Mid-Term-Questionnaire

Questions – Lecture slides – Answers

Question 1

1. How are systems useful in the context of some human activities supported by the software, which is run on some hardware, called?

• Computer systems
• Information systems
• Software-intensive systems
• Software systems
Software-Intensive Systems

• **Software (on its own) is useless**
  - Software is an abstract description of a set of computations
  - Software only becomes useful when run on some hardware
    • we sometimes take the hardware for granted
  - **Software + Hardware = “Computer System”**

• **A Computer System (on its own) is useless**
  - Only useful in the context of some human activity that it can support
    • we sometimes take the human context for granted
  - A new computer system will change human activities in significant ways
  - **Software + Hardware + Human Activities = “Software-Intensive System”**

• ‘Software’ makes many things possible
  - It is complex and adaptable
  - It can be rapidly changed on-the-fly
  - It turns general-purpose hardware into a huge variety of useful machines

1. How are systems useful in the context of some human activities supported by the software, which is run on some hardware, called?

   • Computer systems
   • Information systems
   • **Software-intensive systems**
   • Software systems
Question 2

2. Which software system development lifecycle is primarily used to understand the requirements for the user interface, to examine feasibility of a proposed design approach and/or to explore system performance issues?

- Prototyping
- Spiral model
- V model
- Agile model

Prototyping lifecycle

- **Prototyping is used for:**
  - understanding the requirements for the user interface
  - examining feasibility of a proposed design approach
  - exploring system performance issues

- **Problems:**
  - users treat the prototype as the solution
  - a prototype is only a partial specification
2. Which software system development lifecycle is primarily used to understand the requirements for the user interface, to examine feasibility of a proposed design approach and/or to explore system performance issues?

- Prototyping
- Spiral model
- V model
- Agile model

Question 3

3. What are things in the application domain that are true or not when we ever build the proposed systems?

- Requirements
- Specification
- Domain properties
- Assumptions and expectations
What are requirements?

- **Domain Properties:**
  - things in the application domain that are true whether or not we ever build the proposed system
  - Synonyms: Assumptions and expectations

- **Requirements:**
  - things in the application domain that we wish to be made true by delivering the proposed system
  - Many of which will involve phenomena the machine has no access to

- **A Specification:**
  - is a description of the behaviours that the program must have in order to meet the requirements
  - Can only be written in terms of shared phenomena!

3. What are things in the application domain that are true or not when we ever build the proposed systems?

- Requirements
- Specification
- **Domain properties**
- Assumptions and expectations
Question 4

4. What is the part of the system environment relevant for defining, understanding and interpreting the system requirements?

• System boundary
• Requirement engineering
• Requirements specification
• System context
4. What is the part of the system environment relevant for defining, understanding and interpreting the system requirements?

- System boundary
- Requirement engineering
- Requirements specification
- **System context**

---

**Question 5**

5. What aspects do concern the operational or technical environment in which the system is deployed?

- Subject facet
- Usage facet
- Development facet
- IT system facet
5. What aspects do concern the operational or technical environment in which the system is deployed?

- Subject facet
- Usage facet
- Development facet
- IT system facet
6. Which requirements engineering activity does help to achieve progress in the content dimension by detailing information about existing requirements?

- Requirements validation
- Requirements elicitation
- Requirements management
- Requirements specification

Achieve progress in the content dimension by eliciting new requirements as well as detailed information about existing requirements.

- Elicit all requirements at the level of detail for the system to be developed.
6. Which requirements engineering activity does help to achieve progress in the content dimension by detailing information about existing requirements?

- Requirements validation
- **Requirements elicitation**
- Requirements management
- Requirements specification

---

**Question 7**

7. Which requirements artefacts do specify requirements at the required level of detail, the desired properties and features of the system to be developed?

- Solution-oriented requirements
- Goals
- Scenarios
- Domain properties
7. Which requirements artefacts do specify requirements at the required level of detail, the desired properties and features of the system to be developed?

- **Solution-oriented requirements**
- Goals
- Scenarios
- Domain properties
Question 8

8. Is elicitation difficult because of the existing bias?

- Yes
- No
- Yes, and also because of the thin spread of knowledge and tacit knowledge
- No, it is difficult because of the thin spread of knowledge and tacit knowledge

Difficulties of Elicitation

- **Thin spread of domain knowledge**
  - The knowledge might be distributed across many sources
    - It is rarely available in an explicit form (i.e. not written down)
    - There will be conflicts between knowledge from different sources
    - Remember the principle of complementarity!
- **Tacit knowledge (The “say-do” problem)**
  - People find it hard to describe knowledge they regularly use
- **Limited Observability**
  - The problem owners might be too busy coping with the current system
  - Presence of an observer may change the problem
    - E.g. Probe Effect; Hawthorne Effect
- **Bias**
  - People may not be free to tell you what you need to know
  - People may not want to tell you what you need to know
    - The outcome will affect them, so they may try to influence you (hidden agendas)
8. Is elicitation difficult because of the existing bias?

