

Gårdagens testroller takler ikke dagens utfordringer

Magnus Halvorsen og Erik Rogstad





Eksempel: Testutlysning fra fortiden

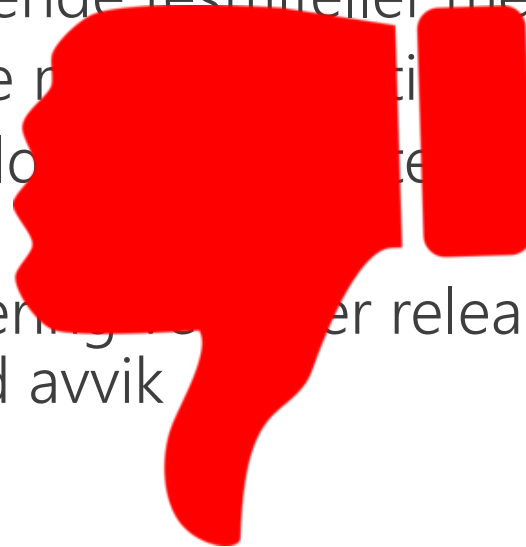
- Arbeidsoppgaver
 - Utarbeide testtilfeller basert på kravspesifikasjon
 - Gjennomføring av systemtest og brukertestet
 - Vedlikehold av manuelle testtilfeller
 - Defect management
 - Overvåke og rapportere jevnt testfremdrift
- Ønsket kompetanse
 - Sertifisert innen HP ALM/Quality Center og HP Sprinter





Eksempel: Forsøk på automatisering

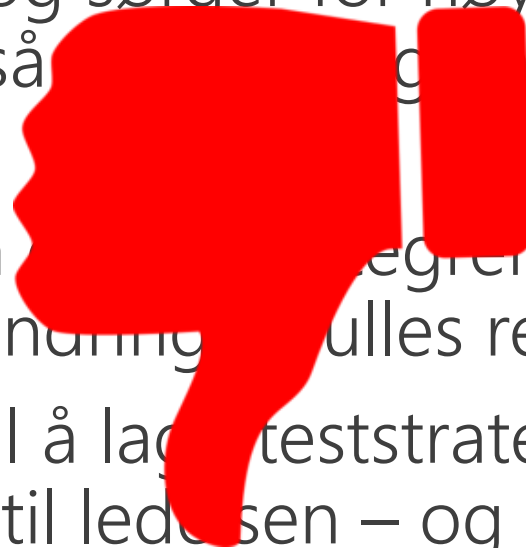
- Arbeidsoppgaver
 - Automatisere eksisterende testtilfeller med Selenium IDE
 - Vedlikeholde test suite med automatiserte regresjonstester
 - Oppsett og vedlikehold av testmiljø for automatiserte tester
 - Følge opp testeksekvering etter release og rapportere feil tilbake til utvikling ved avvik





Eksempel: Testing uten testkompetanse

- Vi jobber i tverrfaglige team hvor de fleste utviklerne er sertifisert innen TDD og sørger for høy enhetstestdekning for all kode. De er også veldig gode til å gjennomføre parvis kodegjennomgang.
- DevOps-tankegangen er implementert, alle prosesser er automatisert og alle endringer sendes rett ut i produksjon.
- Trenger en testleder til å lage teststrategi, overordnede planer, fine rapporter til ledelsen – og å kalle inn til møter





Eksempel: Testansvarlig

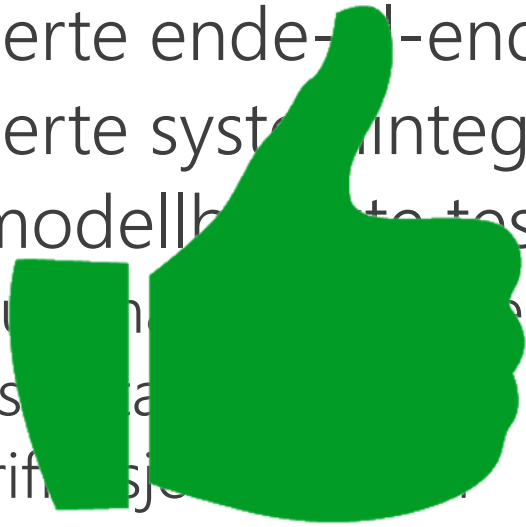
- Vi søker etter en **testansvarlig** til et tverrfaglig team:
 - Hovedansvar: automatisering av tester på systemtestnivå
 - Legge til rette for kontinuerlig levering ved å påse at alle testnivåene blir dekket av automatiserte tester iht testpyramiden
- Ønsket kompetanse:
 - Testmetode og testteknikker
 - Erfaring med praktisk utarbeidelse av automatiserte tester
 - Erfaring fra utviklingsprosjekter med komplekse integrasjoner
 - Erfaring fra prosjekter som benytter smidig utviklingsmetodikk med tett dialog med utvikler, forretningside og andre fagkompetanser
 - Det er en fordel å ha bakgrunn som utvikler





