beteiligten klar wurde, dass diese Erwartungen auf absehbare Zeit nicht erfüllt werden konnten.

Seit dem Platzen der Dotcom-Blase hat sich die Situation, nahezu unmerklich, verändert. Die technologische Leistungsfähigkeit von Hard- und Software hat ein Niveau erreicht, dass für einen Großteil der Anwendungsfelder mehr als ausreichend ist. Speicherkapazität und Rechenleistung sind im Überfluss vorhanden genauso wie preiswerte Endgeräte für alle Kundenschichten. Hinzu kommt die Etablierung des Internets als massenverfügbares und leistungsfähiges Kommunikationsmedium für stationäre und insbesondere tragbare Endgeräte. Im Endergebnis führte dieser Wandel dazu, dass die technischen Möglichkeiten seit einigen Jahren wesentlich stärker gewachsen sind als die Erwartungen an Software. Man kann sogar sagen, dass Software heute viel mehr leisten kann, als die Kundenseite von ihr erwartet. Die Software-Entwicklung ist allerdings an vielen Stellen noch nicht so weit, wie die technischen Möglichkeiten. Hinzu kommt, dass Unternehmen immer noch schwer fällt neue Prozesse, Services oder Geschäftsmodelle zu erdenken und zu formulieren, die ohne Software gar nicht möglich wären. Dies ist aber heute unabdingbar, denn es geht schon lange nicht mehr nur darum, existierende Abläufe zu digitalisieren. Dieses Phänomen wird auch gerne unter dem Schlagwort der Digitalen Transformation zusammengefasst.



Dieser Wandel verändert das Aufgabenspektrum der Softwareentwicklung fundamental. Die implizite Annahme, dass die Kundenseite der Softwareentwicklung sagt, was sie zu entwickeln hat, trifft in vielen Vorhaben nicht mehr oder nur noch in Teilen zu. Stattdessen kann die Kundenseite bestenfalls vage formulieren, was die Benutzer mit dem System tun wollen. Leider werden all zu oft nur Analogien bemüht (»Es muss so sein wie bei Google«), um die eigenen Vorstellungen zu beschreiben. Diese geänderte Situation trifft auf eine Softwareentwicklung, die vollständig darauf fokussiert ist, möglichst genaue Beschreibungen darüber zu erhalten, was zu realisieren ist. Zwangsläufig musste dies zu Problemen in der Softwareentwicklung und zu suboptimalen Ergebnissen und Produkten führen. Die Verbreitung der zuvor bereits genannten agilen Entwicklung als neues Vorgehensmodell wurde durch diese Situation vermutlich erst möglich gemacht oder zumindest stark begünstigt. Aktuelle Standards (vgl. bspw. Bourque und Fairley 2014) bezeichnen die agile Entwicklung als eine der zentralen Methoden zur Entwicklung von Software.

Für die bisherigen Betrachtungen sind die folgenden Kerngedanken der agilen Entwicklung relevant: intensive regelmäßige Kommunikation zwischen Entwicklung und Kundenseite, ein zentraler Ansprechpartner (je nach Modell zum Beispiel als »Onsite Customer« oder »Product Owner« bezeichnet) und schnelle Feedbackzyklen auf Basis realisierter Software. Durch diese Ideen konnten Management und Kunden der Softwareentwicklung ihr bisheriges Weltbild (»Ein Anderer definiert, was zu tun ist«) beibehalten, ohne dass auf Kundenseite ein konkretes Verständnis über die zu realisierende Software bestehen musste. Durch die intensive Kommunikation und regelmäßiges Feedback kann sich jetzt auf beiden Seiten ein konkreter werdendes Verständnis über die entstehende Software entwickeln. Auch diese Form der Entwicklung kommt an ihre Grenzen, sobald sich auf Kundenseite kein konkretes Verständnis über die Nutzung der geplanten Software einstellen will. In solchen Situationen kann das iterative Vorgehen nicht hin zu einer akzeptierten Lösung konvergieren.

### 1.3. Das Gestaltungsvakuum in der Softwareentwicklung

Gestartet sind unsere Überlegungen mit dem Thema Digitalisierung/Digitale Transformation und der Feststellung, dass Softwareentwicklung als Disziplin scheinbar keine aktive Rolle bei diesem Thema spielt, obwohl Software das zentrale Element der Digitalisierung ist. Vielmehr scheint es eine tiefe Kluft zwischen beiden Welten zu geben. Die Ursache hierfür wird in der Historie der Disziplin vermutet. Zusammengefasst führt diese Kluft dazu, dass ein Gestaltungsvakuum entsteht und relevante Entscheidungen in Bezug auf die Gestaltung von Software eher zufällig getroffen werden.

Der Begriff »Gestaltung« bzw. der Begriff »Design« wird in der Literatur sehr unterschiedlich definiert (vgl. bspw. Erlhoff und Marshall 2008). In diesem Dokument werden die Begriffe »Gestaltung« und »Design« synonym verwendet und wie folgt verstanden: Gestaltung umfasst sämtliche Aspekte von System, Softwareprodukten und -services verstanden, die ein Endnutzer erleben kann, d.h. die Form (bspw. Oberfläche oder verwendetes Endgerät), die Funktion (d.h. die Fähigkeiten bzw. der Zweck, den die Software im Kontext erfüllt), emotionale Eigenschaften (wie z.B. Ästhetik) und qualitative Eigenschaften (bspw. Reaktionsgeschwindigkeit). Diese Aspekte

werden natürlich maßgeblich durch die technologischen Möglichkeiten von Software beeinflusst, da die Technologie diese Möglichkeiten bereitstellt. Unter Gestaltung an sich wird der kreative Schaffensprozess verstanden, bei welchem durch die Arbeit des Gestaltenden eine Sache (ein materielles Objekt, eine Struktur, ein Prozess, eine Situation, ein Gedankengut usw.) verändert wird, d. h. erstellt, modifiziert oder entwickelt wird und dadurch eine bestimmte Form oder ein bestimmtes Erscheinungsbild verliehen bekommt oder annimmt (vgl. Davis 2003).

