EXAM METHOD ENGINEERING
ANSWERS

March 1, 2011

11:00 –13:00

Educ-Alfa

NAME:	STUDENTNR.:

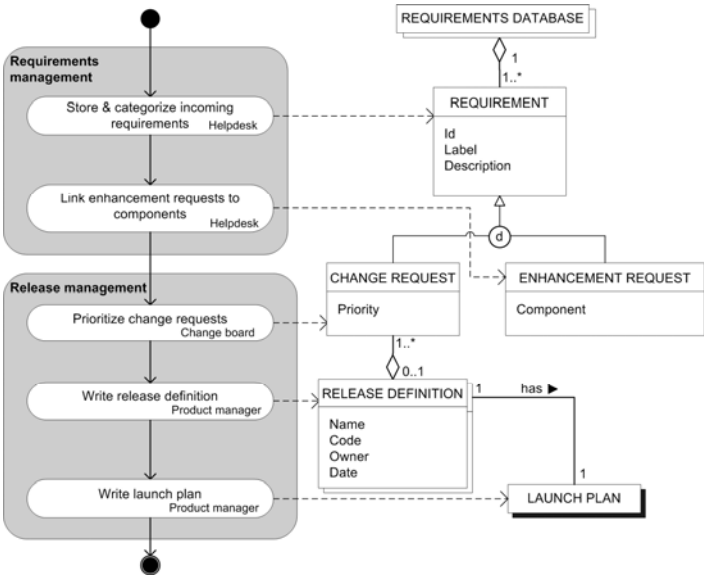
- This exam consists of 5 questions on 11 pages. Please first check whether you have properly obtained **all** pages.
- Enter the answers in the space allocated. In case you need more space you can use the back of the pages. Make a proper reference to such an extra part on the back.
- When you have finished the exam you should submit the complete package stapled in the correct order.
- The results of the exam will be communicated to you through the website of the course as soon as possible.

1	12	
2	15	
3	15	
4	28	
5	30	
Total	100	
Grade		

Good luck!

1. PDD constructs (12 points)

Given the following process-deliverable diagram:



a. What are the open and closed concept(s) in this process-deliverable diagram?
(3 points)

Open: Requirements database, Release definition
Closed: Launch plan

b. Explain the notions of closed and open concepts.
(3 points)

Open concept: subconcepts are known in this context: i.e. the meta-model details can be found in this or another meta-model of the method engineering study at hand.
Closed concept: subconcepts are not known or not relevant in this context

c. What does the 'd' in the generalization-construct mean? Explain.
(3 points)

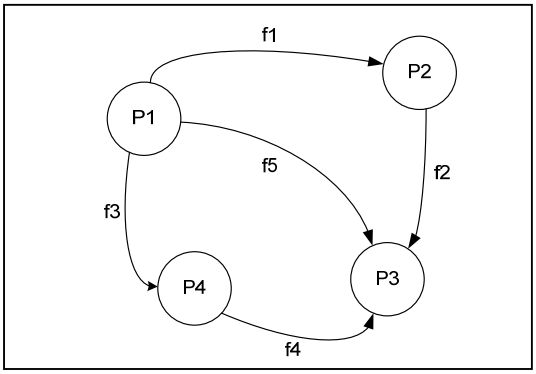
Disjoint: occurrence of one concept is incompatible with the occurrence of the other concept

d. Name and explain the two other ways to specify the type of generalization.
(3 points)

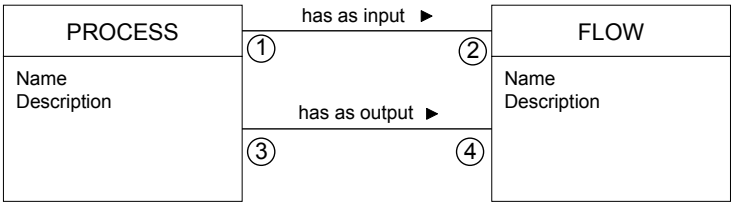
o = overlapping: occurence sof the concepts may overlap with the occurences of the other concept(s)
c = categories: the disjoint concepts have no occurrences in common and that the decomposition is exhaustive

2. Multiplicity (15 points, 3 points per sub question, for a-d: 1 error = -2, 2 errors = -3)

Shown below is a generic example of a Data Flow diagram. This Data Flow diagram only has Processes and Flows, and no Stores, nor Control flows.



