DAT230/DAT231/DIT276 Requirements Engineering

Exam

Monday, October 29th, 2018

Allowed tools / material
- Pen/pencil
- Eraser
- English dictionary

General information
Numbers within parentheses show the maximal points awarded for each question. Maximal points can be given if:
- The answer is correct.
- The presentation of the answer is readable and clear.
- The answer is given in English.

Each question must be answered on a new sheet of paper. Hence, the answers to Question 1 and 2 must NOT be written on the same sheet of paper. You are NOT allowed to write answers to questions on the backside of a sheet of paper!

Grading
The grades on this exam are based on your total score on the questions. For Chalmers students:
0 – 39   points: Fail
40 – 55 points: 3
56 – 67 points: 4
68 – 80 points: 5

For GU students:
0 – 39 points: Fail
40 – 67 points: G (Pass)
68 – 80 points: VG (Pass with distinction)

Review
Will be announced on Ping Pong

Institutionen för Data- och informationsteknik
CHALMERS TEKNISKA HÖGSKOLA & GÖTEBORGS UNIVERSITET

Part 1 – Multiple Choice (30p)

Question 1: Multiple-choice problems (30p)

This part consists of multiple-choice problems. These problems consist of pairs of *propositions* and *reasons*. For each problem, you shall answer by using one of the following answers:

A Both the proposition and the reason are correct statements. In addition, the reason explains the proposition in a correct way, i.e. the reason explains why the proposition is correct.

B Both the proposition and the reason are correct statements, but the reason does not explain the proposition.

C The proposition is a true statement, but the reason is false.

D The proposition is false, but the reason is a true statement.

E Both the proposition and the reason are false.

Correctly answered problems give **1 point**, while incorrect or missing answers give 0 points, regardless if you are partially correct in your answer!

	Proposition	*Reason*	*Answer A|B|C|D|E*
1) (1p)	Requirements engineering is a systematic approach to reduce the likelihood to develop the wrong system, e.g. one that does not solve the problem of a user.	A requirement is a condition or capability needed by a user to solve a problem or achieve an objective.	
2) (1p)	With the term "tacit requirements" we refer to requirements that should not be part of a specification.	Tacit knowledge is knowledge that a stakeholder takes for granted or for other reasons does not explicitly mention.	
3) (1p)	Symmetry of ignorance is a challenge of requirements elicitation that makes it difficult to map a problem to a solution.	Symmetry of ignorance refers to the fact that requirements engineers should ignore solutions and focus on problems instead.	
4) (1p)	When a customer is inexperienced in a domain and a supplier is experienced a domain, the main goal of requirements engineering should be to reduce cost.	Since the supplier is more experienced, elicitation of customer requirements does not provide value and the potential to satisfy the customer can be maximized if the supplier is just providing their solution directly.	
5) (1p)	Identification of stakeholders is an important part of requirements elicitation.	Stakeholder needs must be analyzed and refined. Without that, they are considered to be "raw requirements", not necessarily being requirements according to definition.	

6) **(1p)**	If a stakeholder appears to talk about too detailed requirements, we should ask "why" questions.	"Why" questions raise the level of abstraction and allow to discuss the underlying problem.	
7) **(1p)**	If a stakeholder uses unexpected or ambiguous wording during an interview, this should be ignored during the interview.	Missing information from an interview can easily be provided during analysis.	
8) **(1p)**	It is crucial to satisfy some issues from each stakeholder group when processing the raised issues from focus groups.	All stakeholder groups have to feel that they get something.	
9) **(1p)**	Questionnaires are a good elicitation technique in the early phases of a project.	They are easy to construct and easy to interpret.	
10) **(1p)**	Verbose requirements should be avoided.	Verbose requirements are hard to read and often not thought through.	
11) **(1p)**	Component names, database fields, and similar details in a requirement indicate that it is problem-based.	Considerations of solutions and SW design should be avoided when specifying requirements.	
12) **(1p)**	Context diagrams are useful for defining the project scope.	Context diagrams show the system under consideration in relation to other external entities such as other systems, datastores, or stakeholders. By showing critical interfaces it can define the context for use cases and task descriptions.	
13) **(1p)**	Feature requirements can lead to a false sense of security.	Most stakeholders think that they understand feature requirements, but often, they misinterpret them.	
14) **(1p)**	Feature requirements are useful when managing requirements for a large system.	Feature requirements are relatively short and independent descriptions of product functions and related data. It is easy to store them in databases and they can have many administrative attributes.	
15) **(1p)**	Task descriptions describe the interaction between a user and a system in detail.	The main scenario of a task descriptions usually alternates steps that a user executes with responses or actions that the system executes.	
16) **(1p)**	"Enter guest name" is a good task description.	Good task descriptions are closed (they end with a reaching a goal, with a pleasant feeling) and describe one session (a small set of related tasks).	
17) **(1p)**	All non-functional requirements are quality requirements.	A non-functional requirement is either an attribute or a constraint of a system.	
18) **(1p)**	Quality requirements are an exception to the rule that all requirements should be verifiable.	Since quality requirements are cross-cutting, criteria about whether they are met is better related only to the functional requirements they affect.	