Ohne eine strukturierte Gestaltungskompetenz kann die Softwareentwicklung ihre eigentliche Stärke, nämlich das Wissen um die Leistungsfähigkeit, die Möglichkeiten und die Grenzen von Software zur Besserung von existierendem Business oder zur Erschaffung von neuem Business nur unzureichend oder zu einem viel zu späten Zeitpunkt im Vorhaben im Kontext der Digitalisierung einbringen kann. Viele Vorhaben bleiben so vermutlich weit hinter ihren Möglichkeiten zurück bzw. schlagen gänzlich fehl.

Vor dem Hintergrund dieser Entwicklung hat sich die Taskforce »Software-Gestalter« formiert, um innerhalb des Bitkom und in Kooperation mit den Arbeitskreisen des Kompetenzbereichs Software zu beleuchten, ob das zuvor skizzierte Gestaltungsvakuum im Software Engineering noch zeitgemäß ist und wie sich das Software Engineering in Zukunft mit Blick auf Gestaltungskompetenz positionieren sollte. Grundsätzliche Positionen können sein: besser aktiv als Software-Gestalter oder weiterhin eher passiv als Software-Umsetzer.

## 1.4. Aufbau des vorliegenden Berichts

Der vorliegende Bericht fasst die Ergebnisse der Taskforce Software-Gestalter zusammen und ist wie folgt aufgebaut. Zunächst erfolgt eine Betrachtung zum Stand der Praxis in der Softwareentwicklung (SE) und zur aktuellen Ausbildungslage zum Thema Gestaltung in der Softwareentwicklung. Anschließend wird über die Ergebnisse eines Austauschs mit anderen Gestaltungsdisziplinen zum Thema Gestaltung in der Softwareentwicklung berichtet. Als Fazit dieser Ergebnisse wird der »Digital Designer« als neues gestaltungsübergreifendes und idealisiertes Rollenbild hergeleitet, definiert und beschrieben. Dieses idealisierte Rollenbild soll als Brückenschlag zwischen bestehenden Disziplinen des Software Engineering, des Software Management und dem Kunden, sowie als Ausblick auf die weitere Entwicklung dienen. Abschließend wird ein Ausblick auf weitere Aktivitäten und nächste Schritte zum Thema »Digital Designer« gegeben. Im Verlauf des Textes erfahren Sie auch, was Digital Design mit pi zu tun hat.

# 2

Stand der Praxis zur Gestaltung  
im Software Engineering –  
Wahrgenommene  
Ausgangssituation

## 2 Stand der Praxis zur Gestaltung im Software Engineering – Wahrgenommene Ausgangssituation

Betrachtet man andere etablierte Disziplinen, so haben sich dort eigenständige Rollenbilder oder sogar ganze Kompetenzfelder entwickelt, die sich der Gestaltung widmen. Prominente Beispiele sind hier der Architekt aus dem Bauwesen und der Industriedesigner aus der Produktentwicklung. Zentrale Merkmale dieser beiden Rollen sind, dass sie das Bindeglied zwischen den Kunden/Nutzern und der Herstellung/Realisierung bilden und die Hauptverantwortung für das Endergebnis (ein Gebäude oder ein Produkt) übernehmen. Die Entstehung dieser Rollenbilder motivierte sich aus der Tatsache, dass die Tätigkeiten im jeweiligen Feld (Bauwesen/Produktentwicklung) einen Grad an Komplexität erreicht haben, der von einer einzelnen Person kaum noch zu bewältigen war.

Parallelen kann man durchaus auch in der Softwareentwicklung beobachten. Die Komplexität von Software als »Materie« führte zur Ausprägung verschiedenster Spezialfelder. Beispielsweise listet der aktuelle IEEE Software Engineer Body of Knowledge 15 Kompetenzfelder für die Softwareentwicklung. Hervorhebenswert ist der Teil »Software Design« (SWEBOK V3.0, vgl. Bourque und Fairley 2014, Seite 2-2):

Software design is generally considered a two-step process:

- Architectural design (also referred to as high-level design and top-level design) describes how software is organized into components.
- Detailed design describes the desired behavior of these components.

Diese Definition richtet sich primär nach innen, d.h. auf die zu entwickelnde Software und ihren Aufbau. In diesem Zusammenhang wird gerne auch von Software-Architektur und Software-Architekten gesprochen. Der Software-Architekt kann an dieser Stelle üblicherweise nicht mit der Rolle des Architekten aus dem Bauwesen verglichen werden.

Das SWEBOK zählt auch die Gestaltung der Benutzungsschnittstelle (User Interface Design) zum Software Engineering. Für die Benutzungsschnittstelle als Teilaspekt von Software haben sich weitere Disziplinen mit Gestaltungsfokus entwickelt, z.B. User Experience, Usability Engineering, Service Design, Information Architecture und Interaction Design. Diese Disziplinen können als etabliert betrachtet werden, sind aber nicht in einem Maße präsent oder führend in der Softwareentwicklung wie Architekten im Bauwesen bzw. Industriedesigner in der Produktentwicklung. Für die Ausprägung einer solchen ganzheitlichen und verantwortlichen Gestaltungsrolle für Software bestand bisher vermutlich keine Notwendigkeit, da diese Aufgabe durch die Kundenseite übernommen wurde (siehe Kapitel 1).

