

# Softwarequalitätssicherung durch Unit Testing

---

10. Mai 2022

Daniel Heuser & Melvin Weiershäuser

Web Technologien (WT)  
Medieninformatik Master  
Sommersemester 2022  
TH Köln

# Agenda

---

## Grundlagen

- SQS und (automatisiertes Testing)
- Motivation
- Testing Habits
- Testingarten
- Given, When, Then
- Unit Testing mit dem Jest Framework

## Hands-on Teil 1

## Vertiefung

- Code Coverage
- Test Driven Development

## Hands-on Teil 2

## Resumé

- Testing Tools
- Take-Aways

# Grundlagen

Was ist Softwarequalitätssicherung?

Was ist Software Testing?

- Prozess, zur Bewertung der Funktionalität einer Softwareanwendung

- Überprüfung, ob spezifizierte Anforderungen erfüllt werden

- Mängel werden identifiziert

Was erreicht man durch Software Testing?

Software Testing spart Geld

- Konzeption, Entwicklung und Testing
- Erkennung von Fehlern in frühen Phasen ist meist günstiger
- Tests so früh wie möglich implementieren -> TDD

### Mehr Sicherheit

- Vertrauenswürdigkeit der Produkte
- Personenbezogene Daten sollten vertraulich behandelt werden

### Bessere Produktqualität

- Einhaltung der Produktanforderungen
- Aufgaben können schneller, einfacher und effizienter erledigt werden

### Was ist ein Unit-Test?

- Unit-Tests (=Komponententest) überprüfen Komponenten
- Häufige Ausführung der Komponententests in agilen Methoden:
  - ... Ziel wird erreicht durch vollständige Automatisierung der Tests
  - ... Der Test kann in anderer Sprache als das Testobjekt geschrieben werden

... und wer entwickelt diesen?

- Team für Qualitätssicherung bei schwergewichtigen Methoden
- In agilen Vorgehensweisen ist eine Trennung nicht vorgesehen:  
... Entwickler schreiben die Unit-Tests zu den zu testenden Komponenten selbst



### Was zeichnet gute Unit-Tests aus?

- ➡ Sind isoliert
- ➡ Sichern jeweils genau eine Eigenschaft ab
- ➡ Vollständig automatisiert
- ➡ Leicht verständlich und kurz
- ➡ Hohe Codequalität
- ➡ Testen relevanten Code
- ➡ Vor dem zu testenden Code entwickelt

