

# Usability Engineering

## Vorlesung 7

VU 183.123

Christoph Wimmer

# Usability Engineering Lifecycle & User-Centered Design Prozess

# Usability Engineering als Prozess

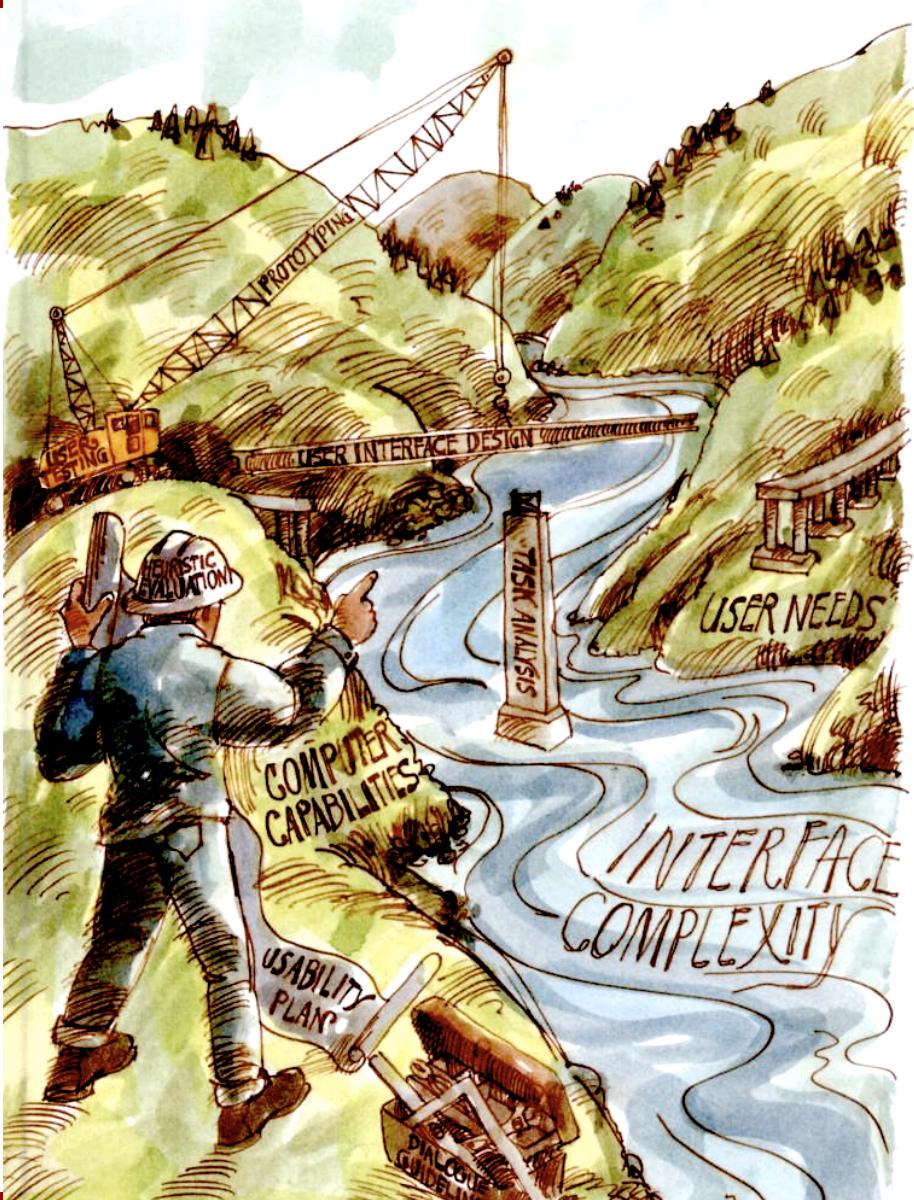
- „Usability engineering is a **set of activities that ideally take place throughout the lifecycle of the product**, with significant activities happening at the early stages before the user interface has even been designed.“ (Jakob Nielsen)
- Keine Sammlung von Einzelmethoden, sondern ein **Prozess**
- Von Requirements Analyse bis zu Installation, Betrieb und Support

# User-Centered Design

- „Human-centred design is an approach to interactive system development that focuses specifically on **making systems usable**.“  
(ISO 9241-210: Human-centred design for interactive systems)
- „The User-centered design (UCD) process outlines the phases throughout a design and development life-cycle all while focusing on **gaining a deep understanding of who will be using the product**.“

