Advanced Requirements Engineering (232085)
2003-2004

January 30, 2004

Contents

1 Course Structure

Size: 5 cr + n*5 credits with n = 0, 1, 2, ...

Advanced Requirements Engineering (ARE) is an optional course for BIT Masters students. ARE consists of a kernel of 5 credits, followed by anyone who chooses ARE, and a set of specialization subjects of 5 credits each. The student can choose to do any or none of these specializations, resulting in a total of 5, 10, 15, ... of credits for ARE.

The ARE kernel treats the methodology of applied research, which is research in which you try to solve action problems. An action problem is a problem in which stakeholders want to change something in the world. You learn how to identify action problems, identify requirements, and specify and evaluate a solution.

Each specialization part treats a different aspect of action problem analysis and solution specification.

2 Kernel: Analysis of action problems, 5 credits

We treat techniques for solving action problems:

- Difference between an action problem and a knowledge problem; recursive relation between the two; ideal types.

- Techniques for problem analysis
- Problem structures
- Research design of a knowledge problem

Examination consists of writing an essay that summarizes the techniques, and an application to an action problem (project) in which the student has been involved earlier.

Literature

[1] P. Verschuren and H. Doorewaard. *Designing a Research Project.* Lemma, 1999.

[2] P. Verschuren and H. Doorewaard. *Het Ontwerpen van een Onderzoek.* Lemma, 2003.

[3] R.J. Wieringa. Methodologies for requirements engineering research and practice. In V. Gervasi, D. Zowghi, S. Easterbrook, and S. E. Sim, editors, *Comparative Evaluation in Requirements Engineering (CERE'03)*, 2003. Workshop at 11th IEEE International Requirements Engineering Conference.

[4] R.J. Wieringa. Requirements engineering course handout. Technical report, University of Twente, 2003.

[5] R.J. Wieringa and H. Heerkens. Requirements engineering as problem analysis: Methodology and guidelines. Technical report, University of Twente, 2003.

[6] R.J. Wieringa and H. Heerkens. Research methodology in requirements engineering: A biased survey and analysis. Technical report, University of Twente, 2003.

3 Specialization: Analysis of ill-structured problems (5 cr)

Problems in real life always occur in problems and they are usually ill-structured or even messes. often, the problem is to find out what the problem is.

- Ill-structured problems, wicked problems, messes.
- Techniques for problem identification
- Techniques to analyze ill-structured problems
- Cognitive aspects of ill-structured problems

The student studies a selection of the literature. Examination consists of writing an essay that summarizes the techniques studied, and applies it to a problem bundle that the student has experienced in an earlier project he or she was involved in.

Literature

[1] P. Checkland and J. Scholes. *Soft Systems Methodology in Action.* Wiley, 1990.

[2] D. Dörner. *The Logic of Failure. Recognizing and Avoiding Error in Complex Situations.* Perseus Books, 1996. Translation of *Logik des Misslingens,* Rowohlt, 1989.

[3] D.C. Gause and G.M. Weinberg. *Are Your Lights On?* Dorset House, 1990.

[4] G.E. Lindblom. The science of 'muddling through'. *Public Administration Review*, 19:79–88, 1959.

[5] J.G. March. *A Primer on Decision-Making.* Free Press, 1994.

[6] W. M. Newman and M. G. Lamming. *Interactive System Design.* Addison-Wesley, 1995. Chapter 2 (Defining the Problem).

[7] G. Polya. *How to Solve it. A New Aspect of Mathematical Method.* Princeton University Press, second edition, 1985. First edition 1945. Pages 149-154 (Practical Problems).

[8] H.W.J. Rittel and M.M. Webber. Dilemmas in a general theory of planning. *Policy Sciences*, 4:155–169, 1973. Republished as "Planning problems are wicked problems", *Developments in Design Methodology*, N. Cross (ed.), Wiley, 1984. Pages 135—144.

[9] H.A. Simon. The structure of ill-structured problems. *Artificial Intelligence*, 4:181–201, 1973.

[10] A.B. VanGundy. *Techniques of Structured Problem Solving.* Van Nostrand Reinhold, 1988. 2nd edition.

4 Specialization: Problem Framing (5 cr)

Software design problems can be classified and structured according to problem frames. Each problem can be classified in different ways, and the interaction of these multiple classification is unpredictable. In other words, the frames are noncompositional. The student studies a number of problems frames and applies them to nontrivial examples, based on selected literature. Examination is by means of an application of problem frames to a nontrivial industrial problem, and writing a report about problem frames in general and this application in particular.

Literature

[1] M.A. Jackson. *Problem Frames: Analysing and Structuring Software Development Problems.* Addison-Wesley, 2000.

[2] B.L. Kovitz. *Practical Software Requirements: A Manual of Content and Style.* Manning, 1998.

5 Specialization: Goal-Oriented RE (5 cr)

Software requirements often are derived from business goals and problems. In this derivation, many patterns can be identified. In this specialization, students make a catalog of known goal patterns and investigate a number of real requirements projects to identify additional patterns. Students will also investigate the interaction of goal patterns in real projects.

Examination consists of writing an essay about the goal pattern catalog and a presentation of the results of applying this to real RE projects.

