



Projektlednings metodik



Föreläsning 4

Dagens Upplägg

Stand up meeting, vi delar med oss

Att arbeta med Scrum; Retro & SPIKE

Skapa tydlighet i Agila projekt; Logiskt nätverk, Är våra mål SMARTA?

Stand up meeting, vi delar med oss



Retro

En del i strävan att alltid förbättra



Skapar
Engagemang, reflektion & perspektiv

För
Teamet



Retro en del i strävan att alltid förbättra

Varför? Syfte

- *Agilt tänk* - en strävan att alltid förbättra
- Kunskap om vad som *gått bra & vad vi vill förbättra*
- *Förstärka* och belysa de delar som gått bra
- Få med allas röst. *Alla är viktiga*
- Inga ryggsäckar, *vi går framåt*
- En strävan att teamet ska arbeta för att *bli ännu bättre*



Retro en del i strävan att alltid förbättra

Engagemang kommer;

- När vi driver vår egen process - och äger våra lösningar
- När frågorna är relevanta för vår grupp
- När vi ser att vi kan påverka



Förändringen blir *mer bestående*

Större chans att vi tillsammans genomför det, när vi själva har tagit fram lösningen



Sannolikheten att förbättringar *faktiskt genomförs & accepteras* ökar när vi gemensamt reflekterar.

Verksamheten/Företag utvecklas & rör sig framåt,
vilket gör dem konkurrenskraftiga

Retro en del i strävan att alltid förbättra

Inte ett diskussionsforum

Man pekar aldrig ut någon personligen eller klagar. Allt är konstruktivt & med en vilja att förbättra.

3 punkter:

- Vad har fungerat bra
- Vad kan förbättras
- Förslag på vad vi skulle kunna göra bättre framåt - *Action Points*



För att inte påverka varandra visar vi inte vad vi skrivit förrän alla är klara

1. Alla teammedlemmar skriver ned sina tankar (var och en för sig) på sina lappar; Kort, en sak/ per lapp. Ingen limit på vad man tar upp eller hur många saker som tas upp
 - Både allt som fungerat bra & det man vill förändra
2. I turordning visar man sina lappar/sätter upp dem på väggen. Med en kort förklaring vad man menar med sin lapp. Du har tolkningsrätt på din lapp, ingen kan/får ifrågasätta vad du upplever)
3. Teammedlemmarna lyssnar laget runt

Retro en del i strävan att alltid förbättra

Exempel på hur whiteboarden etc kan se ut

Bra saker -att fortsätta med



Vill förbättra



Tips! Bra att ha olika färger på sina lappar eller text. Så att vi vet vems lapp det är

Retro en del i strävan att alltid förbättra

Action Points - vi har ett resultat

När alla har fått komma till tals

1. **Sorterar vi/kategorisera** vi våra lappar.
 - Finns det lika åsikter? Står olika ord för liknande saker? = Vi sätter ihop dem
2. Vi namnsätter våra kategoriseringar
3. Rösta (2 st röster per person) på en eller 2 kategorier. Vilken eller vilka kategorier man anser är viktigast att ta tag i först.



Retro en del i strävan att alltid förbättra

Kommunikation



5 st röster

Möten/struktur



2 st röster

Överlämning av arbeten



1 röst

- Nu kan vi *gemensamt brainstorma* fram olika förslag för förbättringar. Sk “*action-points*” (åtgärdspunkter) Max 3 st
- Dessa Action Points läggs in i Jira (Retro går att göra i Jira då är de där från början)
- Nästa gång tittar teamet igenom sina action points och utvärderar dem hur det gått.
- Tanken är action pointsen ska bli en del av de *nya rutinerna*



Överblick

1. Har man haft ett retrospekt-möte tidigare så börjar man först med att snabbt gå igenom *förra mötets beslut/Action-points* och diskutera om det har hjälpt eller inte.
2. **Ta er tid.** Var och en skriver ned sina tankar. Både bra och det som man vill förändra (utan att visa, utan att diskutera)
3. Gå laget runt. **Låt var och en berätta** vad just den personen menar mitt sitt/sina ord/beskrivningar (här sker ingen diskussion, bara förklaringar)
4. Innan ni går vidare. **Läs noga igenom allt bra ni som team gör.** Ta till er & kom ihåg. Det här ska ni fortsätta med
5. **Merga** ihop/kategorisera (förbättringsdelen främst)
6. **Rösta**, vad tycker teamet är viktigt
7. **Action Points/Åtgärdspunkter** skapas (max 3 st)

Övrigt

Vid långa projekt kan man även ha olika teman i retron



Retro en del i strävan att alltid förbättra

Varför tar vi inte upp allt som kommer upp?

