Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

Bakgrund

Detta dokument syftar till att ge en introduktion till RUP och bemöta argument såväl för som emot processen.

För att kunna diskutera om man skall använda RUP eller inte måste man dock ta ett steg tillbaka och först bestämma sig för vad man anser att RUP är för något. På detta område finns det troligtvis lika många åsikter som det finns självutnämnda IT-profeter och det är tyvärr alltför sällan som ett företag som bestämt sig för att införa RUP tar sig tid att fundera vad man egentligen menar med ”att införa RUP”. Vilka förändringar vill man åstadkomma

Vad är RUP?

RUP presenteras ofta som ett smörgåsbord där projektmedlemmar kan välja och vraka mellan de olika verktyg som processen tillhandahåller för att öka gruppens effektivitet. Själva kärnan i RUP är dock i grund och botten ett strukturerat arbetssätt med vissa fokusområden. Dessa fokusområden är sällan några nyheter för erfarna projektdeltagare eftersom de identifierats genom studier av ett stort antal projekt där vissa varit framgångsrika och andra mindre framgångsrika. Utifrån dessa studier har ett stort antal vanligt förekommande problem identifierats. Det visade sig senare att mer eller mindre alla framgångsrika projekt hanterade dessa problem på ett likartat sätt och det är dessa problemlösningsmetoder som ligger till grund för RUPs fokusområden. Det är viktigt att komma ihåg att dessa metoder är empiriska, inte teoretiska. IBM kallar dem för programvarupraxis (eng. Best practises), ett bättre ord är beprövade erfarenheter. Metoderna är:

- Iterativ och inkrementell utveckling
- Visuell modellering
- Strukturerad kravhantering
- Medveten ändringshantering
- Kontinuerligt kvalitetsarbete
- Komponentbaserad arkitektur

En **iterativ och inkrementell** utvecklingsprocess delar upp utvecklingsprojektet i mindre tidsenheter (iterationer) för att under varje tidsenhet utföra ett visst arbete. Resultatet ifrån detta arbete benämns inkrement.

Under varje tidsenhet så genomlöpes i princip samma arbetsmoment gång på gång, även om fokus för projektgruppen kommer att variera med tiden. Detta innebär att i tidiga iterationer så kommer mer kravarbete att utföras, medan i senare iterationer kommer det att vara relativt sett

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

mer implementationsarbete. Det är dock viktigt att poängtera att hela utvecklingskedjan (krav, analys/design, implementation, test och driftsättning) förekommer i varje iteration. Enkelt uttryckt så kan man se ett iterativt projekt enligt RUP som en serie av små vattenfallsprojekt, där varje iteration motsvarar ett vattenfallsprojekt. Målet inom en tidsenhet är hela tiden att producera någonting körbart, även om detta inte alltid uppnås tidigt under projektet. Syftet med att dela upp projektet i mindre tidsenheter att bättre kunna hantera de risker som projektet ställ inför. Detta sker bland annat genom att iterationer:

- underlättar planeringsarbetet eftersom komplexiteten blir mindre ju kortare tidshorisont man arbetar med
- ger mätbara resultat så tidigt som möjligt under projektet
- fokuserar på eliminering av stora risker
- gör att projektledaren kan anpassa bemanningen av projektet efter den förståelse som uppnåtts för uppgiften som skall lösas
- möjliggör produktionssättning av åtminstone någonting även i de fall projektet läggs ned innan slutleveransen
- skapar en flexibel projektform där möjligheterna till situationsanpassning är stora
- förbättrar förmågan att tidsuppskatta arbetet

Tanken bakom **visuell modellering** istället för text är att projektet inte skall bli alltför dokumentationstungt. Modellerna skapas som ett sätt att beskriva och diskutera problem och lösningar, men det fina i kråksången är att dessa senare också kan användas som system-dokumentation. Tillräckligt mycket modeller ska skapas under utvecklingsarbetet för att ett tillräckligt bra system skall kunna konstrueras. Det innebär sällan att systemet är fullständigt dokumenterat så ett visst efterarbete kan vara nödvändigt för att tillfredställa krav från förvaltning, användare, FDA m.fl. Förhoppningen är dock att det inte skall vara fullt så mycket som det hade varit utan modellerna…

