

Kursus Introduktion

Anne Haxthausen

ah@imm.dtu.dk

Informatics and Mathematical Modelling
Technical University of Denmark

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 1

Agenda for i dag

- Praktiske oplysninger:
 - Hvem er vi?
 - Kursusaktiviteter
 - Kursusmateriale
- Introduktion til faget *software engineering*
 - Faget som en ingeniør-disciplin
 - Kursusmål og -indhold
 - Et udviklingseksempel (separat foil-set)
- Kravspecifikation (separat foil-set)
- Øvelser i E-databaren i bygn. 341, rum 003 og 015:
Løs opgave 1, som kan downloades fra kursets hjemmeside.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 2

Hvem er vi?

- 100 studerende med forskellig baggrund:
 - software teknologi
 - It og Kom
 - andre retninger
- Lærere:
 - Anne Haxthausen, lektor ved IMM
 - Hubert Baumeister, lektor ved IMM
 - 3 hjælpelærere

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 3

Kursuslærerne

Lektor Hubert Baumeister:

- Forskningsinteresser: service-orientation, software udviklings-processer (især agile processes), og formelle aspekter af software engineering teknikker.
- Dipl.-Inf., Dortmund University 1991, Ph.D. i Comp. Science, Saarland University 1999.
- Forsker ved Max Planck Institute for Computer Science i Saarbrücken (1991-1998).
- Post doc ved Munich University (1998-2006).
- Ansat på DTU (IMM) siden 2006, sektionsleder for Software Engineering sektionen siden 2008.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 4

Kursuslærerne

Lektor Anne Haxthausen:

- Forskningsinteresser: formelle aspekter af software engineering, især design af specifikationssprog og verifikationsteknikker.
- Civilingeniør, DTU 1985; Ph.D. DTU 1989.
- Software engineer hos DDC og CRI 1988-1994.
- Ansat ved DTU (ID, IT, IMM) siden 1995.
- Gæsteprofessor ved et japansk forsknings-lab 1993 og et universitet i Paris i 2005/06.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 5

Kursusaktiviteter

Første 9 uger:

- Forelæsninger her mandage kl. 13 - 14:45.
- Øvelser i E-databaren i 341 rum 003 og 015 mandage efter forelæsningerne.
 - Hjælpelærerne vil være tilstede kl. 15-17.
 - Øvelserne skal færdiggøres hjemme.

Sidste 4 uger:

- Projektarbejde i grupper.
- Konsultationstid hos hjælperlærerne.
- Aflevering af projektrapporter + give demo'er.

Ingen skriftlig eksamen.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 6

Kursusmateriale og oplysninger

- Hjemmesiden www2.imm.dtu.dk/courses/02161/ indeholder:
 - praktiske oplysninger: lokaler, planer mm
 - kursusmaterialer
- CampusNet: vil blive brugt til at udsende meddelelser.
- Bøger:
 - *UML Distilled*, Third Edition, Martin Fowler, Addison-Wesley, 2003. ISBN 0-321-19368-7. Kan købes i Polyteknisk Boghandel.
 - ...
- Noter: vil løbende blive gjort tilgængelige fra hjemmesiden.
- Overheads: vil løbende blive gjort tilgængelige fra hjemmesiden.
- Opgaver: vil løbende blive gjort tilgængelige fra hjemmesiden.
- Løsninger: udleveres generelt ikke.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 7

Agenda for i dag

- Praktiske oplysninger:
 - Hvem er vi?
 - Kursusaktiviteter
 - Kursusmateriale
- Introduktion til faget **software engineering**
 - Faget som en ingeniør-disciplin
 - Kursusmål og -indhold
 - Et udviklingseksempel (separat foil-set)
- Kravspecifikation (separat foil-set)
- Øvelser i E-databaren i bygn. 341, rum 003 og 015:
Løs opgave 1, som kan downloades fra kursets hjemmeside.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 8

Software Engineering som en disciplin

Software Engineering

- den disciplin, der vedrører alle aspekter ved produktion af programmel, herunder specifikation, konstruktion, test, udvikling, drift og vedligeholdelse.

Begrebet 'software engineering' blev indført i forbindelse med en konference i 1968.

Emne 'the software crisis'

Mål programmeringskunsten skulle blive til en ingeniørdisciplin

Det er, trods fremskridt, stadig udfordrende at lave programmel,

- der opfylder brugernes forventninger, og
- er afleveret til tiden og indenfor budgettet.

ung og umoden disciplin

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 9

Nogle definitioner

- F.L. Bauer, from *Information Processing*, 1972.

"The establishment and use of sound engineering principles (methods) in order to obtain economically software that is reliable and works on real machines"

- IEEE (Institute of Electrical and Electronics Engineers)

"Software engineering.

(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

(2) The study of approaches as in (1)

"

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 10

Nogle forhold om programudvikling

- store programmer
- korrekte programmer
- hyppige modifikation af programmer
- konstrueret af flere, muligvis store, udviklings teams.
- ...

