

TNM094 – Medietekniskt kandidatprojekt

Introduktion till systemutveckling

Några nyheter om systemfel

- 2017 Det nya röntgensystemet i Region Kronoberg blev en katastrof
 - Inget stöd för personnummer eller remisshantering, och systemet gick ned
- 2016 Svårt sjuk fick sitta hel helg utan mat efter bugg i planeringssystemet
- 2014 Kändisars privata bilder läckta efter bugg i iCloud
- 2013 Telia och Halebop läcker samtalslistor
- 2013 Pilot-projektet i Norrköping med PUST inom Polisen stoppas (igen)
- 2013 Störningar på IT-systemet på Karolinska universitetssjukhuset
 - Telefonväxel, mejl och journalsystem slutar fungera under hela dagen.
- 2013 IT-haveri igen på Karolinska universitetssjukhuset.
- 2012 Logica/CGI hackas och läcker därmed 16 GB potentiellt känsliga data.
- 2012 Bugg i ett datorsystem hos SKL störde korssökning av DNA

Systemutveckling

- Systemutveckling är inte programmering
 - Ni kan redan programmera
 - Systemutveckling är programmeringens kontext
- Producera en produkt, ett system
 - Med rätt egenskaper, i tid och inom budget

Kontext – företag som utvecklar

National-/makroekonomi

Företagsledning och -strategier

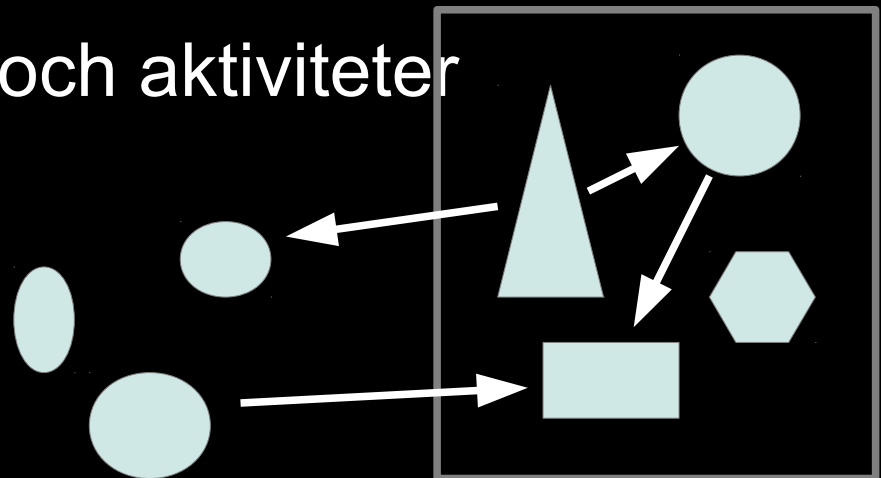
Projektleddning och -strategier

Utvecklingstekniska principer, tekniker och verktyg

Programmering

Ett *System*

- Programvaru-perspektiv
 - Programvara som kör på maskinvara
 - Kommunikerar med tredje-part genom gränssnitt
- Systemgräns-perspektiv
 - interna tillstånd, objekt och aktiviteter
 - externa tillstånd, objekt och aktiviteter
 - gränser och gränssnitt



Några viktiga termer

programvara: programkod som i kompilerad eller tolkad form körs på maskinvara eller annan programvara

maskinvara: fysisk entitet som kör eller kommunicerar med programvara

enhet: maskinvara, programvara eller del därav som kan betraktas separat

modul: kombination av enheter som hör samman och kan betraktas som grupp

Systemutveckling är problemlösning

- Uppfyll angivna behov
 - vilka behov? angivna av vem? varför?
- Producera en högkvalitativ produkt
 - vilken typ av kvalitet? för vem?
 - kvalitet bara nu eller även i ett längre perspektiv?
- Under begränsningar
 - tidsbegränsning
 - begränsade resurser
 - andra begränsningar

Kvalitetsaspekter

- För slutanvändaren
 - Systemet uppfyller syftet
- För utvecklaren
 - Systemet uppfyller specifikationen
 - Systemet uppfyller kontraktet
- För IT-support
 - Systemet behöver lite underhåll och utbildning
- För kunden
 - Systemet genererar inkomst (ekonomiskt värde)
 - Systemet skapar PR-värde

Några viktiga termer

intressent: någon som har intresse i eller på annat sätt påverkar systemets utveckling

slutanvändare: den person eller organisation som förväntas använda systemet

kund: den person eller organisation som betalar för utvecklingen av systemet

Att attackera ett problem

- Steg-för-steg

- 1) Vad är mitt problem?
- 2) Vad behövs för att lösa problemet?
- 3) Hur ska problemet lösas?
- 4) Lös problemet!
- 5) Fungerar lösningen?
- 6) Är hela problemet löst?
- 7) Finns det något nytt problem?

