

Testautomatisierung

Märchen, Möglichkeiten und
praktischer Nutzen



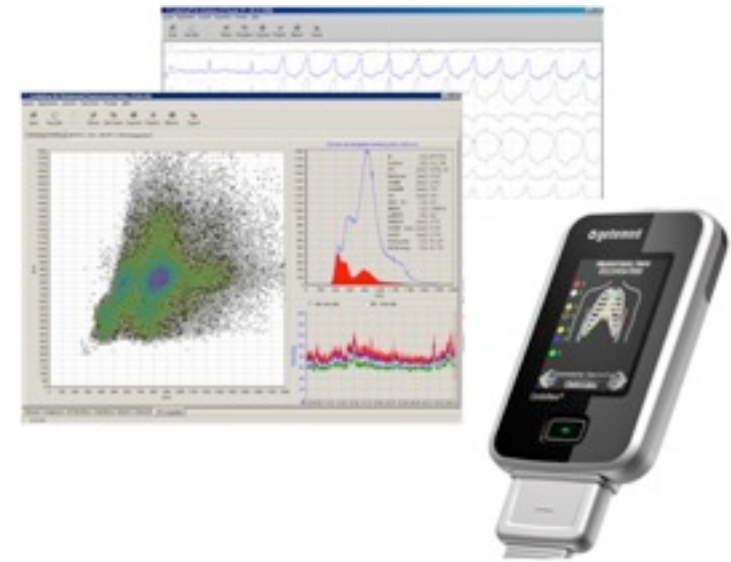
- Teltow, Brandenburg
- Gegründet 1984
- 68 Mitarbeiter (26 F&E)
- Entwicklung, QM, Herstellung, Vertrieb



reddot design award
honourable mention 2010



Vitalfunktions-Monitoring



Kardiologische Funktionsdiagnostik



Telemonitoring





- Strukturierter Software-Test ist etabliert
- ISTQB®
- Klassische vs. agile Vorgehensmodelle
- Reife erlaubt Fokus auf die Effizienz



- Systematischer Test mit großen Datenmengen
- Regressionstests
- Nicht-funktionale Tests



- nutzen nicht das volle Potential eines Testers
- machen betriebsblind
- kosten Zeit



Wiederkehrende Abläufe...

sind das, was *Automatisierung* am Besten kann!



Was automatisieren?

- Nicht nur die Testdurchführung
- Vom Testsupport bis zum Testmanagement
- Vom Script zum Framework



Zeitersparnis

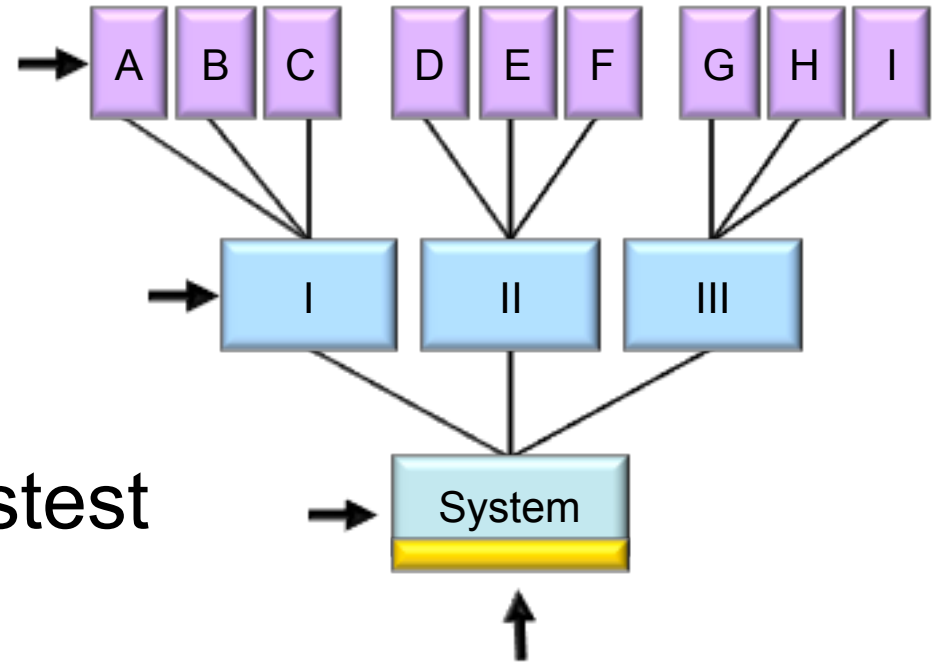
Phase	Ersparnis
Testdesign	-25 %
Testplanentwicklung	55 %
Testdurchführung	95 %
Ergebnisanalyse	50 %
Fehlerstatusüberprüfung	80 %
Auswertungserstellung	83 %
Gesamt	75 %