Eksempel: Testutvikler

- Vi etterspør en **testutvikler**, som skal:
 - Utvikle automatiserte ende-til-ende-tester
 - Utvikle automatiserte systemintegrasjonstester
 - Kunne anvende modellbaserte testteknikker for:
 - systematisk og automatisk generering av tester
 - generering av testdata
 - automatisere verifikasjon





Eksempel: Testarkitekt

- Vi har et stort og komplekst system, hvor test er svært utfordrende
- Vi trenger en teknisk og operasjonell helhetstenker innen test
- Som **testarkitekt** vil du
 - finne ut hva og hvordan man bør jobbe på ulike testnivåer og i hvilke miljøer
 - heve blikket og stake ut kursen gjennom mylderet av teknologier og muligheter
 - sørge for helhetlig test på tvers av teamene
 - sette premissene for effektiv og forsvarlig håndtering av testdata





Hva rører seg innen systemutvikling?

- Hva er hovedtrendene innen systemutvikling?
- Hvordan påvirker dette testingen?
- Hvordan kommer test til å organiseres?
- Hvilke typer test blir spesielt viktig fremover?
- Hva blir viktig kunnskap å ha?
- Hvordan ser fremtidens tester/testleder ut?



Hovedtrender innen systemutvikling

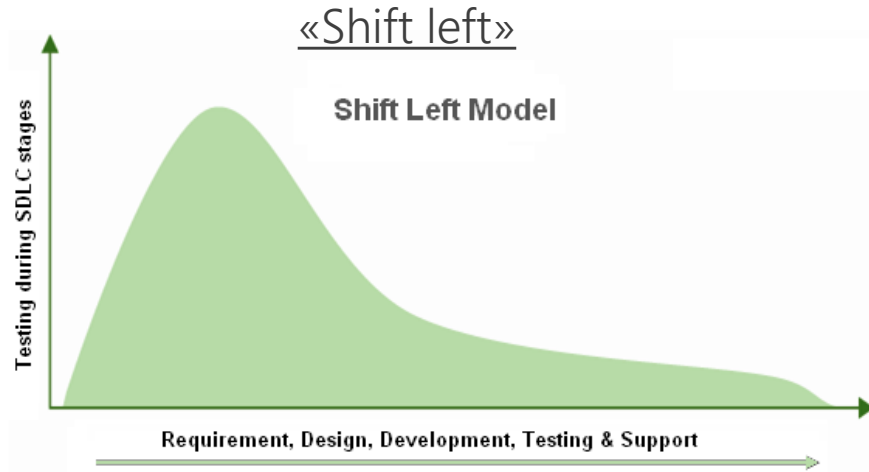
- «Shift left», smidig, tverrfaglige team og BDD
- Kontinuerlig levering og DevOps
- Web og apper overalt
- Fra silo-systemer til mikrotjenester
- PaaS og skytjenester

Felles for disse:

