

# Einführung in Requirements Engineering

## Schnuppervorlesung

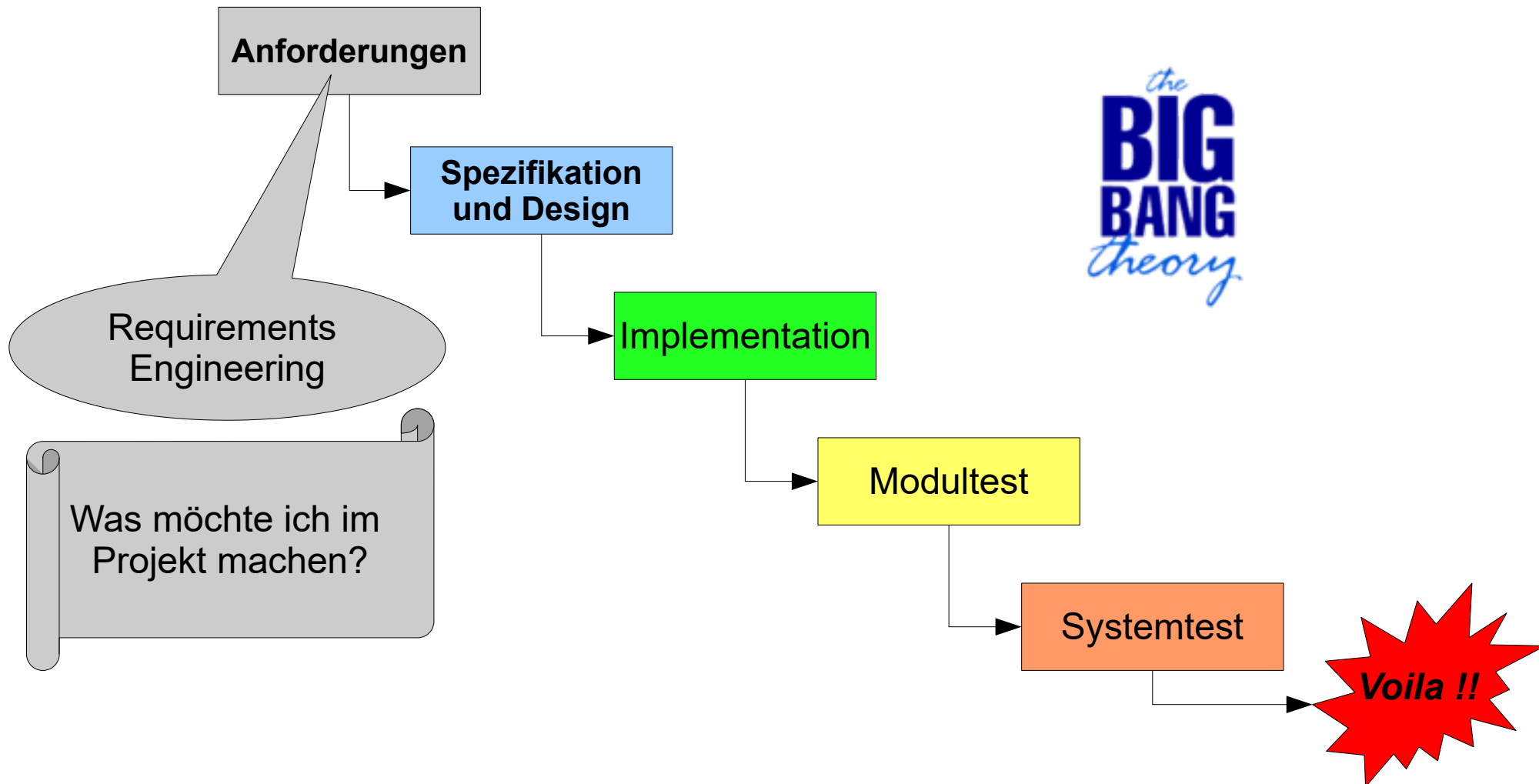


# **Einführung in Requirements Engineering**

---

- Einleitung & Motivation
- Was sind Software- und Requirements Engineering?
- Funktionale und nicht-funktionale Anforderungen
- Struktur und Organisation von Anforderungsspezifikationen
- Zusammenfassung