# Die Kluft überbrücken

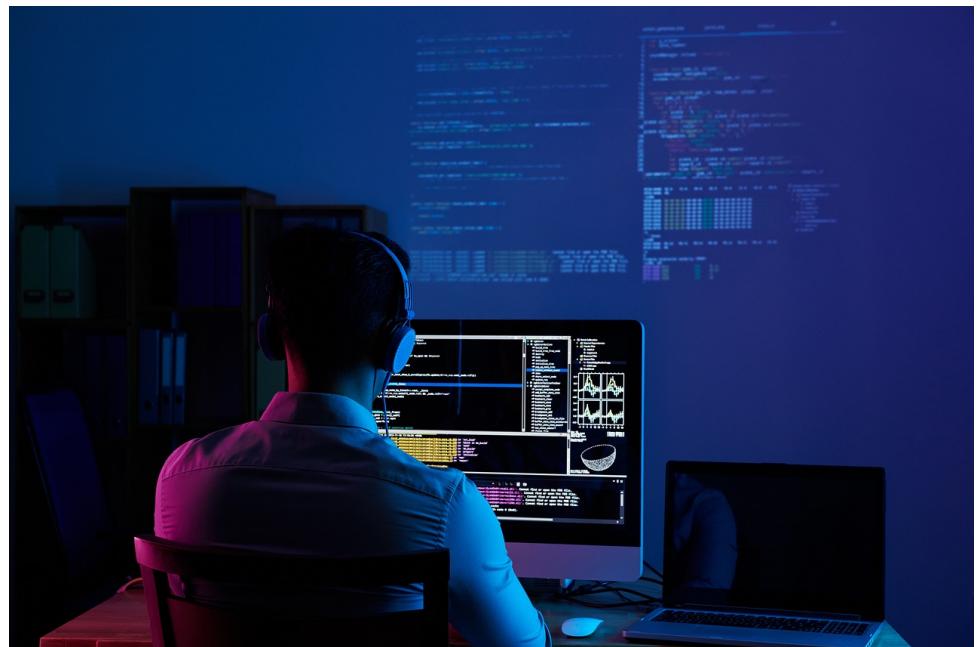
- Zwischen den Fähigkeiten und Anforderungen der Menschen und den Fähigkeiten des Computers
- Zwischen EntwicklerInnen und BenutzerInnen



# Kluft zwischen Entwickler und Benutzer

Level 1: **Entwickler ist Benutzer.**

Wenn man etwas entwirft, was man nur selbst verwenden wird.



Bildquelle Pixabay

# Kluft zwischen Entwickler und Benutzer

## Level 2: Entwickler verstehen das Produkt

Aber Achtung:

Dev. Team ≠ Typische User

DesignerInnen und EntwicklerInnen...

**wissen zu viel** über das Produkt

**sind zu geübt** im Umgang mit Computern

**hängen zu sehr** an ihren Entwürfen



Bildquelle Pixabay

# Kluft zwischen Entwickler und Benutzer

## Level 3: Design für eine fremde Domäne

Gut ausgebildete **BenutzerInnen**  
mit spezialisierten **Aufgaben**  
und **Experten-Wissen**  
in unbekanntem **Nutzungskontext**.



Bildquelle Pixabay

# User-Centered Design Prozess

Merkmale eines UCD Prozesses:

## 1. Früher Fokus auf BenutzerInnen und deren Aufgaben

- EntwicklerInnen müssen die BenutzerInnen kennen und verstehen
- Bei der Arbeit beobachten
- BenutzerInnen in den Design-Prozess einbinden