# 3

Betrachtung der aktuellen  
Ausbildungslage im Software  
Engineering mit Blick auf  
Gestaltungskompetenz

### 3 Betrachtung der aktuellen Ausbildungslage im Software Engineering mit Blick auf Gestaltungskompetenz

Zur Beantwortung der Fragestellung, in welchen Studiengängen bereits heute weitgehende Gestaltungskompetenzen im Zusammenhang mit der Ausbildung in der Softwareentwicklung vermittelt werden, wurde auf Basis der öffentlich zugänglichen Datenbank »Hochschulkompass« ([↗www.hochschulkompass.de](http://www.hochschulkompass.de)) eine Statistik über Informatikstudiengänge mit einem bedeutenden Anteil an der Lehre von Gestaltungsdisziplinen<sup>1</sup> erstellt. Diese Datenbank wurde unter anderen Portalen für die Information über Studiengänge als gute Grundlage ausgewählt, da sie zu der offiziellen Präsenz der Hochschulrektorenkonferenz (HRK) gehört, in der 268 staatliche und staatlich anerkannte Hochschulen aus dem gesamten Bundesgebiet Mitglied sind. Insgesamt werden durch diese Hochschulen ca. 94 Prozent der in Deutschland eingeschriebenen Studierenden repräsentiert (vgl. Hochschulrektorenkonferenz, 2017a).

Studiengänge der Informatik	Anzahl	Anteil
..., alle, grundständig	672	100%
... mit Gestaltung, grundständig	42	6,3%
..., alle, weiterführend	504	100%
... mit Gestaltung, weiterführend	21	4,2%
..., alle, grundständig und weiterführend	1176	100%
... mit Gestaltung, grundständig und weiterführend	63	5,4%

Daten basierend auf [↗www.hochschulkompass.de](http://www.hochschulkompass.de), Stand: 22.12.2016

Tabelle 1: Studiengänge der Informatik und Studiengänge der Informatik mit einem bedeutenden Anteil an Gestaltungsdisziplinen in Deutschland

Wie aus Tabelle 1 ersichtlich ist, beträgt der Anteil an Studiengängen mit einem bedeutenden Anteil an Gestaltungsdisziplinen 5,4%. Studiengänge, die mit einem Bachelor oder einem Diplom (FH) abschließen (grundständig), nehmen einen Anteil von, 6,3% und Studiengänge, in denen ein Masterabschluss erworben werden kann (weiterführend), einen Anteil von, 4,2% ein. Diese absoluten Zahlen erscheinen als gering und liefern somit einen ersten Eindruck über die bestehende Situation.

1 Im Zusammenhang dieser Erhebung wurde »Design« und »Gestaltung« und »Designdisziplin« und »Gestaltungsdisziplin« als jeweils gleichbedeutend verwendet (siehe auch Erlhoff und Marshall (2008), S. 176 ff.)

Um eine quantitative Aussage treffen zu können, inwieweit ein Verbesserungsbedarf besteht, sind weitere Faktoren zu berücksichtigen und ist ein Referenz- oder Zielwert zu bestimmen. Weitere Faktoren können die Anzahl der Absolventen pro Studienjahr und Studiengang und die Anzahl der Absolventen sein, die eine Tätigkeit in der Softwareindustrie aufnehmen. Ein Referenzwert kann auf Basis von Erhebungen in Unternehmen ermittelt werden.

### Studiengänge der Informatik mit und ohne Gestaltungsdisziplinen in Deutschland

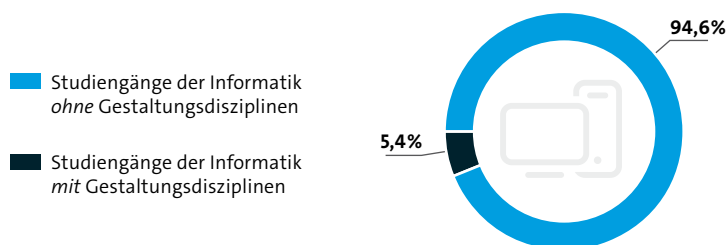


Abbildung 1: Studiengänge der Informatik mit und ohne Gestaltungsdisziplinen in Deutschland (basierend auf [www.hochschulkompass.de](http://www.hochschulkompass.de), Stand 22.12.2016)

In Abbildung 1 ist der Anteil der Informatikstudiengänge in Deutschland mit und ohne einem bedeutenden Gestaltungsanteil gegenübergestellt. Dabei machen die Informatikstudiengänge mit der Vermittlung von Gestaltungskompetenz einen Anteil von 5,4% von allen Informatikstudiengängen in Deutschland aus. Es wurden hier nicht nur Studiengänge der Softwaretechnik, sondern allgemeiner Studiengänge der Informatik betrachtet, da auch Absolventen allgemeiner Informatikstudiengänge potenziell in der Softwareentwicklung arbeiten und arbeiten können. Die Aussage des Ergebnisses ist damit allgemeiner, der Gestaltungsanteil in der kleineren Gruppe von Studiengängen der Softwaretechnik könnte jedoch höher sein.

Einem Studiengang wurde genau dann ein »bedeutender Gestaltungsanteil« zugeschrieben, wenn in der Schlüsselwortsuche der verwendeten Datenbank eine Gestaltungsdisziplin in einem Informatikstudiengang genannt wurde. Es ist davon auszugehen, dass in diesem Fall die betreffende Gestaltungsdisziplin grundlegend und intensiv gelehrt wird. Nicht erfasst sind Studiengänge, in denen ein geringer Anteil an Gestaltungsdisziplinen gelehrt wird, so dass dieser nicht bedeutend genug war, um in die Liste der Stichworte für die Studiengangsbeschreibung und -klassifizierung aufgenommen zu werden.