UML är ett modelleringsspråk som används i RUP. Det används för att specificera, visualisera och dokumentera olika typer av system. Det kan alltså användas för såväl systemutveckling som för verksamhetsutveckling. Som alla språk så har UML en tydligt definierad notation (beteckningar), syntax (regler för hur två saker kopplas ihop med varandra) och semantik (betydelsen av någonting). Språket ägs och förvaltas av Object Management Group (OMG) som är en branschorganisation. Det finns dock flera andra modellspråk som projektet mycket väl kan välja att använda sig av, om det känns mer komfortabelt.

Kraven utgör det kontrakt som systemleverantören skall uppfylla och systemkraven specificerar de egenskaper som systemet skall uppfylla. En **strukturerad kravhantering** går ut på att på ett systematiskt sätt identifiera, organisera, beskriva och komma överens om systemkraven och sedan även vidmakthålla denna överenskommelse. För de projekt som behöver stöd i detta arbete tillhandahåller RUP en process för som är centrerad kring användningsfall. Användningsfall är en teknik för att specificera funktionella systemkrav som ursprungligen beskrevs av Ivar Jacobson och användes vid utvecklingen av Ericssons Axe-växel. Användningsfall används genom hela processen för kravhantering och senare även som utgångspunkt för analys, design och test samt för projektplanering. RUP arbetar med successiv

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

detaljering, d.v.s. kraven detaljeras inte fullt ut med en gång, utan först arbetar projektet på bred front med att identifera och göra en grov beskrivning av alla kraven, sedan prioriteras kraven. Först därefter detaljeras kraven ett och ett.

Projektgruppen måste vara medveten om, och acceptera, att det inte går att undvika ändringar på redan utfört arbete. Det är dock viktigt att dessa ändringar införs på ett säkert sätt (hur och när) samt att alla får tillgång till de ändrade artefakterna och att det går att spåra ändringarna till sin källa. En **medveten ändringshantering** säkerställer detta.

Det går inte att testa in kvalitet i en produkt utan kvalitet är resultatet av ett gemensamt och **kontinuerligt kvalitetsarbete** som tar sin början i användarnas delaktighet i kravspecificerandet och går via avstämningar och diskussioner kring användargränssnittsprototyper till genomförandet av väl definierade tester med genomtänkt testdata. En iterativ process tillhandahåller dessutom många tillfällen till diskussioner och granskningar.

System som skall leva över tiden tenderar att också förändras i takt med omvärlden. För att konstruera system som är lätta att förändra och vidareutveckla förespråkar RUP en **komponent-baserad arkitektur**. D.s.v. ett system som består att ett antal sinsemellan oberoende komponeter så att man kan byta ut en komponents innehåll (så länge man lämnar gränssnittet oförändrat) utan att behöva ändra någon annanstans i systemet. Detta underlättar också förändringsarbetet under projektet…

Vart och ett av ovanstående områden är bra för sig, men de förstärker också varandra på så sätt att visuell modellering underlättar iterativ utveckling o.s.v. Ett plus ett blir med andra ord tre…

Utifrån ovanstående kan RUP definieras som ett strukturerat sätt att arbeta som tar tillvara på beprävade erfarenheter från andra projekt.