# User-Centered Design Prozess

## 2. Empirische Beobachtung

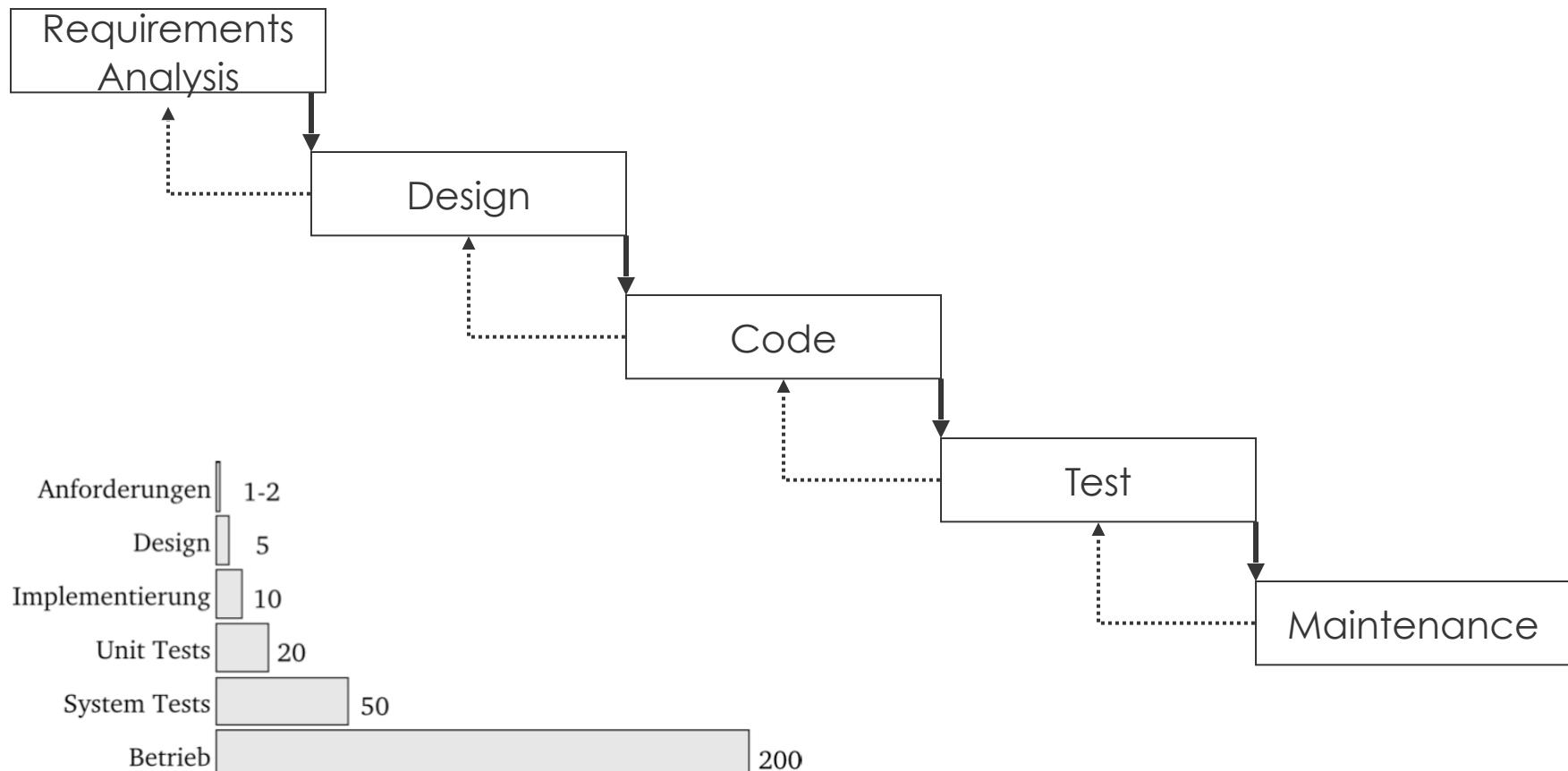
- User Research
- Reaktionen auf Szenarien und Storyboards in frühen Phasen
- Später Evaluierung mit Prototypen, User Testing, Feldstudien