# Software Engineering: Das Wasserfallmodell



# Komplexität von Software

---

- Komplexe Systeme sind exponentiell teuer
  - Vgl. <http://www.visualcapitalist.com/millions-lines-of-code/>
  - Moderne Systeme sind komplex und werden immer komplexer
  - In jedem Bereich wendet man disziplinierte Prozesse an, um die Komplexität – und die Kosten – unter Kontrolle zu bringen
- In der Informatik nennt man diese disziplinierten Prozesse “Software Engineering”

Not including backend **code**, **Facebook** runs on 62 million **lines of code**. With the advent of sophisticated, cloud-connected infotainment systems, the car software in a modern vehicle apparently uses 100 million **lines of code**. This is according to Wired magazine. All Google services combine for a whopping 2 billion **lines**. 08.02.2017

**facebook**

Visual Capitalist › millions-lines-of-code

Infographic: How Many Millions of Lines of Code Does It Take?

## Der Anfang eines IT-Projekts

---



Was die Kundin sagt:

*„Wir wollen OBCs durch TMAs,  
NMAs und Ärzte planen, mit  
Berücksichtigung realistischer  
Zeitfenster und verfügbarer  
Ressourcen...“*

Was der Informatiker hört:

„Wir wollen blah blah blah **planen** blah  
blah **Zeitfenster** und blah  
**Ressourcen...**“



## Der Anfang eines IT-Projekts

---

Der Informatiker antwortet:

*„Wir empfehlen eine MVC-Architektur basierend auf der Java EE. Der Web 2.0 Client verwendet JAX und alle Kommunikation mit dem Server ist per XML.“*



Die Kundin versteht

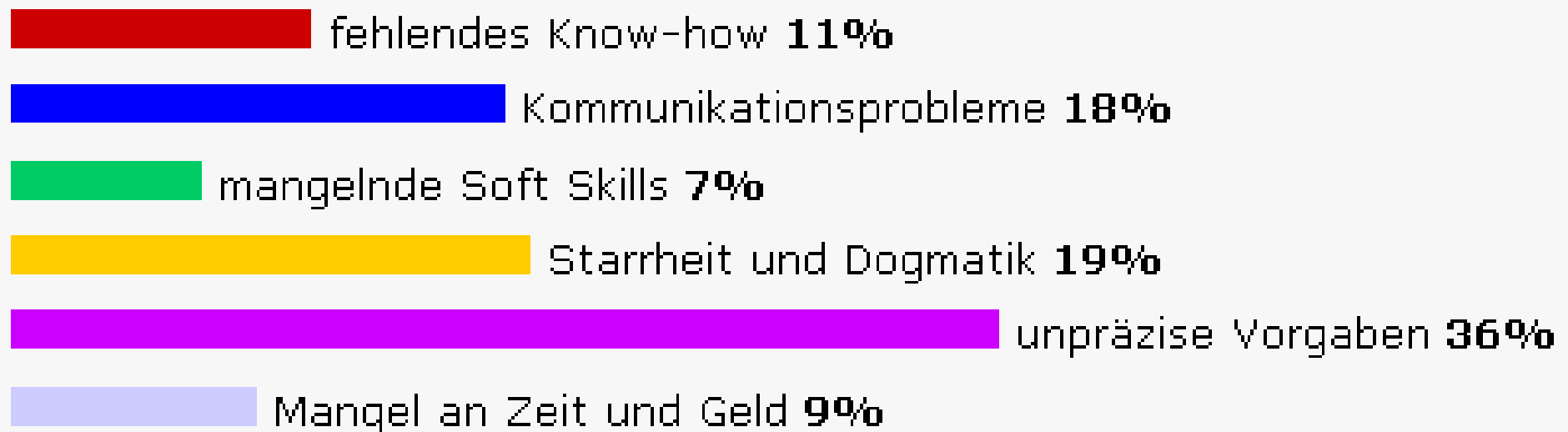
*„Wir empfehlen blah blah blah Web 2.0  
blah blah blah“*

Alles klar – los geht's!