- Yes
- No
- Yes, and also because of the thin spread of knowledge and tacit knowledge
- No, it is difficult because of the thin spread of knowledge and tacit knowledge

Question 9

9. What are the major stakeholder interests?

- Financial interests
- Observational interests
- Development interests
- Usage interests
Stakeholders

- **Stakeholder analysis:**
  - Identify all the people who must be consulted during information acquisition

- **Example stakeholders**
  - **Users**
    - concerned with the features and functionality of the new system
  - **Designers**
    - want to build a perfect system, or reuse existing code
  - **Systems analysts**
    - want to "get the requirements right"
  - **Training and user support staff**
    - want to make sure the new system is usable and manageable
  - **Business analysts**
    - want to make sure "we are doing better than the competition"
  - **Technical authors**
    - will prepare user manuals and other documentation for the new system
  - **The project manager**
    - wants to complete the project on time, within budget, with all objectives met.
  - **"The customer"**
    - Wants to get best value for money invested!

9. What are the major stakeholder interests?

- **Financial interests**
  - Customer

- **Development interests**
  - Designer, System analyst, technical author, etc.

- **Usage interests**
  - Users
Question 10

10. Which elicitation techniques are used for summarization and feedback, i.e., to conclude on a set of the requirements, to discuss the results of the information gathering, etc.

- Meetings
- Interviews
- Groups elicitation techniques
- Joint/rapid application development

Meetings

- **Used for summarization and feedback**
  - E.g. meet with stakeholders towards the end of each stage:
    - to discuss the results of the information gathering stage
    - to conclude on a set of requirements
    - to agree on a design etc.
  - Use the meeting to confirm what has been learned, talk about findings
- **Meetings are an important managerial tool**
  - Used to move a project forward.
  - Every meeting should have a clear objective:
    - E.g. presentation, problem solving, conflict resolution, progress analysis, gathering and merging of facts, training, planning,...
  - Plan the meeting carefully:
    - Schedule the meeting and arrange for facilities
    - Prepare an agenda and distribute it well in advance
    - Keep track of time and agenda during the meeting
    - Follow up with a written summary to be distributed to meeting participants
    - Special rules apply for formal presentations, walkthroughs, brainstorming, etc.
10. Which elicitation techniques are used for summarization and feedback, i.e., to conclude on a set of the requirements, to discuss the results of the information gathering, etc.

- Meetings
- Interviews
- Groups elicitation techniques
- Joint/rapid application development

Question 11

11. What concerns should be included to requirement specification (document)?

- Functionality
- Performance
- Attributes
- Design constraints imposed on and implementation
Specification Contents

- Specification should address:
  - Functionality
    - What is the software supposed to do?
  - External interfaces
    - How does the software interact with people, the system's hardware, other hardware, and other software?
    - What assumptions can be made about these external entities?
  - Performance
    - What is the speed, availability, response time, recovery time of various software functions, and so on?
  - Attributes
    - What are the portability, correctness, maintainability, security, and other considerations?
  - Design constraints imposed on an implementation
    - Are there any required standards in effect, implementation language, policies for database integrity, resource limits, operating environment(s) and so on?

11. What concerns should be included to requirement specification (document)?

- Functionality
- Performance
- Attributes
- Design constraints imposed on and implementation
Question 12

12. Which criterion of good requirements does *often* require quantification?

- External observability
- Understandability
- Cohesiveness
- Testability

Criteria for Writing Good Requirements

- **What, not how (external observability)**
  - Avoid premature design or implementation decisions
- **Understandability, clarity (not ambiguous)**
- **Cohesiveness (one thing per requirement)**
- **Testability**
  - Somehow possible to test or validate whether the requirement has been met, clear acceptance criteria
  - Often requires quantification, this is more difficult for security than e.g. for performance
    - "The response time of function F should be max 2 seconds"
    - "The security of function F should be at least 99.9 %"
12. Which criterion of good requirements does *often* require quantification?

- External observability
- Understandability
- Cohesiveness
- Testability

**Question 13**

13. Why is it important to resolve requirements conflicts?

- For the acceptance of the system by stakeholders
- For the successful (i.e., timely, within the budget, and etc.) completion of the project
- For the interdependence and clarity
- For the complete specification of the requirements
Conflicts

- **Conflict (in RE)**
  - Exists if the needs and wishes of different stakeholders (or groups of stakeholders) regarding the system contradicts each other,
  - Exists if some needs and wishes cannot be taken into account

- **Risks of unresolved conflicts**
  - Compromise acceptance of the system by stakeholders
  - If conflict disregarded or suppressed, some stakeholders may not support development of the system
  - May result in failure of the project

- **Involve relevant stakeholders**

13. Why is it important to resolve requirements conflicts?

- **For the acceptance of the system by stakeholders**
- **For the successful (i.e., timely, within the budget, and etc.) completion of the project**
- **For the interdependence and clarity**
- **For the complete specification of the requirements**
Question 14

14. Which requirements negotiation activity might include decision (by authority) making?

- Conflict identification
- Conflict analysis
- Conflict resolution
- Conflict documentation

Resolving Conflicts:

**Decision**

- Higher **authority** makes a decision
  - in favour of one conflicting party

- Example
  - The client is involved as a higher authority. The client decides that the detection range shall be 500 m.