- Et mangfold av teknologier å forholde seg til
- Høy grad av automatisering



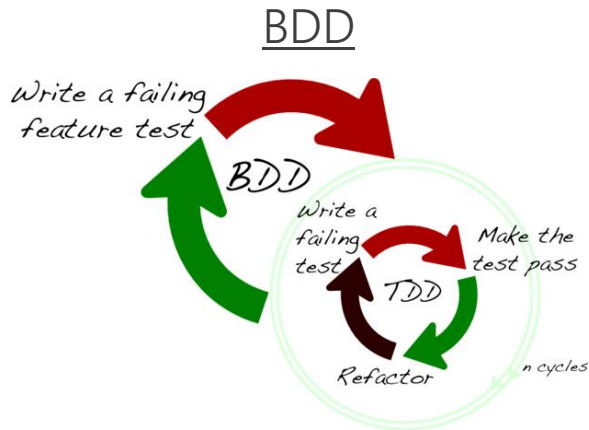
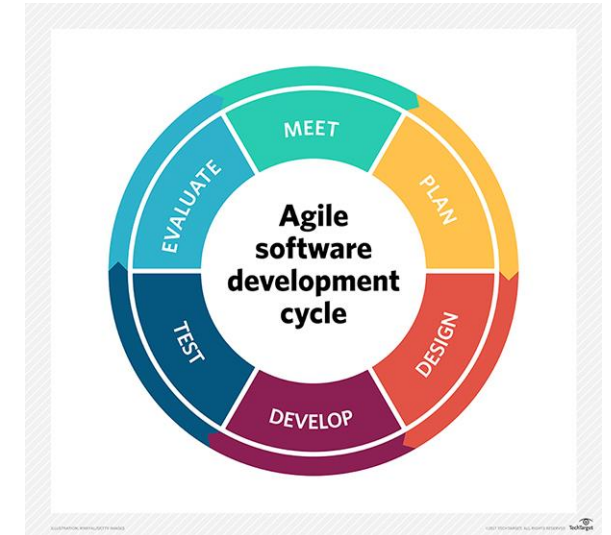
«Shift left», smidig, tverrfaglige team og BDD



Tverrfaglige team



Smidig utvikling





«Shift left», smidig, tverrfaglige team og BDD

Innebærer:

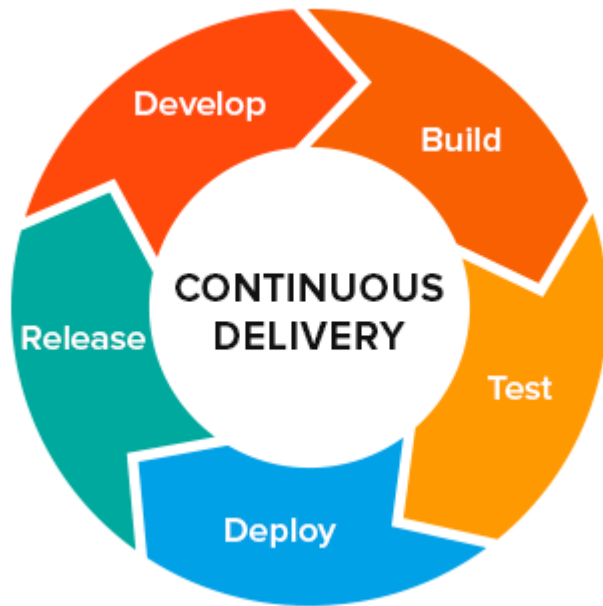
- Hyppige leveranser
- Selvgående team – tar ansvar for å levere ferdig funksjonalitet
- Just-in-time-tankegang
- Test flyttes stadig tettere på spesifikasjon og utvikling
- Spesifikasjon = test = dokumentasjon

Konsekvenser for testing:

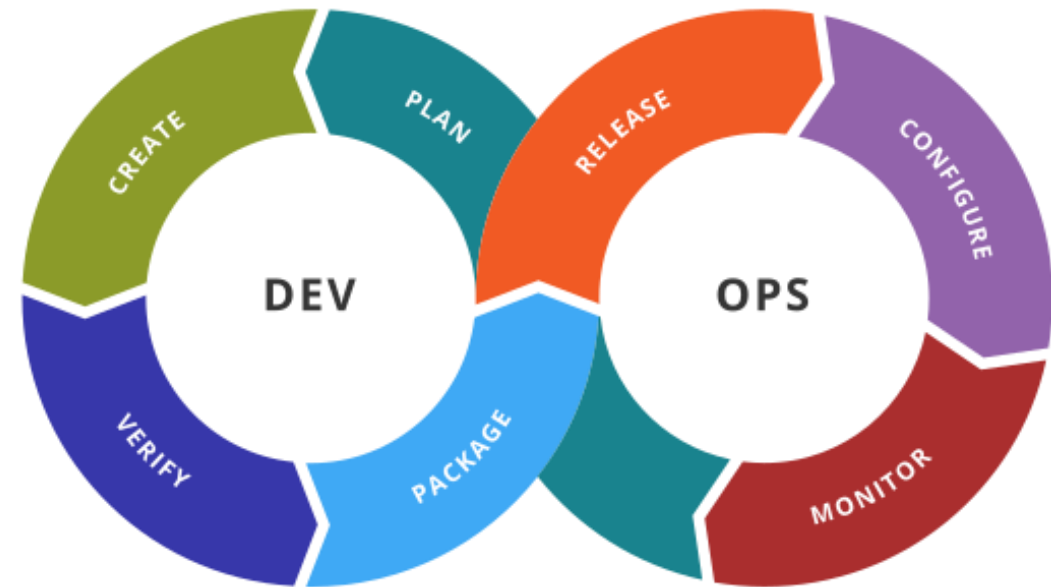
- Gir test større innflytelse
- Gir økt sporbarhet fra krav til test
- Felles ansvar for testkodebasen på alle testnivåer
- Må kunne kommunisere og samarbeide godt med både utviklere, funksjonelle og driftere
- Test av uferdige produkter som krever mer teknisk innsikt og forståelse
- Test på bakgrunn av ustabile spesifikasjoner