- Att förändra beteenden och vanor *tar tid och energi*. Vi har helt enkelt inte kapacitet att ta för många förändringar på samma gång
- Det betyder inte att inte *allt som tas upp är viktigt*
- Målet är att era *action-points ska bli till nya rutiner*. För att ersättas med nya action points.



För att kommer överens



- Det är först när alla har fått komma till tals man kan *gemensamt fatta beslut* om eventuella förbättringar som kan eller bör genomföras. Sk “action-points” (åtgärdspunkter)
- Har man haft ett retrospekt-möte tidigare så börjar man först med att snabbt gå igenom förra mötets beslut och diskutera om det *har hjälpt eller inte*.
- Det här är ett sätt att hela tiden *få bort orosmoment eller delar som stör* arbetet



Varför Retro & inte tex brainstorming direkt?

Ett sätt att få fram *fler och mer nyanserade saker*. Att vara en individ i gruppen och inte ett kollektiv

När vi startar med egen reflektion

- Kommer det oftare upp mer nyanserat vad som är viktigt för individen
- Att allas röst kommer fram. Även de som är tysta, mer reflekterande av naturen, blyga mm
- Vi mår bra av att få tid att reflektera & titta tillbaka



Brainstormar vi för tidigt risker vi

- Att få fram en kollektiv syn & inte en individuell syn på de delar som tas upp.
- Att vissa delar inte tas upp, missas, vågas tas upp eller inte kommer på andra delar som man själv tänker men inte gruppen tänkt på



Grupparbete/Gruppdiskussion



Retrospektiv

- Gör en retro för er vecka som gått
- Ämne; Komma igång/arbeta i grupp
- Vad har gått bra? Vad vill jag förbättra? -> Action Points

SPIKE -Ett sätt att komma vidare

Främst för utvecklare - Men vi kan ta inspiration ifrån hur de löser låsningar - Ett tankesätt



För att komma vidare

När vi inte är överens om hur vi ska arbeta framåt

- Tex vilken teknik vi ska använda
- Inte vet vilken lösning som får projektet framåt
- Inte har erfarenhet/kunskap/ngt nytt
- En user story som av olika anledningar inte är tydlig

Alla i teamet kan göra detta, när som helst

Räck upp handen - Ropar "SPIKE"



SPIKE är

- En testperiod som är tidsbestämd påbörjas. Timeboxa.
- Perioden går ut på att testa olika varianter och gemensamt utvärdera dem.
- Målet är att definiera user storyn, estimera den & hitta lösningar framåt
- Erfarenheter/ lärdomar tas fram

Efter det tas ett beslut om vilken väg teamet tar.

För att komma vidare

Ibland är detta den snabbaste vägen framåt

- Att göra en ”mini” utredning gör att vi kan förstå bättre
- När vi förstår bättre kan
 - Vi gör bättre beslut
 - Förtydliga
 - Estimera
 - Vi testar oss framåt, för att se vilka vägar som fungerar
 - Göra färre missar på vägen



Ge SPIKE:en tid - Timeboxa

- Bestäm i förväg hur lång
- Kan vara allt från ett par dagar & framåt.
- Tänk på att det inte får ta för lång tid. Funkar det inte så funkar det inte. Då kanske vi behöver tänka om helt.
- Glöm inte att diskutera den i ditt team på retron, utvärdera ert arbete

För att komma vidare

Översatt från engelska

- En spik är en *produktutvecklingsmetod* som kommer från **Extreme Programming** som använder det enklaste möjliga programmet för att utforska potentiella lösningar.
- Den används för att bestämma hur mycket arbete som krävs för att lösa eller kringgå ett programvaruproblem.

