



UNIVERSITY OF APPLIED SCIENCES ROTTERDAM / CMI

Methods and Techniques of Software Engineering

INFANL01-5 (FT) or INFANL225A (PT)

Number of european credit points: 4
Course Manager: Tanja Ubert Bsc RI

Approved by:

(on behalf of the exam committee)

Date:

Course description

Course name:	Methods and Techniques of Software Engineering (M&T of SE)
Course code:	INFANL01-5 FT (or INFANL225A (PT))
Number of european credit points and study hours:	This course is valued at 4 ECTS which is the equivalent of 112 hours of study load. This includes all scheduled classes, self-study, research, team and individual assignments etc.
Prerequisites:	First year projects.
Teaching methods:	Combined lectures and work groups.
Exam:	Assignment reports.
Learning tools and materials:	Software engineering, Ian Sommerville, Ninth edition, Publisher: Pearson education, 2011, ISBN-10: 0-13-705346-0, ISBN-13: 978-0-13--705346-9 This book is also available as an ebook: http://www.coursesmart.co.uk/IR/5192267/9780133032482?hdv=6.8
Contributes to competency:	<ol style="list-style-type: none"> 1. Managing, level 2 (Beheren, Niveau 2) 2. Consulting, level 2 (Adviseren, Niveau 2) 3. Designing, level 2 (Ontwerpen, Niveau 2)
Learning Goals:	<ul style="list-style-type: none"> - You can give a methodical justification of project approach and choose the solution appropriate methodology, operations and products - You will acquire basic knowledge about the most common principles and methods of system engineering - You will learn generic software development models including activities - You will get an introduction to software architectural principles and patterns - You will learn the relationship between requirements engineering, modelling and systems development. - You will acquire basic knowledge about software management approach.
Content:	<p>This course covers:</p> <ul style="list-style-type: none"> - Basic principles and methods of system development - Basic generic software development models and the associated activities. - An introduction to software architectural principles and patterns - Relationship with software engineering requirements engineering and modelling techniques. - Basics of software management - Agile Manifesto, Agile Principles
Remarks:	
Course manager:	T. Ubert Bsc RI
Date:	July 21st, 2015

1. General description

The course **INFANL01-5** in year two of our program is an introduction to commonly used methods for system and application engineering within the professional environment.

We discuss recognizing types of software and associated development approaches, requirements engineering, system modeling, system architecture, software design and implementation, software evolution, configuration management and process improvement.

The focus in this course is on the basic principles behind the methods.

1.1 Introduction

In professional software development it is not just important to create code. Code must also be produced as efficiently and reliably as possible. Software engineers must comply with high common standards and best practices. Because of the importance of software to businesses today the quality of code is vital to an organizations' success.

Using correct methods and techniques is essential to be a modern professional software engineer.

1.2 Relationship with other program units

This model aids this trimester's project.

1.3 Materials

Mandatory

- Software engineering, Ian Sommerville, Ninth edition, Publisher: Pearson education, 2011, ISBN-10: 0-13-705346-0, ISBN-13: 978-0-13--705346-9
This book is also available as an ebook:
http://www.coursesmart.co.uk/IR/5192267/9780133032482?_hdv=6.8
- All materials provided or prescribed via electronic learning environment (ELO), in classes or in lecture presentations.

Recommended

- Agile manifesto: <http://agilemanifesto.org/principles.html>

Additional literature and IT resources

The book has an extensive bibliography and pointers to Internet resources. The student is additionally made to the ACM Digital Library that is accessible to students via the [website of the library](#).

2. Program

In this chapter, the program is detailed including assignments per week.

Below is the program for all classes.