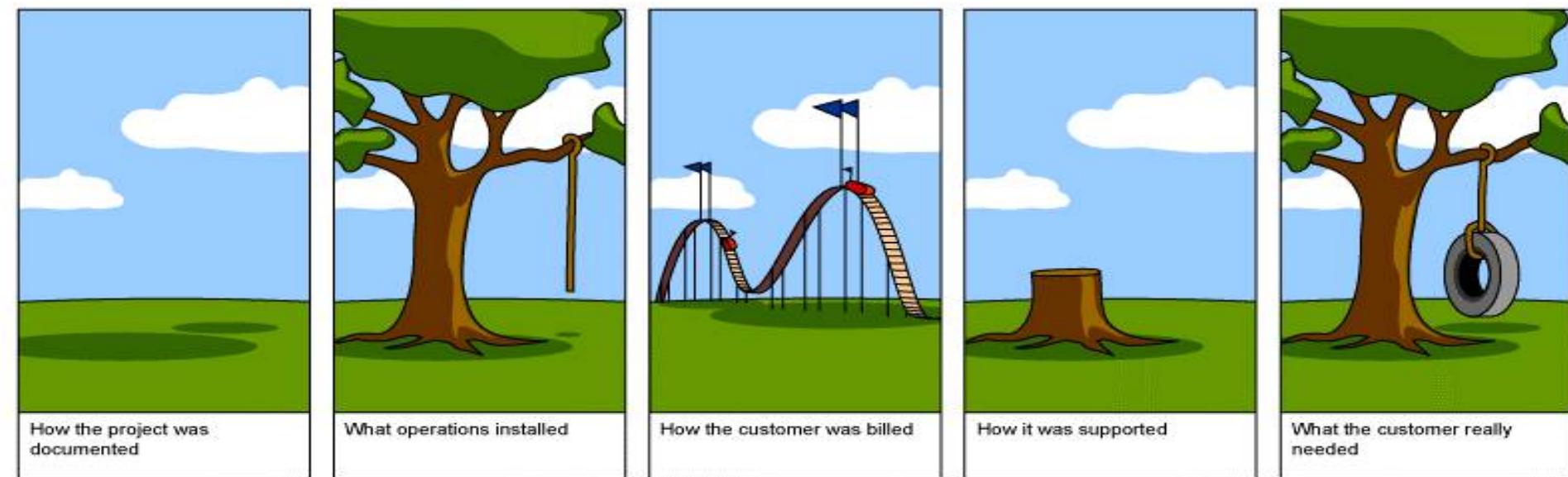
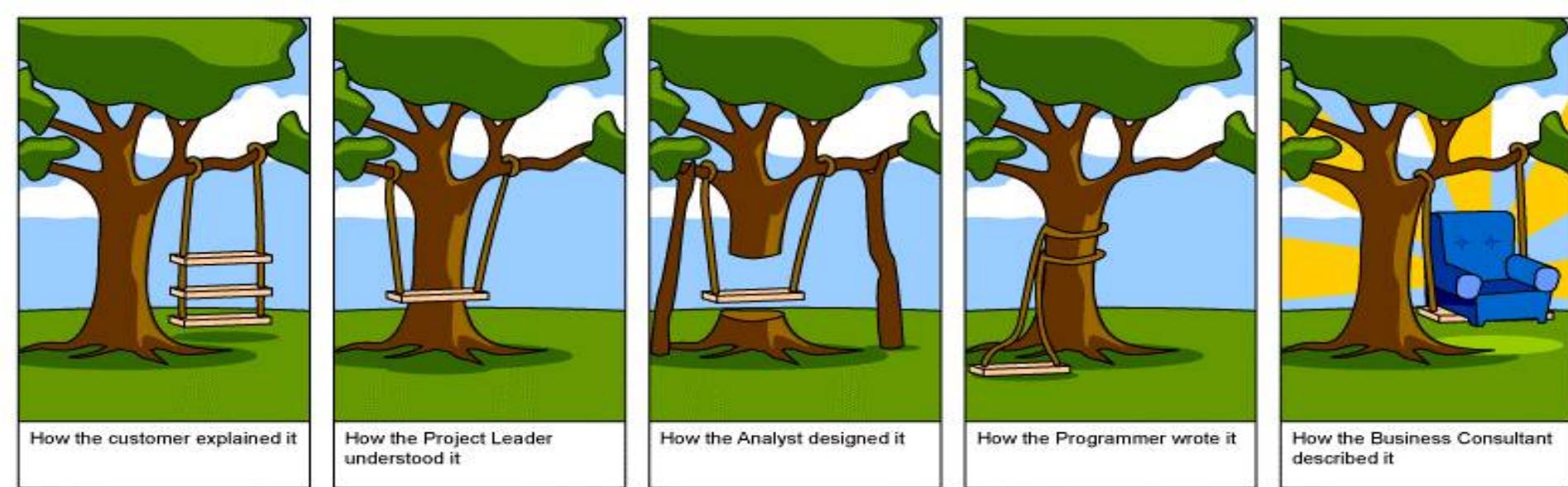
# User-Centered Design Prozess