## Warum scheitern IT-Projekte?

---

### Woran scheitern die meisten IT-Projekte?



Source: [www.gulp.de](http://www.gulp.de) (Umfrage)

## Warum scheitern IT-Projekte?

---

- Das Ziel der Anforderungsanalyse ist, die Kundenwünsche zu dokumentieren

- Dies setzt voraus:

- Der Kunde weiss, was er will

 unpräzise Vorgaben **36%**

- Die Kundenwünsche sind sinnvoll

 Starrheit und Dogmatik **19%**

- Die IT Firma versteht die Wünsche des Kunden

 Kommunikationsprobleme **18%**

## Requirements Engineering: Mehr als nur Anforderungsanalyse

- Woher – und wie – bekommen wir die Anforderungen?
  - Von Personen (Stakeholder)
  - Von bestehenden IT-Systemen (Systemarchäologie)
  - Das jetzige „System“ ist oft informell ergänzt worden: Auf Papier, Excel-Bastlereien, Access-Anwendungen, usw.
- Inwiefern kennt der Kunde seine Anforderungen?
  - „Das, was wir jetzt haben, nur besser!“
  - Die aktuellen Geschäftsprozesse sind nirgendwo aufgeschrieben
  - Arbeitsprozesse sind verinnerlicht und kaum erklärbar

# Geschäftsprozess und Anforderung

- Geschäftsprozess
  - Folge von Aktivitäten
  - Beschreibt, welche (Teil-)Prozesse automatisiert werden
  - Beispiel: CRM-System
  - (Teil-)Prozesse: Dateneingabe, Vertriebsprognosen, Analyse mit Dashboards, Messaging zwischen Mitarbeitern, E-Mail-Integration etc.
- Anforderung
  - Beschreibt eine *konkrete* Eigenschaft oder Fähigkeit eines Systems
  - Funktionale Anforderungen
    - ♦ Beispiele: Eingabe neuer Kunde (mit vordefinierten Datenfeldern), Auflisten aller Kunden im System, Löschen eines Kunden, Export Kundenliste in PDF etc.
  - Nicht-Funktionale Anforderungen
    - ♦ Beispiele: Das System soll alle Anfragen in weniger als 3 Sekunden abarbeiten, Das System soll in weniger als 24h für alle erlernbar sein etc.

# Übung: Prozesse und Anforderungen (Hotelplattform)

**Booking.com** CHF Recently Seen My Lists Manage booking Sign in

home → usa (58,891 properties) → new york state (2,091 properties) → new york city (629 properties) → search results

**Search**  
Destination/Hotel Name: New York City  
Work Leisure  
 I don't have specific dates yet  
Guests: 2 adults, 0 children  
**Search**

**608 properties are available in and around New York City**  
3 Reasons to Visit: people watching, food & entertainment  
There are 99 people looking at New York City right now.

Lock in a great price for your stay on these dates:  
Sept 18 — Sept 19    Sept 23 — Sept 24    Sept 20 — Sept 21    Sept 24 — Sept 25

Order by: We Recommend **Stars** Distance from downtown Review Score

**Park Lane Hotel on Central Park** ★★★★★  
2014 10428  
New York City – Nearby Subway/Metro Access  
Great savings! 98 deals booked here this week.  
The Park Lane Hotel is a deluxe hotel with a European atmosphere, providing views of Central Park and the New York skyline. [More](#)  
Booked 43 times today

**The New York EDITION** ★★★★★ 106  
New York City – Nearby Subway/Metro Access  
Located in the iconic landmark Metropolitan Life clock tower 22 metres from Madison Square Park, The New York EDITION offers a Zagat-rated restaurant and free WiFi access. [More](#)  
Booked 6 times today

**Excellent 8.9**  
25 reviews  
**Select your room**  
or [Save to a list](#)

**I'm looking for properties with...**

- ▼ **Star Rating**
  - 1 star (26)
  - 2 stars (121)
  - 3 stars (146)
  - 4 stars (202)
  - 5 stars (59)
  - Unrated (54)
- ▼ **Property Type**
  - Hotels (363)
  - Apartments (201)
  - Hostels (15)
  - Guesthouses (14)

a) Was sind hier die Geschäftsprozesse?

b) Nennen Sie beispielhaft konkrete Anforderungen für Geschäftsprozesse.