Quelle: „QA Quest“, Newsletter des Quality Assurance Instituts, November 1995



Automatisierung auf verschiedenen Teststufen

- Komponententest
- Integrationstest
- System- / Regressionstest
 - Transaktionen
 - Datenvergleiche
 - GUI
 - Schnittstellen

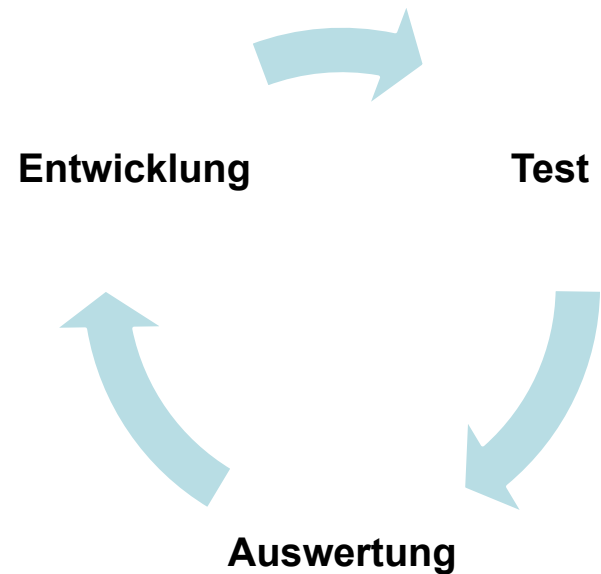




- Unit- und Unitintegrations-Test
- Anwendung der Testmethoden
- TDD - Test Driven Development



- Regressionstest
- BDD - Behavior Driven Development





- Tester können mit Unvorhergesehenem gut umgehen... ...Automatisierung nicht!
- Viele Fehlerquellen
 - Fehler in der zu testenden Applikation
 - Fehler im Testfall
 - Technische Änderung der Applikation
 - Fehler in den Testdaten
 - Fehler im Framework / Werkzeug / Testumgebung...
 - Fachliche Änderung der zu testenden Applikation



Was kann Automatisierung (nicht) leisten?

Sicherheitsnetz	Keine Garantie, dass keine Regressionen existieren	Bestätigung, dass bestimmte Fälle gewissen Gesichtspunkten entsprechen
Aufwandsersparnis	Automatisieren bedeutet zu Beginn mehr Aufwand als eine manuelle Durchführung	Das Team wird von wiederkehrenden Tests befreit, kann sich auf aktuelle Themen fokussieren
Fehlerfindung	Automatisierung findet sehr wenige Fehler	Automatisierung findet Seiteneffekte, für die ein manueller Test zu aufwändig wäre
Stabilität	Automatisierung muss laufend angepasst werden	Automatisierung zeigt Änderungen zuverlässig auf, wird nicht betriebsblind



Wie funktioniert Testautomatisierung?

- End-User wird simuliert
- Objekte werden anhand von Kriterien erkannt



Gewichtete Erkennung von Objekten

Eigenschaft	Erwarteter Wert	Gewichtung
Text	Bestätigen	45
Höhe	50	30
Breite	200	20