## 3. Iteratives Design

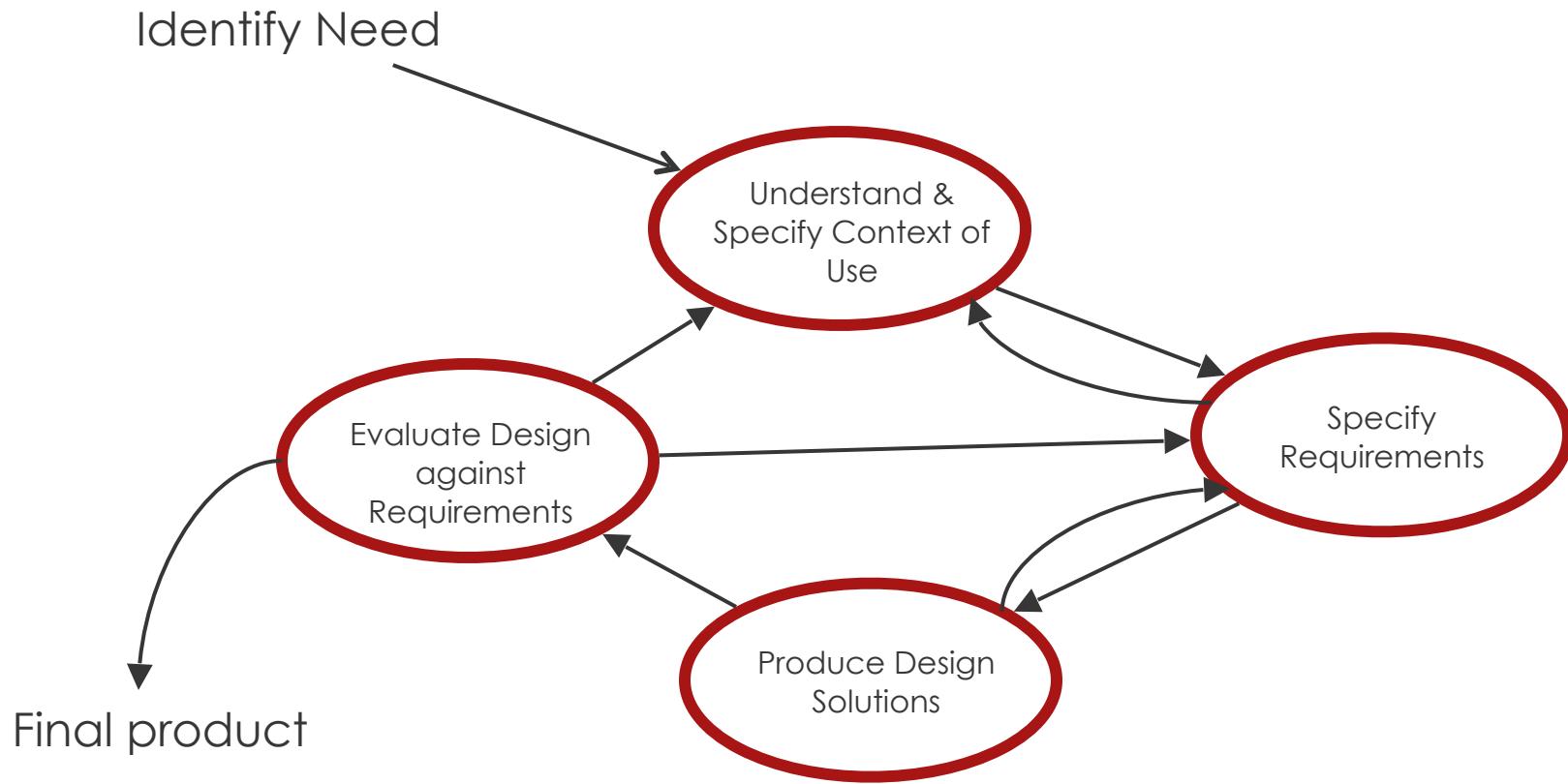
- Probleme können in jeder Phase im Design- und Entwicklungs-Prozess auftreten
- Deshalb: Iteratives Design