# Definitionen

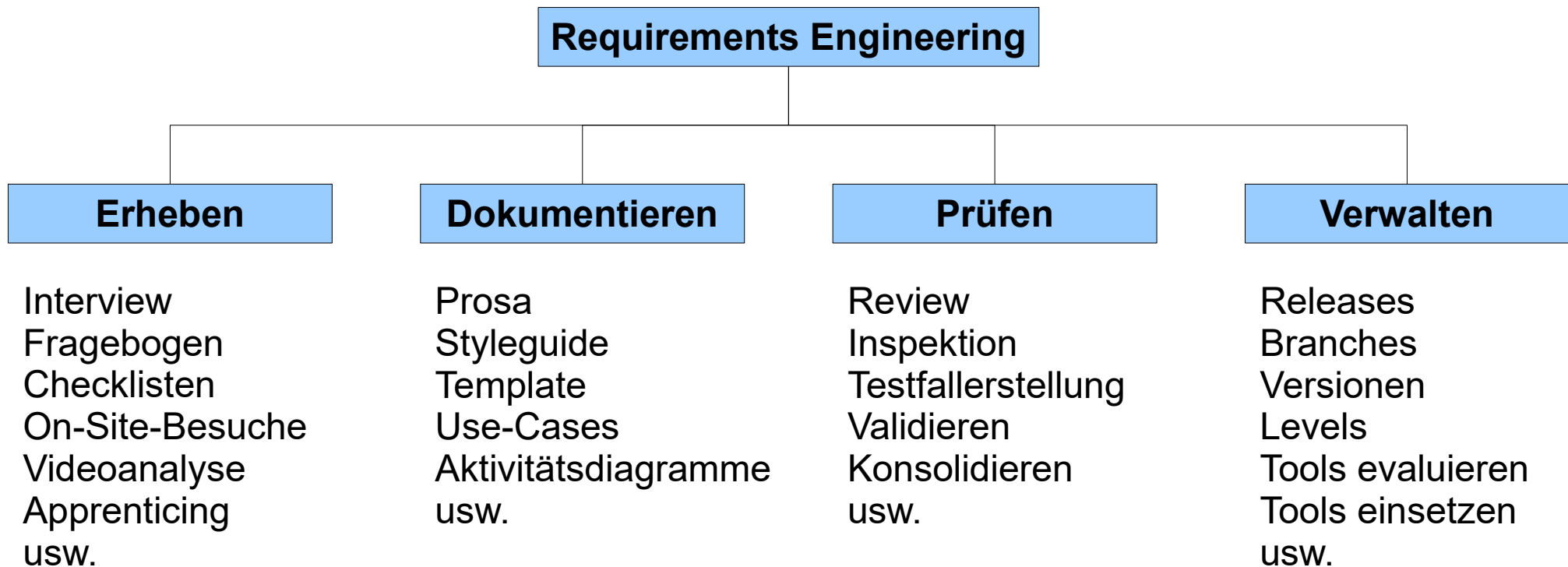
## **Was ist Requirements Engineering?**

---

- IREB: Das Requirements Engineering ist ein kooperativer, iterativer, inkrementeller Prozess, dessen Ziel es ist zu gewährleisten, dass
  - alle relevanten Anforderungen bekannt und in dem erforderlichen Detaillierungsgrad verstanden sind,
  - die involvierten Stakeholder eine ausreichende Übereinstimmung über die bekannten Anforderungen erzielen,
  - alle Anforderungen konform zu den Dokumentationsvorschriften dokumentiert bzw. konform zu den Spezifikationsvorschriften spezifiziert sind.

# Aspekte des Requirements Engineering

---



## Was versteht man unter „Requirements Management“?

- Verwaltung der durch RE produzierten Resultate (Dokumente, usw.)
- Einsatz aller Aspekte des REs für neue bzw. geänderte Anforderungen

Quelle: Requirements-Engineering and Management, Chris Rupp, Hanser Verlag

# FIS Heer

---

IT-Projekt FIS Heer

## Führungssystem ohne Funkgerät

11.1.2017, 08:51 Uhr

Das **umstrittene** IT-Projekt FIS Heer des Verteidigungsdepartement wird nicht mit mobiler Telekommunikation ausgebaut. Die Funkgeräte könnten nicht mit der Datenflut umgehen, befand das VBS. Damit müssen 125 Millionen Franken abgeschrieben werden.



*(sda)* Das Verteidigungsdepartement (VBS) kämpft bereits seit längerem mit Informatikproblemen. Das **700 Millionen Franken teure Führungsinformationssystem (FIS) Heer** für die vernetzte Einsatzführung funktioniert zwar an fixen Standorten und im halbmobilen Betrieb. Die Funkgeräte haben aber eine zu geringe Bandbreite, um im mobilen Bereich die Datenmenge in der notwendigen Zeit zu übertragen.