# 4

Design meets IT – eine kleine  
Bestandsaufnahme zum  
Zusammenspiel von IT und Design



# 4 Design meets IT – eine kleine Bestandsaufnahme zum Zusammenspiel von IT und Design

Um den Austausch zwischen den Design- und IT-Rollen in Gang zu bringen, fand am 04.05.2017 im Dortmunder U die Bitkom-Veranstaltung <sup>↗</sup>»Design meets IT« statt. Konkret ging es dabei um die Fragestellung »Was kann die IT-Branche von Designern« lernen. 219 Teilnehmer aus den Bereichen Produkt-/Softwareentwicklung, UX und Design diskutierten auf Basis einiger Impulsvorträge gemeinsam mit den Referenten Andrea Augsten, Andreas Enslin, Marc Hassenzahl, Frank Jacob, Uwe Kemker und Bernd Wiesenauer über mögliche Antworten.

## 4.1. Impulsvorträge

Die Impulsvorträge sollten zum einen Einblick in die Arbeitswelt und Denkweise von Designern bieten. Sie zeigten darüber hinaus aber auch auf, welchen Wert Designer in Unternehmen stiften können:

- Andrea Augsten (Gründerin von design:transfer) stellte in ihrem Vortrag das menschenzentrierte Gestalten von Produkten, Services und Organisationen in den Vordergrund. Sie griff kritisch und explorativ aktuelle Phänomene auf und stellte Eindrücke aus der Praxis vor.
- Andreas Enslin (Leitung Designcenter, Miele & Cie. KG) beschäftigte sich in seinem Beitrag anhand von Beispielen aus seiner täglichen Praxis mit guter Produktgestaltung. Er beschrieb die interdisziplinäre Zusammenarbeit zwischen Design, Business und Technik dabei als einen wesentlichen Erfolgsfaktor für gute, relevante und erfolgreiche Produkte.
- Marc Hassenzahl (Professor für »Ubiquitous Design/Erlebnis und Interaktion« an der Universität Siegen) rief in seinem Beitrag zum Wohlbefinden und Erleben zur Versöhnung von »Erlebnis haben« und »Technik nutzen« im Gestaltungsprozess auf. Also nicht nur Dinge zu automatisieren oder Techniken einzusetzen, weil es machbar ist, sondern sich immer zu fragen welches Bedürfnis befriedigt wird und darauf zu achten, dass das Erlebnis nicht zu flach und damit bedeutungslos wird.
- Frank Jacob (Professor für Interfacedesign an der Muthesius Kunsthochschule in Kiel und Inhaber der HID Human Interface Design GmbH) widmete sich in seinem Vortrag der Rolle und der Bedeutung des Entwerfens in der täglichen Arbeit von Designern. Aus seiner Sicht können sich IT und Design beflügeln, wenn sie in einer Werkzeugkette zusammenarbeiten und auf Basis der jeweiligen Arbeitsergebnisse aufsetzen anstatt diese über gegenseitige Dokumentationen und Erklärungen zu vermitteln.
- Uwe Kemker (Head of Industrial Design, Vorwerk & Co. KG) stellte den Designprozess bei Vorwerk vor und ging auf die Herausforderungen ein, vor denen Designer stehen, wenn sie Kundennutzen durch eine gute Produktgestaltung erreichen wollen.
- Bernd Wiesenauer (Senior Manager User Experience, Robert Bosch GmbH) berichtet in seinem Vortrag über die Erfahrungen, die er bei Bosch bei der Einführung von User Experience und Design Thinking im Rahmen des digitalen Wandels gemacht hat.

## 4.2. Podiumsdiskussion

Die anschließende Diskussion im Panel und im Publikum zeigte sehr deutlich auf, dass die Ränder der Disziplinen in IT und Design noch unschärfer werden müssen, um in der schnelllebigen IT-Welt bessere und erfolgreichere Ergebnisse zu erreichen. Eine Wortmeldung aus dem Publikum fasste den Status Quo provokant wie folgt zusammen:

*»Die Lösungen, die entstehen, wenn Design und Entwicklung nicht miteinander reden, sind halt einfach Scheiße, wenn man ehrlich ist.«*

Die Spezialisierungen der Disziplinen von IT und Design sind zwar notwendig, damit die notwendige Expertise in den Projekten vorhanden ist, aber alle Disziplinenvertreter müssen über ein gemeinsames Überblickswissen über die anderen beteiligten Disziplinen verfügen. Die Schaffung des gemeinsamen Wissens ist eine große Herausforderung für die Ausbildung der Disziplinen in der IT. Eine Wortmeldung aus dem Publikum fasste dies wie folgt zusammen:

*»Reden Sie mit Ihren ITlern. Die beißen nicht, die wollen nur spielen.«*

Die Vorträge und die Diskussionen zeigten zum einen auf, dass es aktuell schon Bestrebungen gibt, die diesen Brückenschlag versuchen und auch erste Erfolge damit erzielen. Sie zeigten aber auch klare Defizite in der Ausbildung von Software-Entwicklern auf. Ein möglicher Weg, das zu adressieren, wäre es, verstärkt auf Projektstudiengänge zu setzen. Weiterhin sollten sich mehr Masterstudiengänge im Design-Bereich für Studenten aus anderen Disziplinen öffnen.

Um die Disziplinen näher zusammenzubringen, muss auch die Industrie ihren Teil zu dieser Veränderung beitragen, indem sie weniger nach hochspezialisierten Experten in IT und Design sucht. Wenn mehr Profile mit »unscharfen« Rändern in den Kenntnissen und als Mischform zwischen Design und IT ausgeschrieben werden würden, würde sich die Ausbildung von selbst anpassen.