# Wasserfall-Modell





# User-Centered Design



# Iterative vs. lineare Prozesse

**Iterative**, inkrementelle,  
zyklische Prozesse: z.B.  
Agile Methoden, Scrum



**Lineare Prozesse:**  
Wasserfall,  
V-Modell

**Adaptiv**  
Flexibel bzgl Anforderungen,  
Agil, iterativ, inkrementell

≠ chaotisch

**Prädiktiv**  
Einfach planbar, aber das  
lineare Prozessmodell  
muss zutreffen

≠ diszipliniert

# Agile vs. UCD

- Agile favors increments:  
Incrementally add artifacts, but avoid re-work



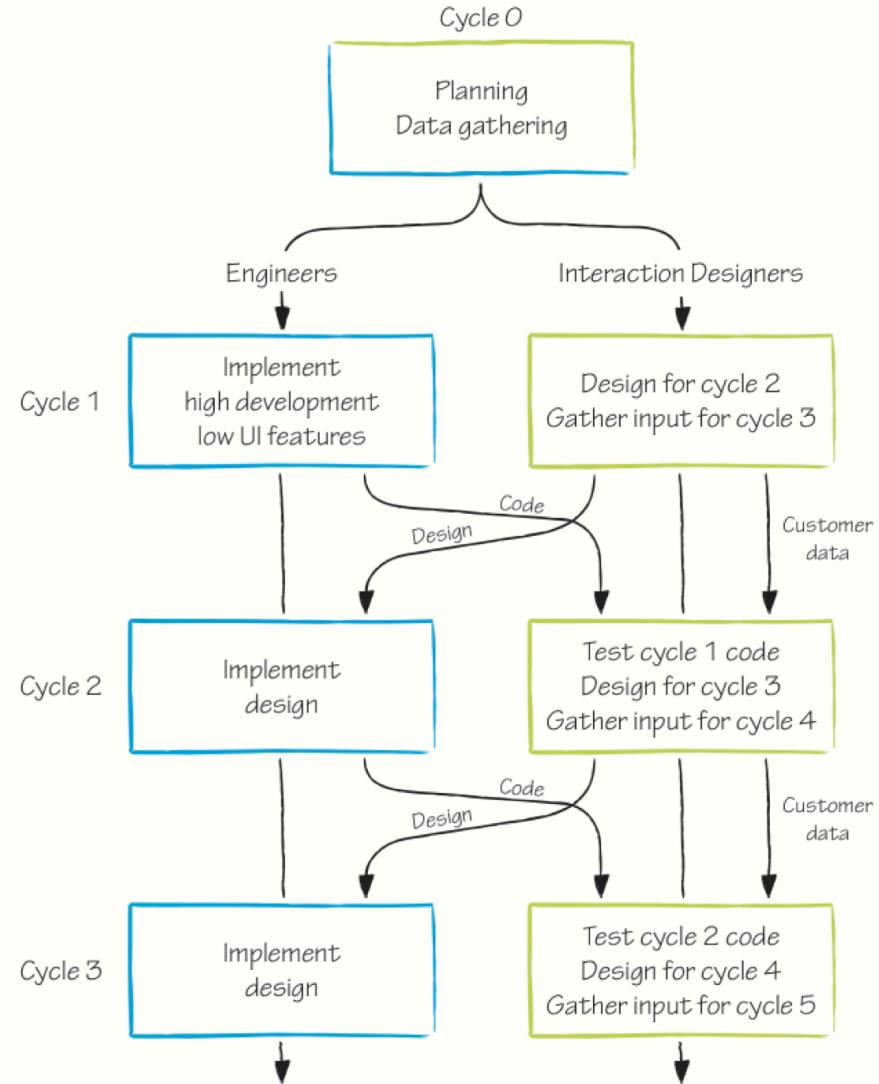
# Agile vs. UCD

- UCD wants iterations:  
Iteratively reworking, changing, and improving the same artifact



# Agile UCD

- „Little Research and Design Up-Front“
- Parallel Tracks
- One Sprint Ahead
- Methoden
  - Schnell
  - Flexibel
  - Informell
  - Geringer Overhead
- Kommunikation statt Dokumentation



# UCD ...aber: Users are not Designers!

- „No amount of data analysis can make up for a lack of talent.“ (Jeffrey Zeldman)
- „Users (and their data) should be there to inform designers, not substitute for them.“ (Dan Saffer)
- Es geht nicht darum, DesignerInnen zu ersetzen
- Es ist nicht Aufgabe der User, Probleme zu lösen

# The user is (not) always right

„When people tell you something's wrong or doesn't work for them, they are almost always right.

