Welcome to
OO Project Course 2017

Joachim von Hacht

Staff

Lecturer, Course responsible, Examiner

- Joachim von Hacht, hajo@chalmers.se

Assistants

- Adam Waldenberg
- Jacob Jarmar
- Christer Carlsson
- Joel Hultin

Students

Course has 2 major target audiences
- Chl/IT programme <u>year 1</u>
- GU/CS <u>year 2</u>

Will handle this as a <u>year 1</u> course!
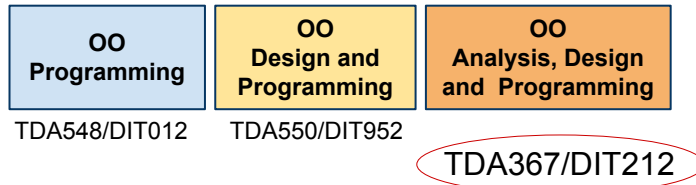- Others possibly will find tempo a bit slow

Must have passed TDA548/DIT012 or TDA550/DIT952.

If failed both preceding OO course, this is <u>not a good course to take</u>
- Take any programming (or other useful) course instead!
- IT-program has been informed

Course Position

Course is part of the OO-trail

OO Programming	OO Design and Programming	OO Analysis, Design and Programming

TDA548/DIT012   TDA550/DIT952

TDA367/DIT212

Will use knowledge from DAT216 (GUI)

In real life reversed
- First OOA …
- then OOD…
- and finally OOP
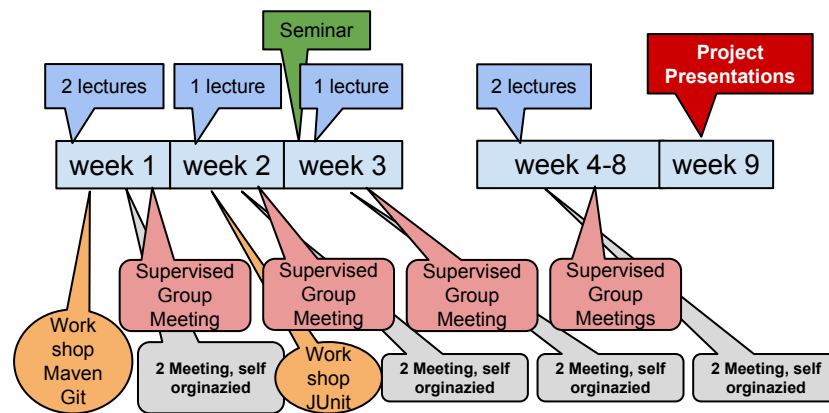
Course Content

See Course Page > Projekt och Grupper > Projekt PM

Participating in a group (of 4 students) you will follow a software process to deliver an application (of your choice).
- Expected application type is a <u>standalone or mobile with a GUI</u>

Course Organization

Seminar

2 lectures   1 lecture   1 lecture   2 lectures

Project Presentations

| week 1 | week 2 | week 3 | week 4-8 | week 9 |

Work shop Maven Git

Supervised Group Meeting

Supervised Group Meeting

Supervised Group Meeting

Supervised Group Meetings

2 Meeting, self orginazied

Work shop JUnit

2 Meeting, self orginazied

2 Meeting, self orginazied

2 Meeting, self orginazied

Roadmap on course page

Lectures

Will run "the course" software process
- Introduces concepts
- Remarks on design …
- Slides after lecture

Not many new programming concepts
- In this course you should exercise your programming skills (so far)
- More on design: Modules, Dependencies, MVC, …

Group Meetings with Assistant

Groups meets assistant 1h/week
- Help with process, advice in design, documentation, etc. …
  <u>not</u> a bug fixer
- You are supposed to push!
- Any problems: Contact me!

Meetings Mandatory (and on time)!

Group Meetings Self Organized

Groups should have a 2 documented
meeting/week
- I will help group, … keep focus

Mandatory!

Workshops

Workshops
- Kickstart for tools
- 2h/sessions, <u>optional</u>
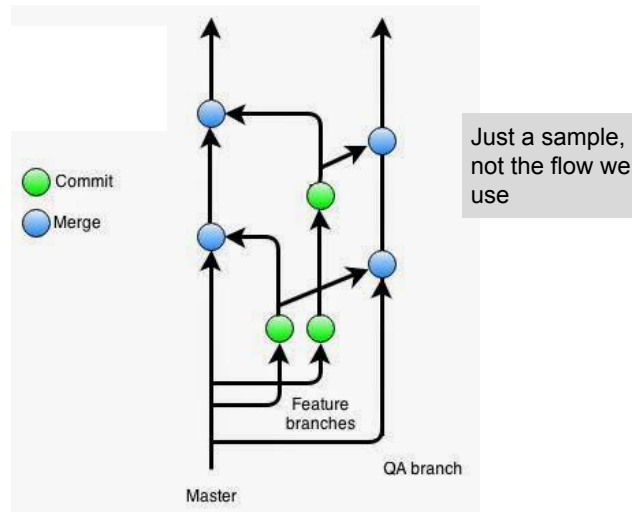
Week 1: Git and Maven
- Use: UTF-8!

Week 2: JUnit

Other nice tools (no workshop)
- Quality tools like: STAN, Findbugs, PMD, Jacoco, …
- Process support: Task lists, UML drawing, etc.

Git Workflow

Commit
Merge

Just a sample, not the flow we use

Feature branches

QA branch

Master

Git mandatory.
- Must use Git in a disciplined way
- All members should use a (the same) workflow (more in workshop)

Maven and Gradle

```xml
<dependency>
        <groupId>com.thoughtworks.xstream</groupId>
        <artifactId>xstream</artifactId>
        <version>1.4.7</version>
        <type>jar</type>
    </dependency>
    <dependency>
        <groupId>org.projectlombok</groupId>
        <artifactId>lombok</artifactId>
        <version>1.16.6</version>
    </dependency>
    <dependency>
        <groupId>commons-io</groupId>
        <artifactId>commons-io</artifactId>
        <version>2.2</version>
    </dependency>
</dependency>
```

Modern software development is highly dependent on libraries (API)
- First thing to investigate if some need: Is there a library/API?
- Potentially very complex to handle because of transitive dependencies (libraries depend on other libraries, that depends on ...)
- By using build automation software to handle libraries/APIs needed by our application we greatly reduce the problems!
- We accept: Maven and Gradle

Seminar

Seminar
- Early week 3
- All groups present their (first) domain model
- 10 min/group
- Quiz Based (i.e. Guess what this is ...)

Project Presentation

- Demo run application
- Technical walkthrough
- Answer questions (from audience)
- Act as opponents for other group
- ~ 20 min./group

For IT: In collaboration with LSP 310

Questions