Fördelarna med att använda RUP

En sammanfattning av ovanstående ger följande fördelar med RUP:

- underlättar planeringsarbetet eftersom komplexiteten blir mindre ju kortare tidshorisont man arbetar med
- ger mätbara resultat så tidigt som möjligt under projektet
- fokuserar på eliminering av stora risker
- gör att projektledaren kan anpassa bemanningen av projektet efter den förståelse som uppnåtts för uppgiften som skall lösas
- möjliggör produktionssättning av åtminstone någonting även i de fall projektet läggs ned innan slutleveransen
- skapar en flexibel projektform där möjligheterna till situationsanpassning är stora
- förbättrar förmågan att tidsuppskatta arbetet
- mindre dokumentationstunga projekt
- gemensamt språk

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

- mindre missförstånd
- kontrollerad och medveten ändringshantering
- många tillfällen till kvalitetskontroll
- många personer engagerade i kvalitetsarbetet
- genomarbetade lösningar

Fem vanliga missuppfattningar om vad RUP är?

1. En vanlig missuppfattning är att RUP är en uppsättning namn på roller och dokument som skall användas i ett projekt. RUP tillhandahåller en uppsättning namn på roller och dokument som projektet/organisationen kan välja att använda sig av om det underlättar arbetet. Ett system för att identifiera dokument och roller är alltid bra att ha och i de fall det redan finns etablerade beteckningar i projektet på de nödvändiga dokumenten och rollerna kan det vara bra att använda dessa. Saknas beteckningar behöver man inte fundera så mycket på vad man skall kalla saker och ting utan kan istället använda sig de som RUP föreslår.

2. En annan vanlig missuppfattning är att RUP är ett oplanerat arbetssätt, att man vid iterativ utveckling inte behöver planera sitt arbete. RUP innebär inte på något sätt oplanerade projekt. Däremot så planerar man på en ganska övergripande nivå lite längre fram i tiden, och på en mer detaljerad nivå för den närmaste framtiden. Precis hur detaljerade planer man görs styrs bl.a. av projektets storlek (ju fler personer som är inblandade och därmed också beroende av varandras arbete desto mer detaljerade planer tenderar vara nödvändiga), av projektdeltagarnas erfarenhet (ju mindre erfarenhet en person har desto mer tenderar en tydlig plan att vara ett stöd) och av projektgruppens distribution (kan inte deltagarna träffas så ofta så tenderar arbetsplanen att vara ett viktigt verktyg för att synkronisera arbetet). Hur pass plan- och regelstyrt samt dokumentationstungt arbetet i projektet blir styrs alltså inte i sig av RUP utan av projektets behov.

RUP kan aldrig implementeras som ett vattenfallsprojekt men mer eller mindre agilt (lättrörligt). Agil systemutveckling är ena ändan av det iterativa angreppssättets spektra. Den andra änden är planstyrd systemutveckling.

Agil systemutvecling	⟷	**Planstyrd systemutveckling**
Låg risk		Hög risk
Seniora utvecklare		Juniora utvecklare
Förändringsbenägna krav		Stabila krav
Liten projektgrupp		Stor projektgrupp
Decentraliserad organisationskultur		Toppstyrd organisationskultur

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

3. En tredje vanlig missuppfattning är att man i ett RUP-projekt börjar programmera omedelbart och utan eftertanke (a.k.a. happy hacking). RUP betonar vikten av att man tidigt kommer fram till testbar (exekverbar) kod som kan demonstreras för kravställarna. Programmeringsarbetet föregås dock alltid av en övergripande diskussion kring vilka krav som finns. Kraven prioriteras utifrån verksamhetsnytta och teknisk komplexitet. Först när denna prioritering är klar så kan design och programmering påbörjas. Hur pass formellt ovanstående görs styrs av projektets förutsättningar.

