TDDI02

Programmeringsprojekt, Föreläsning 1

Filip Strömbäck

Med utgångspunkt i tidigare slides av Jonas Lindgren

Kursledning

Kursledare	Filip Strömbäck
Kursassistent	Viktor Olsson
Handledare	Viktor Olsson Johan Frimodig Malin Eklund Aspelin
Etikmoment	Elin Palm Johanna Romare
Examinator	Klas Arvidsson
Kursadministratör	Madeleine Häger Dahlqvist

Kommentarer från tidigare år

- OOA/OOD fokuserar nu på ert eget projekt i stället för ett godtyckligt exempel
- Ändrat ett par otydliga frågor i hemtentan

Viktigt att ni planerar er tid då kursen har en hög arbetsbelastning. Projektet innefattar ca 140 timmar/person.

Kursupplägg

Kursens mål

- Kodning av lite större volym
- Programutveckling i metodisk projektform, inkl. dokumentation
- Kunskap om några element inom Software Engineering

Förkunskaper

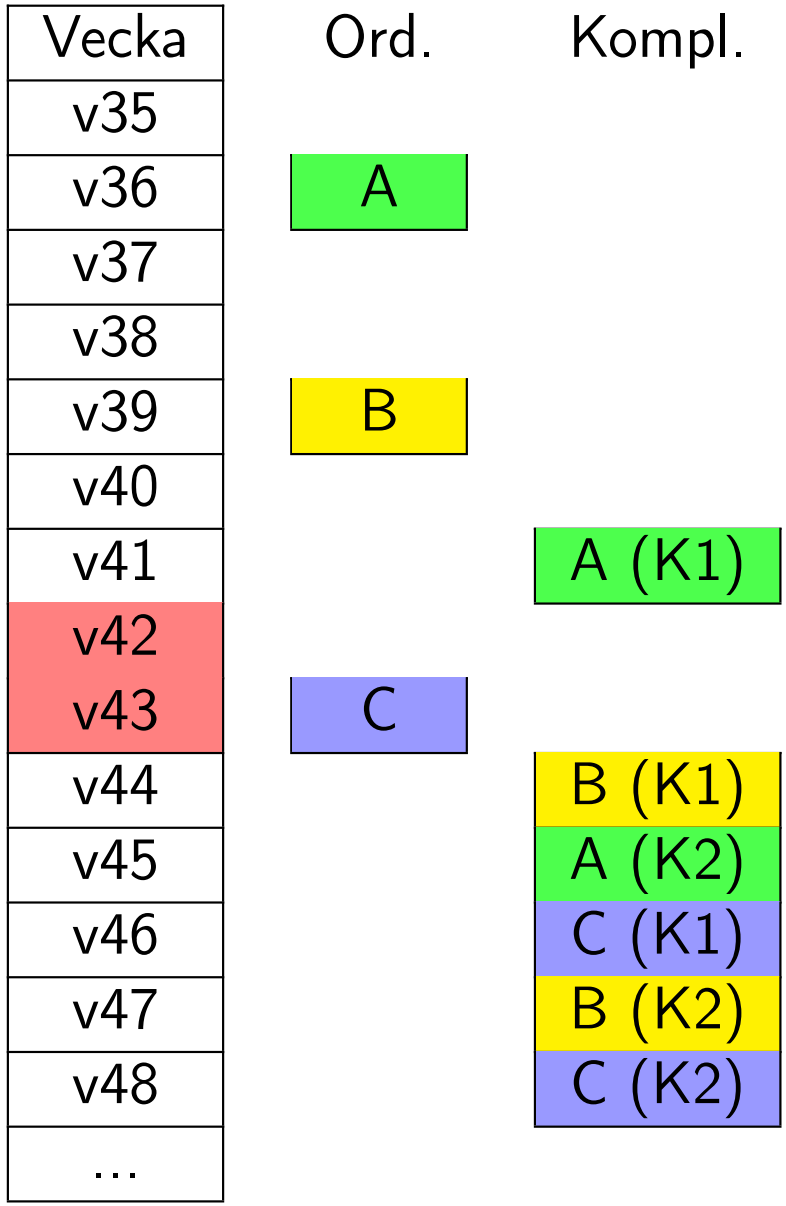
- God förtrogenhet med något högnivåspråk
- Praktisk kunskap inom datastrukturer och algoritmer