Einen ausführlichen Bericht über die Veranstaltung und die Impulsvorträge können Sie auf dem User Experience Blog von Ulf Schubert (Datev eG) nachlesen (<http://www.user-experience-blog.de/tag/dmit17/>). Folien der Vorträge finden sich in Auszügen unter <https://www.bitkom.org/designmeetsit/>

# 5

Fazit – Der »Digital  
Designer« als  
Gestaltungsrolle

# 5 Fazit – Der »Digital Designer« als Gestaltungsrolle

Die Ergebnisse der Taskforce zeigen deutlich, dass das in der Einleitung beschriebene Gestaltungsvakuum bzw. die Fragmentierung der Gestaltungskompetenz in der Softwareentwicklung ein echtes Problem ist und die Ausbildungslage in Deutschland mit Blick auf Gestaltung in der Softwareentwicklung unterentwickelt ist (Stichwort 5%). Schaut man auf die Bedeutung von Software für die heutige Gesellschaft und Wirtschaft (siehe oben, Stichwort »Digitale Transformation«), ist die Ausprägung einer ganzheitlichen und integrierenden Gestaltungsrolle aus der Softwareentwicklung heraus (analog zum Architekten oder Industriedesigner) ein nächster konsequenter Entwicklungsschritt. Der »Digital Designer« soll diesen Entwicklungsschritt darstellen.

## 5.1. Herleitung des Begriffs »Digital Designer«

Der Begriff »Digital Designer« mag auf den ersten Blick ein wenig befremdlich wirken, besonders für Menschen mit einem technisch geprägten Hintergrund. Das Wort Digital bezieht sich ja auf die Verwendung von binären Werten zur Repräsentation von Information. Schaut man jedoch auf das Wort »Digitalisierung« und seine gewandelte Bedeutung in der Gesellschaft (siehe Kapitel 1), so scheint der »Digital Designer« als Begriff schon passender. Der Begriff »Software Designer« wäre eine denkbare Alternative, aber greift zu kurz, da im Kontext von Digitalisierung nicht nur Software gestaltet wird, sondern auch das Umfeld der Software (bspw. Geschäftsprozesse).

Als Beispiel für gelungenes Digital Design kann das Unternehmen und die zugehörige Webseite Airbnb herangezogen werden. Airbnb ist eine Plattform auf der Privatpersonen Zimmer mieten bzw. vermieten können. Dabei tritt Airbnb als Vermittler auf und wickelt die Zahlung ab. Hierdurch ist ein neues wirtschaftliches System zur Vermittlung von privatem Wohnraum als Alternative zum klassischen Hotelgeschäft entstanden. In diesem Sinne wurde durch Airbnb nicht nur eine Software gestaltet, sondern neues digitales Geschäftsmodell.

Zur konkreten Definition wird auf das Vorbild des Industriedesigners zurückgegriffen: »Digital Designer« gestalten und optimieren digitale Produkte, Systeme und Dienstleistungen. Sie berücksichtigen dabei das Spannungsfeld zwischen den Wünschen und Bedürfnissen der Nutzer, den wirtschaftlichen Rahmenbedingungen und den technischen Möglichkeiten. Digital Designer führen den Entwicklungsprozess durch Skizzen, Modelle, Spezifikationen und Prototypen. Sie arbeiten dabei in multidisziplinären Gruppen mit dem Management, dem Marketing, der Entwicklung und dem Betrieb von Software.

Hierbei ist besonders die Visionsfähigkeit herauszustellen. Der Digital Designer ist in der Lage, über den aktuellen Stand der Möglichkeiten hinaus zu denken, neue Konzepte und Anwendungen zu entwickeln, die bisher noch nicht möglich schienen. Er kennt aber genauso die Grenzen des (derzeit) Machbaren. Damit steht das Entwerfen gegenüber dem Planen im Vordergrund.

## 5.2. Digital Designer als idealisiertes Bild und Brückenkopf

Der Digital Designer muss als idealisiertes Rollenbild verstanden werden. Genauso wie es nicht den Industriedesigner oder den Architekten im Bauwesen gibt, wird es auch nicht den präzise definierten Digital Designer als Rolle im Projekt geben. In der Praxis werden die Verantwortlichkeiten sicherlich abhängig vom Kontext verteilt werden.

Dennoch ist dieses Idealbild in Anbetracht der aktuellen Situation in der Softwareentwicklung von besonderer Bedeutung. Das beschriebene Gestaltungsvakuum und die fragmentierte Gestaltungskompetenz kann nur dann sinnvoll überwunden werden, wenn stabile Brücken zwischen den Disziplinen gebaut werden können. Stabile Brücken können nur dann entstehen, wenn klare Positionen in Form von Idealbildern definiert werden, anhand deren sich handelnde Personen in der Praxis orientieren können. Weiterhin soll das Idealbild polarisierend wirken und so den Diskurs über die Gestaltung in der Softwareentwicklung anregen, sowohl innerhalb als auch außerhalb der Softwarebranche.