4. En fjärde vanlig missuppfattning är att man aldrig behöver frysa kraven i ett RUP-projekt utan kan ändra dessa när som helst och hur som helst. RUP är framtaget för professionellt arbete och betonar vikten av en professionell attityd till uppgiften. Det innebär att det visserligen är tillåtet att ändra kraven, men detta kommer att få konsekvenser i termer av tid och pengar. Dessa konsekvenser måste identifieras och förklaras för kravställarna innan beslut kan fattas. Det är alltså inte tillåtet att ändra kraven **hur** som helst, utan enbart i enlighet med en fastställd ändringshanteringsrutin. Hur pass formell denna rutin är brukar variera med hur pass nära produktion projektet är; ju närmare produktion desto mer formell ändringshantering. För att skapa en arbetsro för projektet så är det heller inte, i normalfallet, tillåtet att ändra de krav som är under implementation i den aktuella iterationen. Kraven fryses därmed iteration för iteration och det är alltså inte tillåtet att ändra kraven **när** som helst heller.

5. En femte vanlig missuppfattning är att RUP är ett färdigt recept för alla projekt att följa. I själva verket är det precis tvärtom. RUP är beskrivet som en generell process där varje projekt måste ta ställning till de olika delarna utifrån sina egna förutsättningar, ungefär som en kokbok som enbart beskriver olika kryddor och råvaror. Kocken måste sedan själv komponera sin egen måltid.

Vad är skillnaden mellan RUP och andra processer?

Med ovanstående resonemang; inte mycket! Alla metoder har det gemensamma syftet att öka projektets effektivetet och underlätta ett bra resultat samt förespråkar iterativ systemutveckling. Metoder/filosofier som SCRUM, DSDM (Dynamic System Development Method), XP (eXtreme Programming), Crystal, ASD (Adaptive Software Development), FDD (Feature Driven Development), spiralmodellen, RAD (Rapid Application Development) och Lean Software Development är beskrivna av andra personer och för andra typer av projekt men det grundläggande angreppssättet och principerna är desamma.

Några saker som gör att beskrivningen av RUP avviker från beskrivningen av ovanstående alternativa processer är dels att RUP täcker in väldigt stora delar av väldigt stora projekt, dels att RUP är generellt beskriven och utgår ifrån det största tänkbara och mest komplexa projektet.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

Hur veta om RUP är rätt lösning?

En duktig läkare ordinerar ingen medicin utan att veta vilken sjukdom patienten har och operarar aldrig en frisk person. Börja därför med att definiera problemet/problemen som finns i organisationen och fundera sedan på om RUP underlättar lösandet av dessa.

Nedan är några vanliga problem som en bra implementation av RUP hjälper projektgruppen att fokusera på. I slutändan är det dock alltid projektgruppen själv som måste lösa problemen…

Tio vanliga problem som RUP kan hjälpa till att lösa

1. System som är dåligt dokumenterade – Bristfällig dokumentation uppstår ofta antingen med anledning av otydliga krav/kravställare på dokumentation eller på grund av att tid för dokumentationsarbete saknas i projektet. RUP adresserar detta problem bl.a. genom att tillhandahålla en metod för att identifiera krav och kravställare på dokumentationen och genom ett arbetssätt där dokumentationen skapas i förväg, som ett sätt att beskriva/kommunicera problem och lösningar.

2. System som är dåligt designade – En dålig design tar sig ofta uttryck i låg skalbarhet och system med många interna beroenden. RUP adresserar denna risk bl.a. genom att beskriva hur man kan använda sig av ett modellspråk (UML) för att beskriva designen och därmed möjliggöra såväl rena kvalitetsgranskningar som en allmän diskussion kring designen. Det iterativa angreppssättet gör också att fokus ligger på en enkel lösning som fungerar. En enkel lösning bör ge en enkel design och en enkel design är ofta en bra design.

3. Driftstörningar i produktionsmiljö – Driftstörningar i produktion är ofta resultat av okända buggar. Att buggarna är okända beror i sin tur delvis på för lite testande. RUP adresserar detta problem genom sitt iterativa arbetssätt. Varje iteration avslutas med systemtest vilket innebär att de centrala delarna av systemet testas flera gånger under projektet (i varje iteration).