Kontinuerlig levering og DevOps

Kontinuerlig levering



DevOps





Kontinuerlig levering og DevOps

Innebærer:

- Drift og utvikling smelter sammen
- Prosessforbedring og helautomatiserte prosesser
- Hyppige leveranser

Konsekvenser for testing:

- Økt fokus på og bedre tilrettelagt for automatisering
- Automatiserte tester på alle testnivåer blir en integrert del av leveransen
- Testere må bry seg om byggkjeder m.m.
- Press på å kutte ned på / droppe tester som ikke så lett kan automatiseres – mer «testing i produksjon»

✓ Web og apper overalt



Web og apper overalt

Innebærer:

- Stort mangfold av bruksmønstre og klienttyper
- Alt er på nett hele tiden
- Økt sannsynlighet for og større konsekvenser av sikkerhetsbrister

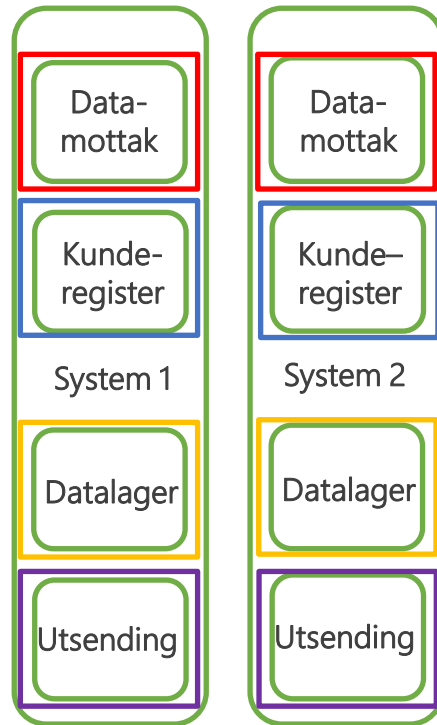
Konsekvenser for testing:

- Viktig å teste kompatibilitet på tvers av enheter og nettlesere
- Ta hensyn til universell utforming
- Sikkerhetstesting må bli bedre integrert med annen testing