MEISTGELESEN IN DIESEM RESSORT

Schweizer Kavallerie

### Die letzte Reiterschlacht Europas

Marc Tribelhorn / 27.2.2017

Gemeindeammann von Oberwil-Lieli

## Insieme

---

### Insieme (Informatikprojekt)

---

**Insieme** war ein **Informatikprojekt**, welches von der **Eidgenössischen Steuerverwaltung** der **Schweiz** ins Leben gerufen worden war, um die getrennt laufenden Informatiksysteme der **Stempel-** und **Mehrwertsteuer** abzulösen und zu vereinen.<sup>[1][2]</sup> Insieme sollte als «einheitliches Gesamtsystem» die veralteten Informatiksysteme der Schweizerischen Steuerverwaltung ersetzen und die Abläufe auf den Kunden ausrichten.<sup>[1]</sup>

Im Jahr 2006 zerstritten sich die Steuerverwaltung und die Auftragnehmerin **Unisys** und der Auftrag wurde widerrufen. Im Anschluss wurde das Insieme-Projekt in zahlreiche kleinere Projekte aufgeteilt und an unterschiedliche Auftragnehmer vergeben. In den Folgejahren lief das Projekt finanziell aus dem Ruder und **Eveline Widmer-Schlumpf**, Vorsteherin des Eidgenössischen Finanzdepartements, brach im September 2012 das Projekt ab. Bis zu diesem Zeitpunkt hatte das Grossprojekt **115,9 Millionen Franken** gekostet.<sup>[3]</sup>

Im Frühjahr 2013 wurde das Nachfolgeprojekt *Fiscal-IT* lanciert.<sup>[3]</sup>

# Fiscal-IT

---

## EFK: Fiscal-IT hat Performanceprobleme, unklare Betriebskosten

### Der Prüfbericht zum Insieme-Nachfolger zeigt grosse Probleme.

Das 2012 gestoppte Informatikprojekt Insieme der Eidgenössischen Steuerverwaltung war ein Millionengrab. Das Nachfolgeprojekt Fiscal-IT ist inzwischen in Betrieb -- scheint aber ebenfalls ein Sorgenfall. Die Systeme sind unzuverlässig und zu wenig leistungsfähig.

Das hat die Eidgenössische Finanzkontrolle (EFK) in einem gerade veröffentlichten Bericht festgestellt. Die alte Anwendung für die direkte Bundessteuer wurde im Oktober 2017 durch das neue System DIFAS abgelöst. Dieses wies damals noch zahlreiche Mängel auf, die bis Ende Jahr nicht behoben werden konnten.

Gemäss dem Bericht dauerte zum Beispiel das Öffnen eines Dokuments bis zu 20 Sekunden. Es fehlten Filtermöglichkeiten, die Prüfung von Rückerstattungen lief nicht automatisch. Wegen solcher Hindernisse bearbeiteten die Mitarbeitenden in den ersten Monaten nach der Umstellung nur rund einen Drittel der üblichen Anzahl Formulare.



Quelle: <https://www.inside-it.ch/articles/52644>

# Healthcare.gov (“Obamacare”)

## Rückblende auf Obamacare

Für seine zweiten Amtszeit als US-Präsident hatte sich Barack Obama vorgenommen, den US-Amerikanern endlich ein Gesundheitssystem zu geben, was diesen Namen verdient. Obwohl das Projekt innenpolitisch höchstumstritten war (und ist), setzte sich der Präsident durch, das Projekt „Obamacare“ erhielt Grünes Licht.

Um eine Krankenversicherung zu bekommen, muss man sie beantragen können. Dafür musste eine Website her, groß und schön! Das Ziel war schnell beschrieben: gesucht wurde eine Website mit dem Look’n’Feel von Amazon und der Einfachheit des iTunes-Stores. Dort sollten allerdings keine Bücher oder Lieder verkauft werden, sondern Versicherungen. Und damit es nicht zu einfach ist, sollte die Website möglichst viel automatisieren. D.h., die Login-Daten sollten von Meldeämtern übernommen werden, Versicherungsgesellschaften sollten direkt Policen verkaufen, Preisvergleiche mussten möglich sein u.v.m.

Jedes Detail für sich genommen wäre sicherlich lösbar gewesen. Sämtliche Funktionen gleichzeitig auszurollen, musste in einem Disaster enden – und so kam es dann auch.