Resultat

- När vi har fler möjligheter och fler lärdomar kan vi göra ett nytt beslut
- Ofta blir det en blandning av de olika testerna
- Vart vi ska
- Hur? Vilken teknik? Etc
- Om vi kan estimera storyn bättre
- Om vi kan förtydliga den utifrån lärdomarna

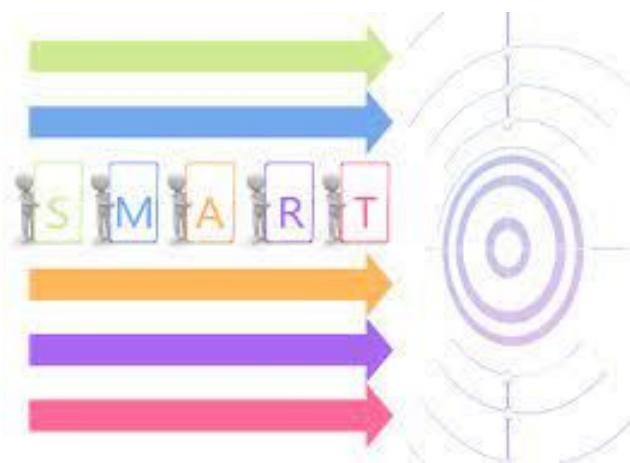


SPIKE - Ett sätt att komma vidare



Fler frågor & funderingar

Är våra mål, SMARTA?



När använder vi oss av SMART:a mål?

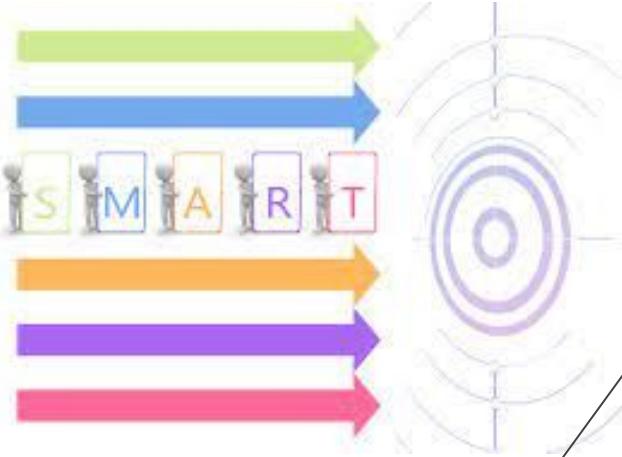
När?

- Planering av projekt
- Planering långsiktiga & kortsiktiga mål
- Planering av Epics
- Planering av user story
- Planering av en sprint
- Etc

Hur?

- När det behövs dokumenterar vi dem
- Vi skriver inte alltid ned våra mål eller dokumenterar (enbart vid behov)
- Men vi har dem i ”ryggraden” & tänker utifrån dem i många situationer
- Det är ett sätt att tänka & planera

Är våra mål, SMARTA? Kontrollera



- Ett realistiskt mål bör du – eller gruppen det gäller för – kunna nå på egen kraft.
- Tydligt när målen ska vara uppfyllda

- Specifikt
- Mätbart
- Accepterat
- Realistiskt
- Tidsbundet

- **Vad** vill vi uppnå,
- **Varför** är det här viktigt,
- **Vem** eller vilka är inblandade,
- **Var** görs det här,
- **Vilka** resurser och begränsningar gäller.
- **Hur** ska ni mäta? Under vilken tidsperiod eller när? Vad ska ingå? vem mäter?
- Resurserna som krävs finns.
- Alla förstår målet och ser det som realistiskt.
- Målet är kommunlicerat och accepterat av de som är inblandade

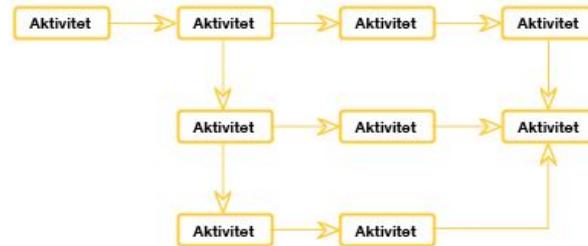
Grupparbete/Gruppdiskussion



SMART:a mål

- Vad har StädaFint för mål?
- Gör dem SMART:a

Logiskt nätverk



Logiskt nätverk

Vad är ett logiskt nätverk, nätplan etc

- Ett verktyg
- En visuell karta
 - Vad som ska göras (större block, mer åt milstolpe hållit, kan vara mer övergripande)
 - I vilken ordning, vad måste göras först
 - Vad kan göras parallellt
 - Beroenden
 - Tiden för ett projekt. Visar det logiska sambandet mellan kalendertid och aktiviteter