Genomförande

- 4 föreläsningar, varav en om verktyg
- 1 laboration – **förslag om innehåll**
- 1 lektion om objektorienterad analys/objektorienterad design
- 1 gästföreläsning av Kristian Sandahl
- Etikmoment med 1 föreläsning av Elin Palm samt 1 seminarie
- **Hemtenta** i 3 delar
- **Projekt**

Examination - Hemtenta

- Individuell
- 3 delar
- 2 kompletteringsmöjligheter per del
- Föreläsningar översiktliga, tentan på djupet
- Läs relevant litteratur
- Ange källor
- Skickas ut via e-post
- En vecka per del
- Lämnas in via Inla



Examination - Projekt

- Grupper om 4 personer
- Projektplan
  - Tidsplanering, milstolpar
- Kravspecifikation
- Designspecifikation
  - Presentation
  - Granskning
- Erfarenhetsrapport
- Leverans
  - Demonstration
  - Kodinlämning
- Testrapport (eventuellt)

Examination - Projekt

- Minst ett möte med handledare i veckan
  - Semat Alphas för att förmedla status
  - Tidsrapport via e-post (sikta på 140 timmar/person)
- Innan 31/8:
  - Bilda grupper á 4 personer, varken mer eller mindre
  - Anmäl gruppen i Webreg!
  - Projektval klart
- Första handledarmötet 31/8 efter föreläsningen, då måste gruppen vara bildad, samt projektval klart.
- Krav - Minst ett externt bibliotek, ex:
  SFML, Qt, SQLite, ENet, ZeroMQ, Box2D, Bullet etc.
- Krav - använd Git och Gitlab
  Repositories skapas åt er.

Projektförslag

- Spel av något slag
- Ljudspelare/mixer/synthesizer
- Chattapplikation
- Bibliotekshantering
- Webapplikation
- Program till Iphone/Android/Windows Phone
- Ny version av BookIt?

Info

- Webreg - anmäl er snarast!
- Gitlab - efter första handledarmötet
- Tidsbokning - BookIt, minst en arbetsdag i förväg

Projekt

En tillfällig kraftsamling som genomförs för att skapa en unik produkt, tjänst eller resultat.

- Ett *definierbart* ändamål
  - Definieras i en kravspecifikation: funktionalitet, prestanda, uppträdande, etc..
- Ett *unikt* åtagande
  - inte rutinarbete, avser inte något som gjorts identiskt tidigare
- En *tillfällig* aktivitet
  - Det finns en tydlig början och ett tydligt slut.

Kort definition:

Ett projekt är en kombination av resurser som förs ihop för att skapa något som inte fanns förut. [1]

[1] Cleland och Ireland, 2002

Utgångspunkt

Ett projekt löper allmänt i ordningen:

1. Förstå problemet
2. Planlägg lösningen
3. Genomför planen
4. Utvärdera resultatet

Utgångspunkt

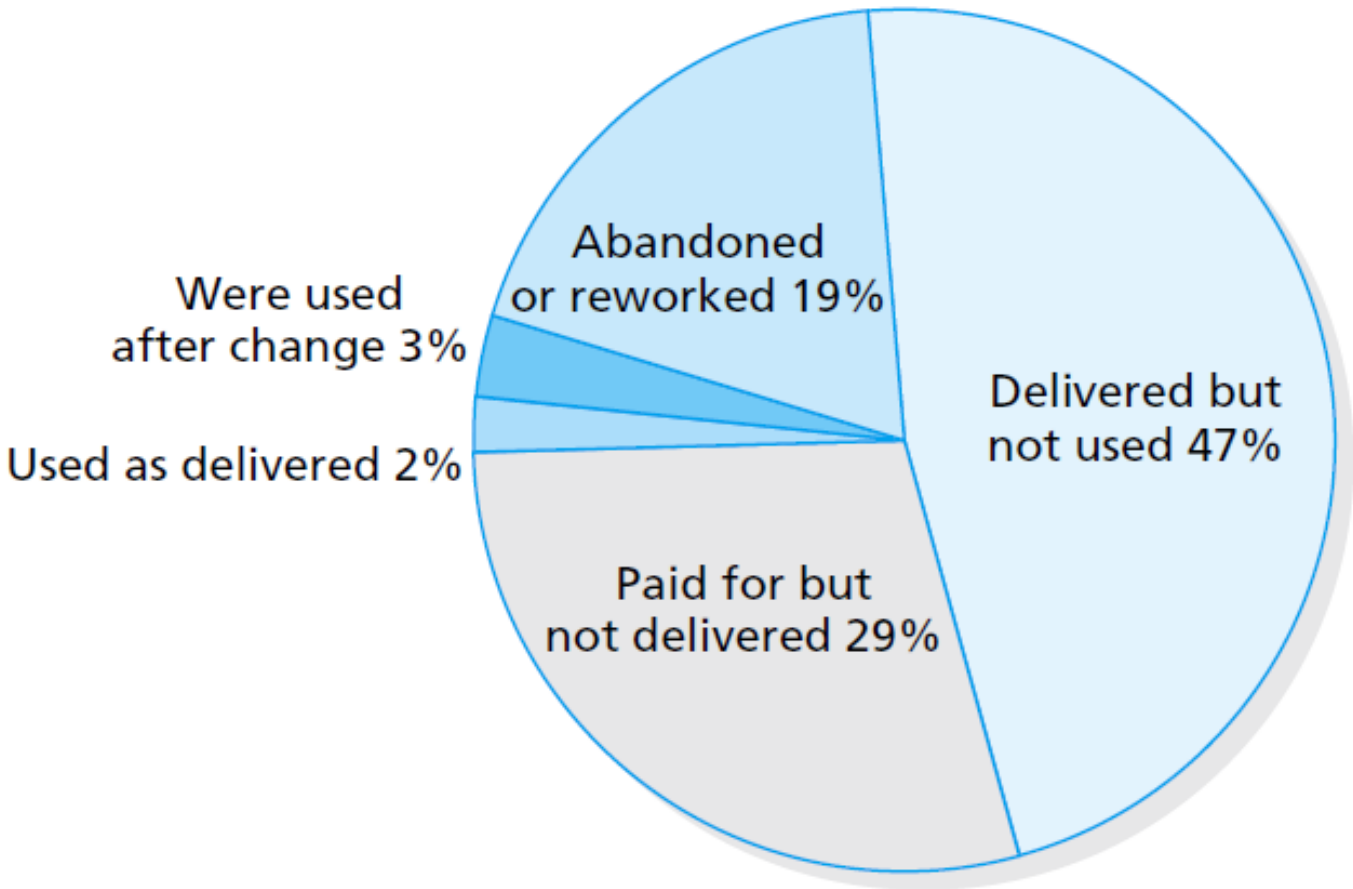
Programmering (=kodning) och design(=konstruktion) är teknikområden.

Framställning av stora/komplexa system kräver dessutom tex:

- Många programmerare/grupper
- Specialister, utbildning, ersättare, etc.. (personalfrågor)
- Marknadsföring, kontrakt, etc.. (externa frågor)
- Kvalitetssäkring (processer, produkter, etc..)
- Dokumentation av olika slag
- Management (ledning, uppföljning, resursfördelning, etc..)