## Bauchlandung

Als healthcare.gov am 1. Oktober online ging, ging sie auch gleich wieder aus. Die Server kämpften mit massiver Überlastung, rund 19 Millionen Menschen zeigten Interesse, der Ansturm war einfach zu groß. Nach dem Serverausfall folgte der Regierungs-Shutdown aufgrund des Haushaltsstreits, während dessen die Probleme mit der Website in den Hintergrund gerieten. Doch bereits ein paar Tage später war das Problem wieder in den Medien und es wurde klar: die Website funktioniert schlichtweg nicht.

Sowohl die Gesundheitsministerin als auch der Präsident höchstselbst mussten die technischen Fehler eingestehen und sich allerhand kritische Fragen gefallen lassen. Immerhin hatte die Kern-Website gut drei Jahre Vorlaufzeit bekommen und über 172 Millionen US-Dollar Steuergeld gekostet, **bis zu 400 Millionen Dollar** wurden insgesamt bisher für das Projekt verwandt.

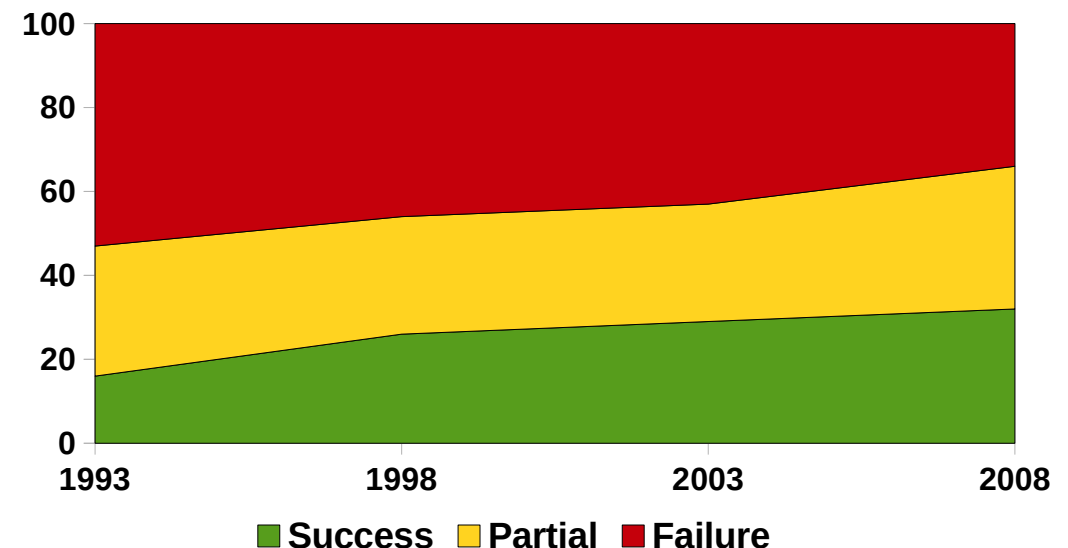


Quelle: <http://tagwerk.tripuls.de/2013/11/erkenntnisse-aus-dem-obamacare-website-desaster/>, Zugriff: 2018-02-28