The following diagram is an incomplete meta-model of a Data Flow diagram:



There are four standard multiplicity values for associations in a meta-model: **0..1** , **0..*** , **1** and **1...*** . Give the position (1, 2, 3 or 4) and value of the multiplicity for the following rules.

a. A flow is output from precisely one process.
   Position: (**3)**
   Value: **1**

b. A flow may point to at most one process.
   Position: (**1)**
   Value: **0..1**

c. A process may have any number of input flows.
   Position: **(2)**
   Value: **0..***

d. A process may have any number of output flows.
   Position: **(4)**
   Value: **0..***

e. Rule c and d imply that dangling processes are allowed, i.e. processes without input flows and output flows. Suppose we want to forbid dangling processes in Data Flow diagrams. What should be changed or added to the meta-model to include this?
   Impossible to model, so add an additional textual rule: A process has <u>at least</u> one input <u>or</u> one output flow.

3. Meta-data modeling (15 points)

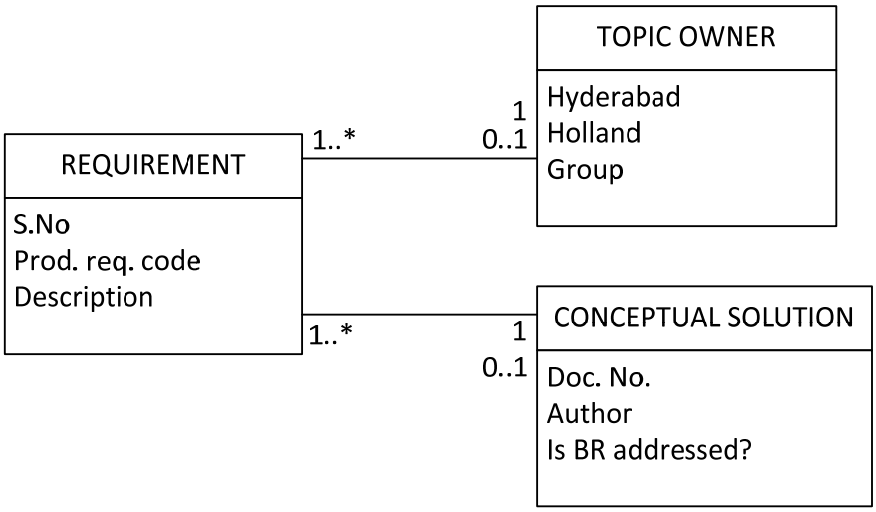
Below, a traceability matrix for tracing product requirements is presented. In this matrix, product requirements are linked to topic owner and conceptual solution.

Traceability between Product Requirement & Conceptual Solution								
S.No	Product Requirement	Description	Topic Owner			Conceptual Solution		
			Hyderabad	Holland	Group	Doc. No.	Author	Is BR addressed?
1	PR1-101002	Project Estimation	Vijay	Ernico	TPH	D1295 US	Vijay	No
2	PR1-101244	Bottom-up estimating with amounts, without the use of cost objects	Vijay	Ernico	TPH	D1296A US	Rao	Yes
3	PR1-101245	Both types of estimating within one estimate	Kumar	Ernico	TPH	D1296A US	Rao	Yes
4	PR1-101247	Schedule estimation	Vijay	de Bruin	SGB	D1296B US	Rao	Yes
5	PR1-101012	Integration Project with Microsoft Excel	Vijay	de Bruin	SGB	D1296B US	Rao	Yes

Create the meta-data model for this traceability matrix, consisting of concepts and their properties, and the relationships between these concepts. You do not have to include the concept table.

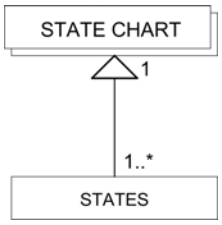
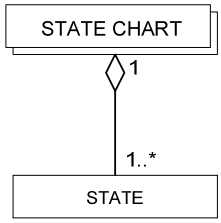
4 points for correct concepts
3 points for correct properties
4 points for correct associations
4 points for correct multiplicity (multiple correct answers)

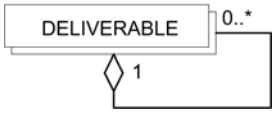
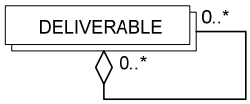
for each unnecessary construct: -2
for small errors: -1
no capitals: -1

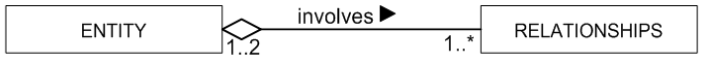
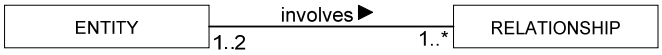


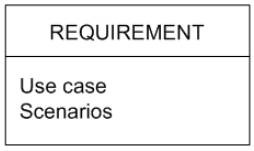
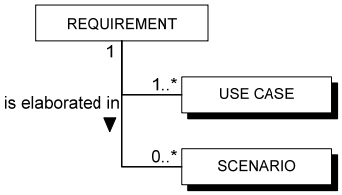
4. Find the error (28 point, 4 per sub question (2 for error(s), 2 for correct fragment))

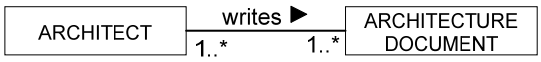
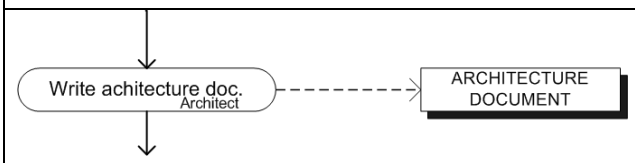
In this assignment, we show some method fragments, containing one or more errors. Explain every error, and draw a correct version of the method fragment next to the erroneous one.