Tips

- Bryt ner projektet - få en överblick
- Då vet vi redan vilka delar som ingår



Logiskt nätverk

Vad är ett logiskt nätverk, nätplan etc

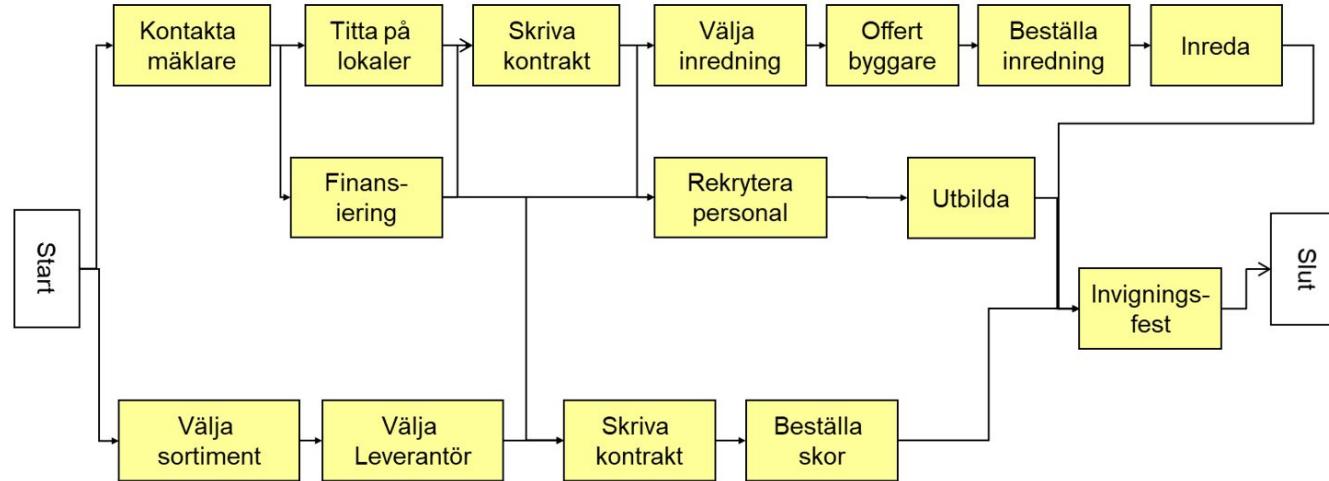
Att tänka på

- Agilt projekt - vi behöver hantera förändringar
- Nätverket/nätplanen behöver ha en struktur vi kan ändra/uppdatera
 - Tillräckligt detaljerad för att kunna få en förståelse för projektet
 - Se tiden över de olika delarna
 - Men inte gå för djupt på detaljnivå- Det finns det ingen mening för
 - Skapa en enkel mall/modell som du enkelt kan uppdatera vid behov



Nätplan (logiknät)

- Logisk ordningsföljd
- Beroenden mellan aktiviteter, behov av överlämningar



Logiskt nätverk

Projekt: Bygga en webbaserad applikation

Aktiviteter och beroenden

1. Planeringsfas

- A1: Kravinsamling (0–1 vecka)
- A2: Utveckling av projektplan (0–1 vecka)

2. Designfas

- A3: Skapa UI-design (1–3 vecka)
- A4: Skapa databasschema (1–2 vecka)

3. Utvecklingsfas

- A5: Utveckla frontend (3–6 vecka) (beroende av A3)
- A6: Utveckla backend (2–5 vecka) (beroende av A4)
- A7: Implementera API:er (2–5 vecka) (beroende av A4)