When they tell you exactly what they think is wrong and how to fix it, they are almost always wrong.“

- Neil Gaiman

# Grenzen und Möglichkeiten von UCD

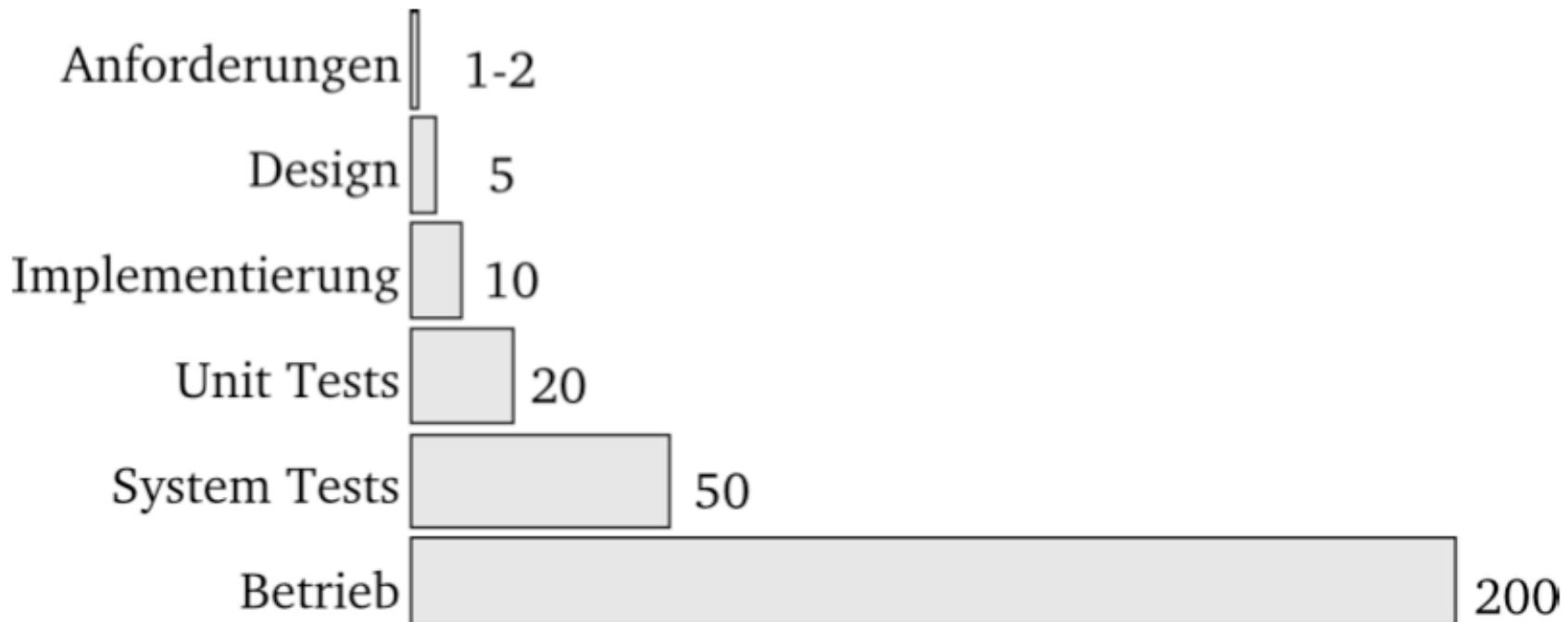
- „If I had asked people what they wanted, they would have said a faster horse.“ - Henry Ford
- Kein Ersatz für Grundlagenforschung und technologische Durchbrüche
- Schlecht geeignet für radikale Innovation
- Gut geeignet für iterative Verbesserung
- Einarbeitung in unbekannte Problemdomäne

# Erfolgsfaktor Usability Engineering

# Erfolgsfaktor Usability Engineering

- Einbindung der BenutzerInnen ist einer der bedeutendsten Faktoren für Erfolg oder Scheitern eines Software-Entwicklungsprojekts
- Standish Group CHAOS Report:
  - Project Success Factor #1: User Involvement
  - Project Challenged Factors:
    - Lack of User Input
    - Incomplete Requirements & Specifications
    - Changing Requirements & Specifications