Week	Subject	Study	Hand in (Before deadline is appreciated)
1.02	Introduction Software Engineering	Chapter 1-3	
1.03	Requirements Engineering (Guest lecture/ workshop)	Chapter 4	Week 3 Monday before 13:00 h: Assignment 1
1.04	Requirements Engineering/ System Modelling	Chapter 5	
1.05	Architecture 1 (Guest lecture/ workshop or week 6)	Chapter 6	Week 5 Monday before 13:00 h: Assignment 2
1.06	Architecture 2 (Guest lecture/ workshop or week 5)	Chapter 19	
1.07	Design and Implementation	Chapter 7	Week 7 Monday before 13:00 h: Assignment 3
1.08	Software Evolution	Chapter 9	
1.09	Configuration Management and Process Improvement	Chapter 25-26	Week 9 Monday before 13:00 h: Assignment 4
1.10	Repair assignments meetings		Week 10 Monday before 13:00 h.

2.1 Rules for submitting assignments

Each assignment report is named in the following manner: <Education Abbreviation> <Class_group> <student number (s)> <name / name> Assignment <number> <short document description>. Also include this in the footer of the document. Add attachments to the document or give it a unique name using the same format.

Only students listed on the assignment products will be included in the grading list.

Documents with different naming conventions will NOT be checked and evaluated and considered "not handed in".

Assignments are submitted through n@tschool. Late submitted documents will NOT be checked and evaluated and considered "not handed in".

On the first page of the document state your **program**, **class_group** and the **assignment** and **a title**. Also add in table form a list of student numbers and names of the group members on the first page.

E-mail

For valid e-mail correspondence use the e-mail etiquette in the appendix. Emails that do not meet the etiquette, are considered not sent.

2.2 Assignments

Assignment 1 (20 points)

Software Engineering Basics

Study chapter 1-3 and use this for the next assignment parts

- A. Which *type* of software was project 4? (Ch. 1.1.2)
- B. Choose a different type of software from chapter 1.1.2 and describe how to adjust project 4 to become this type of software.
- C. What ethical dilemmas could be applicable to the 'mental health care system' described in the book (ch. 1.3.2).
- D. Describe the activities for project 4 according to Software process models "Waterfall", "Incremental" and "Re-use oriented" method. This is also called the project method in a professional setting.
- E. How would you (for every project method) add change management activities? Describe this.
- F. Research which parts of RUP could be applicable for your project, specify these and describe why or why not. If (a part of) RUP is not applicable, describe an alternative project management method and specify the activities needed for the project.

Combine your answers in a professional report. Use the correct form, table of contents, format etc. (See appendix for pointers). (Part of assignment 5).

Assignment 2 (12 points)

Requirements

Study chapter 4 and use this for the next assignment parts

- A. Describe the requirements for the project in 2 ways. Use the method in chapter 4.1 to 4.3 and use the method of user stories, use cases and use case descriptions (ch. 4.5.4). Use chapter 4.4 to 4.5.3 to detail the requirements properly.
- B. Use Ethnography at the project company (client) to get requirements (4.5.5).
- C. Compare methods in your group and translate the pro's and con's of those methods for the project group and the client.

Combine your answers in a professional report, fit to present to an external client. Use the correct form, table of contents, format etc. (Part of assignment 5).

Assignment 3 (12 points)

Architecture, Design and Testing

Study chapter 5, 6, 7 and 19 and use this for the next assignment parts

- A. Create the necessary models for the project using UML and use a repository based tool that covers all models (preferably with a code generation feature, like Astah). (Ch. 5)
- B. Create an architectural design for your project, edit your models from assignment 3.a to fit this design if necessary. Note your changes using change management. Use theory from chapter 6.
- C. Design the implementation plan to fit your project design. Note: do not forget to consider open source software.

Combine your answers in a professional report, fit to present to an external client. Use the correct form, table of contents, format etc. (Part of assignment 5).

Assignment 4 (20 points)

Software management

Study chapter 22.1, 25 en 26 and use this for the next assignment parts

- A. Do a risk analysis for the project and describe counter measures.
- B. Create a configuration management plan and apply this to your project.
- C. Describe how process improvement is applied in your project.
- D. Make a professional evaluation and reflection on the assignments for this course. Include also how theories and practices fit with your previous knowledge, the learning goals and the project for all group members.

