

# Testautomatiseringens När, Var, Hur

prolore

## Agenda

### INNEHÅLL

- Vad är testautomatisering?
- Varför ska man testautomatisera?
- För- och nackdelar med testautomatisering?
- När ska man inte automatisera testerna?
- Vad krävs för att lyckas?
- Kommer ROI-kalkylen att hålla?
- Vanliga fällor och fallgropar och hur vi undviker dem
- Vi tar upp praktiska exempel på lyckade och mindre lyckade automatiseringsprojekt
- Framgångsfaktorer

prolore

## Mål

- Att Ni vet vad testautomatisering är, hur det fungerar och vad det kan ge!

prolore

## Vad är testautomatisering?

Kort introduktion till hur ett automatiserat test är uppbyggt och hur det fungerar

- Innebär att ett testautomatiseringsverktyg kör våra manuella funktionella tester (oftast...)
- Testerna är ofta skapade med hjälp av någon inspelningsmekanism i verktyget som genererar kod
- Denna kod beskriver för verktyget vad den ska utföra, exempelvis starta applikationer, klicka på knappar, fylla i fält, kontrollera värden och data, läsa i filer, etc. (d.v.s. Samma saker som en manuell testare...)
- Mycket kort demo...

prolore

## Varför ska man testautomatisera?

Testautomatisering har några fundamentala (och attraktiva) grundegenskaper

- Testar snabbt  
I princip så snabbt som den testade klienten klarar av
- Testar konsekvent  
Ett testverktyg glömmer aldrig att verifiera ett fält, läser aldrig fel, etc. (tar å andra sidan inga egna initiativ...)
- Testar konstant  
Ett testverktyg blir aldrig trött, vill inte äta lunch och jobbar gärna 24\*7 (kan å andra sidan uppskatta en omboost emellanåt...)
- Testar utan interaktion  
En bra testautomatisering är driftsäker och kan hantera de vanligaste problemen som kan uppkomma under testexekvering

prolore

## Varför ska man testautomatisera?

Bonusegenskap!!

- Testar datadrivet  
Indata (och kanske utdata/facit) i excelark eller databaser vilket möjliggör att enkelt köra samma testfall med många indata varianter.

prolore

## Varför ska man testautomatisera?

Kan vi nyttja de grundegenskaperna på ett bra sätt har vi lagt grunden för en lyckad testautomatisering och de vinster det *kan* innebära genom....

- Att man testat klart snabbare (spara tid)
- Att man testat med mindre resurser (spara pengar)
- Att man testat mer (bättre kvalitet)

Oftast någon sorts kombination av ovanstående även om varje organisation har sitt egna fokusområde.

prolore

## Fördelar med automatisering

- Snabbare information om en leverans kvalitet
  - Ökar testteamets produktivitet
  - Möjliggör parallell testexekvering (med flera datorer)
  - Möjliggör hög testtäckningsgrad med många indatavarianter
  - Tar över repetitiva och tråkiga testfall
  - Frigör resurser för "intelligenta" tester
  - Kan (skal!) överlämnas till förvaltnings/underhålls projekten
  - Kan (ibland) själv räkna ut förväntat resultat
- Bonusfördelar: Kan användas för att tvätta testdata, återställa databaser, generera personnummer etc.

prolore

## Nackdelar med automatisering

- Kunskapströskel
- Investeringskostnader (Verktyg, HW, Kunskap)
- Initiala automatiseringskostnader

prolore

## Vad ska man inte automatisera?

- Tester som kräver intelligens och känsla
- Applikationer som ännu inte är stabila (för tidigt i livscykeln)
- Applikationer som verktyget har svårt att stödja
- Testfall som inte har gått bra manuellt
- Testfall inom organisationer som är omogna inom manuell funktionstestning d.v.s. har man inte bra ordning på krav och test kommer automatiseringen bara göra "fel saker" fortare...

prolore

## Vad krävs för att lyckas?

- Engagemang
- Kunskap och/eller utbildningsinsatser
- Uthållighet och kontinuitet

prolore

## Kommer ROI-kalkylen att hålla?

Ta reda på/estimera :

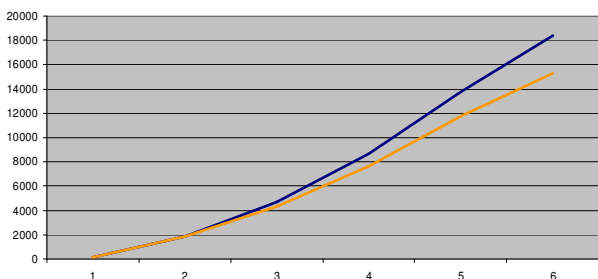
- antal testfall
- antal testfaser (tänk livscykel)
- antal omtester i resp. fas
- automatiseringsgrad
- antal indatavarianter per testfall
- exekveringstid per testfall
- underhållstid per testfall

+ ev. investeringskostnader (ska dessa ligga utanför projektet?)

prolore

## ROI forts...

Akkumulerad tidsåtgång (25% TA vs M)



prolore

## ROI forts...

Vad förstör ROI kalkylen:

- 1 Underhållsarbete åter upp tidsvinsten
- 2 Underhållsarbete åter upp tidsvinsten
- 3 Underhållsarbete åter upp tidsvinsten
- 4 Underhållsarbete åter upp tidsvinsten
- 5 Underhållsarbete åter upp tidsvinsten
- 6 Underhållsarbete åter upp tidsvinsten
- 7 Underhållsarbete åter upp tidsvinsten
- 8 Underhållsarbete åter upp tidsvinsten
- 9 Underhållsarbete åter upp tidsvinsten
- 10 Underhållsarbete åter upp tidsvinsten
- 11 Färre iterationer än planerat (har detta hänt?)
- 12 Automatiseringskunskapen "försvinner" (har hänt!)

