

8 questions

+ Many good questions

- (1) Assuming you mean project type COTS purchase, not developing a COTS product. (7) User acceptance is also important, since it leads to success in the listed factors, i.e. indirectly important.

•[Lau:5,7] 1 problem (Special interfaces, Requirements in the product life cycle)

Proposition: From a suppliers point of view, a COTS contract is one of the most easy contracts to handle.

Bonus: 4

Reason: The supplier delivers a product of the shelf and the contract is more or less the products specification. If the customer has a requirement specification he can use it as a guideline to choose the right product.

Correct answers: A

Motivation: Since the customer chooses a product of the shelf guided by his needs the supplier only needs to deliver what is already on the shelf. Therefore the supplier does not often take responsibility to fulfill all of the customers requirements.

Reference: Lau chap 7, page 295

Learning objectives: 3, 4, 5

Main responsible: Erik Alstersjö (adi10eal)

•[Lau:6, QUPER] 2 problems (Quality requirements, QUPER)

Proposition: Usability tests are a popular method for acceptance testing.

Reason: Usability tests consist of letting a test person try to carry out realistic tasks using the system.

Correct answers: D

Motivation: "[Usability tests are] not something you do at the end of development to prove that the system is user-friendly". This is most probably because, as Lausen himself says: "Usability problems often require large changes or the complete redesign of central parts of the user interface".

Reference: Lau. Chap 6, pp. 252-254

Learning objectives: 1, (3,) 4, 5,

Main responsible: Andreas Coroiu (dat11aco)

Proposition: The best way to get a good quality requirement is to use the open target approach.

Reason: Open target approach is a method where you let the supplier specify details in the requirement.

Correct answers: D

Motivation: Open target approach has a problem: Often the supplier has no idea of how important the requirement is for the customer, which means that he will spend the wrong amount of resources on it. To complement the Open target approach, the customer can specify his expectations.

Reference: Lau. Chap 6, page 228-229

Learning objectives: 1, 3

Main responsible: Carl Rygart (elt11cry)

•[Lau:9, INSP] 2 problems (Checking & Validation, Inspection)

Proposition: Lauesen states that when validating completeness of a software requirement specification, it is important to ensure that user interface details are covered.

Reason: User interface details are important, as statistics show that requirements concerning the user interface are the most common source of problems.

Correct answers: D

Motivation: User interface details are indeed important, but unfortunately really difficult to specify logically. Instead the user interface should be specified indirectly (through User Tasks e.g.) or through careful design and testing.

Reference: Lau. Chap 9, page 376

Learning objectives: 1, 2, 3

Main responsible: Carl-Johan Heinze (ada10che)

Proposition: Using CRUD matrices instead of event checks when validating system requirements leads to a more stable product, which is more likely to have all intended functionality and less serious bugs after the first release.

Reason: CRUD matrices ensure that entity classes in the data model be Created and Deleted by an event or function or user task. They even validate if the fields in the data model are Read and Updated by an event/function/user task.

Correct answers: D

Motivation: Even though the reason argument is a correct statement, CRUD matrices do not cover the same scenarios as event checks. Even though the two methods may occasionally overlap for a system, event checks revise that system events are handled by a function or user task. Hence these event checks cannot be replaced by a CRUD matrix.

Reference: Lau. Chap 9, page 386-388

Learning objectives: 1, 3, 4, 5

Main responsible: Marcel Tovar (elt11mto)

•[MDRE+PRIO+RP] 2 problems (Prioritization, Release Planning, Market-Driven Requirements Engineering)

Proposition: While doing prioritization it is important to have the three perspectives of customer, developer and financial representative in mind

Reason: The developer stakeholder should be the main source of information, while the other two is good to have in mind to get a guideline for the developers to follow.

Correct answers: C

Motivation: To get a complete view of a project the three stakeholders cover a good basis, and should all be taken in consideration while doing a prioritization of something in a project. Each of the stakeholders can possibly provide with vital information that each of the other two parts have not thought of. That's why it is not only important to have the developers view in mind but also atleast the costumer and finansial representative.

Reference: [PRIO]Requirements Prioritization, Patrik Berander and Anneliese Andrews, part 4.5.4

Learning objectives: "Be able to apply several different techniques for requirements prioritisation"

"be able to consciously see the problem in the relation between requirements engineering and economical aspects of product development"

Main responsible: Johan Westerlund (elt11jwe)

Proposition: For Market-Driven Requirements Engineering success is measured in sales, market share and product reviews. Bespoke Requirements Engineering on the other hand places importance in customer satisfaction and user acceptance.

User acceptance is also important sine it leads to success in the listed factors, i.e. indirectly important.

Reason: The primary object of Market-Driven Development is to deliver the right product at the right time, whereas Bespoke Development focuses on contract fulfillment and compliance with a requirements specification.

Correct answers: A

Motivation: Market-Driven Requirements Engineering focuses on pleasing a broad set of customers, with the intention of as widespread adoption as possible. Bespoke Requirements Engineering generally aims to please one customer based on agreed upon terms.

Reference: [MDRE] Market Driven Requirements Engineering for Software Products, Björn Regnell & Sjaak Brinkkemper, 13.2.2 Characteristics of MDRE

Learning objectives: 1, 6

Main responsible: Joakim Klasman (ael10jkl)

•[AGRE+INTDEP] 1 problem (Agile Requirements Engineering, Requirement Interdependencies)

Proposition: Release planning is a large part in market-driven software development.

Reason: Release planning is basically prioritizing requirements and then select them on basis of their priorities, dependencies and existing resources, and then develop them for release. A proper release planning will show what should be delivered when during the development cycle.

Correct answers: A

Motivation: Release planning is a crucial part of a market-driven development cycle since it decides what should be delivered when. If there are none or few interdependencies between requirements, the release planning is pretty much a matter of prioritizing requirements and then develop them according to this prioritization. The prioritization however, can be affected by available resources and delivery dates.

Reference: INTDEP, (An Industrial Survey of Requirements Interdependencies in Software Product Release Planning)

Learning objectives: 1,(5,6)

Main responsible: Tomas Harryson (dic12tha)

weak  
reason,  
but in  
motiva-  
tion