Typiska aktiviteter

- **Kravhantering** (requirements)
vad är mitt problem? identifiera och konkretisera systemet funktioner
- **Systemarkitektur** (system architecture)
vad behövs för att lösa problemet? strukturera och distribuera systemet
- **Modul-design / programdesign** (module design / program design)
hur ska problemet lösas? planera och strukturera implementationen
- **Implementation** (implementation)
lös problemet! realisera planen
- **Enhetstest / modultest** (unit testing / module testing)
löstes problemet? testa lösningen
- **Systemtest** (system testing)
är hela problemet löst? testa hela systemet
- **Leverans** (delivery)
- **Underhåll** (maintenance)
finns nya problem som behöver lösas?

Kravhantering

- Synkronisering mellan
 - Krav från kund
 - Behov från andra intressenter
 - Plan för utvecklingsprojektet
- Syfte
 - Förstå problemet, vad systemet måste kunna göra
 - Bestämma vad systemet ska överlåta till andra att göra
 - Ska *inte* bestämma *hur* problemet ska lösas

Arkitektur och system-/programdesign

- Systemarkitektur
 - Bestämma vilka delar systemet behöver ha (datorer, processer, moduler, etc)
 - Specificera kommunikationsvägar och medium
 - Förstå beroenden och uppdelning av arbete
- Modul-/programdesign
 - Analysera och identifiera data, funktioner och processer
 - Bestämma anropssekvenser och dataflöden
 - Bestämma algoritmer, klasser, funktioner och data

Implementation

- Syfte
 - realisera planen i körbar samt läsbar programkod
 - skapa test som validerar programkoden

Testning

- Syfte
 - Hitta fel, för att senare kunna rätta dem
 - Undvika "det-fungerar-ju-på-min-dator"
- Enhetstestning
 - Identifiera problem i en del av programmet (metod, funktion, klass, modul, anropsgränssnitt, etc)
- Integrationstestning
 - Hitta fel i kommunikationen mellan enheter
- Systemtestning
 - Hitta ouppfyllda krav, felaktigt beteende, säkerhetshål, etc (funktioner, prestanda, kompatibilitet, användbarhet, etc)

Underhåll

- Vidareutveckla systemet
 - Lägg till ny funktionalitet
 - Gör systemet mer användbart
- Rätta till fel i systemet
 - Felaktigt beteende (buggar)
 - Förändrade regler

Genomförande

- Steg 1
 - Vi behöver veta vad vi ska göra
 - Aktiviteter
 - Funktioner
- Steg 2
 - När ska vi göra vad?
 - Utvecklingsprocess

Utvecklingsprocessen/-er

- Guidande principer och restriktioner
 - För att optimera arbete och kommunikation
 - För att hitta problem tidigt
- Processbeskrivning
 - För att hitta inkonsekvens
 - För att hitta redundans
 - För att identifiera ouppfyllda behov
 - För en gemensam förståelse
 - För att återskapa en fungerande process