# Motivation Softwarequalitätssicherung und Unit Testing & Abfrage Testing Habits

### Testing

75%

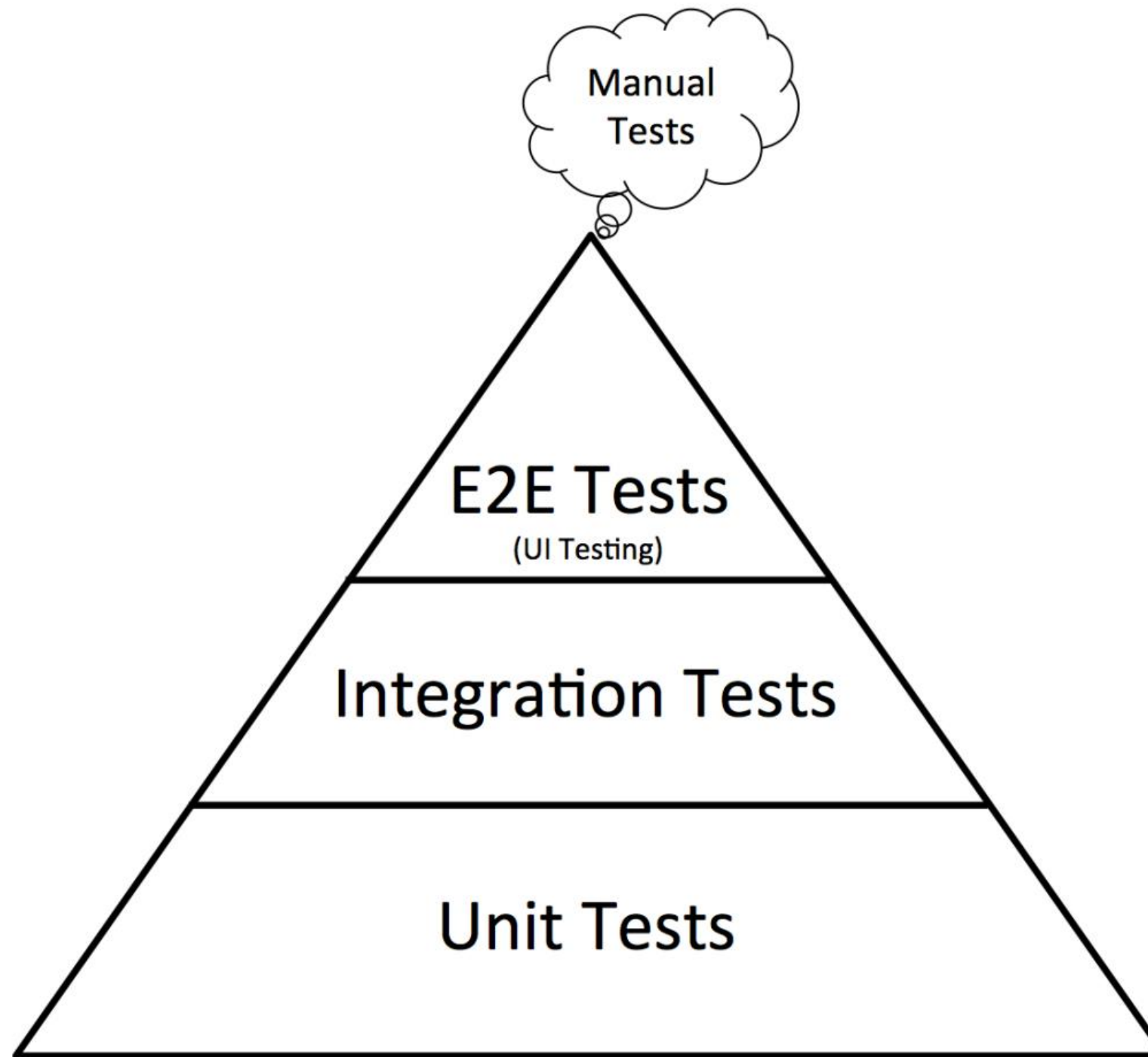
of all respondents say testing plays an integral role in their development.



44%

of all respondents are involved in the testing process.





Bildquelle: <https://www.ministryoftesting.com/dojo/lessons/the-mobile-test-pyramid>

# Given, When, Then

---

- Struktur, in der Testfälle spezifiziert werden.
- Ursprung im BDD

## Given

- Beschreibt Ausgangssituation
- Beinhaltet Schaffen der Ausgangssituation (Test-Setup)

## When

- Beschreibt das zu testende Verhalten.
- In der Regel wird der zu testende Code ausgeführt.

## Then

- Beschreibt die erwarteten Änderungen, die durch das getestete Verhalten entstanden sein sollen.

# Jest

---

„Delightful JavaScript Testing Framework“

- Open Source
- Vorangetrieben durch Facebook

Let's have a look!

Vertiefung

# Code Coverage

---

„Testabdeckung“ → Wie viel meines Codes wurde getestet?