Desværre er det ofte sådan at

- tidsplaner skrider (meget)
- omkostningerne skrider (meget)
- grænser flyttes
- vanskeligt at producere software af god kvalitet
- ...

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 11

Hvorfor er SE svært?

- Krav ændres og tilføjes hyppigt
- Vanskeligt at indfange de rigtige krav
- Svært at overskue andres og egne programmer
- Svært at gøre rede for programmers korrekthed
- Begrænsninger i metoder til test af programmel
- Begrænsninger i metoder til vedligeholdelse af programmel
- Vanskeligt at lave programdokumentation af høj kvalitet
- Vanskeligt at vedligeholde programdokumentation
-
-
-

Hvad kan man gøre?

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 12

Typiske faser i et projekt

- **Analyse:** Hvad er problemet – definer **krav** til løsningen.
→ **Kravs-specifikation**
- **Design:** Hvordan er løsningen på problemet – beskriv opdeling i komponenter, disses grænseflader og opførsel. Model (abstraktion) af løsningen. → **Designspecifikation**
- **Implementering:** Konstruer løsningen. → **Teknisk dokumentation**
- **Test:** Evaluer løsningen – afprøv programmet.
→ **Test dokumentation: plan og resultater**

Mange andre typiske faser findes, f.eks. **vedligeholdelse**, men kun de fire ovennævnte vil optræde i 02161.

Der findes en række paradigmer for den process det er at udvikle software. I disse **udviklingsprocesser** kombineres faserne på forskellige måder.

Udfordring: rød tråd (sporbarhed) fra krav til kode

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 13

Teknikker og notationer

Der findes mange teknikker og notationer til at udføre og dokumentere resultaterne af de enkelte faser.

I vil lære nogle af disse i dette kursus. Bl.a. vil i lære UML notationen.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 14

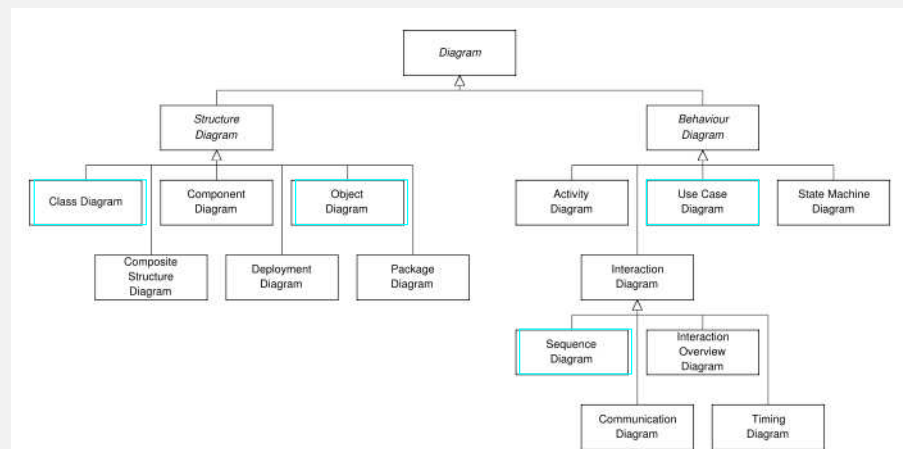
Hvad er UML?

- UML = **Unified Modeling Language**
- Er en familie af grafiske, objekt-orienterede notationer til at specificere, visualisere og dokumentere modeller af objektorienteret programmel.
- UML kan også bruges til at modellere andet end programmel, f.eks. begreber.
- UML er ikke en udviklingsmetode.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 15

13 forskellige typer UML diagrammer

Forskellige typer diagrammer beskriver forskellige aspekter (views) af et system.



02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 16

Kursusmål

- At kunne gennemføre et “mindre projekt” ud fra en “uformel og åben beskrivelse” til et produkt
- At kunne udvikle krav, design, implementering og test således at væsentlige begreber kan spores igennem hele forløbet

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 17

Kursus indhold

- Analyse:
 - use cases
 - begrebslister
 - UML klassediagrammer
- Design:
 - CRC
 - UML klassediagrammer
 - UML sekvensdiagrammer
 - Design-by-contract
 - Design patterns
 - Usability engineering
- Implementering: i Java (brug af viden I allerede har)
- Test:
 - systematiske teknikker til *planlægning* af test
 - værktøj til automatisk at *udføre* test (JUnit)

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 18

Agenda for i dag

- Praktiske oplysninger:
 - Hvem er vi?
 - Kursusaktiviteter
 - Kursusmateriale
- Introduktion til faget *software engineering*
 - Faget som en ingeniør-disciplin
 - Kursusmål og -indhold
 - Et udviklingseksempel (separat foil-set)
- Kravspecifikation (separat foil-set)
- Øvelser i E-databaren i bygn. 341, rum 003 og 015:
Løs opgave 1, som kan downloades fra kursets hjemmeside.

02161 Software Engineering 1 ©Anne Haxthausen, Spring 2010 – p. 19