Till stor del icke-tekniska frågeställningar!

Vad är Software Engineering?



Software Engineering for Students: A Programming Approach, D. Bell

Vad är Software Engineering?

Mål:

- Konstruera stora/komplexa programvarusystem
- I enlighet med användares/beställares avsikter
- Inom föreskrivna budget- och tidsramar
- Med hänsyn till kvalitets- och underhållsaspekter

Alltså behövs:

- Metod
- Verktyg
- Riktlinjer
- Attityder
- ...

Vad är Software Engineering?

Mest övergripande mål:

- Förmåga att urskilja och rätta sig efter kunders önskemål och krav (dvs. behovsstyrt, inte teknikstyrt)
- Användande av ingenjörsmässiga principer, idéer, kvaliteter och attityder

Vad innebär ingenjörsmässighet?

Konstruktion av programvara är inte (längre) ett ad-hoc jobb utfört av enstaka kreativa individer i den mörka källaren, utan ett välorganiserat, metodbaserat teamwork, baserat på känd teknik.

Begreppsdistinktioner

Principer:

- Ett enskilt förfaringssätt, sätt att arbeta
  - Parprogrammering
  - Planning poker

Metod:

- Ett konkret, detaljerat, förfaringssätt, inklusive verktyg och principer
  - Scrum
  - Extreme Programming (XP)

Metodologi:

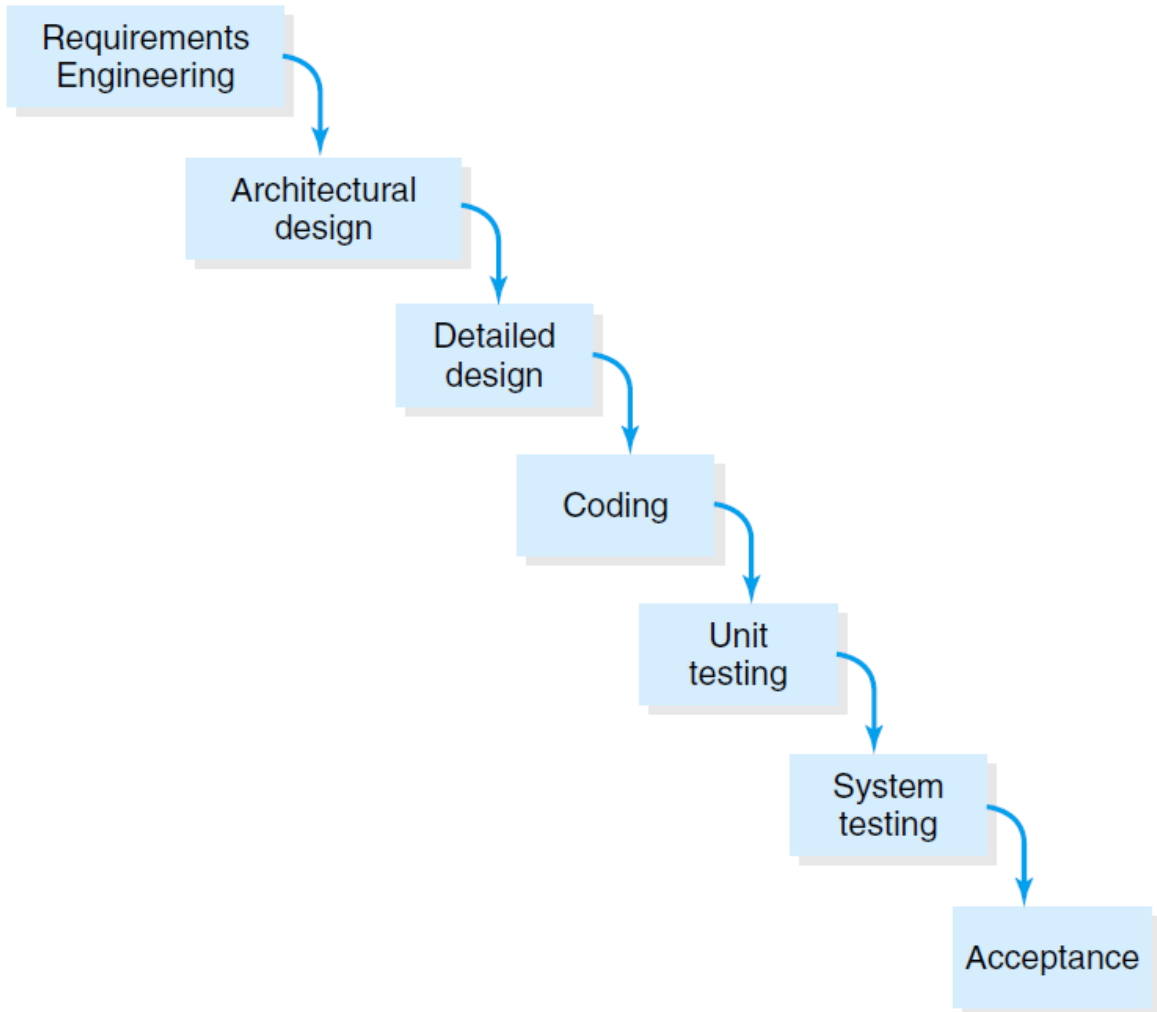
- Läran om hur metoder konstrueras, kan värderas, dess generella egenskaper.