4. Missförstånd (pga språket) – RUP bygger på etablerade begrepp som hämtats från branschen och som idag lärs ut på högskolor och universitet över hela världen samt är vanligt förekommande på Internet och i facklitteratur. Missförstånd kan alltid uppstå när någon hör ett nytt begrepp för första gången, men i längden bör RUPs begreppsapparat minimera antalet missförstånd.

5. Saker och ting faller mellan stolarna – En orsak till att saker och ting faller mellan stolarna och inte blir gjorda i ett projekt är en otydlig ansvarsfördelning. RUP Adresserar detta problem med etablerade och väl beskrivna roller. Alla dokument som tas fram och alla aktiviteter som utförs enligt RUP har en roll kopplad till sig. RUPs iterativa angreppssätt där man har tydliga acceptanskriterier för varje iteration gör också att ogjorda saker upptäcks tidigt i projektet.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

6. Dyra projekt – Genom sitt iterativa angreppssätt där man tidigt verifierar att systemet är möjligt att bygga möjliggör RUP att man lägger ned "omöjliga" projekt innan de har kostat allt för mycket pengar. De extrakostnader som ofta förknippas med RUP i form av utbildning, testautomatiseringsarbete och beställarmedverkan finns ofta med i traditionella projekt också även om de där inte är planerade och därför inte ingår i den ursprungliga budgeten. Externt stöd i form av konsulter och mentorer bör betraktas som en försäkringskostnad och är i sig ingen förutsättning för ett RUP-projekt utan ett resultat av att man saknar viss, nödvändig, kompetens i företaget. Verktygskostnaden bör tjänas in av projektet. Används RUP dessutom i kombination med objektorienterad design så ökar detta möjligheterna till återanvändning av tidigare arbete/kod (bl.a. genom inkapsling och polymorfi). RUP kan inte garantera att systemen blir billiga, men rätt använt så är processen ett stöd för att projekten inte ska bli dyrare än nödvändigt…

7. Dåliga beslutsunderlag – Traditionellt så bygger många beslut på subjektiva värderingar av hur långt projekten har kommit, och hur långt man har kvar samt hur mycket tid/pengar projektet har gjort av med. Kravspecen är klar, designen är klar, koden är klar etc. I ett RUP-projekt strävar men efter objektiva beslutsunderlag dvs testrapporter. Ett användningsfall är testat och godkänt, två användningsfall, tre användningsfall etc. Nackdelen med det förstnämnda är att beslutsfattarna kan aldrig vara säkra på att kravspecifikationen verkligen är kvar (det kan tillkomma något krav, det kan finnas missförstådda krav), designen kanske är felaktig/bristfällig etc. Med det sistnämnda angreppssättet så minimerar man den typen av problem.

8. Dålig kontakt med beställaren – RUPs iterativa angreppssätt där beställaren involveras tidigt och får frekventa demonstrationer av arbetsresultatet skapar en arbetsmiljö där alla inblandade lär känna varandra och drar år samma håll.

9. System som är svåra att använda – System som är svåra att använda uppstår ofta när ingen användare haft något att säga till om vid utvecklingen av systemet. RUP involverar kravställarna (som helst bör vara slutanvändare) igenom hela projektet vilket möjliggör användbarhetsstudier och användbarhetstester.

10. Hög förvaltningskostnad – System som är väl dokumenterade och har få buggar tenderar att ha en lägre förvaltningskostnad än system utan dessa egenskaper… Dessutom så ugår man i ett RUP-projekt utifrån verksamhetens behov och skapar system som baseras på verksamhetens grundläggande informationsstruktur. Denna struktur är ofta den som förändras minst i ett företag och system som baseras på denna blir därför relativt framtidssäkra vilket också det leder till en relativt låg förvaltningskostnad.