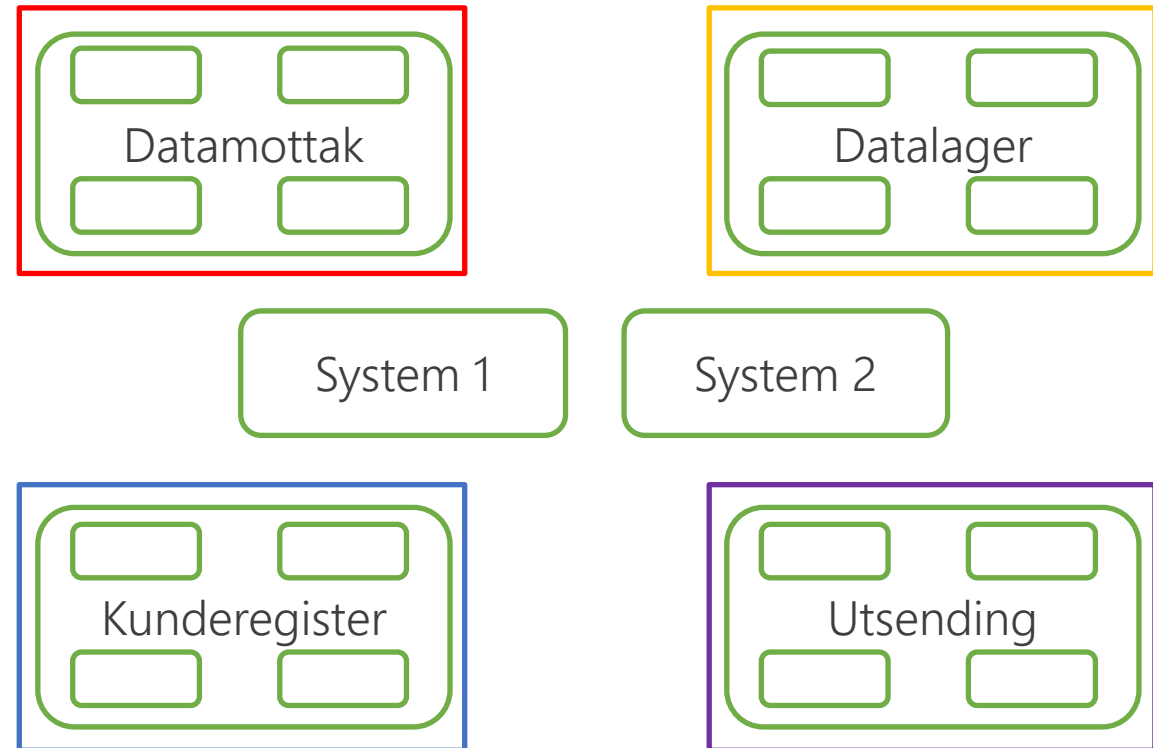


Fra silo-systemer til mikrotjenester

Silo-systemer



Komponentbasert systemarkitektur med mikrotjenester





Fra silo-systemer til mikrotjenester

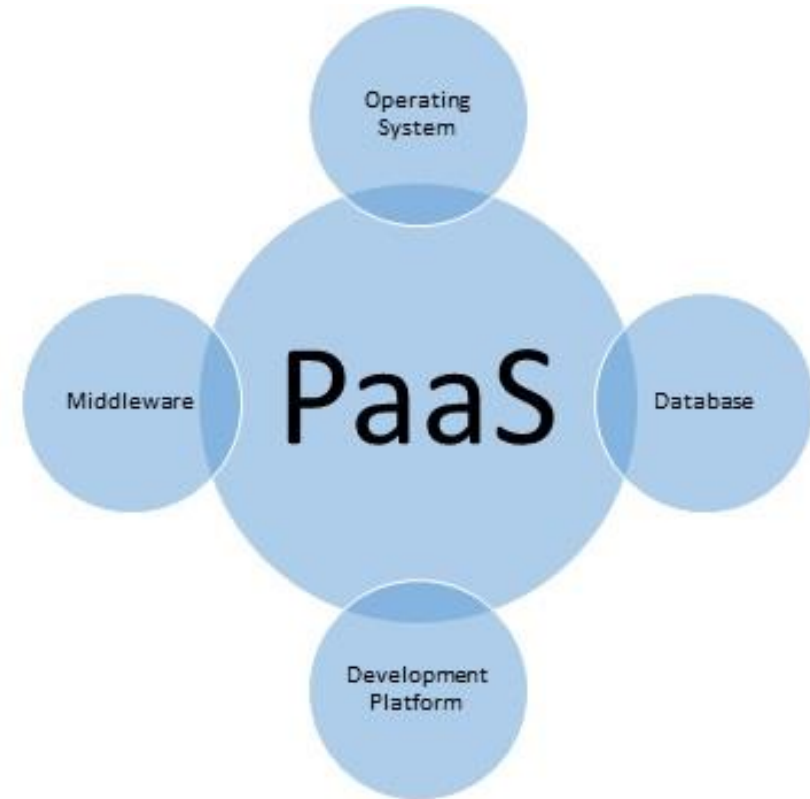
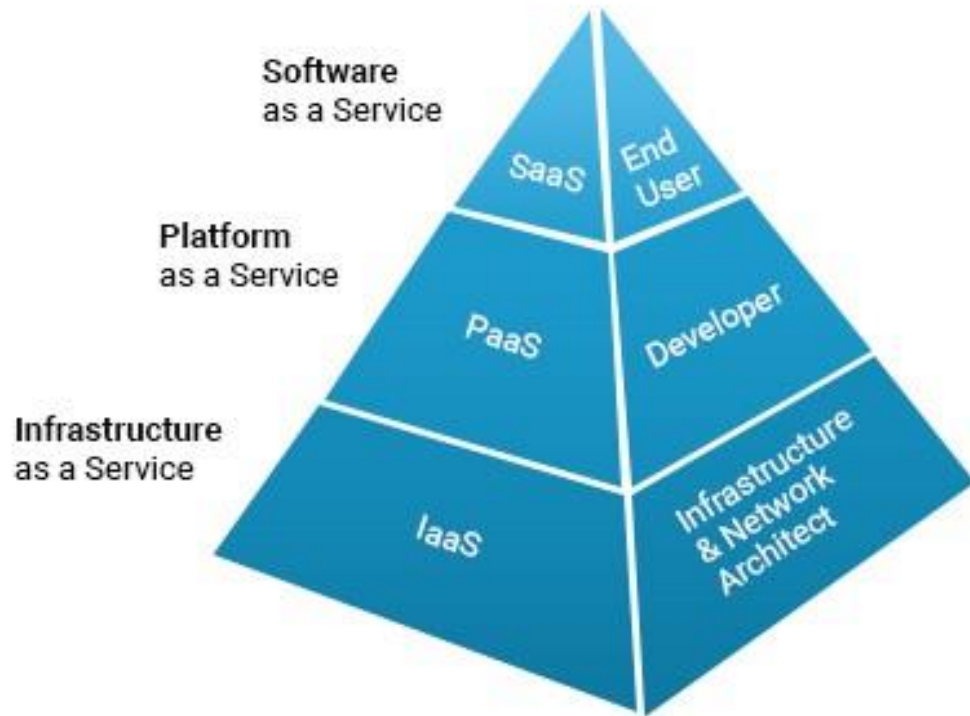
Innebærer:

- Hver komponent blir enklere, men totalen blir mer kompleks
- De klassiske systemgrensene viskes ut – alt henger sammen med alt
- Større avhengigheter på tvers av team og prosjekter
- Feilsituasjoner er mer normalt og krever bedre spesifisering

Konsekvenser for testing:

- Flere, men mindre «svarte bokser» å teste -> økt testbarhet
- Krevende å sette opp og administrere komplette testmiljøer, inkludert testdata
- Økt behov for koordinerte testaktiviteter og systemintegrasjonstesting
- Isolerte systemtester (tjenestesimulering) blir veldig sentralt
- Økt fokus på robusthetstest og test av gjenopprettingsmekanismer

✓ PaaS og skytjenester



PaaS og skytjenester

Innebærer:

- Et nytt abstraksjonslag
- Skalerbarhet og redundans
- Større krav til selvdiagnose og gjenoppretting etter feilsituasjoner
- Automatisering av miljøoppsett

Konsekvenser for testing:

- Økt behov for ikke-funksjonell testing
- Testmiljøer kan settes opp og rives ned med et tastetrykk
- Nye muligheter for automatisert test i byggkjeden
- Testlabben kan leveres over Internett



Oppsummering av hovedtrender

- **Testautomatisering:** Automatiserte tester på alle testnivåer er en integrert del av leveransen
- **Kontinuerlig levering:** Hele teamet må ta ansvar for test
- **Isolert systemtest:** Test av komponenter/systemer i isolasjon blir viktig for å effektivisere testarbeidet tilstrekkelig
- **Robusthetstest:** Arkitektur-, og infrastruktur-trender skjerper krav til robusthet
- **Sikkerhetstest:** Stadig økende eksponering av tjenester over internett stiller strengere krav til sikkerhet
- **Kompatibilitetstest og universell utforming:** Web og apper overalt



Hvordan organisere test for fremtiden?

- Slipp til test enda tidligere – ikke bare på utviklingsstadiet, men også på arkitekturstadiet
 - Ha testere som kan være bindeledd mellom utvikling og spesifisering
 - Ha testere med teknisk kompetanse, gjerne utviklerbakgrunn
 - ... men bevar testkompetansen!
-
- Noen må tenke test «i stort»
 - Eks: arkitektur, verdikjeder, testdatasett
 - Noen må tenke test «i smått»
 - Eks: brukerhistorier, BDD / testbar spesifisering, APler

Fremtidens testroller

1. Jobbe teknisk og utviklernært med verdikjeder – teknisk tester erstattes av **testutvikler** (utvikler med testkompetanse)
2. Ta det overordnede ansvaret for test og kvalitet i tverrfaglig team (mini-testleder) – **testansvarlig**
3. Løfte blikket, abstrahere, se sammenhengene og tenke helhetlig testing (testbarhet, testnivåer, vedlikeholdbarhet av tester) – **testarkitekt**
4. Teststrategi, teambygging, overordnet fremdriftsplan og rapportering i ledergruppen, risikovurderinger, koordinering utad - **testleder**