19) (1p)	The quality grid is useful to describe trade-offs between different quality requirements.	The quality grid provides a matrix of the relevant quality requirements. This allows to define trade-offs between two quality requirements in the cells.	
20) (1p)	When relying on "Open Target" when specifying quality requirements, the customer indicates that the degree of fulfillment of this quality requirement is not important to them.	It is often difficult to decide the value needed for some quality attribute. "Open Target" gives the supplier the responsibility to investigate a suitable target value.	
21) (1p)	"Combinatorial creativity" combines both exploratory and transformational creativity.	A combinatorial creativity technique is generally a two-phase approach, where first, ideas are generated based on an exploratory technique, before these ideas are then discussed related to a transformed solution space.	
22) (1p)	Requirements validation relates to the question "Are we building the right system".	Requirements validation is defined as the process of checking whether documented requirements match the stakeholders' needs.	
23) (1p)	Since it is important that all requirements are verifiable, it is implied that all requirements must be atomic.	If requirements were not clearly identifiable and would not allow to reason about them individually, it would be impossible to state whether they are met.	
24) (1p)	Consistency checks are done to compare different parts of a requirements specification, looking for inconsistencies.	CRUD checks are done to assess whether each entity in the data model is created, read, updated, and deleted, by an event, or a user task.	
25) (1p)	A prototype is only useful for verification, but cannot be used for validation.	A prototype only allows to answer whether we are building "the system correctly", but not whether we are "building the right system".	
26) (1p)	Relying on face-to-face communication instead of written specifications can lead to incomplete and wrong understanding of requirements.	Agile projects are not always able to achieve high-quality interaction with the customer organization.	
27) (1p)	Agile methods promise to mitigate communication gaps in requirements engineering.	Agile RE promotes regular interaction with customer and between teams.	
28) (1p)	In large-scale agile development, most requirements analysis takes place on program level.	In large-scale agile, requirements analysis is spread out over portfolio, program, and team level activities.	
29) (1p)	It is important to elaborate and document each user story in detail.	A user story is the main mechanism to specify requirements in agile methods.	
30) (1p)	Requirements engineering relates to the trade-off between common product goals and autonomy of agile teams.	The fine balance between decisions on portfolio, program and team level must be maintained through strong communication of requirements-related knowledge.	

Part 2 – Applying RE Knowledge (10p)

Question 2. Usability requirements and risk (10p)

Ms. Taylor has been in contact with Requirements Engineered Ltd., a company specialized in Requirements Engineering and bespoke development, to realize one of her project ideas. The company has worked for several months to come up with a specification. Now, Ms. Taylor would like to move to the next step and sign a contract for development of the resulting specification. She has asked you, as an expert in Requirements Engineering, to give her a second opinion on the following requirements.

R1	At most 1 of 5 novices shall encounter critical problems during tasks Q and R. At most 5 medium problems shall be encountered in total.
R2	Novice users shall perform tasks Q and R in 15 minutes. Experienced users shall perform tasks Q, R, and S in 2 minutes.
R3	Recording breakfast shall be possible with 5 keystrokes per guest. No mouse.
R4	80% of users shall find system easy to learn. 60% shall recommend system to others.
R5	Show 5 users 10 common error messages, e.g. Amount too large. Ask for the cause. 80% of the answers shall be correct.

Which issues and risks do you see in these requirements? Name one issue/risk per requirement, argue whether it applies to supplier or customer, and motivate why you think it is problematic/risky.

Part 3 – Essays (40p)

For each topic below, write an essay that includes all of the listed concepts within the given maximum number of pages. The essays are graded based on

(a) how well the topic is described using the concepts in the list under each topic (thus judging the general flow and argumentation), and
(b) how well the concepts in the list are described and/or exemplified in the essay (thus, points are given for each concept that is mentioned and explained in sufficient depth).

Please make an effort to write readable. Essays that are difficult to read or difficult to understand will render a deduction of received points. An unreadable essay will receive zero points! Each essay should start on a new sheet of paper.

Question 3: Topic: Completeness of Elicitation **(10p)**
Max 1 page!
List of concepts: 1 example of an early-phase elicitation technique, 1 example of a mid-phase elicitation technique, 1 example of a late-phase elicitation technique, tacit knowledge, effect of asking why

Question 4: Topic: Understanding Quality Requirements **(10p)**
Max 1 page!
List of concepts: Quality Grid, Planguage, Open Metric, Constraint, Attribute

Question 5: Checking and Validation **(10p)**
Max 1 page!
List of concepts: Contents check, Structure check, Consistency checks, Reviews, Tests

Question 6: Quality Requirements in Agile Requirements Engineering **(10p)**
Max 1 page!
List of concepts: Portfolio level, program level, team level, agile estimation, architecture