

Technische Universität Dresden

Software Engineering - Software Evolution im Sommersemester 2013 (30 Lehrstunden in englischer Sprache)

Referent: Harry M. Sneed (MPA)

Dates: Thursday and Friday, 27th, 28th June & 4th, 5th, 11th, 12th, 18th, 19th July with 4 double hours and two exercises per week plus a final exam on the last day 19th July.

Lecturing Schedule

Thursdays from 14:50 –18:10 and Fridays from 14.50-18:10.

Thursday, 27 June: Software Maintenance

1. Lecture: The Nature of Software Maintenance (MAIN)
2. Lecture: Software Maintenance Techniques (CHNG)
3. Lecture: Procedural Software Maintenance (PROC)
4. Lecture: Object-oriented Software Maintenance (OBJT)
1. Exercise: Revising an existing program (PLI or COBOL & C++, C# or Java)

Friday, 28 June: Software Reverse Engineering

5. Lecture: Software Evolution Modelling (EMOD)
6. Lecture: Software Reverse Engineering (REVE)
7. Lecture: Software Redocumentation Techniques (RDOC)
8. Lecture: Software Repository Management (REPO)
2. Exercise: Redocumenting an existing program (PLI or COBOL & C++, C# or Java)

Thursday, 4 July: Software Reengineering

9. Lecture: Technical Debt Prevention (DEBT)
10. Lecture: Software Reuse (REUS)
11. Lecture: Software Restructuring (REST)
12. Lecture: Software Refactoring (REFA)
3. Exercise: Reengineering an existing program (PLI or COBOL & C++, C# or Java)

Friday, 5 July: Software Retesting

13. Lecture: Software Regression Testing (REGR)
14. Lecture: Regression Testing Techniques (RETS)
15. Lecture: Regression Test Automation (AUTO)
16. Lecture: Retesting a migrated Application System (WKOT)
4. Exercise: Retesting a reengineered program (C++, C# or Java)

Thursday, 11 July: Software Evolution Management

17. Lecture: Software Evolution Laws (LAWS)

18. Lecture. Software Maintenance Organization (MORG)
19. Lecture: Software Maintenance Processes (MPRO)
20. Lecture: Software Product Line Management (SPLM)
- 5. Exercise: Defining a Software Evolution Process.**

Friday, 12 July: Software Evolution Cost Management

21. Lecture: Software Maintenance Economics (MECO)
22. Lecture: Value-Driven Software Maintenance (VALU)
23. Lecture: Software Product Measurement (MESS)
24. Lecture: Product Maintenance Cost Estimation (MEST)
- 6. Exercise: Estimating the costs of a new Release.**

Thursday, 18 July: Software Product Quality Management

25. Lecture: Software Quality Assurance (QUAL)
26. Lecture: Software Error Projection (EPRO)
27. Lecture: Software Product Measurement (MEAS)
28. Lecture: Software Product Evaluation (EVAL)
- 7. Exercise: Measuring and Evaluating Software Code Quality.**

Friday, 19 July: Software Service Management

29. Lecture: IT Service Management (ITSM)
30. Lecture: Software Problem Management (PMGT)
31. Lecture: Software Change Management (CMGT)
32. Lecture: Software Configuration Management (SWCM)

8. Exercise: An exam with 40 questions taken from the 300 self-control questions plus 4 exercises in code changing, redocumenting, refactoring and cost estimation. Each question is weighted with 2 points, each exercise with 5 points. The exam lasts circa one hour.

Exercises: Students will create teams of 2-3 persons. Each team solves two exercises per week and submits them in pdf format by the following Thursday to the instructor. The grade for the team is inherited by all members of that team. Exercise results should be sent to the following email address:

Harry.Sneed@t-online.de

Docent: Harry Sneed is currently working as a tester and auditor for the ANECON GmbH in Wien & Dresden and is teaching Software Engineering for Wirtschaftsinformatiker at the University of Regensburg, Software Maintenance, Test and Measurement for Phd students at the University of Szeged, Software Evolution and Product Management for the Fachhochschule Hagenberg and Test Automation and Software Metrics for the Fachhochschule Wien.

Seminar Material: Circa 25 power-point foils for each topic = 750 foils plus the 7 exercises with 6 code samples. To this comes a self-study check list of 300 questions on the subject of software evolution. This material will be made available to the students on electronic media. In addition students are given the **manuscript of Harry Sneed's latest book on Software Evolution** now being published by the dpunkt Verlag. As supplementary literature, the following books from Harry Sneed are recommended:

- Sneed, H.: Software Management, Rudolf Müller Verlag, Köln, 1987
- Sneed, H.: Software-Qualitätssicherung, Rudolf Müller Verlag, Köln, 1988

- Sneed, H.: Softwarewartung, Rudolf Müller Verlag, Köln, 1989
- Sneed, H. Softwaresanierung, Rudolf Müller Verlag, Köln, 1990
- Sneed, H.: Objektorientierte Softwaremigration, Addison-Wesley, Bonn, 1999
- Sneed, H./Winter, M.: Test objektorientierter Software, Hanser Verlag, 2001
- Sneed,H./Sneed,S.: Webbasierte Systemintegration, Vieweg Verlag, 2003
- Sneed, H./ Hasitschka, M./ Teichmann, M.-T.: Software Produktmanagement, dpunkt Verlag, Heidelberg, 2004
- Sneed, H.: Software Projektkalkulation, Hanser Verlag, München-Wien, 2005
- Sneed,H./Baumgartner, M./Seidl,R.: Der Systemtest, Hanser Verlag, München, 2006
- Sneed, H./ Wolf, E./ Heilmann, H.: Softwaremigration, dpunkt Verlag, Heidelberg, 2010
- Sneed, H.: Software in Zahlen, Hanser Verlag, München-Wien, 2011
- Sneed,H./ Winter, M. u.A.: Der Integrationstest, Hanser Verlag, München-Wien, 2012.

Die Klausurnote beträgt 60% der Gesamtnote. Die Gruppenarbeit beträgt 40% der Gesamtnote. Jeder in einer Arbeitsgruppe erhielt die Note der Gruppe. Die 7 Übungen sind von Gruppen von 2 bis 3 Personen zu erledigen.