**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**LEKTION 1:**

**Presentation från Lek1**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Handlar om utvecklingsprocessen för mjukvara:

* **Analys:** Skaffa sig uppfattning om VAD ett system ska göra
* **Design:** Före implementationen bestämmer HUR man ska implementera det

**Vad vi går igenom:**

1. **Utvecklingsmetoder:** Vilka faser som ingår i processen
2. **Krav:** Vad är ett TYDLIGT krav? Hur bestäms kraven?
3. **Analys:** Tolka formella krav och sammanfatta vad systemet bör innehålla
4. **Design:** Utifrån analys kunna bestämma hur det bäst implementeras
5. **Design patterns:** Best practice
6. **UML:** Dokumentering av analys och design på ett ”*gemensamt språk*”

**OOA-metodik:**

1. Hitta objekten
2. Organisera objekten
3. Beskriv hur objekten interagerar
4. Definiera hur objekten beter sig
5. Definiera objektens interna attribut

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**LEKTION 2:**

**Projektstyrning**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Typer av projektstyrning:**

1. **Vattenfallsmodellen**
2. **RUP**
3. **Agila metoder:**
   1. Scrum
   2. (Kanban)
   3. (XP)
   4. (Mobb)

**Roller i projekt:**

1. **Administrativ projektledare**
2. **Teknisk projektledare**
3. **Arkitekter:**
   1. Systemarkitekt
   2. Lösningsarkitekt
   3. Infrastrukturarkitekt
4. **Utvecklare**
5. **Testare**
6. **Kravhanterare**
7. **Driftstekniket/DevOps**
8. **Beställare -produktägare**
9. **Styrgrupp**

**Vattenfallsmetoden:**

* **Tanken är att man ska vara helt färdig med varje steg innan man går vidare till nästa**
  + Svårt att hantera förändringar när man väl lämnat inledningsfasen
  + Blir dyrt att göra förändringar senare i projektet

**RUP:**

* **Bygger på iterativt sätt (fast inte agilt):** Gör det möjligt att gå tillbaka och ex. ändra på systemets krav och planera om de följande iterationerna
* **Kontrollerar förändringar**

**SCRUM (agil metod):**

* **Målen kan inte sättas från början (= lättrörliga projekt)**
* Beskriver en stafettliknande process

**VAD MAN GÖR:**

* Delar in organisationen i mindre grupper, team
* Splittar uppgiften till mindre deluppgifter (konkretiserade uppgifter)
* Splittar tiden in till mindre fixed tidsomlopp

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**LEKTION 3:**

**Kravhantering**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

* **Typer av krav:**

1. **Funktionella krav**
2. **Icke funktionella krav**

* **User cases:**
* **User stories**
* **Wireframes**

**Vad innebär kravhantering:**

* Som utvecklare är man oftast inte med i denna fas
* Man **fångar upp och beskriver** de krav som ställs på systemet från beställaren
* Görs i början av projektet

**Typer av krav:**

1. **Funktionella krav:**

* Hur systemet ska fungera

EX: Vad händer när jag sparar?

EX: Vad händer när jag trycker på navigeringslänken?

1. **Icke funktionella krav:**

Krav som inte direkt har med systemets funktioner att göra men som ändå påverkar hur systemet ska designas

EX: Prestanda

EX: Säkerhet

EX: Vilka behörigheter som ska finnas

**User cases (mer detaljerade):**

1. **Bra att ha för att beskriva vilka funktioner ett system ska ha**
2. **Används för att specificera krav**

* Beskriver ett antal steg som en aktör gör för att uppnå en specifik sak i ett system
* **Aktör:** Kan ex vara en användare (kan finnas olika typer av användare i use caset)
* **Huvudfall:** Beskriver ett huvudfall
* **Alternativscenario:** Kan göra Use case på undantagsfallen och vad som händer i dessa lägen

**EXEMPEL USE CASE GMAIL:**

**Huvudscenario**

1. Användaren öppnar webbplatsens första sida.

2. Användaren anger sitt användarnamn

3. Användaren anger sitt lösenord och trycker Logga in

4. Systemet verifierar att inloggningen motsvarar ett användarkonto och att lösenord är korrekt.

5. Systemet godkänner inloggningen och mailboxens visas

**Alternativt scenario**

5. Ett felaktigt användnamn eller lösenord har angetts. Systemet godkänner inte inloggningen

6. Systemet meddelar användaren om att det är fel och webbplatsens förstasida visas igen.

**User stories (är mer övergripande):**

1. **Mindre detaljer än use cases**
2. **Utgår från en viss roll(aktör) och beskriver vad som krävs utifrån den rollen för att lösa en uppgift i systemt**

* Liksom use case så vill man här fånga samverkan mellan användare och systemet
* Används för att specificera krav

**UPPDELNING:**

1. **Actor:** Den som äger user storyn (oftast användaren)
2. **Action:** Vad en actor vill göra (Vill… eller Måste…)
3. **Achivement:** Vad en actor vill uppnå i en action

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
USER STORY:**

* **Som en** <actor, type of user>
* **Vill jag** <action, some goal>
* **För att** uppnå <achivement, some reason>

1. **Condition of Satisfaction**
2. **Business Value**
3. **Examples**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**DÅLIGT EXEMPEL USER STORY:**

* **Som en** kund vill jag kunna söka på bokens titel och sortera på pris
* **För att** få fram de böcker som jag letar efter
* **Och se vilka** av dom som är billigast.
* **Som en** kund
* **Vill jag** kunna söka på bokens titel
* **För att** få fram de böcker som jag letar efter.
* **Som en** kund
* **Vill jag** kunna sortera ett sökresultat
* **För att** se vilka som är billigast.

**Exempel: User stories**

* **Estimering:** Man brukar ge poäng eller S/M/L på storlek om hur stor tasken är
* **EX User stories:**
  + Display high scores
  + Print report
  + Twitter login
  + Find friends
* Feature: What to do
* Scenario: When something happens
  + Given: What is the settings
  + (And): maybe some other settings
  + When: The action
  + Then: The result
* Feature: Refund item
* Scenario: Jeff returns a faulty microwave
  + Given that Jeff has baught a microwave for 100
  + And he has the receipt
  + When he returns the microwave
  + Then Jeff should be refunded 100

**Wireframes:**

* **Modell som beskriver uppbyggnaden av en websajt utan layout (bara navigering och uppbyggnad)**
* Ger en känsla av hur sajten ska vara uppbyggd och bra för att jämföra med kunden och dens krav