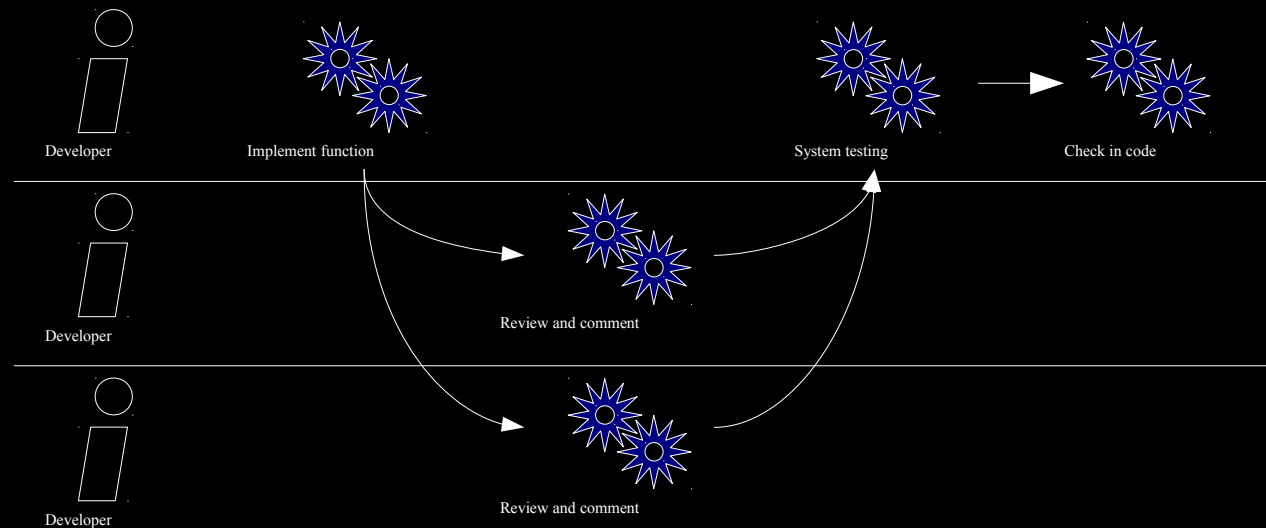
Process-modellering

- Beskriver eller föreskriver processen

- aktiviteter
- resurser
- artefakter
 - prototyper
 - produkter
 - dokument

- Kontrollerar

- sekvenser
- processer och underprocesser
- organisation
- policyer



Några viktiga termer

aktivitet: en syssla begränsad i tid och med ett specifikt syfte

process: en sekvens av aktiviteter med ett specifikt syfte

rutin: ett bestämt tillvägagångssätt för att uppnå ett visst syfte

artefakt: någonting som producerats t ex programvara, dokument eller maskinvara

Planering!

”Plans are worthless,
but planning is everything”

Dwight D. Eisenhower, 1957

Projektplan

- Vi har ett uppdrag
 - En kund har ett behov → projektets syfte
 - Visar på plan för att uppfylla syftet
- Formaliserar planen
 - Anger projektets syfte (kundens behov)
 - Anger vad som innefattas i projektet och målbild
 - Anger schema, team och organisation
 - Anger processer, aktiviteter och rutiner
 - Anger infrastruktur och andra stödfunktioner

Några viktiga termer

leverabel: en artefakt som produceras i ett projekt med ett specifikt syfte

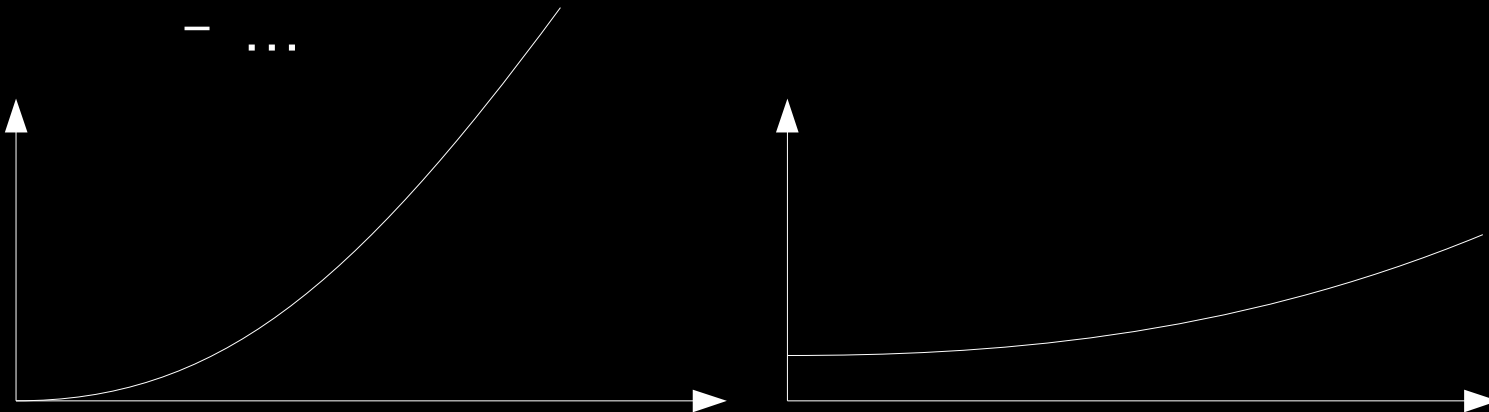
milsten: ett datum då leverabel ska vara klar och/eller beslut ska fattas

infrastruktur: fast, underliggande stöd för genomförande av aktiviteter

overhead: administrativa och andra indirekta kostnader i tid eller pengar

Overhead i utvecklingsprojekt

- Ökar med ökad storlek och komplexitet
 - Fel som behöver rättas
 - Strukturer som behöver implementeras
 - Delar som behöver förklaras
- Minskas av tekniker inom systemutveckling
 - Planering, processer och rutiner
 - Strukturering, modellering och dokumentation
 - Möten
 - ...



Övergripande process

