Agilt projektarbete – Martin Haagen

**Projektmodeller**

* Ge kontroll, struktur och tydliggöra ansvarsområden
* Var befinner sig projektet på sin livslinje
* Förutsägbarhet – Tid, budget m.m

**Agila metoder** – Kommer från verkstads industrin, det blev mycket mer effektivt

* Ofta för program-, system- och mjukvaruutveckling
* Ofta inkrementell och iterativa Flera faser samtidigt
* Snabbare resultat, testa och verifiera tidigare ger hög kvalité
* Hanterar förändringar och krav bra

Några av dom vanligaste agila metoderna

* Scrum
* sXtreme Programming – Par programmering
* Kanban
* Mf

**Lean**

* Många olika projekttyper
* Hela organisationen inblandad
* Minska slöseri med resurser

**Lean – Varianter**

* Lean
* Kaizen
* Six Sigma

**Project Management Professionals (PMP)**

* Project Management Professionals – Certifiering
* Stort i USA och världen ( Project Management Institute, PMI)
* Sekventiell – ett projekt passerar igenom olika faser

**Projects in Controlled Environements ( PRINCE2)**

* Projects IN Controlled Environments (Certifiering)
* Stort i Europa

**Hur väljer man?**

* Vilka begränsningar finns?
* Första gången?
* Vilka är intressenterna?
* Vilka risker finns?
* Hur komplext är projektet?
* Hur stort är projektet och vad kommer det att kosta?
* Företagskultur
* Vad man brukar använda

**Agilt – Lätt rörligt, föränderlig**

* Lean har rötterna i TPS – Toyota Production System
* Ford – löpande band
* Stora industrier
* Gott om material
* Gott om lågavlönad personal
* Svinn/Slagg/Avfall -> Återanvändning

**Vad gjorde Toyota annorlunda?**

* Ta tillvara på resurser, främja delaktigheten och insyn i arbetsprocessen
* Optimera processer: mindre material, mindre mänskliga resurser utan att minska mervärdet som skapas
* **Syfte** – vilket problem försöker jag lösa
* **Tillämpning** – hur säkerställs alla stega i processen är värdeskapande
* **(Mänskliga) resurser** – Hur optimeras varje process i enlighet med värderingar och resurser

**Lean Meat – without fat**

**Agila metoder – det finns några att välja mellan**

* Lean
* Scrum
* Kanban
* eXtreme Programming (XP)
* Scrumban
* SAFe mm

**Agile inom utveckling**

Vad utmärker en Agil arbets/projektledningsmetod?

Man sitter i sprintar, har möten om och om igen.

Det agila manifestet

**Fungerande** programvara

**Kundsamarbete** framför kontraktsförhandling

**Anpassning till förändring** framför att följa en plan

De 12 principerna

1. Vår högsta prioritet är att tillfredsställa kunden genom tidig och kontinuerlig leverans av värdefull programvara.
2. Välkomna förändrade krav, även sent under utvecklingen. Agila metoder utnyttjar förändring till kundens konkurrens
3. Leverera fungerande programvara ofta, med ett par veckors till par månaders mellanrum, ju oftare desto bättre.
4. Verksamhetskunniga och utvecklare måste arbeta tillsammans dagligen under hela projektet.
5. Bygg projekt kring motiverande individer. Ge dom miljö och det stöd de behöver. Och lita på att de får jobbet gjort.
6. Kommunikation ansikte mot ansikte är det bästa och effektivaste sättet att förmedla information, både till och utom
7. Fungerande programvara är främst måttet på framsteg
8. Agila metoder verkar för uthållighet. Sponsorer utvecklare och användare skall kunna hålla jämt utveckling under obegränsad tid
9. Kontinuerlig uppmärksamhet på förstklassig teknik och bra design stärker anpassningsförmågan
10. Enkelhet . konsten att maximera mängden arbete som inte gärs – är grundläggande
11. Bästa arkitektur, krav och design växer fram med självorganiserande team
12. Med jämna mellanrum reflekterar teamet över hur det kan bli mer effektivt och justerar sitt beteende därefter.

**Scrum**

* Vanligaste projektmodellen som används
* Få som gör Scrum gör faktiskt inte Scrum
* Cyklisk, självorganiserande och lärande
* Värdestyrd

**Kanban**

* Kontinuerlig process
* Begränsa arbete
* Mindre team eller service teams

**Vattenfall**

* En ”traditionell projektledningsmodell
* Sekventiell metod
* Olika faser att gå igenom
* Man går inte ”tillbaka i processen

Kravspecifikation

* Går igenom projektets mål, projekt, krav och begränsningar
* Allt Kravspecifikationen i ett dokument: Kravspecifikationen
* Kravspecifikationen används sedan under hela projektet
* Illa formulerade krav – Dåliga resultat
* Välskrivna krav – Bra resultat

Analys

* Allt material från kravspecifikationen gås igenom
* Projektets ramas
* Är projektet genomförbart
* Vad kommer det kosta (resurser)
* - Hur lång tid kommer det ta?
* Man tar fram modeller, schema och policy
* Plan som beskriver hur man skall gå tillväga
* Frågor tillbaka till beställaren

Design

* Skapar en arkitektur för hur projektet skall lösas
* Beskriver hur man i detalj skall ta fram sin produkt
* Vilka material och resurser behövs

Implementering

* Produkten skapas
* Det är här koden skrivs om det är mjukvara
* Planen och arkitekturen genomförs

Testning

* Man testar det som har producerats
* Uppnår det resultatet i kravspecifikationen
* UTA – User Acceptance Test
* När produkten/projektet har accepteras kan det levereras till kunden/beställaren

Lessons Learned

* Vid slutet av en fas
* Eller
* Vid slutet av projektet

Förändringar

* Farligt eftersom dem medför risk
* Tid/Schema
* Pengar/Budget
* Mål / Resultat / Kvalitet
* Ändra en så påverkas dem andra

Förändringsprocess

* Förändringar till ett projekt måste hanteras
* Change Control Boards – CCB – Analyserar och beslutar om en förändring

Dokumentation

* Varje fas genererar mycket dokumentation
* Ger i sig inget värde för den slutliga produkten / resultatet
* Kan användas som källmaterial till framtida projekt

Risker och hot

* Risk-listor och analyser
* Hur minskar man riskerna?
* Vad gör man om man trots allt råkar ut för ett problem

Tidsplan

* Vad kommer projektet att kosta
* När kommer det vara klart?
* Hur påverkar det andra projekt och verksamheter?
* Hur ser tillgängligheten ut för kritiska resurser?
* Är projektet i tid och inom budget?

Work Breakdown Structure (WBS)

* Vilka uppgifter finns?
* Hur lång tid tar det?
* Vem gör de?
* Hur är de relaterade?

WBS – Verktyg

* Papper och penna
* Excel
* MS Project

Fördelar och nackdelar

* Man kan tidigt upptäcka problem i arbetet med kravspecifikationen som visar att projektet inte kan genomföras
* Tydligt var projektet befinner sig – fasen
* Det skapas mycket dokumentation – denna kan ibland användas i framtida projekt

Nackdelar med vattenfall

* Om problem uppdagas sent i projektet är det svårare att åtgärda dem
* Nyckeln till framgång är att allt är tydligt definierat och att all information finns vid start
* Hanterar inte förändringar så bra. Svårare och farliga