Confirm

Text: „Confirm“
Höhe: 50px
Breite: 200px
Score: 50



Bestätigen

Text: „Bestätigen“
Höhe: 50px
Breite: 200px
Score: 95



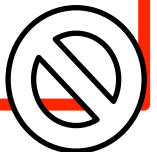
Bestätigen

Text: „Bestätigen“
Höhe: 130px
Breite: 250px
Score: 45



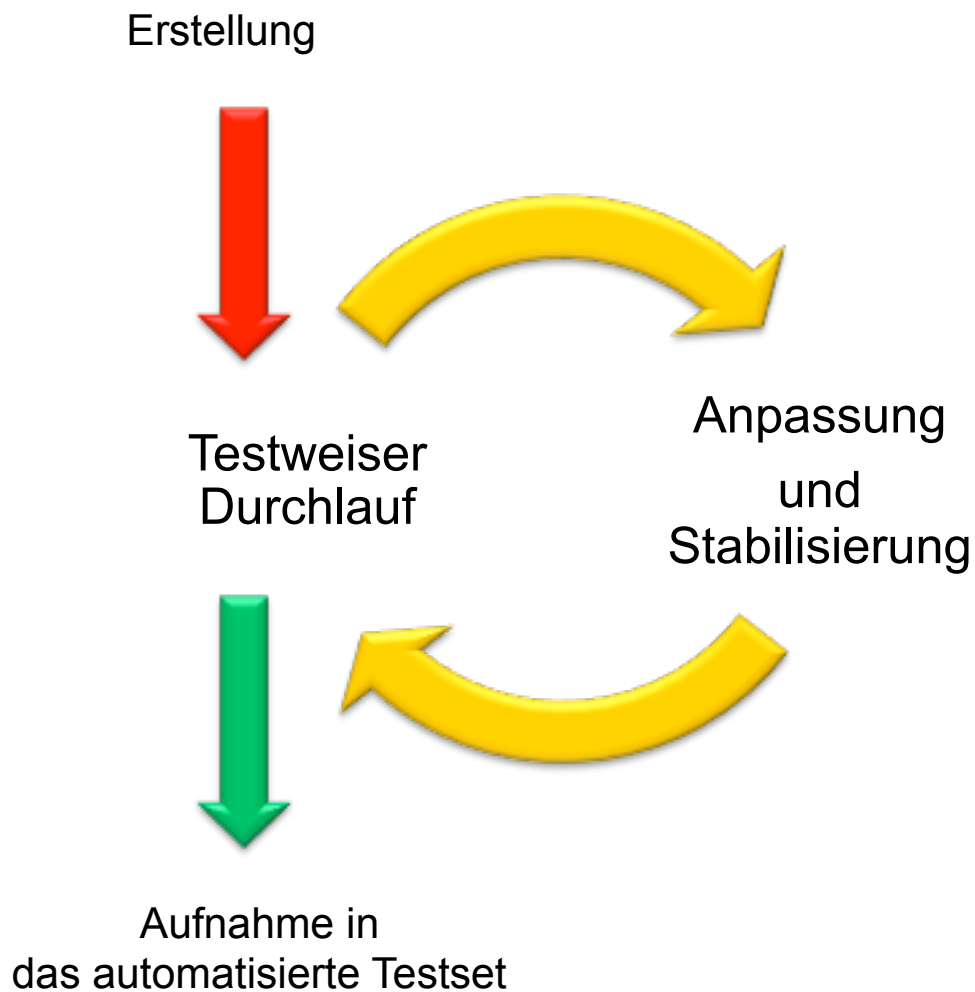
Logout

Text: „Logout“
Höhe: 130px
Breite: 250px
Score: 0





Entwicklung eines Testfalls





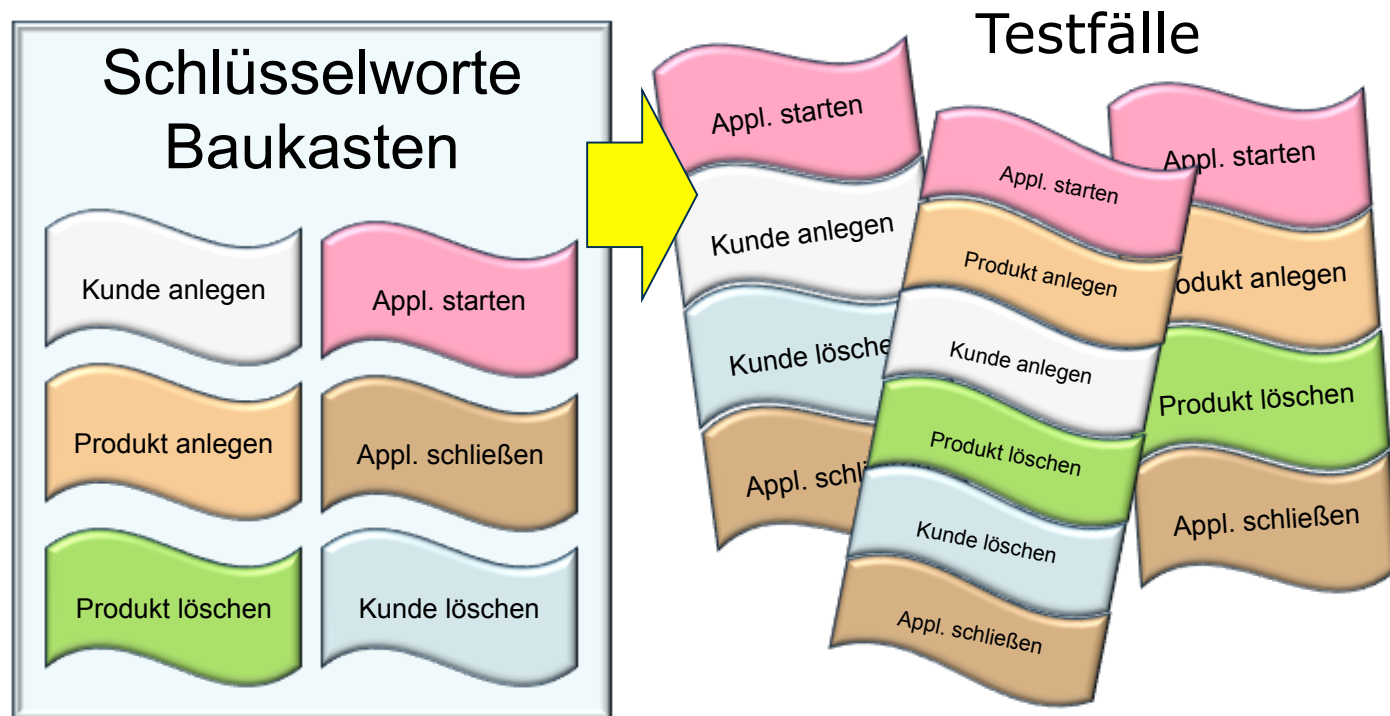
- Idee: Gleicher Ablauf, Daten variieren
- Testdatengenerierung z.B. mit ÄK, GW