prolore

## Vanliga fällor och fallgropar

- Utvecklare Urban:  
"Åh, hur svårt kan det vara, det följer ju med en tutorial! Och jag programmerade i VB redan på mellanstadiet"
- Projektledare Pia:  
"Jaha, då skickar vi Eva-Lena och Klas-Göran på testautomatiseringsverktygskurs så ni kan automatisera när ni kommer tillbaka!"
- Senior Vice President of Program Management Pär-Martin:  
"Vi ska automatisera 100% av våra tester så att vi spar 53 miljoner \$ på årsbasis och blir 37,42% effektivare än Denver kontoret!!"

prolore

## Vanliga fällor och fallgropar

- Utvecklare Urban:  
"Kurs? Åh, hur svårt kan det vara, det följer ju med en tutorial! Och jag programmerade i VB redan på mellanstadiet...Men jag inser vikten av att få en helhetsbild av verktyget för att kunna nyttja hela dess potential och därmed undvika framtida problem"
- Projektledare Pia:  
"Jaha, då skickar vi Eva-Lena och Klas-Göran på testautomatiseringsverktygskurs så ni kan automatisera när ni kommer tillbaka!...Och då kommer även Ann-Katrin att stötta ER eftersom hon testautomatiserade i Beta-Gamma projektet!"
- Senior Vice President of Program Management Pär-Martin:  
"Vi ska automatisera 100% av våra tester så att vi spar 53 miljoner \$ på årsbasis och blir 37,42% effektivare än Denver kontoret!! ...men en rimligare ansats med hänsyn till företagets kompetensprofil och erfarenhet är nog 55,7%, vi sparar då 27,9 m \$ och är ändå 23,97% bättre än Denver!"

prolore

## Vanliga fällor och fallgropar

- Respektera behovet av utbildning i verktygen
- Respektera behovet av stöd i arbetet
- Sätt realistiska mål på vad som ska uppnås

prolore

## Lyckat projekt 1

Korta fakta

- Branch: Finans
- Applikation i produktion
- 65 testfall per förvaltningstest
- Release 4 ggr per år (samt vid akut patchning)

Resultat

- Testexekveringstiden minskade från 2 dgr till 287 sek.

Insats

- 55h

Framgångsfaktorer

- Stabil miljö (redan i drift)
- 65 manuella testfall blev 7 automatiska testfall med flera indatavarianter
- Automatiska testfall kunde själva räkna ut facit

prolore

## Lyckat projekt 2

### Korta fakta

- Branch: Finans
- Applikationsändringar p.g.a. lagkrav, batch orienterat
- ~120 testfall
- Release 2 ggr per år (samt vid akut patchning)

### Resultat

- Testexekveringstiden minskade från 3 dgr till 287 sek.

### Insats

- 52,5h

### Framgångsfaktorer

- Testledaren hade grundläggande testautomatiseringskunskaper
- All testdata fanns i Excel och behövde inte anpassas
- Testerna genomfördes GUI-löst

prolore

## Lyckat projekt 3

### Korta fakta

- Branch: Spedition
- Nyutveckling och förvaltning
- 30 automatiserade testfall med 30 indatavarianter (~70% av tot test)
- Release 2 ggr per år + 8 planerade patchar + 2-3 regressioner

### Resultat

- Extremt snabb feedback vid patch och releaser.
- Minskade testexekvering från 3 dagar till ca 45 minuter

### Insats

- 300h

### Framgångsfaktorer

- Få GUI förändringar
- Relativt stabil miljö
- Generiskt GUI (høgt återutnyttjande)

prolore

## Misslyckat projekt 1

### Korta fakta

- Branch: Finans
- Nyutveckling
- Release: 12 st under utvecklingsprojektet

### Resultat

- Tekniskt avancerade tester föll när arkitekturen ändrades

### Insats

- XXXh

### Anledning

- Påbörjade testautomatiseringen för tidigt
- Utvecklare automatiserade (avancerade script, dålig testmetodik)

prolore

## Misslyckat projekt 2

### Korta fakta

- Branch: Software
- löpande utveckling
- Release: 4 ggr/år + general och private patches

### Resultat

- Tester föll när plattformen ändrades till Java

### Insats

- XXXXh

### Anledning

- Utvecklade automatiserade tester trots ett troligt byte av strategisk plattform

prolore

## Misslyckat projekt 3

### Korta fakta

- Branch: Life Science
- löpande utveckling
- Release: 4 ggr/år + service packs

### Resultat

- Endast 5 av 380 automatiska testfall användes fler än 1 gång.

### Insats

- 3000h

### Anledning

- Tester föll när testautomatiseringarna inte fick tillräckligt utrymme för att driva arbetet framåt. Vilket i sin tur berodde på dålig förankring och förståelse för TA på ledningsnivå.

prolore

## Framgångsfaktorer

- Välj ett verktyg som stödjer era strategiska plattformar och som erbjuder bra support\*
- Hantera utbildningsinsatserna\*
- Hantera behovet av löpande stöd/erfarenhetsutbyte\*
- Demonstrera hur testautomatisering fungerar för utvecklarna\*
- Förankra testautomatiseringen i organisationen
- Överväg att centralisera testautomatiseringskunskapen\*
- Engagemang och uthållighet (det tar tid att spara tid!)

prolore

## Frågor?

- ?

prolore

## Mål

- Att Ni vet vad testautomatisering är, hur det fungerar och vad det kan ge!

prolore

## Tack för mig!

- Presentationen finns på [www.prolore.se](http://www.prolore.se)
- Maila gärna:  
daniel.andersson@prolore.se  
eller ring:  
070-384 06 66

prolore