Sex dåliga argument mot att införa RUP och två bra

1. Vi hinner inte analysera arkitekturen på en iteration – Det är inte tanken… Arkitekturprototypen skall implementeras och testas innan etableringsfasen (fas två av fyra) kan avslutas. I kalendertid brukar detta motsvara ca halva projekttiden.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

2. RUP-projekt blir för dyra – Ett RUP-projekt skall inte kosta mer pengar än andra projekt. Då faller ju hela tanken med processen. Vad RUP däremot gör som ofta får beställare att tycka att det blir dyrt är att processen synliggör kostnaden för kravställarna och svårigheten med att sätta ett exakt pris. Ofta så döljs kostnaden för kravställarna på så sätt att de allokeras till x antal möten i början av projektet. Vad som sedan sker i verkligheten är att efter det att denna tid har gått så ringer projektgruppen och ställer frågor och begär förtydliganden och klargöranden. Kravställarna klämmer då in telefonmöten och lunchmöten etc. med tiden belastar sällan projektet. RUP ställer redan från början krav på att beställarna skall vara tillgängliga under hela projektet.

3. Vi har inte tid – I ett RUP-projekt bör ingenting ta så lång tid att man inte hinner med det… Nyckeln är att tänka tillräckligt bra. Först gör man en övergripande kravanalys som är tillräckligt detaljerad för att man skall kunna identifiera vilka krav som är tekniskt mest riskfyllda och vilka som ger mest verksamhetsnytta. Dessa krav detaljeras sedan ytterligare och därefter går man vidare med tillräckligt mycket analys, design för att man skall kunna implementatera och sedan testa funktionaliteten. Alla krav behöver alltså inte detaljeras innan man väljer ut vilka man skall gå vidare med. Inte heller så behöver man producera alla 13 typerna av uml-diagram för varje tänktbart användningsfallsscenario. Tillräckligt mycket analys och design dokumentation skall göras inte mer, och inte mindre.

4. RUP är inte rätt för oss (passar inte för våra projekt) – Gå tillbaka till gå och läs om från början igen…

5. Det blir för mycket dokumentation – Rätt använt så medför ett RUP-projekt mindre dokumentation eftersom det mesta tas fram som ett sätt att beskriva och lösa problemen. Ställ frågan vilka som behöver vilken dokumentation och ta enbart fram denna. Användare? Förvaltare? Beställare? Testare? Utvecklare? Projektledare? Styrgrupp? Säkerhetsgruppen? Kvalitetsgruppen? Arkitekturgruppen? Metodgruppen? Kravställare? Externa organisationer (t,ex, amerikanska FDA)? Personer som anser att man producerar för mycket dokumentation tänker ofta inte på hur många andra personer det är som faktiskt är intresserade av en eller annan aspekt av systemet.

6. Vi utvecklar inte objektorienterat – RUP har inget med objektorienterad utveckling att göra. Principerna som nämns i inledningen är lika giltiga för såväl objektorienterade som proceduriella system.

7. Det fungerar som det är idag – Ett bra argument!!! Det finns ingen orsak att ändra på ett vinnande recept.

8. Hela metodfrågan är för politisk just nu – Ett bra argument men metodfrågor är alltid politiskt känsliga! Det är därför inget argument mot RUP, utan snarare ett argument för att man skall göra ett genomtänkt införandeprojekt.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

Tio vanliga misstag i ett RUP-projekt

RUP löser inga problem i sig utan är en hjälp till självhjälp som terapeuterna säger… Nedan följer 10 vanliga misstag som ofta görs de första gångerna man försöker sig på att arbeta enligt RUP. Personer som hänvisar till specifika projekt som har använt RUP men ändå inte lyckats brukar ta det som ett tecken på att RUP inte fungerar. Om något av nedanstående är sant för dessa projekt så är det i dessa fall mer korrekt att säga att RUP inte har använts på ett bra sätt.