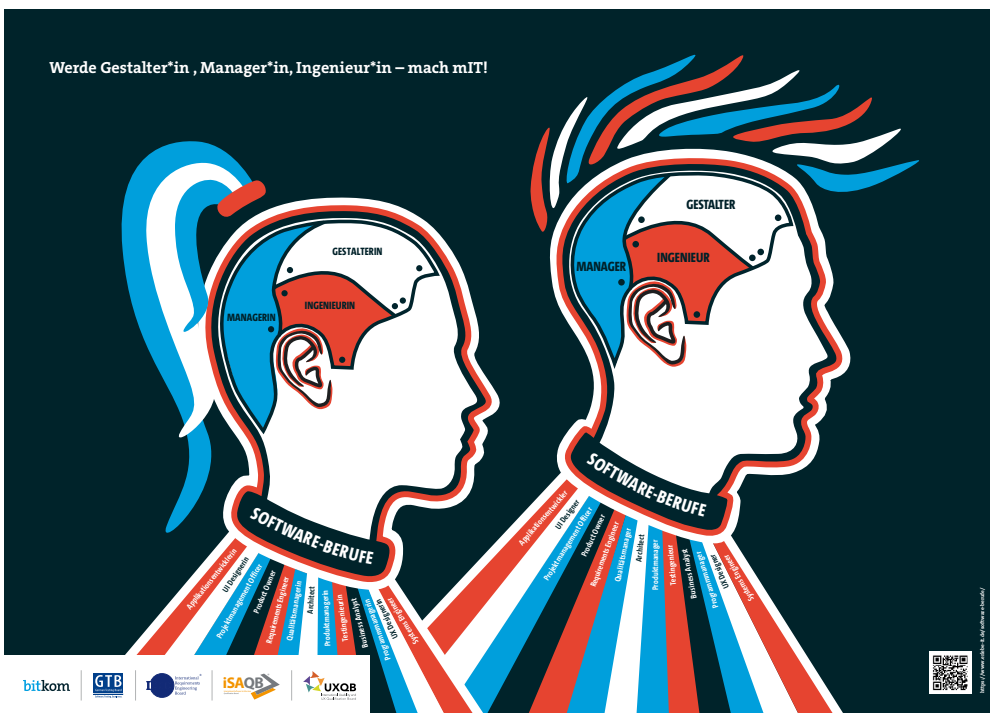
## 5.3. Abgrenzung zu anderen idealisierten Rollenbildern

Neben dem Digital Designer können mit dem Software/System Ingenieur und dem Software Manager zwei weitere idealisierte Rollenbilder in der Softwareentwicklung definiert werden. In Kombination ergibt sich dann der folgende Dreiklang mit klar umrissenen Verantwortlichkeiten:

- Der **Digital Designer** ist für die Gestaltung der für Kunden bzw. Nutzer erlebbaren Aspekte digitaler Produkte, Systeme und Services verantwortlich. Dies sind unter anderem Funktionen, Benutzungsschnittstelle, Qualitätsaspekte (bspw. Geschwindigkeit) und auch die Auswirkungen in der Umwelt/Umgebung. Auf eine kurze Formel gebracht: Der Digital Designer verantwortet und gestaltet alles, was Kunden/Nutzer erleben können.
- Der **Software/System Ingenieur** ist für die Gestaltung und Realisierung der technischen Umsetzung von digitalen Produkten, Systemen und Services verantwortlich. Auch für die Tätigkeiten des Ingenieurs wird der Begriff der Gestaltung verwendet, da digitale Technologie (und vor allen Dingen Software und Systeme) ein sehr hohes Maß an technischer Komplexität (vielfältige Technologien und Möglichkeiten der Umsetzung) aufweist, so dass hier von Gestaltung (dem sogenannten Software-Design, siehe oben) gesprochen wird (vgl. Glass, 2006). Auf eine kurze Formel gebracht: Der Ingenieur verantwortet und gestaltet alles, was »unter der Haube« ist.
- Der **Software Manager** ist für die Gestaltung des Prozesses und die effiziente und ökonomische Durchführung der Umsetzung verantwortlich. Auch hier wird wieder das Wort Gestaltung verwendet, da Vorhaben zur Umsetzung digitaler Produkte, Systeme und Services im Vergleich zu klassischen Konsumgütern einen wesentlich komplexeren Umsetzungsprozess aufweisen. Daher müssen diese Prozesse proaktiv gestaltet und gemanagt werden, um erfolgreich entwickeln zu können. Auf eine kurze Formel gebracht: Der Manager ist für Prozess, Zeit und Budget verantwortlich.

Die Beschreibungen aller drei Rollenbilder machen deutlich, dass Digitalisierung nur dann gelingen kann, wenn alle drei Rollenbilder mit entsprechenden kompetenten Personen zusammenarbeiten und alle drei Rollenbilder ihre Verantwortung ernst nehmen. Überwiegt eine Rolle bzw. wird ein Rollenbild vernachlässigt, dann entsteht ein Ungleichgewicht und ein suboptimales Ergebnis.

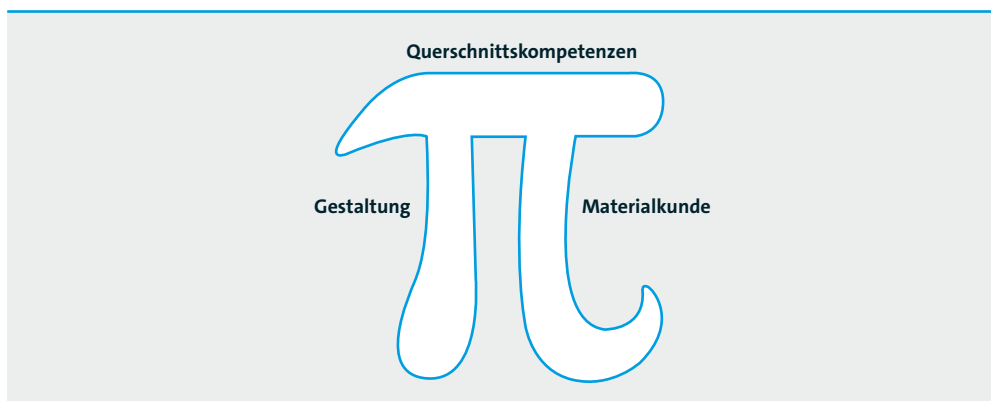
Diese drei Rollenbilder können nicht nur als Brückenköpfe innerhalb der Softwareentwicklung gesehen werden, sondern können insbesondere auch in der Außendarstellung genutzt werden. Ein Beispiel hierfür ist die Seite Software-Berufe der Bitkom-Kampagne erlebe-IT (siehe <https://www.erlebe-it.de/software-berufe/>). Diese Kampagne zeigt, dass in der IT ein breites Tätigkeitspektrum existiert, das verschiedenste Talente und Fähigkeiten erfordert.