- **Voting**

**Third Party Resolution**

- participants appeal to outside source
  - the rule-book, a figure of authority, or the toss of a coin.
  - can occur with the breakdown of either negotiation or competition as resolution methods.

- types of third party resolution
  - **judicial**: cases presented by each participant are taken into account
  - **extra-judicial**: a decision is determined by factors other than the cases presented (e.g. relative status of participants)
  - **arbitrary**: e.g. toss of a coin
14. Which requirements negotiation activity might include decision (by authority) making?

- Conflict identification
- Conflict analysis
- **Conflict resolution**
- Conflict documentation

**Question 15**

15. What type of conflict does exist if stakeholders are wrongly or incompletely informed about the requirements?

- Values conflict
- Data conflict
- Interest conflict
- Analysis conflict
Data conflict

- **Data conflict exists**
  - if stakeholders are wrongly or incompletely informed about the requirement
  - If stakeholders interpret the meaning of the requirements differently

- **Example:**
  R4: The DVD player shall be able to handle re-writable CDs (CD-RW) and DVDs (DVD-RW).

15. What type of conflict does exist if stakeholders are wrongly or incompletely informed about the requirements?

- Values conflict
- **Data conflict**
- Interest conflict
- Analysis conflict
Question 16

16. During which activities requirements inconsistencies can be found?

- Requirements elicitation
- Requirements documentation
- Requirements management
- Requirements validation

Conflicts in different RE activities

- **During the elicitation workshop**
  - Two stakeholders state requirements that contradict each other and hence cannot be realised together

- **When documenting requirements**
  - The stakeholders detect a conflict between two requirements that originate from different interviews

- **Requirements management**
  - Conflict occurs during requirements prioritisation – different stakeholders have different opinions regarding the requirements priority

- **Requirements validation**
  - Conflicts occur while stakeholders check the specified requirements for correctness
  - One stakeholder considers a requirement correct – another objects
16. During which activities requirements inconsistencies can be found?

- Requirements elicitation
- Requirements documentation
- Requirements management
- Requirements validation

Question 17

17. To which activity does phrase “to build right system” refer?

- System development
- Validation
- Verification
- Requirements engineering
Verification and Validation

- **Validation:**
  - “Are we building the right system?”
  - Does our problem statement accurately capture the real problem?
  - Did we account for the needs of all the stakeholders?

- **Verification:**
  - “Are we building the system right?”
  - Does our design meet the spec?
  - Does our implementation meet the spec?
  - Does the delivered system do what we said it would do?
  - Are our requirements models consistent with one another?

17. To which activity does phrase “to build right system” refer?

- System development
- **Validation**
- Verification
- Requirements engineering
Question 18

18. What should be validated during requirements engineering?

- Requirements activity inputs
- Requirements competitors
- Execution of requirements activities
- Requirements activity outputs

**Validation Goal**

- Check whether the **outputs** of activities fulfill defined quality criteria
- Check whether the execution of activities adheres to process definitions and activity guidelines
- Check whether the **inputs** of activities fulfill defined quality criteria
18. What should be validated during requirements engineering?

- Requirements activity inputs
- Requirements competitors
- Execution of requirements activities
- Requirements activity outputs

Question 19

19. During which activity does requirement engineer establish requirements traceability, prioritise requirements, and manage changes of requirements artefacts?

- Requirements representation
- Requirements validation
- Requirements management
- Requirements documentation
19. During which activity does requirement engineer establish requirements traceability, prioritise requirements, and manage changes of requirements artefacts?

- Requirements representation
- Requirements validation
- **Requirements management**
- Requirements documentation
Question 20

20. How is the development of cost effective solutions to practical problems through the application of scientific knowledge called?

- Live-cycle
- Building
- Engineering
- Application

What is engineering?

“Engineering is the development of cost-effective solutions to practical problems, through the application of scientific knowledge”

“...Cost-effective...”
- Consideration of design trade-offs, esp. resource usage
- Minimize negative impacts (e.g. environmental and social cost)

“... Solutions ...”
- Emphasis on building devices

“... Practical problems ...”
- solving problems that matter to people
- improving human life in general through technological advance

“... Application of scientific knowledge ...”
- Systematic application of analytical techniques
20. How is the development of cost effective solutions to practical problems through the application of scientific knowledge called?

- Live-cycle
- Building
- Engineering
- Application