Combine your answers in a professional report, fit to present to an external client. Use the correct form, table of contents, format etc. (Part of assignment 5).

Assignment 5 (4 points)

This assignment addresses the meticulousness of the report.

Assignment 6 (4 points)

Excellence. You can earn extra credit with the following type of actions:

- Out of the box (create original, innovative solutions)
- Use (if appropriate) humor, anecdotes, metaphors, etc.
- Write a catchy beginning and / or closing

3. Exams and grading

3.1 Procedure

This course is completed with a series of assignments. The totaled grade is your course grade. You can only get this grade if **all** assignments are handed in. Furthermore for **each** assignment *at least half of the possible credits* must be earned. If one of the assignments is not handed in a ND will be noted in the grading system (Osiris). The pass mark (cesuur) of this course is 5,5 (43 points). Criteria can be found in the assessment form appendix.

Practical assignments hand in

During the practical you show your work to the lecturer and you clarify your work. (**All** members of the group need to attend). Every member of the project must be able to clarify **all** parts of the assignment report. So prepare with your group before showing your work.

To test your knowledge and insights the lecturer can ask you to do an extra assignment during class. If necessary the lecturer can book an additional meeting with the group to talk about the assignment report. The whole group needs to attend this meeting.

You get feedback from the lecturer and take notes on this during feedback. You will apply the feedback and hand in the notes (scan) and revised assignment in N@tschool in. The complete set of assignments will be graded to get the final course grade.

The appendix includes the assessment form. It can also be found on N@tschool.

3.2 Repair assignments

Repair of assignments will be done according to the Education Examination Regulations in the Study Guide of our program. If your assignment report is insufficient, the lecturer can define a repair assignment. You can get a maximum of **3** repair assignments. If you fail the repair assignments you must retake this course, next year according to the schedule.

3.2 Inspection of assignments

Inspection of assignments is done during the practicals. If necessary, a consult hour will be planned to discuss the last assignment.

Appendix 1: Exam matrix

The exam matrix reflects the relationship between the learning objectives and the questions or assignments of a test. The exam matrix also indicates the level at which the learning outcome is tested. The program has chosen to use the HBO-I guidelines for this and has developed competency levels.

	Learning goals	Competency / Level	Reference to assignment / query / criteria
1.	You can give a methodical justification of project approach and choose the solution appropriate methodology, operations and products	Management and consulting/ level 2	Assignment 1
2.	You will acquire basic knowledge about the most common principles and methods of system engineering	Management/ level 2	Assignment 1
3.	You will learn generic software development models including activities	Designing/ level 2	Assignment 2
4.	You will get an introduction to software architectural principles and patterns	Designing/ level 2	Assignment 3
5.	You will learn the relationship between requirements engineering, modelling and systems development.	Designing/ level 2	Assignment 3
6.	You will acquire basic knowledge about software management approach.	Designing/ level 2	Assignment 4

Appendix 2: assessment form

Assessment form INFANL01-5 of INFANL225A Methods SE

Group name:

Student names/ student numbers:

*

*

*

*

Circle for each aspect the number of credits in the category column of your judgment