1. Att göra för mycket i första iterationerna – Att jobba iterativt innebär inte att bygga hela systemet i första iterationen… Den första iterationen är ofta svårast eftersom projektets syfte och krav fortfarande är otydliga, eftersom projektet kanske använder utvecklingsverktyg som är nya för projektdeltagarna och eftersom projektdeltagarna ännu inte lärt känna varande samt kanske beroende på att arbetssättet är nytt. Använd den första iterationen till att få igång projektgruppen; låta dem vänja sig vid utvecklingsmiljön och lära känna varandra. Låt den största arbetsinsatsen fokusera på kraven och deras prioritering samt eventuella gränssnittsprototyper. Alla tekniska risker behöver inte vara lösta efter den första iterationen!

2. Att inte producera någon kod – Att tidigt producera, testa och demonstrera exekverbar kod är RUPs hjärta och själva grundbulten i iterativ utveckling. Det möjliggör validering av system-kraven, verifiering av användbarhet och en objektiv utvärdering av projektets status samt skapar ett engagemang hos kravställarna.

3. För långa/korta iterationer – En iteration skall resultera i en meningsfull del av systemet. Korta iterationer ställer stora krav på projektgruppens erfarenhet, vana och omgivning. Vid för långa iterationer försvinner fokuset på ”tillräckligt bra” och det blir en serie vattenfallsprojekt av det hela. Ju längre iterationer desto svårare blir det ofta att lära av misstagen som gjorts.

4. Överlappande iterationer – Iterationerna används för att synkronisera projektets olika delar, till att bygga ihop systemet. Schemaläggs iterationerna så att de överlappar missar man denna funktion.

5. Arkitekt och projektledare är samma person – Arkitekten är oftast den person som är bäst insatt i hur lång tid utvecklingen kommer att ta och vilka tekniska risker som finns. Det gör att det ofta känns som ett naturligt val när man skall tillsätta projektledarrollen i ett mindre projekt. Det är dock inte alltid så lyckat eftersom arkitekten ofta behöver fokusera på utvecklingsarbetet, medans projektledarens huvudfokus ofta ligger på projektets externa gränssnittet (gentemot beställare och styrgrupp). Arkitektens och projektledarens intressen kolliderar också ibland i det att arkitekten ansvarar för systemets kvalitet medans projektledaren ansvarar för tid och budget.

6. Fokuserar inte på risker – Kent Beck har en gång sagt att all utvecklingsmetodik bygger på rädsla. Med det menas att allt vi gör i projektet gör vi för att vi tror att om vi inte gör det så kommer vi att få problem. Kostnaden för dessa problem vägs mot kostnaden för att undvika dem. Vid iterativ utveckling är tanken att man snabbt skall identifiera de stora riskerna och börja med att lösa dem för om inte projektgruppen lyckas med det så är det ingen idé att fortsätta med projektet. Det är därför viktigt att inte projektdeltagarna sticker huvudet i sanden och blundar för

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

de problem som de identifierar. Det är en bra sak att hitta risker och problem för det är först när man känner till dem som de kan lösas.

7. Fokuserar på dokument inte artefakter – Artefakter och dokument är inte samma sak. Det förstnämnda är ett arbetsresultat och det sistnämnda är dokumentationen av resultatet i ett textdokument. Det som är viktigt är att man gör det arbete som en artefakt kräver och att det är dokumenterat. Om man sedan väljer att lägga allting i ett och samma jättedokument eller om man väljer att dokumentera varje resultat i ett eget dokument eller väljer en tredje indelning är inte relevant. Vad som är viktigt är att alla hittar det de behöver när de behöver det.

8. Detaljplanering av alla iterationer i förväg – I ett RUP-projekt arbetar ofta projektledaren med två planeringshorisonter parallellt, en övergripande för hela projektet och en detaljerad för den närmaste iterationen. Det är viktigt att inte projektledaren lägger ned för mycket tid och möda på att detaljplanera iterationer som ligger längre fram i tiden eftersom dessa planer troligen kommer att behöva ändras och då är det bra om det inte ligger för mycket prestige och arbete i dem. En grov plan för iterationer längre fram i tiden skall dock projektledaren skaffa sig.