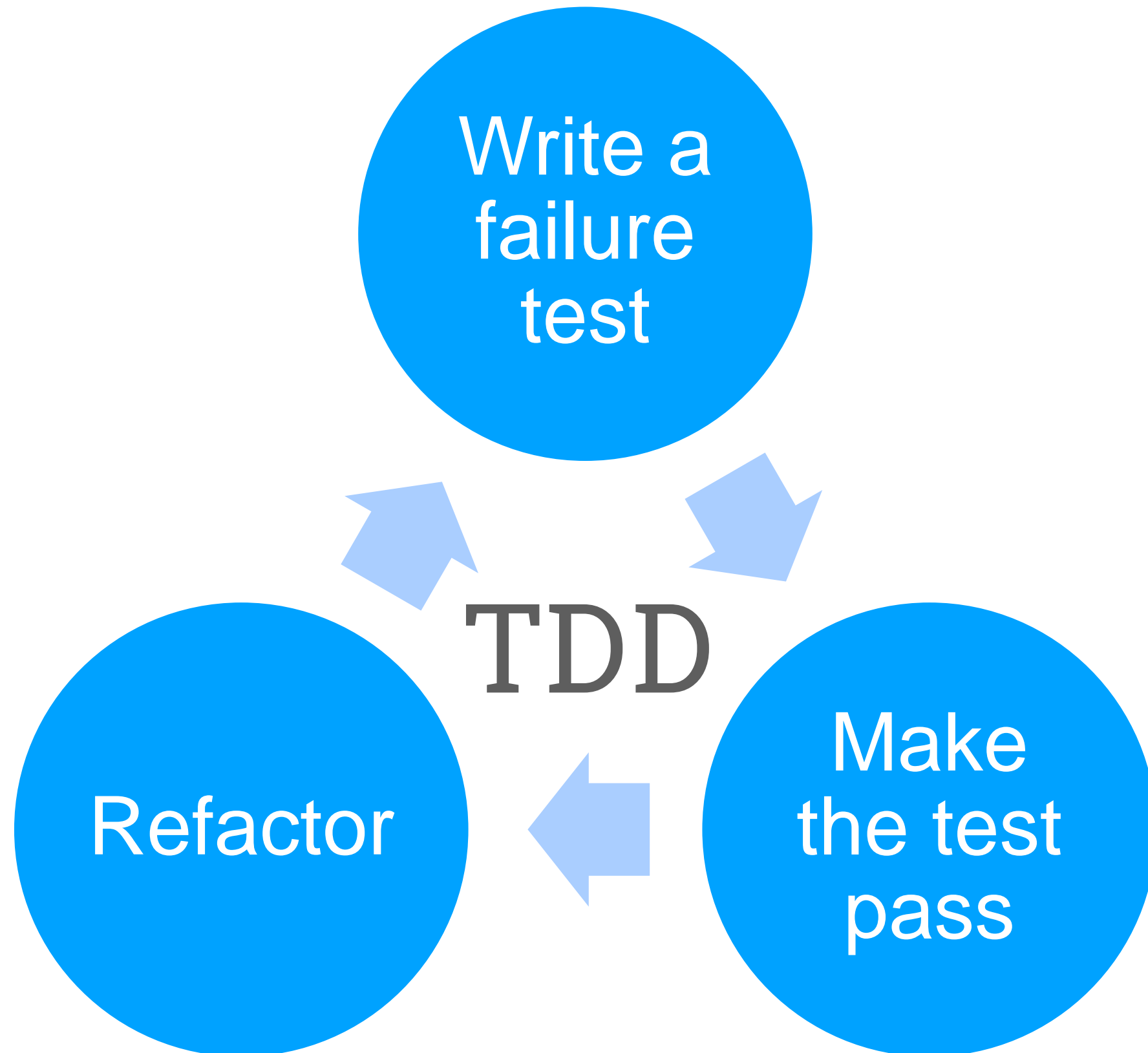
Function coverage

Statement coverage

Line coverage

Branch coverage





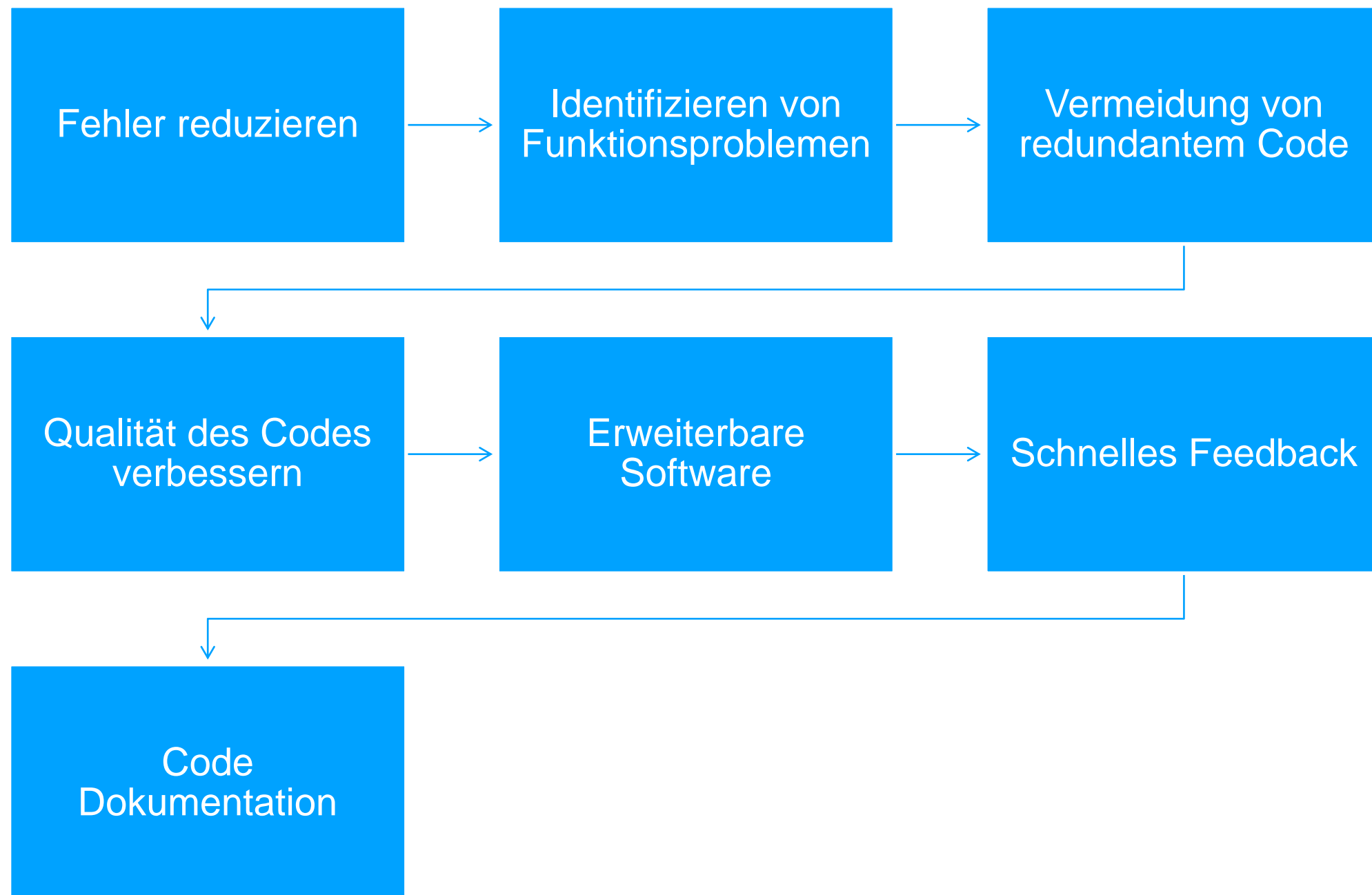
# Test Driven Development

---

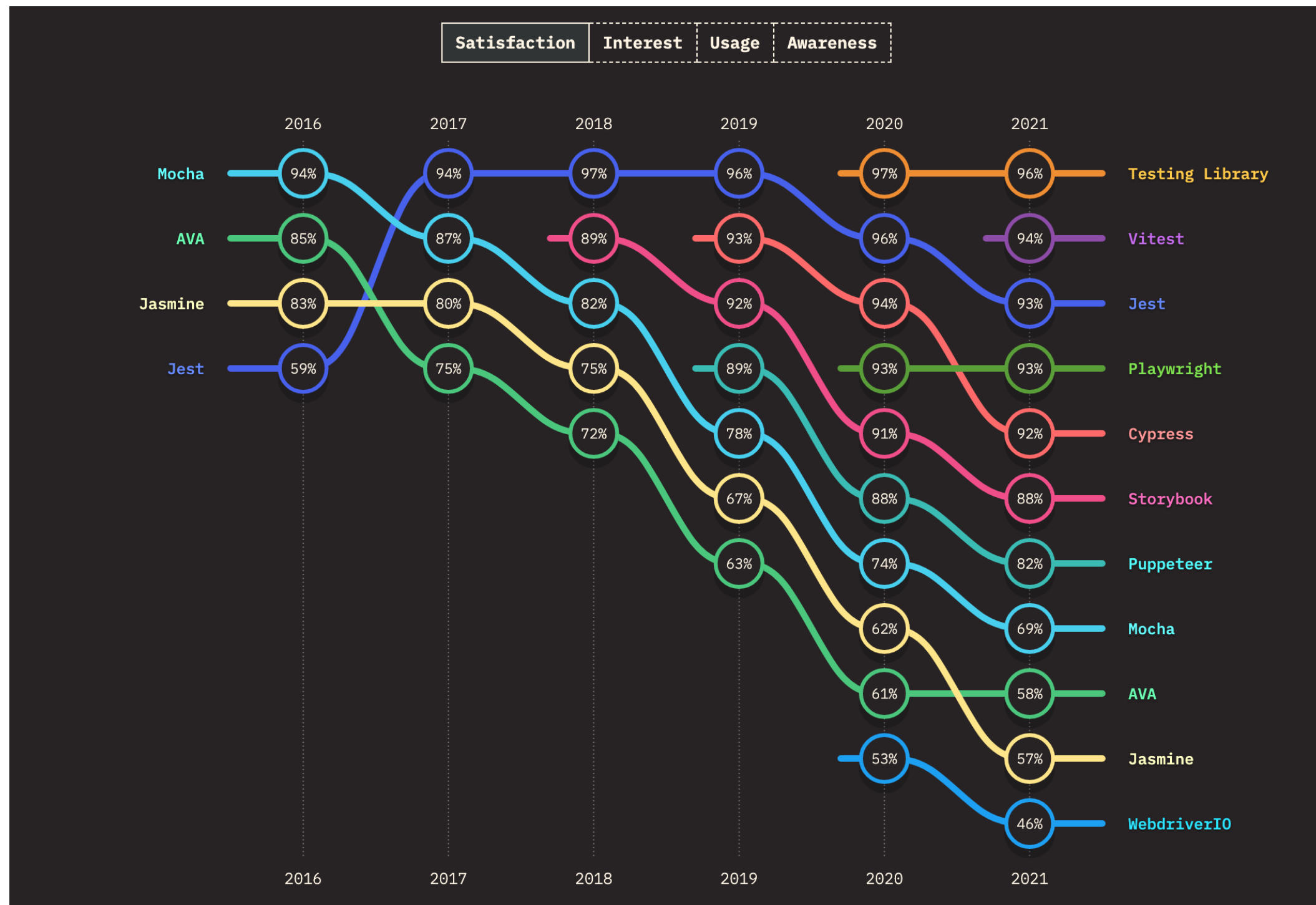
Test Driven Development konzentriert sich auf den Code, der notwendig ist, um den Testfall zu bestehen und reduziert ihn auf das Wesentliche.

# Test Driven Development

---



# Testing Tools



Bildquelle: <https://2021.stateofjs.com/en-US/libraries/testing>

# Take-Aways

---

- SQS bei so gut wie allen Softwareprojekten relevant.