	bad	insufficient	sufficient	good	CREDITS
1. a: Which type of application was project 4? - Is the theory of Ch. 1.1.2 used? - Are the right arguments used? Yes: 4 points, No. 1, wrong type, and incorrect arguments: 2, right type, no arguments: 3, right type, insufficient arguments: 3	1	2	3	4	
b: Choose a different type of application from chapter 1.1.2 and show how you edit your project 4 to become this chosen type of application - Is the theory of Ch. 1.1.2 used? - Are the right arguments used? Yes: 4 points, No. 1, wrong type, and incorrect arguments: 2, right type, no arguments: 3, right type, insufficient arguments: 3	1	2	3	4	
c: What ethical dilemmas might play in the "mental health care system" from the book. - Is the theory of ch. 2. used? - Is the Software Engineering Code of Ethics and Professional Practice (INT) researched and used? Or the Dutch "richtlijnen van het VRI"? Yes: 4 points, No. 1, wrong type, and incorrect arguments: 2, right type, no arguments: 3, right type, insufficient arguments: 3	1	2	3	4	
d: In Ch 2.1. shows the software process models: "Waterfall", "Incremental" and "Re-use oriented" method. Describe, for each model, per phase, the activities apply to project 4. (Motive)	1	2	3	4	
- Is the theory of ch. 2.1. used? - Did they describe the correct activities? Yes: 4 points, No. 1, incorrect activities: 2 correct activities for 2 models: 3	1	2	3	4	
e: How would you, for each process model, add, change management activities? (How does the process model deal with change, it is handled (how)? If not, what should be added?) - Is theory from ch. 2.3 used? - Did they describe the correct activities? Yes: 4 points, No. 1, incorrect activities: 2 correct activities for 2 models: 3	1	2	3	4	
2. a: Describe the requirements for the project in two ways. - Is the method described in Chapter 4.1 / 4.3 with user stories, use cases and use case descriptions (4.5.4) used? (USER STORY) - Is the method in chapter 4.4 t/m 4.5.3 used to the requirements clear? (SRS) - All requirements prioritized? Yes: 4 points, No. 1, incorrect method: 2, requirements not SMART: 3	1	2	3	4	
b: Use at the company, Ethnography method to determine requirements. (Ch. 4.5.5). - Have the processes described actually take place. - Requirements that are derived from cooperation and awareness of activities of other people. For example, the controllers can be aware of the work of other controllers to predict the number of aircraft that will enter their control sector. Yes: 4 points, No: 1, described the wrong set of processes: 2, the right processes, but not related to the environment: 3	1	2	3	4	
c: Compare methods with the group and translate what the advantages and disadvantages of both methods are for the project and the client. - Proper characteristics to compare selected? - Comparison valid? Yes: 4 points, No. 1, not the correct characteristics selected: 2, opted to choose insufficient characteristics for valid comparison: 3	1	2	3	4	
3. a: Create UML models needed for the project. Use a tool that supports all the models with a repository and a code generation function (return). (for example, Astah). - Knows Schedule Techniques	1	2	3	4	

<ul style="list-style-type: none"> - Schedule technique applied correctly - Repository based tool used <p>Yes: 4 points, No: 1, UML misapplied: 2 does not use UML, but applied correctly: 3</p>					
<p>b: Create an architectural design for the project. Edit the models created to match this design. Identify what your changes, by means of change management. Use the instructions in chapter 6.</p> <ul style="list-style-type: none"> - Appropriate architectural designs. - Model adapted from assignment 3a based on architecture requirements, if necessary? (Argue why or why not). - Change management from Chapter 6 correctly applied? <p>Yes: 4 points, No: 1, no appropriate architecture: 2, no adjustments were made, while chosen this architecture does require 3</p>		1	2	3	4
<p>c: Create an appropriate implementation plan (Chapter 7.3) for the design of the project. Don't forget to bring open source software into consideration.</p> <ul style="list-style-type: none"> - Is the project implementation plan complete? - If not, are adjustments to the standard format properly motivated? <p>Yes: 4 points, No: 1, With proper project implementation: 2 suitable plan, not always correct or incomplete motivation for choices: 3</p>		1	2	3	4
<p>4. a: Create a risk assessment for the project and describe how you will manage this risk (Ch. 2.1).</p> <ul style="list-style-type: none"> - Risk identification on project related properties. (NOT project member becomes ill and the like) - Risk analysis and classification - Risk decomposition - Risk reduction <p>Yes: 4 points, No: 1, only identification: 2, developed more than 2 of the risk aspects: 3</p>		1	2	3	4
<p>b: Create a configuration management plan and apply it to the project.</p> <ul style="list-style-type: none"> - Recognized configuration management (CM), and it may be locations within the used method, or may indicate which method could be used in a given situation. - Past CM or right to indicate how it could be applied correctly. <p>Yes: 4 points, No: 1, UML misapplied: 2 does not use UML, but applied correctly: 3</p>		1	2	3	4
<p>c: Describe how process improvement is a part of the project.</p> <ul style="list-style-type: none"> - Recognize the process method used or could be used (motivate). - Describe how it is applied (or should be applied) <p>Yes: 4 points, No: 1, Method not recognized: 2 recognized, not properly applied: 3</p>		1	2	3	4
<p>d: Provide a professional evaluation/ reflection on the process of all four previous assignments. Also include how lessons learned are in line with previously acquired knowledge, the learning objectives and the project, for all group members.</p> <ul style="list-style-type: none"> - Professional critical attitude? (what activities were going well, why, what are you going to do again? What is to be improved, what will you do differently next time? Do you have tips and tips for teammates, where you or they will benefit from to actually become better SE's). - Aware of learned material? - Can apply this knowledge in the context of study and (future) career? <p>Yes: 4 points, No: 1, not critical enough: 2, cannot relate knowledge to career and study</p>		1	2	3	4
<p>5. Meticulousness of the report?</p> <p>Things such as:</p> <ul style="list-style-type: none"> - Introduction group - Table of Contents, cover sheet, etc. Chapter structure is logical - Essence is clear. Business like writing, flawless use of language. Concisely written. 		1	2	3	4
<p>6. Excellent</p> <p>Things such as:</p> <ul style="list-style-type: none"> - Think "outside the box" (original, innovative) - Use, if possible, humor, anecdotes, metaphors, etc. - Catchy start and (or) slot 		1	2	3	4
Total					0