9. Inget mål definieras – Ibland så tas iterativ utveckling som en ursäkt för att man inte behöver fundera på vad som ska göras så att man istället kan börja och så kommer det nog att visa sig vad man vill. Det är väldigt sällan detta fungerar! Under de första iterationerna arbetar man med systemvisionen där man försöker beskriva vilken typ av system man vill ha, vilket problem det skall lösa, hur målgruppen ser ut etc. Visionen är viktig för att man senare skall kunna prioritera på ett konsekvent sätt.

10 Tror att man skall lyckas i första försöket… - De flesta av oss är rädda för att misslyckas och hoppas att vi genom att läsa denna typ av korta guider och andra längre böcker skall kunna göra rätt i första försöket. Det fungerar tyvärr inte så. Såväl varje organisation som varje projekt är unikt och därför gör vi ofta fel i första försöket. Tanken är att man använder sig av iterationsutvärderingarna för att ta reda på vad man gjorde fel denna gång. Kommer man på det firar man med tårta och bestämmer sig för hur man kan förändra processen till nästa iteration.

Vanliga frågor angående RUP

Avslutningsvis så finns det ytterligare några frågor angående RUP som förtjänar svar.

Hur vet jag om vår RUP-implementation är rätt/bra?

Nedanstående sex tecken tyder på en väl fungerande RUP-implementation.

1. Inget dokument skapas utan att det sedan kommer till nytta.

2. Mängden nya buggar som identifieras vid test minskar iteration för iteration.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

3. Arkitekturprototypen är exekverbar.

4. Mängden omarbete minskar iteration för iteration.

5. Kravställarna känner sig delaktiga.

6. Testarbetet börjar redan i den första eller andra iterationen.

Hur vet jag vilka artefakter, roller etc. vi skall använda?

För varje artefakt/roll ställer ni nedanstående två frågor:

- Vilket problem löser jag med denna?
- Vem skall använda den och till vad?

Löser inte artefakten/rollen något problem och/eller ingen tänker använda sig av den så behövs den inte. Upptäcker ni sedan att ni trots allt får vissa problem i projektet är det alltid bra att gå igenom sin anpassning och titta på vad man tagit bort. Kanske var det något som trots allt var bra att ha…

Hur vet jag hur jag ska anpassa RUP?

Ställ följande frågor:

- Vilka problem har vi idag?
- Vad finns det i RUP som kan hjälpa oss att lösa våra problem?
- Vilken organisationskultur har vi?
- Vilken typ av projekt bedriver vi?

Organisationskulturen är viktig att känna till när man skall anpassa RUP eftersom det är den som styr hur pass formell RUP bör vara samt hur pass finfördelade rollerna och artefakterna bör vara.

Projekttypen (storlek, risk, komplexitet, tidspress, volatilitet etc.) är viktigt att känna till eftersom det är denna som styr vilka roller och artefakter som är nödvändiga.

Systemvaruhuset AB Solna strandväg 78 171 54 Solna
Tel 08 50 52 10 42 Fax 08 50 52 10 10 Org. Nr 556665-7614 E-post info@systemvaruhuset.se www.systemvaruhuset.se

Hur vet jag om jag bör eller inte bör använda RUP

De allra flesta projekt dra fördel av att arbeta i enlighet med de principer som beskrivs i början men om man sedan väljer att gå ytterligare ett steg och faktiskt tala om att man arbetar enligt RUP och ersätter de egna rollnamnen och dokumentnamnen med RUPs är till största delen ett politiskt beslut.

Nedanstående tre frågor är bra att ställa sig inför detta beslut:

- Vilken attityd har projektdeltagarna gentemot RUP visavi andra metoder?
- Hur mycket erfarenhet har projektdeltagarna av RUP (alt. andra metoder)?
- Hur mycket kunskap har projektdeltagarna av RUP (alt. andra metoder)?