

Agil systemforvaltning

Metoder og teknikker

Poul Staal Vinje

Certified Scrum Master

Certified Scrum Practitioner

poul.vinje@gmail.com

www.agile-metoder.dk

Indhold

Agil systemforvaltning	1
1. Indledning	2
2. Teknikker og metoder	2
2.1 Scrum	2
2.2 Testdrevet forvaltning	2
2.3 Extreme Programming (XP)	3
2.4 Automatiseret test	4
2.5 Agile processer	4
2.6 Teams	5
2.7 Issue Management	5
3. Hasterettelser	5
4. Best Practice	5

1. Indledning

Agile metoder og teknikker er velegnede til systemforvaltning. De introducerer et passende niveau af fælles processer. For meget bureaukrati vil få forvaltningen til at sande til. For lidt vil ende i en personaliseret form, hvor forvaltning afhænger af enkeltpersoner.

Agile metoder giver adrætte og spændstige processer der kan deles af alle forvaltere på tværs af produkterne der forvaltes. Og gøre det muligt at hjælpe hinanden når det strammer til i en periode.

2. Teknikker og metoder

Figur 1 viser de vigtigste metoder og teknikker i agil systemforvaltning.

1. Scrum
2. Testdrevet forvaltning
3. Extreme Programming
4. Automatiseret test
5. Agile processer
6. Teams
7. Issue Management

Figur 1: Teknikker og metoder

2.1 Scrum

Scrum bruges til styring af agil forvaltning. Scrum bruger tre roller, tre værktøjer og tre ceremonier. Alle ni dele bidrager konkret til styringen af forvaltningen. Scrum sprints bruges til iterationer og som byggesten i releases. Scrum bruges også til at sikre vidensdeling og formidle Best Practices. Vidensdeling reducerer virksomhedens tavse viden.

2.2 Testdrevet forvaltning

Når udviklere bruger testdrevet udvikling (TDD), giver det et holdbart mikrodesign af det nye system. Når brugere anvender Accepttestdrevet udvikling (ATDD).specificerer de accepttesten sammen med kravene til ændring.

Exploratory Testing er nyttig til ATDD.

Når forvaltere bruger TDD får de tilsvarende et holdbart mikrodesign af ændringen. Og de får specificeret accepttesten når de selv bruger ATDD. Forvaltere kan også bede opgavestillerne om at levere testeksempler sammen med ændringsønsket.

I forbindelse med fejlrettelser spiller det testdrevne en særlig rolle. Når testen specificeres før rettelsen kan den bruges til at genskabe fejlen. Og leve videre som automatiseret test.

Men de tre andre forvaltningskategorier, tilpasninger, udbygninger og saneringer har også stor glæde af testdrevne principper.

2.3 Extreme Programming (XP)

Det bliver til Extreme Maintenance når XP bruges i forvaltningen. Figur 2 viser de 13 XP discipliners nytte i de fire forvaltningskategorier.

Forvaltning og Extreme Programming	R	T	U	S
Teamet (Whole Team)			Yellow	Blue
Planlægning (Planning Game)				Blue
Brugertest (Customer Tests)			Yellow	Green
Iterativ udvikling (Small Releases)				Blue
Enkelthed i design (Simple Design)	Red		Yellow	Green
Parprogrammering (Pair Programming)	Red		Yellow	Green
Testdrevet udvikling (Test-Driven Development)	Red		Yellow	Green
Trinvis forfining (Design Improvement /Refactoring)	Red		Yellow	Green
Løbende integration (Continuous Integration)	Red			
Fælles eje af koden (Collective Code Ownership)	Red		Yellow	Green
Kodestandard (Coding Standard)	Red		Yellow	Green
Metafor (Metaphor)				
Bæredygtigt arbejdspress (Sustainable Pace)	Red			

Figur 2: Forvaltningskategorierne og XP's 13 discipliner

R = Rettelser
T = Tilpasninger
U = Udbygninger
S = Sanering (Refactoring)

De farvede felter peger på de dele af XP der tilbyder ekstra store fordele for den pågældende forvaltningskategori. Der er ingen betydning af de fire forskellige farver. De er blot forskellige for adskillelsens skyld. De ufarvede felter for en forvaltningskategori kan også give fordele. De farvede er blot særlig vigtige for kategorien.

Brugertest er selvfølgelig også en fordel for kategorien rettelser. Men den primære testcase er den der genskaber fejlen. Viser at den er der. Det er en anden test end testdrevet

videreudvikling, der har til formål at give testeksempler på forventninger til videreudvikling og tilpasninger.