Eingabe ins Textfeld „Name“: X
Klick auf Button „OK“

X = „Max Mustermann“
X = „Martina Musterfrau“
X = „Peter Mayer“
X = „Anna Müller“
...



- Schlüsselwörter bezeichnen Schritte im Geschäftsprozess
- Testfälle sind Sequenzen von Schlüsselwörtern
- Schlüsselwörter werden als Testscripte implementiert





- Fachexperten können Testfälle entwerfen
- Testmethoden: z.B. zustandsbasierter Test
- Kombination von Keyword- und Data-Driven



- Capture/Replay-Tools
- „agile“ Werkzeuge



- HP Quicktest Professional
- IBM Rational Functional Tester
- Tricentis Tosca
- Micro Focus Silktest
- Selenium
- Fitnesse



- Anforderungsdefinition
- Proof-of-Concept

- Kompatibilität
- Schnittstellen
- Objektidentifikation
- Programmierung

Die 10 + 1 Gebote der Testautomatisierung

Kardiologische Funktionsdiagnostik



Vitalfunktions-Monitoring



Telemonitoring





I. Hör' auf die Tester

- Tester haben einen Blick für Risiken
- Tester kennen das Testobjekt
- Gute Testfälle sind essentiell

„If you automate a mess, you get automated mess.“



II. Verwende die passenden Werkzeuge

- Kein Werkzeug deckt alle Bedürfnisse ab
- Es gibt spezifische Aufgaben
- Die Benutzer müssen damit zurechtkommen



III. Definiere Geschäftsprozesse

- Halte technische Details aus Deinen TF raus
- Verwende eine klar definierte „Sprache“
- Data Driven & Keyword Driven



IV. Denke erst über Dein Vorgehen nach

- Schreibe zunächst die Anforderungen
- Leite daraus eine technische Lösung ab
- Sei offen für verschiedene Testarten



V. Bilde nicht die zu testende Software nach

- Es ist leichter die Testdaten anzupassen als die Testskripts
- Testfall-Beispiel:
 - Nicht so: $A = 5$, $B = 3$, Ergebnis: $A+B$
 - Sondern: $A = 5$, $B = 3$, Ergebnis :8



VI. Definiere einen Ausgangszustand

- Definierter Start- und Endpunkte
- Räume vor dem Test auf
- Hab' Deine Testumgebung im Griff



VII. Respektiere den Detailgrad

- Zu wissen, was fehlgeschlagen ist, ist wichtig
- Halte Testfälle voneinander unabhängig



VIII. Sei ein Vollzeit-Automatisierer!

- Es ist viel zu lernen (Umgebung, SUT, Tools)
- Testautomatisierung ist keine
Nebensächlichkeit



IX. Verfolge Nutzen und Ziele

- Automatisierung muss Nutzen stiften
- Prüfe ob Fehler gefunden werden
- Automatisierung ist kein Selbstzweck



- Welche TF sind automatisiert?
- Welche Funktionalität wurde damit getestet?
- Wie hoch sind die Abdeckungen?
- Was kann verbessert werden...
 - an der Testautomatisierung?
 - am Testobjekt?



XI.

Don't Panic!

Richard Seidl

seidl@getemed.de



GETEMED Medizin- und Informationstechnik AG

Oderstraße 77

14513 Teltow