Traditionell arbetsgång

Projektfas	**Allmän SE-fas**	**Resultat**
Förstå problemet	Kravanalys	Kravspecifikation
Planlägg lösningen	Planering	Projektplan
Genomför planen	Design & implementation	Designspecifikation, kod
Utvärdera resultatet	Testning	Ny kod, dokument

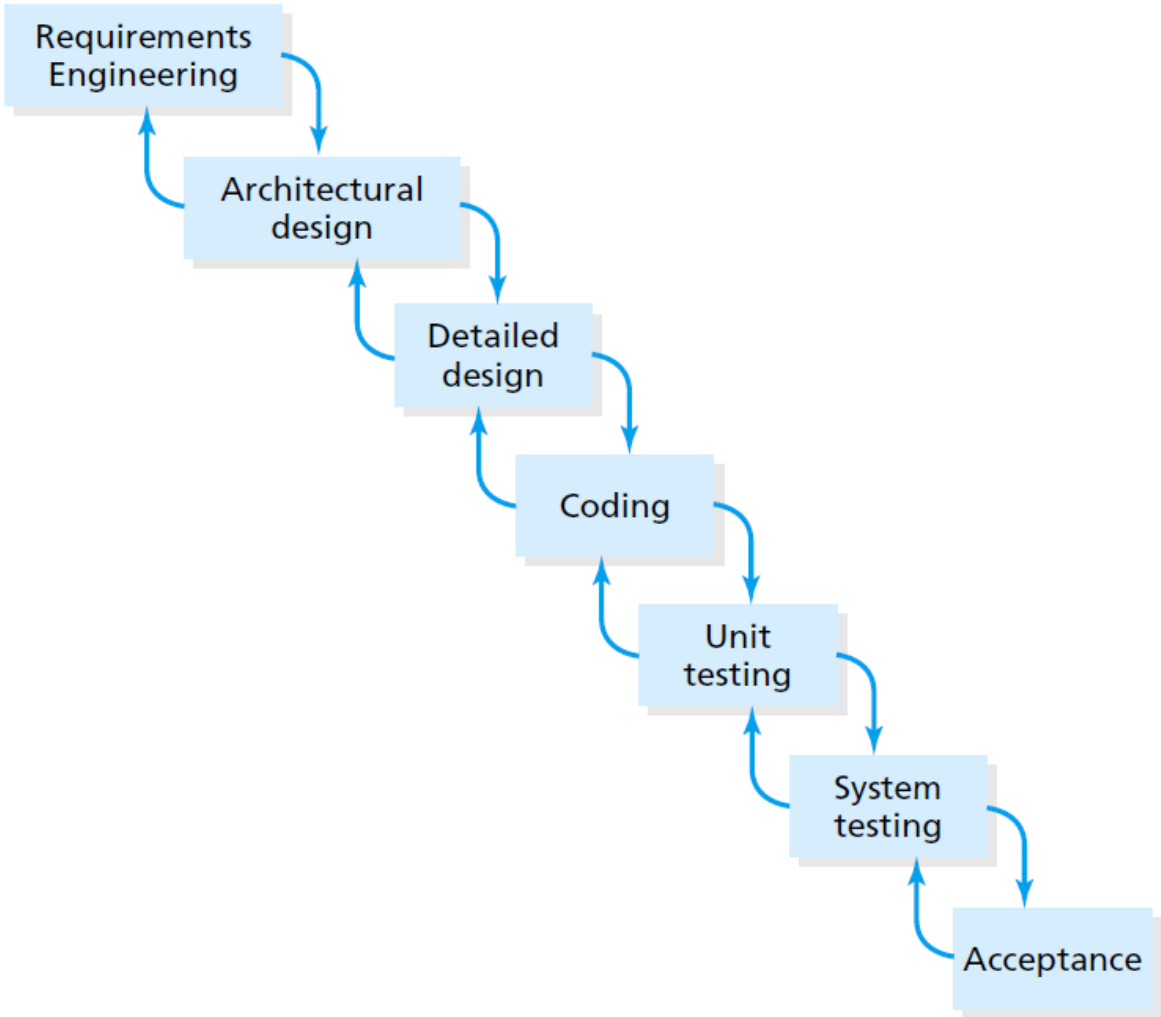
Dessa steg bryts ner i flera, mer preciserade delsteg.