# Kosten von Softwarefehlern



Stuart R. Faulk, 1995

# Cost/Benefit von Usability Engineering

- Usability Engineering ist viel Arbeit, aber...
  - Verringerung von Designänderungen in späten Entwicklungsphasen
  - Verringerung von Trainingskosten
  - Verringerung der Kosten für den Support
  - Verringerung der Fehler der BenutzerInnen
  - Steigerung der Produktivität
  - Positive Reviews und zufriedene KundInnen
- Problem: ROI ist schwer formalisierbar

# Cost/Benefit von Usability Engineering

- Return on Investment von Usability Engineering:
- Reduzierte Entwicklungskosten:

"The rule of thumb in many usability-aware organizations is that the cost-benefit ratio for usability is \$1:\$10-\$100. Once a system is in development, correcting a problem costs 10 times as much as fixing the same problem in design. If the system has been released, it costs 100 times as much relative to fixing in design." (Gilb, 1988)

# Cost/Benefit von Usability Engineering

- Key Performance Indicators:
  - Traffic, time spent, conversion rates, user performance, ...
- Zwei Studien zu Web Usability:
  - Nielsen, 2008: +83% Verbesserung
  - Nielsen, 2002: +135% Verbesserung

„We have now harvested most of the low-hanging fruit from the truly horrible websites that dominated the lost decade of Web usability (approximately 1993–2003).“

„Luckily, current usability ROI is so stupendously big (spend 10% to gain 83%) that it can decrease much more and still be a favorable proposition for business executives.“

# Cost/Benefit von Usability Engineering

- “Success stories”: The \$300 Million Button:

The designers [...] took away the Register button. In its place, they put a Continue button with a simple message: "You do not need to create an account to make purchases on our site. Simply click Continue to proceed to checkout. To make your future purchases even faster, you can create an account during checkout."

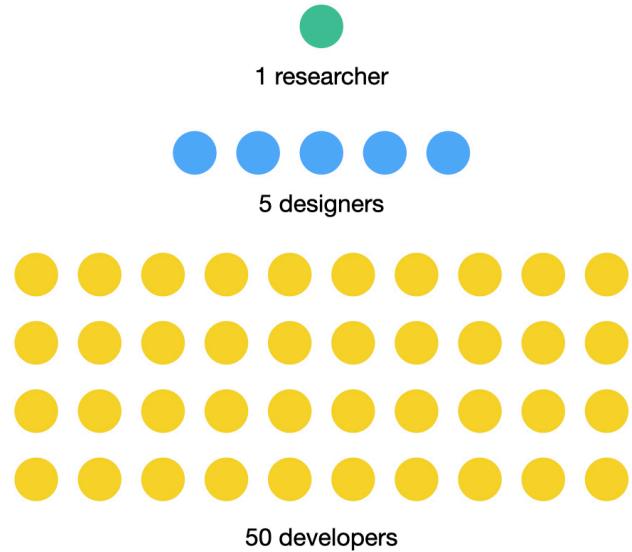
The results: The number of customers purchasing went up by 45%. The extra purchases resulted in an extra \$15 million the first month. For the first year, the site saw an additional \$300,000,000.

- Nützlich und interessant als exemplarische Beispiele, aber mit Vorsicht zu genießen:
  - Ergebnisse sind nicht zwangsläufig verallgemeinerbar
  - Was für ein bestimmtes System funktioniert hat funktioniert nicht zwangsläufig für jedes andere System

# UX-to-Developer Ratio

- 2020 Industry Survey: 1 - 5 - 50 Ratio
  - 1 Researcher pro 5 DesignerInnen
  - 1 DesignerIn pro 10 EntwicklerInnen
- Kein Zusammenhang zwischen UX-Dev-Ratio und UX Impact bzw. Maturity erkennbar
- Typischer Anteil von UX am Gesamtbudget eines Projekts: ~10%

**Most Common Extended Ratio**  
Researchers to Designers to Developers



NNGROUP.COM NN/g

# Discount Usability Engineering

- „Das Beste ist der Feind des Guten“
- Ein Test ist besser als kein Test
- Hallway Usability Testing:  
„A hallway usability test is where you grab the next person that passes by in the hallway and force them to try to use the code you just wrote.“ (Spolsky, 2000)

?



[deco.inso.tuwien.ac.at](http://deco.inso.tuwien.ac.at)