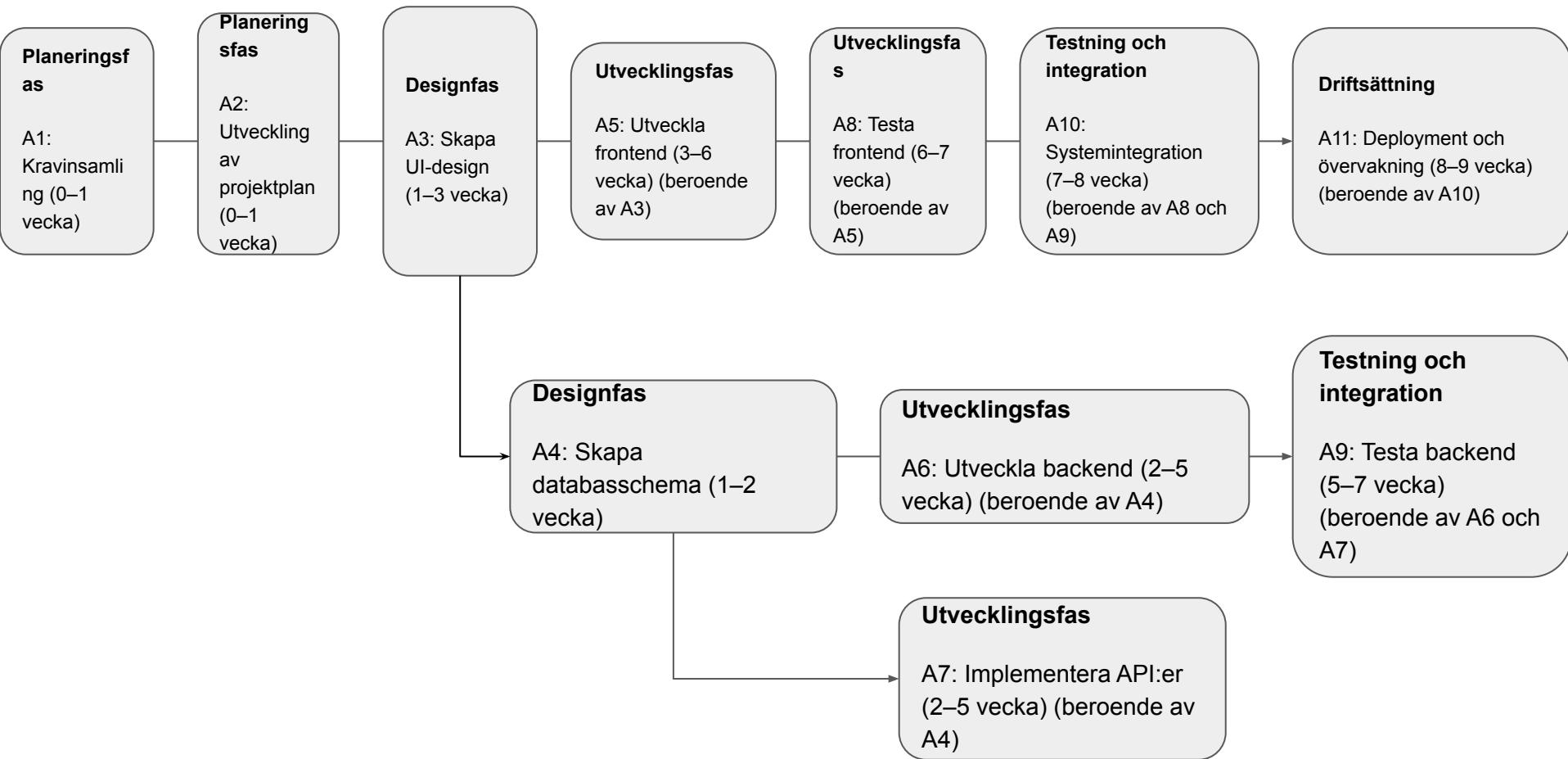
4. Testning och integration

- A8: Testa frontend (6–7 vecka) (beroende av A5)
- A9: Testa backend (5–7 vecka) (beroende av A6 och A7)
- A10: Systemintegration (7–8 vecka) (beroende av A8 och A9)

5. Driftsättning

- A11: Deployment och övervakning (8–9 vecka) (beroende av A10)





Till nästa gång vi ses

Nästa gång vi ses

- Arbeta vidare med ert Case
- Titta på de aktuella introduktionsfilmerna

Instuderingsfrågor

- Del 1 sid 7 & 10 (titta på planeringen för att se vad ni kan fokusera/prioritera)

Till nästa gång vi ses

Stand-up

- Scrum mastern har ordet/Stand-up meeting

Att arbeta med Scrum

- Retro

Att följa upp Agila Projekt

- Burn up and burn down charts

Vad tar jag med mig av dagens genomgång