Vattenfallsmodell

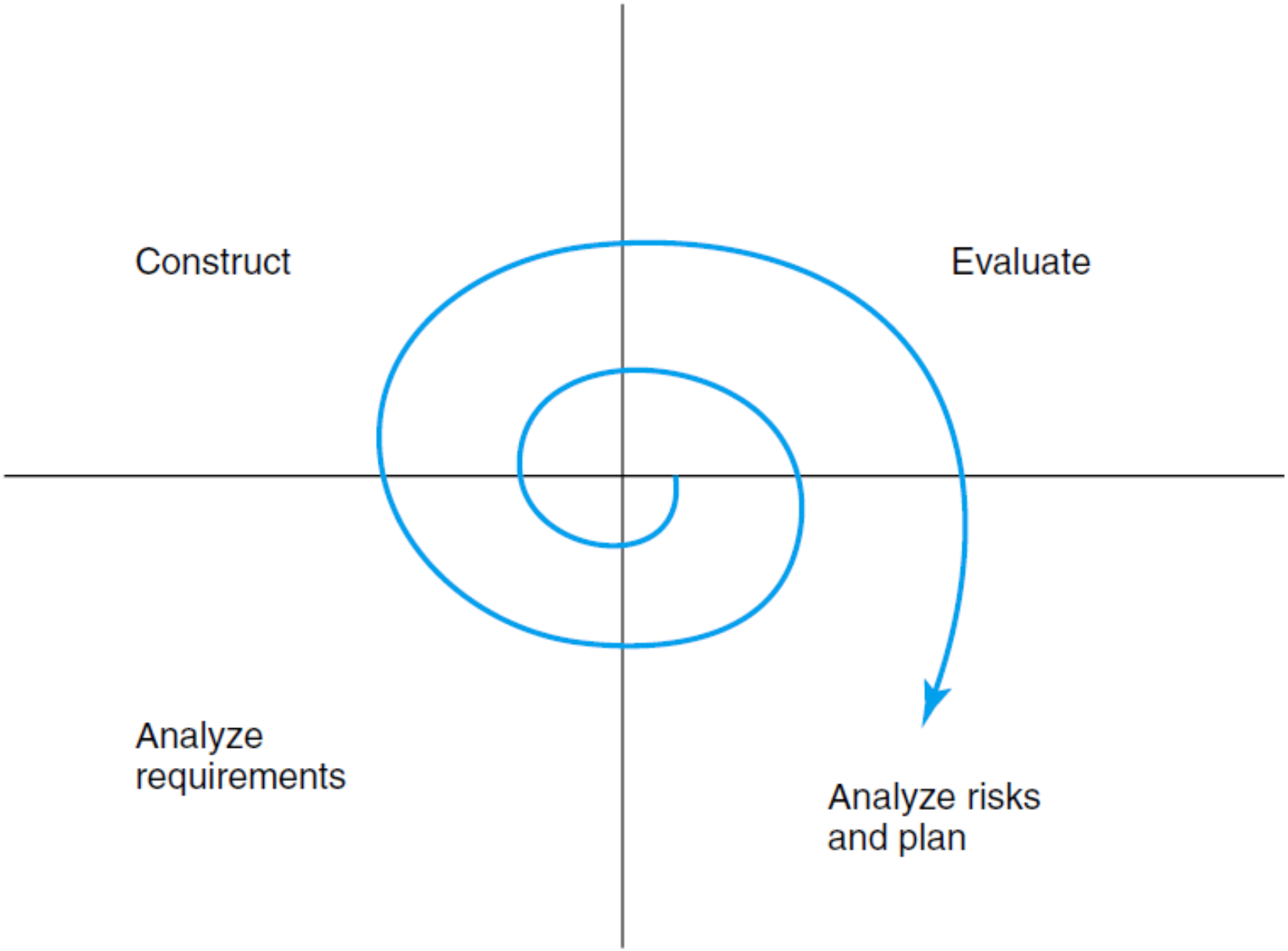


Software Engineering for Students: A Programming Approach, D. Bell

Vattenfallsmodell med återhopp



Spiralmodellen



Software Engineering for Students: A Programming Approach, D. Bell

Scrum

- Scrum är en agil utvecklingsmetod.
- Roller
  - Produktägare
  - Scrum master
  - Utvecklare
- Sprints - liknande spiralmodellen.
  - Varierande längd, ofta 3-30 dagar.
  - Planering
  - Arbete
  - Återkoppling/retrospektiv

Scrum

- Backlog
  - Hanteras av produktägaren.
  - Vad ska göras?
  - Uppdelat i relativt små delar.
- Sprint backlog
  - Del av backlog för aktuell sprint.
- Daglig Scrum
  - Kort statusmöte varje dag.
  - Ofta stående $\Longrightarrow$ kort (max ca 15 min)