## 5.4. Kompetenzprofil »Digital Designer«

Wesentlich für die Charakterisierung des Digital Designers ist die Berücksichtigung des Spannungsfelds aus Nutzerbedürfnissen, Wirtschaftlichkeit und technischen Möglichkeiten in einer Rolle. Die kombinierte Betrachtung dieser drei Aspekte rückt den Digital Designer ins Zentrum jedes Entwicklungsvorhabens und erlaubt die Vermittlung zwischen den drei Aspekten zur Schaffung nachhaltig erfolgreicher Produkte, Systeme und Dienstleistungen. Analog zum Architekten im Bauwesen und zum Industriedesigner ergibt sich aus der Definition des Digital Designers ein breit angelegtes und anspruchsvolles Kompetenzprofil und ein breites Querschnittskompetenzfeld. Die Rolle des Digital Designers geht über die Rolle der "User Experience Designer" hinaus, die in einigen Unternehmen bereits etabliert ist. Das Kompetenzprofil des Digital Designers besteht aus den zwei Schwerpunkten Gestaltung und Materialkunde. Im Vergleich zum "User Experience Designer" finden sich im Schwerpunkt "Materialkunde" des Digital Designers deutlich mehr technische Aspekte. In den Querschnittskompetenzfeldern weist der Digital Designer deutlich mehr Kompetenzen in den Methoden und Vorgehensweisen zur Entwicklung von Software auf.

Die zwei Schwerpunkt zusammen mit der Querschnittskompetenz ergeben das sogenannte pi-shaped Profil. Daher auch das pi als Titelbild. Im Gegensatz dazu stehen die bekannten sogenannten T-shaped Profile mit einem Schwerpunkt und einer Querschnittskompetenz. Diese Form des Kompetenzschnitt ist, wie in diesem Bericht gezeigt, für die Herausforderungen der Digitalisierung und der Digitalen Transformation nicht ausreichend. Es braucht zwei Schwerpunkte als Standbeine und ein breites Querschnittsprofil:



### Schwerpunkt 1 – Gestaltung

- Arbeit mit Anforderungen (Requirements Engineering)
- Konstruktion von Benutzungsschnittstellen (Usability Engineering/Interaction Design)
- Grundlegende Vorgehensweisen zur Gestaltung
- Entwerfen neuer Konzepte
- Explorationsfähigkeit, Fähigkeit zur Durchführung ethnografischer Feldforschung
- Menschenzentriertheit

## Schwerpunkt 2 – Materialkunde

- Wissen um die Möglichkeiten und Grenzen von Software und Hardware
- Wissen um Algorithmen und Grenzen von Algorithmen
- Arten von Systemen: Informationssysteme, Eingebettete Systeme
- Arten von Endgeräten: Desktop, Notebook, Tablet, Smartphone, etc.
- Interaktionsformen: Tastatur, Maus, Touch, Sprache, Geste, etc.
- Wissen über den grundsätzlichen Aufbau von Software (Software-Architekturen)
- Wissen um Form- und Farbgebung

## Querschnittskompetenz

- Wissen um Methoden und Vorgehensweisen zum Management von Entwicklungsvorhaben:
  - Projektmanagement
  - Agile Entwicklung
  - V-Modell
  - etc.
- Wissen um Methoden und Vorgehensweisen zur Entwicklung von Software:
  - Qualitätssicherung/Testen
  - Konfigurationsmanagement
  - Produktprozesse, die gesamte User Journey kennen
- Wirtschaftliche Aspekte zur Gestaltung/Entwicklung von Software
  - Kostenkalkulation
  - Geschäftsmodelle für Software (Pay per Use, ...)
  - Geschäftsprozesse
  - Kreieren von Marken (Brands)
  - Wissen über Organisationen, die softwarebasierte Produkte herstellen
- Fähigkeit zum Arbeiten in interdisziplinären Projekten
- Psychologische Grundlagen zur Realisierung von Software auf Nutzer- und Herstellerseite



# 6 Zusammenfassung und nächste Schritte

## 6 Zusammenfassung und nächste Schritte

Das dargestellte idealisierte Rollenbild des »Digital Designer« stellt einen konsequenten nächsten Schritt in der Weiterentwicklung der Softwareentwicklung als Disziplin dar und kann eine zentrale Rolle in allen Vorhaben im Kontext der Digitalisierung einnehmen. Technologische Kompetenz in der Softwareentwicklung ist eine notwendige Kompetenz für erfolgreiche Digitalisierung und Digitale Transformation, aber keineswegs hinreichend. Die Softwareindustrie in Deutschland konzentriert sich allerdings zu stark auf diese technologische Kompetenz und verspielt durch mangelnde Gestaltungskompetenz einen relevanten Wettbewerbsvorteil: Die Kompetenz zur Gestaltung wirtschaftlich tragfähiger, technisch exzellenter und erfolgreicher Produkte, Dienstleistung und Services.