Course grade:

Grade matrix

Minimum credits	10
Maximum credits	50
Cesuur	25

Credits	Grade	Credits	Grade
17	1	43	55
18	3	44	57
19	5	45	59
20	7	46	60
21	9	47	62
22	12	48	64
23	14	49	66
24	16	50	68
25	18	51	69
26	20	52	71
27	22	53	73
28	24	54	75
29	26	55	77
30	28	56	78
31	30	57	80
32	33	58	82
33	35	59	84
34	37	60	86
35	39	61	87
36	41	62	89
37	43	63	91
38	45	64	93
39	47	65	95
40	49	66	97
41	51	67	99
42	53	68	100

Appendix 3: Study load (standard in ECTS)

Only fill the gray shaded boxes

	Number of weeks	Number of class hours of 50 minutes	Clock hours
<u>Class hours</u>	8	3	24
<u>Self-study</u>			
Reading time	Number of pages		
		3 per hour	0
		6 per hour	0
	312	10 per hour	31
Presentations			
Discussion/ meeting time			
Research time			14
Not scheduled lesson time			
<u>Exam(s)</u>	Prepare Exam		
	Reflection session		
<u>Assignment, report, report, thesis</u>	Research		
	Discussion/ meeting writing		6
<u>Internship Practical assignment</u>	preparation attendance meetings		
Subtotal in clock hours			75
Noise 5%			37,5
Total in clock hours			112,5
Total in european credit points (ects)		here only multiples of 28 hours	4

Explanation of study load (standard in ECTS)*Class hours*

scheduled participation in education related to the unit, such as lessons, exercises, training, working groups (with or without teachers) etc.

Self-study:

NOTE: This is the time that the student spent on education outside the contact hours

Reading time:

Three to ten pages per clock hour (depending on complexity, language, layout, font size)

Presentations:

The actual estimated time involved with preparing presentation activities; The presentation itself is already included in the lessons

Discussion meeting time:

The actual estimated time involved in meeting activities

Research time:

The actual estimated time involved in research activities.

Tests and exams (including preparation, taking the test or exam and feed back)

Actual estimated time

At feed back for tests and exams:

Every student has the right to view/ feed back tests and exams. Schedule at least 1 hour of feed back time.

Creating/ writing papers, reports, theses etc:

The estimated real time accounted on the planned learning output

Internship or other Practical assignment:

Actual estimated time (including transfers) for internships or other practical assignments in a working environment

'Noise Time" and other times:

5% of the study load