**Course overview**

**Methods and techniques for the development of medium- and large-scale software**

Topics:
- software processes
- requirements engineering
- design and development
- verification and validation
- software evolution
- project management

There is a project but this is not a programming module.

**Course details**

- **Web page:** [www.cs.sun.ac.za/rw344/](http://www.cs.sun.ac.za/rw344/)
  - in progress, check regularly
- **Lecturer:** Bernd Fischer (bfischer@cs.sun.ac.za)
- **Lectures** (all in A407):
  - Monday, 12:00-13:00
  - Wednesday, 08:00-09:00
  - Friday, 09:00-10:00
- **Tutorials:**
  - Tuesday, 10:00-13:00, Narga H

---

**Learning objectives**

- knowledge of the **basic terms and concepts**
- understanding of commonly used **methods**
- experience with **design tools**
- appreciation of **social aspects** of software engineering
- experience with **group design and development**
- ability to **learn new techniques** as they emerge

Master the knowledge and skills to begin practice as software engineer.
Course material

- Slides will be posted on the web as we go
- 2013 slides by Jaco Geldenhuys
  - contain more text, but different organization
- Reference textbooks

Reference textbooks:

Evaluation: Continuous assessment

- class tests: 30%
  - 6 class tests, 10 marks each, 5 best count
- design project: 30%
  - requirements, design, code, testing, evolution
  - several deliverables with (roughly) equal weights
- group development project: 40%
  - requirements, design, demonstration
  - 1:1:2 weights

Class tests

- individual submissions
- roughly every second week, details on web page
- mostly multiple choice

Experiment participation

- compare tool developed here against standard tool
  - use tool, answer some questions
- voluntarily, for extra credit
- exposure to software engineering research
- we will collect some data and statistics
  - fully anonymized
  - fill out questionnaire on SurveyMonkey

Design project

- "small group" submissions (2 students)
  - same marks for both group members
  - group assignments posted at the end of the week
- several small "spec, design, and code" deliverables
  - documents
  - models
  - testplans
- set at beginning of tut (or after test), hand in at end
- use Astah tool for modelling, http://astah.net/
  - free community version

Group development project

- "large group" submissions (5-6 students)
  - group marks "moderated" by contribution
  - self-assessment & peer review
- deliberately underspecified
- "spec, design, and code" deliverables
  - requirements document
  - design document
  - implementation demo
- completely self-managed
- group assignments later
**Plagiarism**

- need to submit signed declaration with each assignment (except class tests)
  - for group work: all group members need to sign
  - forms on web page or part of assignment sheet
  - submissions without declaration will not be marked

---

**Preliminary schedule**

- lecture schedule remains in flux (agile planning :-()
  - one lecture in tomorrow’s tut (10:00)
  - no lectures Aug 10 (→ Aug 11), Sep 23/25/28
- potentially company presentations
- class tests on Jul 28, Aug 18, Aug 25, Sep 15, Sep 29, Oct 13
- tuts starting Jul 28
- design project before recess
- group development project (largely) after recess

---

**What is software engineering?**

---

**Questions??**