Prototyping

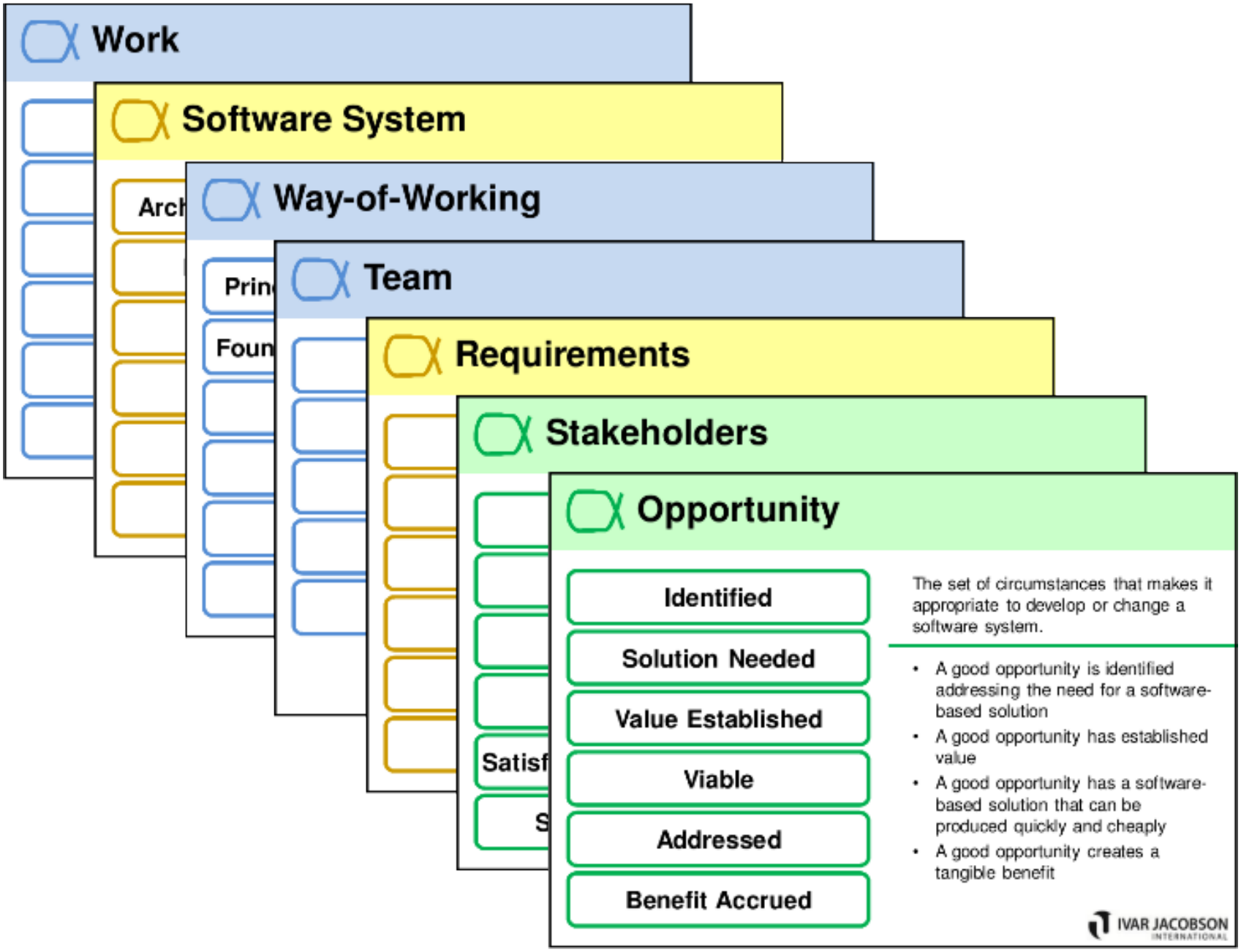
Bygger på att:

- Tidigt skapa en prototyp
- Låta användaren testa denna prototyp
- Förfina kravspecifikationen baserat på feedback
- Förfina prototypen och upprepa

Kan användas på två sätt:

- Evolutionary development
  - Prototypen förfinas tills den utgör den färdiga produkten
- Throwaway prototyping
  - Prototypen kasseras till slut
  - Den färdiga produkten utvecklas från grunden med tidigare lärdomar

SEMAT Alpha Kernels



Tankar inför projektet

- Börja fundera på era principer och er metod
- Finns många fler att välja på än de som nämnts
- Gör milstolpar fram till tenta-p, ange även Semat Alpha-tillstånd för varje milstolpe
- Se till att er tidsplan sträcker sig åtminstone en vecka framåt
- Dokumentera gärna era principer samt metoder

Övrigt

- Kursen kräver stort egenansvar!
- Läs kurshemsidan noga
- Bilda grupper och välj projekt!
- Anmäl er i Webreg!
- Maila förslag till laborationen till `filip.stromback@liu.se`