Die Ergebnisse der Taskforce zeigen deutlich, dass die Etablierung des Digital Designers als neues Rollenbild ein langfristiges Unterfangen ist:

- Zwischen den stark fragmentierten Feldern der Softwareentwicklung müssen Brücken gebaut werden, um ein gegenseitiges Verständnis für die Relevanz der Gestaltung in der Softwareentwicklung zu schaffen und um das Idealbild des Digital Designers in der Softwareentwicklung zu etablieren. Hierzu ist umfangreiche Community-Arbeit und Kooperation mit einschlägigen Berufsverbänden und Organisationen erforderlich.
- Der Stand der Praxis zeigt, dass das Idealbild des Digital Designers in Teilen bereits etabliert ist. Dies zeigt die Etablierung von Rollen aus dem Bereich User Experience, z.B. User Experience Designer. Allerdings decken die handelnden Personen in der Regel nur einen Teil der Kompetenzen ab. Daher müssen Ausbildungsprogramme (bspw. in Kooperation mit entsprechenden Ausbildungsverbänden) entwickelt werden, um die Ausbildung solcher Personen zu vervollständigen.
- Der Stand der Ausbildungslage zeigt, dass in der akademischen Ausbildung ebenfalls starker Veränderungsbedarf besteht. Es müssen Studienprogramme etabliert werden, die eine fundierte Ausbildung zum Digital Designer ermöglichen. Langfristig gesehen sollte der Digital Designer als Berufsbild genauso etabliert sein, wie ein Industriedesigner oder Architekt im Bauwesen. Um dieses Ziel zu erreichen, ist sowohl politische Arbeit als auch Kooperation mit einschlägigen Hochschulverbänden (bspw. Gesellschaft für Informatik) erforderlich.
- Zur Etablierung des Digital Designers in der Industrie muss ein Wandel in den Köpfen in der Unternehmensführung in Gang gebracht werden, damit der Digital Designer als führende Rolle in Unternehmen und Projekten etabliert werden kann.

Mindestens genauso wichtig wie das Etablieren einer führenden Rolle durch den Digital Designer ist die Veränderung der Außensicht auf Software durch den Digital Designer. Bisher wurde Softwareentwicklung ausschließlich als Realisierung betrachtet. Durch den Digital Designer kann sich diese Außensicht positiv verändern und die Softwareentwicklung als Disziplin vollständiger machen und eine wichtige Lücke in der Gestaltungskompetenz schließen.

Der Digital Designer ist zweifelsohne ein Idealbild und muss auch als solches verstanden werden. Die Definition eines solchen Idealbildes ist unumgänglich für einen Wandel in der Softwareentwicklung: Genauso wie im Bauwesen kein Gebäude von ernstzunehmendem Format ohne Architekt realisiert wird, sollte in Zukunft kein ernstzunehmendes Vorhaben im Kontext der Digitalisierung oder der digitalen Transformation ohne einen entsprechenden qualifizierten Digital Designer realisiert werden.

# 7 Literatur

## 7 Literatur

Pierre Bourque, Richard E. (Dick) Fairley (Hg.) (2014). Guide to the Software Engineering Body of Knowledge Version 3.0 (SWEBOOK). A Project of the IEEE Computer Society. IEEE.

Alan M. Davis (2003). System Phenotypes. IEEE Software, July/August 2003, 54-56.

Barry Boehm (2006). A View of 20th and 21st Century Software Engineering. 28th International conference on Software engineering (ICSE '06), 2006, 12-29.

Michael Erlhoff, Tim Marshall (Hg.) (2008). Wörterbuch Design: Begriffliche Perspektiven des Design. Basel, Boston, Berlin: Birkhäuser.

Robert L. Glass (2006). Software Creativity 2.0. developer.\* Books.

Hochschulrektorenkonferenz (2017). Aufgaben und Struktur.

↗<https://www.hrk.de/hrk/aufgaben-und-struktur/>. Abfrage am 9.6.2017, 11:00h.

Winston Royce (1970). Managing the development of large software systems. WESCON, August 1970,1-9.

Bitkom vertritt mehr als 2.500 Unternehmen der digitalen Wirtschaft, davon gut 1.700 Direktmitglieder. Sie erzielen allein mit IT- und Telekommunikationsleistungen jährlich Umsätze von 190 Milliarden Euro, darunter Exporte in Höhe von 50 Milliarden Euro. Die Bitkom-Mitglieder beschäftigen in Deutschland mehr als 2 Millionen Mitarbeiterinnen und Mitarbeiter. Zu den Mitgliedern zählen 1.000 Mittelständler, mehr als 400 Start-ups und nahezu alle Global Player. Sie bieten Software, IT-Services, Telekommunikations- oder Internetdienste an, stellen Geräte und Bauteile her, sind im Bereich der digitalen Medien tätig oder in anderer Weise Teil der digitalen Wirtschaft. 80 Prozent der Unternehmen haben ihren Hauptsitz in Deutschland, jeweils 8 Prozent kommen aus Europa und den USA, 4 Prozent aus anderen Regionen. Bitkom fördert und treibt die digitale Transformation der deutschen Wirtschaft und setzt sich für eine breite gesellschaftliche Teilhabe an den digitalen Entwicklungen ein. Ziel ist es, Deutschland zu einem weltweit führenden Digitalstandort zu machen.

**Bundesverband Informationswirtschaft,  
Telekommunikation und neue Medien e.V.**

Albrechtstraße 10

10117 Berlin

**T** 030 27576-0

**F** 030 27576-400

[bitkom@bitkom.org](mailto:bitkom@bitkom.org)

[www.bitkom.org](http://www.bitkom.org)

**bitkom**