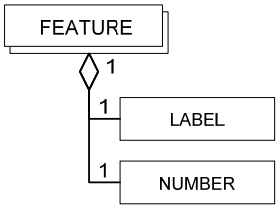
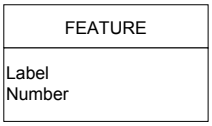
a.	Error(s)	Correct fragment
	• **Generalisation is used, instead of aggregation** • **STATES instead of STATE**	

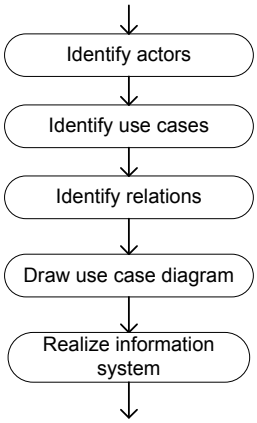
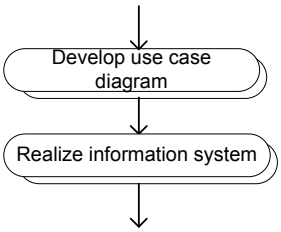
b.	**Error(s)**	Correct fragment
	• **Recursive until infinity, because a DELIVERABLE is always part of another DELIVERABLE (due to the '1' multiplicity)**	

c.	Error(s)
	• **Incorrect use of the aggregation diamond** • **It should be RELATIONSHIP instead of RELATIONSHIPS.** • **Optional: arrow wrong direction**
Correct fragment 	

d.	Error(s)	Correct fragment
	• **Use case and scenarios are not attributes: they are to complex, and there could be a 1 to many relation, which is not modeled now.**	

e.	Error(s)
	• **Roles should not be modeled in the deliverable side**
Correct fragment	
	

f.	Error(s)	Correct fragment
	• **1 to 1 relation → label and number should be attributes**	

g.	Error(s)	Correct fragment
	• **Unbalanced activities**	

5. Creating a PDD (30 points)

In this assignment, we show a typical example of how software companies handle change requests.

Change analysis

Change management is concerned with the processes that are used to manage changes to system requirements. An important part of change management is the change analysis. There are seven basic activities in the change analysis process:

1. The change request is submitted by a stakeholder. It can be in the form of an email, phone call, meeting, etc. How this is documented, depends on the submitter.
2. The requirements engineer checks the change request to see if it is valid. Sometimes, stakeholders misunderstand the requirements and suggest unnecessary changes. In that case, the change request is rejected.
3. If the change request is valid, a change request form is created by the requirements engineer.
4. The requirements engineer lists the requirements that are directly affected by the change. This activity results in a requirements change list consisting of directly affected requirements.
5. The requirements engineer uses traceability information to find dependent requirements which may also be affected by the change. The dependent requirements are also added to the requirements change list.
6. The requirements engineer proposes the actual changes that must be made to the requirement.
7. The project manager estimates the costs of making the changes. This estimate should include both the effort required to make the change and the amount of calendar time needed. The availability of resources to implement the change must also be considered.
8. Negotiations with the requirements engineer, project manager and the stakeholders are held to check if the costs of the proposed changes are acceptable to them. At this stage, it may be necessary to go back to step 5, to propose alternative changes if the stakeholder feels that the change proposal is too expensive. Alternatively, the stakeholder may modify the CR so that the whole process has to be repeated.

During the change management process, information about the change and the system is documented in a change request form. Essential parts of a change request form are:

- Change request id
- Submitter
- Description of the change request
- Date that shows when the change request was submitted
- A list of directly affected and dependent requirements
- A status, which may have values such as ‘rejected, ‘under consideration’ etc.

a. List the sub activities that are carried out in the Change Analysis activity. Also, list the role(s) per sub activity.

6 points, 3 for roles, 3 for sub activities. Each error/missing: -1

Role(s)	Sub activity
Stakeholders	**Submit change request**
Requirements eng.	**Check change request**
Requirements eng.	**Create change request form**
Requirements eng.	**List affected requirements**
Requirements eng.	**Find dependant requirements**
Requirements eng.	**Propose changes**
Project manager	**Estimate costs**
Requirements eng., Project manager & Stakeholders	**Negotiate proposed changes**

b. The concept CHANGE REQUEST FORM consists of sub concept(s) and properties. Explain which parts of this document should be modeled as properties and which should be modeled as (a) separate sub concept(s).