- Unit Testing als etablierte und sinnvolle Maßnahme.
- Hilfreiche Tools (z. B. Jest) und Methodiken vorhanden.
- Unit Testing in Projekten zumindest stets evaluieren!

## Quellen

---

- Facebook, Inc. (o. J.), <https://jestjs.io/>
- Fowler, Martin (2013), <https://martinfowler.com/bliki/GivenWhenThen.html>
- Pittet, Sten (o. J.), <https://www.atlassian.com/continuous-delivery/software-testing/code-coverage>
- Martin, Robert C. (2013), Clean Code - Refactoring, Patterns, Testen und Techniken für sauberen Code
- JetBrains (2021), <https://www.jetbrains.com/lp/devecosystem-2021/testing/>
- Grafe, Thomas (2019), <https://blog.ccc-industriesoftware.de/qualitaetssicherung-in-der-softwareentwicklung/>
- IT-Agile (o.J.), <https://www.it-agile.de/agiles-wissen/agile-entwicklung/unit-tests/>
- Greif, Sacha & Pyltsyn, Alexey (2021), <https://2021.stateofjs.com/en-US/libraries/testing>

# Vielen Dank!

---

Noch Fragen? Gerne jetzt!

Kontakt:

[daniel.heuser@smail.th-koeln.de](mailto:daniel.heuser@smail.th-koeln.de)

[melvin\\_leon.weiershaeuser@smail.th-koeln.de](mailto:melvin_leon.weiershaeuser@smail.th-koeln.de)