Exploratory Testing er fremragende til at få detaljerne frem i forventningerne. Det er normalt en teknik der bruges til afsluttende tests, men so kan bruges til AcceptanceTestDrivenDesign (ATDD).

Bæredygtigt arbejdspress er også vigtigt i alle fire kategorier. Men det får lov at stå frem i særlig grad for rettelser. For at illuminere de risici der er forbundet med rettelser sent om sent natten, eller under andre pressede forhold.

Der bør foretages en bæredygtig gentest af hasterrettelser, og tilsvarende regressionstest af det system hasterrettelsen er foretaget i, så snart det kan lade sig gøre, efter den natlige udskejlse.

XP bruges som værktøjskasse til gennemførelsen af agil forvaltning. Til den konkrete analyse, design, test og ændring. Ikke alle XP-Practices er nødvendige i alle iterationer. Forvalterne kan situationsbestemt vælge de nyttige.

Når der er brug for at forbedre den eksisterende kodebase er XP også det første sted at hente processer til gennemførelse. Altså når forvaltningens formål er at forbedre den generelle vedligeholdelsesvenlighed og udbygningsegnethed,

Refactoring (Design Improvement i figur 2), er afgørende for forvaltningens kvalitet. Og det er reelt to separate teknikker der ligger i punktet. Refactoring af koden er central for den daglige/faglige evolution af koden. Og Refactoring af testcases er den daglige/faglige evolution af forventningerne til koden.

2.4 Automatiseret test

Automatiseret test er nødvendig for at opnå den helt basale produktivitet og kvalitet. Med manuel test bliver forvaltningen langsomelig, når testen skal gentages. Eller mangelfuld fordi der ikke bliver tid til at gentage den.

Det er den automatiserede regressionstest der giver den største nytte. Når man vil sikre sig at der ikke er opstået afledte fejl andre steder i systemet.

Automatiseret test bruges også til at reducere den eksisterende testgæld.

Automatiseret test er en forudsætning for større omstruktureringer af kode.

Automatiseret test kan være som unittest ved hjælp af xUnit. Eller som accepttest ved hjælp af Selenium, Watir eller lignende.

2.5 Agile processer

Der kan være brug for processer til agil systemforvaltning, ud over dem der er en del af Scrum og XP. Se mere om agile processer på www.agile-metoder.dk.

De overordnede kategorier af agile processer, er Styring, Planlægning, Test og Udvikling. For eksempel Agile Modeling (UML) til design, fra kategorien udvikling. Og Planning Poker fra kategorien planlægning.

Med Scrum, XP og et fuldt tilvalg af øvrige agile processer er procesmodenheden i vidt omfang adresseret i hvad der svarer til CMMI level 3.

2.6 Teams

Teams er væsentlige for produktiviteten og kvaliteten. Teknikker og metoder folder sig bedst ud når det gøres i et team.

Teams bruges også til at udfolde samarbejde. Det sker med midler som motivation, gejst, kommunikation, feedback, humor, vidensdeling og mod. Og med de andre værdier der er en naturlig følge af samarbejde mellem mennesker.

Ikke at produktivitet, kvalitet eller samarbejde behøver at mangle i en traditionel systemforvaltning. Det er bare en sikker del af teams der dyrker agil systemforvaltning.

2.7 Issue Management

Issue Management bruges til at mindske afstanden mellem interessenternes behov og systemernes formåen. For dermed at leve op til agil forvaltnings karakteristika om som minimum at være på omgangshøjde, og også gerne være forberedt på fremtiden. Issue Management bruges også til prioritering og Cost/Benefit beregninger.

3. Hasterettelser

Når en ændring gennemføres som en hasterettelse er det XP der leverer teknikker og metoder. Ikke mindst Pair Programming, men også det testdrevne.

Hasterettelser må samles op "dagen efter". Den eventuelle manglende test må udføres, og gøres til en del af den automatiserede regressionstest.

4. Best Practice

Et sæt af fælles metoder og teknikker gør det muligt at hjælpe hinanden. Det er der brug for når forvaltningen af et system prioriteres højt i en periode. Og behovet for ressourcer til forvaltningen af forskellige systemer derfor svinger over tid.

Agile metoder og teknikker er et udtryk for Best Practice. De er velegnede til videreudvikling og vedligeholdelse af såvel meget gamle som helt nyudviklede systemer.

Agile metoder gør det muligt at fokusere anvendelsen af sine forvaltningsressourcer måned for måned. Ved at bruge Sprints som iterationer.