6 points, 3 for properties and 3 for sub concept. Each error/missing: -1

Properties	Explanation
Change request id	**This is just a field in the document.**
Submitter	**Idem**
Description	**Idem**
Date submitted	**Idem**
Status	**Idem**

Sub concept(s)	Explanation
List of directly affected and dependent requirements	**This concept consists of other sub concepts (requirements) and can therefore not be modelled as a property.**

c. List all concepts that can be derived from the text. Describe for each concept whether you think it is a standard, open or closed concept. Explain.

6 points, each error/missing: -1

Concept	Type	Explanation
Change request	**closed/ standard**	**Structure unknown. How the CHANGE REQUEST is documented, depends on the submitter.**
Change request form	**open**	**It has a sub concept (REQUIREMENTS CHANGE LIST)**
Requirements change list	**open**	**It has a sub concept (REQUIREMENT)**
Requirement	**standard**	**It has no sub concepts.**

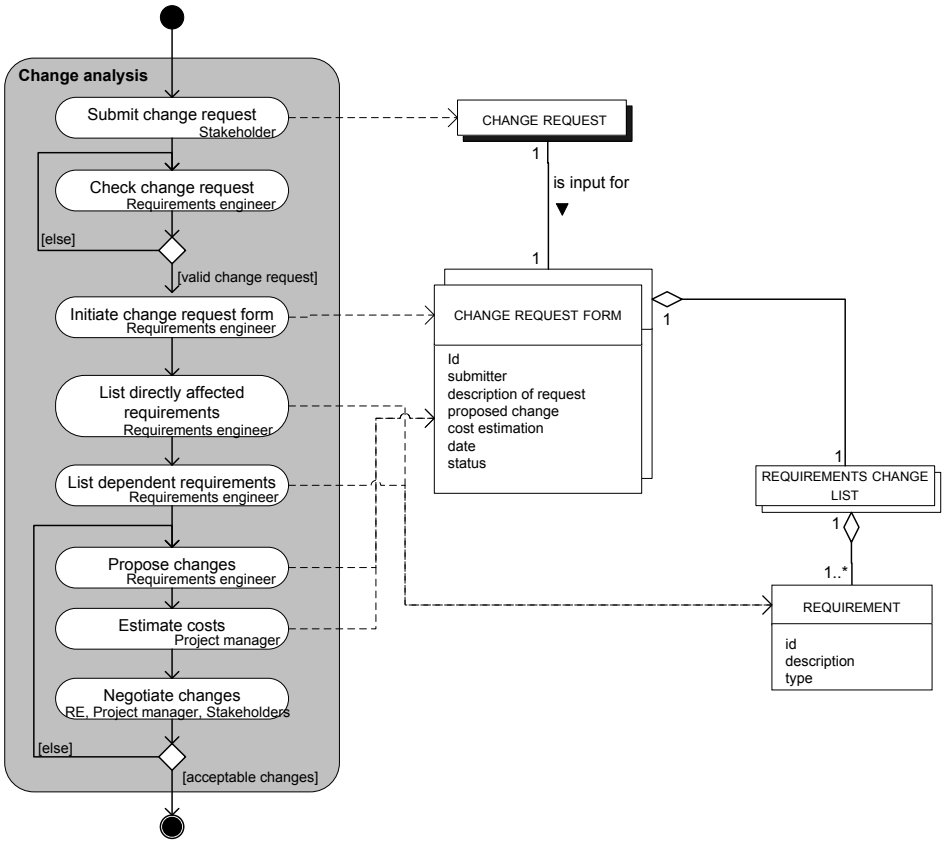
It is also correct to list the two concepts DEPENDENT REQUIREMENTS and AFFECTED REQUIREMENTS, which are then connected to REQUIREMENT with generalisation.

d. Translate the Change Analysis text into a process-deliverable diagram. Make good use of activities, sub activities (sequential, unordered, concurrent or conditional), initial and final states, and roles. Model the concepts (standard, open or closed), concept properties, and the relations between the different concepts (generalization, association or aggregation). It is allowed to abbreviate role names.

12 points:

- **6 points for the process side (2p for (sub) activities, 2p for conditions, 2 p for roles)**
- **6 points for the deliverable side (2 p for correct concepts & properties, 2p for correct relations) & 2 point for the correct output connections.**

For each error: -1



Also correct:

- **Inclusion of the concept Cost estimation**
- **Inclusion of different types of requirements**
- **Inclusion of extra decision moment**