## Software-Projekte sind selten wirklich erfolgreich

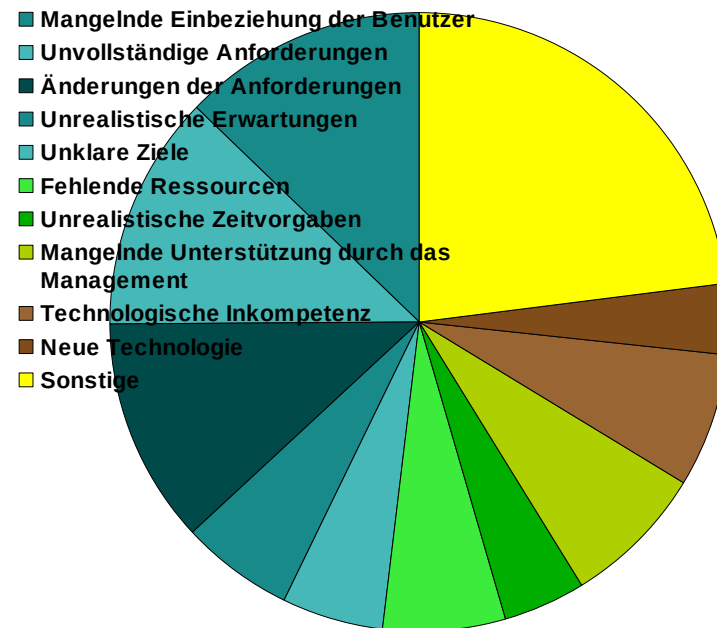
---

- Wie oft ist ein Projekt erfolgreich?
  - Mit voller Funktionalität, einigermaßen im Budget und zeitgerecht geliefert
  - Weniger als 1/3 aller Projekte
- Wie oft ist ein Projekt teilweise erfolgreich?
  - Eingeschränkte Funktionalität, weit über Budget, usw.
  - Ca. 1/3 aller Projekte
- Und wie oft nicht erfolgreich?
  - Abgesagt oder geliefert, aber nie in Einsatz genommen?
  - Mehr als 1/3 aller Projekte



## Warum scheitern Software-Projekte?

- Dazu folgende Studienergebnisse der Standish Group, 1994-2010
  - Die blauen Felder sind Aspekte des Requirements Engineering
  - Die grünen Felder sind mit Requirements Engineering verwandt

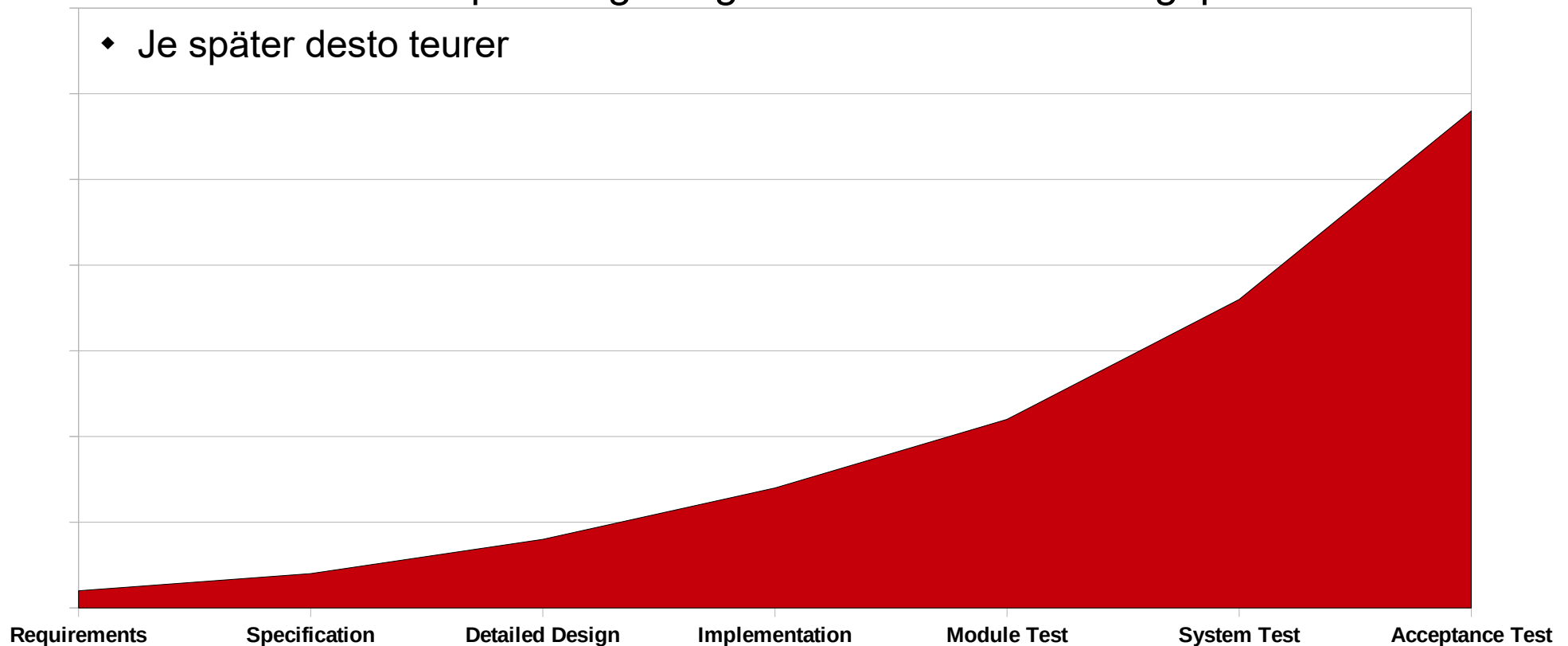


- Viele andere Studien bestätigen: 60% bis 70% der Systemfehler lassen sich auf Fehler in der Anforderungsanalyse zurückführen

