

## *ITU – VOOB – F2002*

### *Udviklingsprojektet*

Lars Thorup  
lars.thorup@acm.org

1

## *Udviklingsprojektet*

- ★ Folkene
- ★ Værktøjerne
- ★ Analyse
- ★ Design
- ★ Test
- ★ Programmering
- ★ eXtreme Programming

2

## *Folkene*

- ★ Gruppen er ofte givet
- ★ Udfyldte forskellige roller
  - \* Resultatorienteret
  - \* Kreativ
  - \* Teknisk
  - \* Overblik
- ★ Kommunikation: snak, email
- ★ Par-programmering

3

## *Værktøjerne*

- ★ Dokumenthåndtering
- ★ Planlægning
- ★ Diagrammering
- ★ Programmering
- ★ Test
- ★ Byg
- ★ Performance

4

## Dokumenthåndteringsværktøjer

- \* Fælles repository
- \* Versionsstyring
- \* Flerbruger
- \* CVS er open source standarden
- \* Microsoft SourceSafe er også udbredt
- \* CVS over SSH gennemgås detaljeret senere idag.

5

## Planlægningsværktøjer

- \* Så man kan bevare overblikket over
  - \* Mandskab
  - \* Kalendertid
  - \* Indhold
  - \* Milepæle
- \* Microsoft Project er almindelig udbredt
- \* Ikke en del af dette kursus

6

## Diagrammeringsværktøjer, I

- \* En tegning siger mere end 1000 ord...
- \* Mange kriterier for valg:
  - \* Understøtte bestemte notationer
  - \* Køre på både Linux og Windows?
  - \* Automatisk vedligeholde forbundne diagrammer (inklusive kode)
  - \* Pris

7

## Diagrammeringsværktøjer, II

- \* Mange valgmuligheder
  - \* Office suite (Microsoft Office eller StarOffice)
  - \* Microsoft Visio
  - \* Together
  - \* Dia som er open source

8

## Programmeringsværktøjer

- ★ Kode browser og editor
  - ★ vælg selv
- ★ Compiler
  - ★ javac
- ★ Debugger
  - ★ jdb?
- ★ Biblioteker inkl. dokumentation
  - ★ jdk

9

## Testværktøjer

- ★ Understøtte automatiseret test
- ★ JUnit er meget udbredt
- ★ JUnit findes også til "alle" andre programmeringsprog
- ★ JUnit bliver gennemgået i 5. forelæsning

10

## Byggeværktøjer

- ★ Automatisere compilering og testkørsler
- ★ Automatisere pakning til release
- ★ Automatisere generering af dokumentation
- ★ Make er meget brugt
- ★ Ant fra Apache er specielt tilpasset Java
- ★ Ant bliver gennemgået i 5. forelæsning

11

## Performanceværktøjer

- ★ Vi vil kunne måle hvor flaskehalsene er
- ★ Fx OptimizeIt til Java
- ★ Vil blive gennemgået i 11. forelæsning

12

## *Værktøjerne, opsamling*

- ★ Dokumenthåndtering
- ★ Planlægning
- ★ Diagrammering
- ★ Programmering
- ★ Test
- ★ Byg
- ★ Performance

13

## *Analyse*

- ★ Emnet for 2. og 3. forelæsning
- ★ Use Cases
- ★ Object Modelling
- ★ Løbende ændring af krav

14

## *Design*

- ★ Emnet for 4., 6. og 7. forelæsning
- ★ Design Patterns
- ★ Design by contract
- ★ Fejlhåndtering

15

## *Test*

- ★ Emnet for 5. forelæsning
- ★ Automatiseret test
- ★ Continuous Integration

16

## *Programming*

- ✧ Emnet for 8., 9., 10. og 11. forelæsning
- ✧ Persistens
- ✧ Trådprogrammering
- ✧ Performancemåling

17

## *eXtreme Programming*

- ✧ XP ting vi gør
- ✧ Par-programmering
- ✧ Automatiseret test
- ✧ Continuous Integration
- ✧ XP ting vi ikke gør
- ✧ The planning game
- ✧ Løbende design & refactoring

18