ESD Intro

Pieter van den Hombergh

Fontys Hogeschool voor Techniek en Logistiek

August 21, 2018

(ESD), Software Architecture for Enterprises and a lot of fun

- The description in this frame title says it all.
- This addresses several aspect which we as teachers think are interesting for you.
- The way of working prepares you best for your bachelor project and beyond.
- You will have a say (because you will do some of it yourselves ) in what we will tackle as topics.
- We will have some guest speakers:
  - We will have a workshop on 2018-09-03, whole day, provided by Cap Gemini, organized by Torsten Rix, former student.
- **There will be NO written exam.**

The fine print

However, we need some rules about grading, assessment, fairness and rewarding differences in effort. These are the rules:

- All students will do time writing for all modules in this semester, including SOFA, GRAP, COM7 and this module, ESD.
- The presence during the scheduled hours is mandatory for all wortkshops to be given, by either guests, the teachers, or fellow students.
- The teachers are available for guidance and discussion during scheduled hours.
- The use of the sebivenlo repository group at github is mandatory. `https://github.com/sebivenlo` Make sure you have a github account with a recognisable name. Send it to me, so I can invite you as commiter to the repo.
- The teachers prescribe the technology of reports and any other documentation. Think LaTeX (for sheets, presentations) and asciidoctor (for github hosted websites).
- The students will produce course material and present it in form of a workshop, with fellow student and teachers as the target audience.
- The workshop will have exercises and solutions to the exercises all in the github repo, using tags.
- The workshop will have a github hosted web page. See the examples.
- For group work, we will assess contributions through peer assessment and git version history.
- You are meant to question everything (except this list) to maximise the learning of all involved. Actually the relevance of your questions will count towards your esteem.

- We want publication quality stuff, meaning *reviews, reviews, reviews*, so that your and our names will be known for the good stuff.

Rules on tools and technology

- All technologies used are open source and available on Linux. You must prove that you materials are properly useable on Linux.
- Use of docker containers for all server side stuf is mandatory. This includes command line based stuff.
- Choice of editors is free, for Java we prefer NetBeans-IDE, because of portablity to earlier semesters.
- The build tool (for Java) is maven, with a possible excursion to gradle.

Final Rule

- This is a choice module, cooperation required but based on mutual trust.
- There are no formal grades. Topics are not comparable in difficulty and effort. You either pass, with a grade that is neutral for you cum laude, or fail (when violating trust).

Topic List 2018

1. Other programming paradigms:
   - Haskell based on the material `http://learnyouahaskell.com/chapters`
   - Erlang based on material `https://learnyousomeerlang.com/content`
2. YesSql, using sql as intended in 2018. (Common table expressions, windows functions, analytic functions). All in postgreSQL.
3. 'Modern' data types in RDBMS, e.g. hstore, json(b) in postgresql. How to easily support with database built in functions.
4. Intro into NO-SQL, e.g. neo4j and (open) cypher intro, advantages and disadvantages.
5. Using web sockets in a JEE-8 container (payara or wildfly), maybe streaming video in browser?
6. Java 9: jigsaw.
7. Implementing Business workflow support with BPMB, using Camunda.
8. Use of OAuth2 and Json web token, prepare for easy use in context of 2nd semester project. (Postgres, JDBC, stored procedures where relevant).

How to get selected

- If you want to have your own topic, present a 5 minute pitch, give it next week.
- For all others: choose your topic from the list.
- Next week we will form groups.

TODO week1

- Get up and running with docker on you machine. Use the Linux Container variant, not windows containers.
  - On Linux, use docker package from distro. Docker can run 'natively' on your machine.
  - For Windows and OSX this implies the use of a virtual machine. Prefer to use VirtualBox.
- See `https://github.com/sebivenlo/dockerstart`