## Wenn Anforderungen sich im Laufe des Projekts ändern

---

- “*To err is human*” - Menschen machen Fehler
  - Anforderungen werden übersehen, werden falsch verstanden
    - ♦ oder müssen aus Geschäftsgründen im Laufe des Projekts geändert werden
  - Die Kosten einer Anpassung hängen von der Entwicklungsphase ab
    - ♦ Je später desto teurer



# Was sind Anforderungen?

---

- Funktionale Anforderungen
  - **Was soll das System tun?**
    - ♦ Welche Daten werden bearbeitet?
    - ♦ Wie soll das GUI aussehen?
    - ♦ Was muss das System produzieren?
- Nichtfunktionale Anforderungen
  - Qualitätsanforderungen: **Wie soll sich das System verhalten?**
    - ♦ Wie performant muss das System sein?
    - ♦ Welche Verfügbarkeit muss das System vorweisen?
  - Weitere Randbedingungen, z.B.
    - ♦ Mit welchen Technologien muss das System realisiert werden?
    - ♦ Auf welchen Plattformen muss das System laufen?

## Kategorien der Qualitätsanforderungen

---

- Typische Kategorien sind
  - Qualität: Sicherheit, Genauigkeit, usw.
  - Zuverlässigkeit: Fehlertoleranz, Verfügbarkeit, usw.
  - Benutzbarkeit: Reaktionszeit, Verständlichkeit, usw.
  - Effizienz: Performance, Ressourcenbedarf, usw.
  - Änderbarkeit: Erweiterbarkeit, Prüfbarkeit, usw.
  - *und weitere Kategorien nach Bedarf ...*

## Zu welcher Art von Anforderungen gehören diese Beispiele?

- Bericht implementieren, der den durchschnittlichen Umsatz pro Kunde nach Altersgruppe präsentiert
- Die IT-Abteilung schreibt vor, dass sämtliche Webanwendungen mit Java EE implementiert werden
- Kunden können im Webmodul ihre vorherige Bestellungen durchstöbern
- Das System unterstützt mindestens 1000 gleichzeitige Benutzer
- Das System wird an der 100. Jubiläumsfeier präsentiert; bis dann muss die erste Version geliefert werden

# Beispiele für Studentenprojekte im RE Modul

<p><b>n w</b> Fachhochschule Nordwestschweiz</p> <p>Web-Erweiterung Webseite des Freiämter Abschlussarbeit Requirements Engineering</p>  <hr/> <p>Autoren: <b>Philipp Landis, Sergio Sutter, Marco Jenny und Aldo Richner</b></p> <p>Version: 1.0</p> <p>Erstell-Datum: 09.03.2018</p> <p>Dozent: Prof. Dr. Rainer Telesko</p> <p>Abgabetermin: 13.06.2018</p> <p>13. Juni 2018      Web-Erweiterung Webseite des Freiämter      1 / 70</p>	 <p><b>Requirements Engineering</b> Zeiterfassungssystem</p> <p>Bytyqi Florent David Samuel Inderbitzin Marco Maurer Jannick Schürch David</p> <p>Work smarter not harder Werkstrasse 14, 3420 Wiler, www.officecare.ch</p> <p>OFFICE CARE AG OFFICE CARE AG</p> <p><b>n w</b> Fachhochschule Nordwestschweiz Hochschule für Wirtschaft</p>	<p><b>n w</b> Fachhochschule Nordwestschweiz Hochschule für Wirtschaft</p> <p>Line-Out - Vereinsplattform Abschlussarbeit Requirements Engineering</p>  <p>Auftraggeber: Rugby Club Würenlos Autoren: Philipp Hartmann, Christian Roth, Daniel Masic, Noel Pineiro Abalo, Thomas Morcinek Dozent: Prof. Dr. Rainer Telesko Olten, 4. Januar 2019</p>
--	--	--

## Zusammenfassung

---

- Requirements Engineering steht am Beginn eines Projektes
- Diese Phase entscheidet über den Erfolg eines Software-Engineering Projektes
- Erhebung von Anforderungen
  - Funktionale Anforderungen (Funktionalitäten des Systems)
  - Nicht-funktionale Anforderungen (Systemverhalten)
- Ergebnis des Requirements Engineering ist eine technische Spezifikation (Dokument)