Literature

[1] A. Antón. *Goal Identification and Refinement in the Specification of Software-Based Information Systems.* PhD thesis, Georgia Institute of Technology, 1997.

[2] A.I. Antón, R.A. Carter, A. Dagnino, J.H. Dempster, and D.F. Siege. Deriving goals from a use-case based requirements specification. *Requirements Engineering*, 6:63–73, 2000.

[3] A.I. Antón and C. Potts. The use of goals to surface requirements for evolving systems. In *International Conference on Software Engineering (ICSE'98)*, pages 157–166. IEEE Computer Society, 1998.

[4] R. Darimont and A. van Lamsweerde. Formal refinement patterns for goal-driven requirements elaboration. In *Fourth ACM Symposium on the Foundations of Software Engineering (FSE4)*, pages 179–190, 1996.

[5] A. van Lamsweerde, R. Darimont, and E. Letier. Managing conflicts in goal-driven requirements engineering. *IEEE Transactions on Software Engineering*, 24(11):908–926, November 1998.

[6] A. van Lamsweerde, R. Darimont, and P. Massonet. Goal-directed elaboration of requirements for a meeting scheduler: problems and lessons learnt. In *RE95*, 1995.

6 Specialization: Quality attributes (5 cr)

Software products that compete in the same market are usually distinguished by quality attributes rather than by their functionality, which is largely similar in these products. Quality attributes are usually motivated from business processes and goals and from user requirements. Quality attributes are notoriously hard to operationalize and, even when operationalized, it is hard to know which design decisions affect which quality attributes.

In this specialization, the student studies a number of classifications and operationalizations of quality attributes, and for some selected attributes investigates which design decisions influence these attributes. This is validated by invesitagting the relationship between architecture and attributes in some real software products.

Examination consists of writing an essay that summarizes the literature and reports about relationships between architecture and attributes found by the student.

Literature

[1] L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos. *Non-Functional Requirements in Software Engineering*. Kluwer, 1999.

[2] D.C. Gause and G.M. Weinberg. *Exploring Requirements: Quality Before Design*. Dorset House Publishing, 1989.

[3] T. Gilb. *Principles of Software Engineering Management*. Addison-Wesley, 1988.

[4] IEEE Computer Society. *IEEE Standard for Software Quality Metrics Methodology*. IEEE Std 1061-1992.

[5] B. van Zeist, P. Hendriks, R. Paulussen, and J. Trienekens. *Kwaliteit van Softwareprodukten: Praktijkervaringen met een kwaliteitsmodel*. Kluwer Bedrijfswetenschappen, 1996.

7 Specialization: Requirements prioritization (5 cr)

If time and money would be unlimited, all requirements would be equally important. With limited time and budget, the requirements engineer must set priorities aming requirements. These priorities must be releted to preferences of the customer. In this specialization, the student studies prioritization techniques used in software and priduct engineering, and applies them to a number of selected products.

Examination consists of writing an essay summarizing the literature and reporting about the application to real products.

Literature

[1] L. Cohen. *Quality Function Deployment: How to Make QFD Work for You.* Addison-Wesley, 1995.

[2] D.C. Gause and G.M. Weinberg. *Exploring Requirements: Quality Before Design.* Dorset House Publishing, 1989.

[3] J.R. Hauser and D. Clausing. The house of quality. *Harvard Business review*, 66(3):63–73, May–June 1988.

[4] J.G. March. *A Primer on Decision-Making.* Free Press, 1994.

[5] N.F.M. Roozenburg and J. Eekels. *Product design: Fundamentals and Methods.* Wiley, 1995.

[6] K. Wiegers. *Software Requirements.* Microsoft Press, 1999. Chapter 13 (Setting Requirements Priorities).

8 Specialization: Systems and Product Engineering (5 cr)

Systems engineering is the discipline of the design of large systems, consisting of people, machines, organizations, and software. Systems engineering is the link between requirements engineering and project management. It supplies techniques to relate problems, goals and stakeholders to to solution properties and solution architectures, and to embed the solution process into a project. In this specialization, the student studies techniques for software systems engineering and industrial product engineering.

Examination consists of writing an essay that summarizes the literature and relates this to project management theory and project experience of the student.

Literature

[1] N.F.M. Roozenburg and J. Eekels. *Product design: Fundamentals and Methods.* Wiley, 1995.

[2] R. Stevens, P. Brook, K. Jackson, and S. Arnold. *Systems Engineering: Coping with Complexity.* Prentice-Hall, 1998.

9 Specialization: Task Modeling (5 cr)

Requirements of a software system often cannot be specified before the user has actually used the system. to escape this catch 22, the user should be observed in his or her daily work, and requirements should be specified based on these observations. This process should be repeated once the product is delivered to the user, so that requirements can be updated in the light of first experiences with the product. Requirements engineering is an evolutionary process of observing the user, identifying new requirements, and implementing these in the product. In this specialization, the student studies task modeling methods and applies them in a real project.

Examination consists in writing an essay about the literature and about its application to a real project.

Literature

[1] H. Beyer and K. Holtzblatt. *Contextual Design: Defining Customer-Centered Systems.* Morgan kaufmann, 1998.

[2] W. M. Newman and M. G. Lamming. *Interactive System Design.* Addison-Wesley, 1995.