Lecture # 2

Software Requirements Engineering (SE – 208)

Dr. Shehnila Zardari

Kinds of Software Requirements

- Functional requirements
- Non-functional requirements
- Domain requirements
- Inverse requirements
- Design and implementation constraints

Non-Functional Requirements

Non-Functional Requirements - 1

- Most non-functional requirements relate to the system as a whole. They include constraints on timing, performance, reliability, security, maintainability, accuracy, the development process, standards, etc.

Non-Functional Requirements - 2

- They are often more critical than individual functional requirements
- Capture the emergent behavior of the system, that is they relate to system as a whole

Non-Functional Requirements - 3

- Must be built into the framework of the software product
- Failure to meet a non-functional system requirement may make the whole system unusable

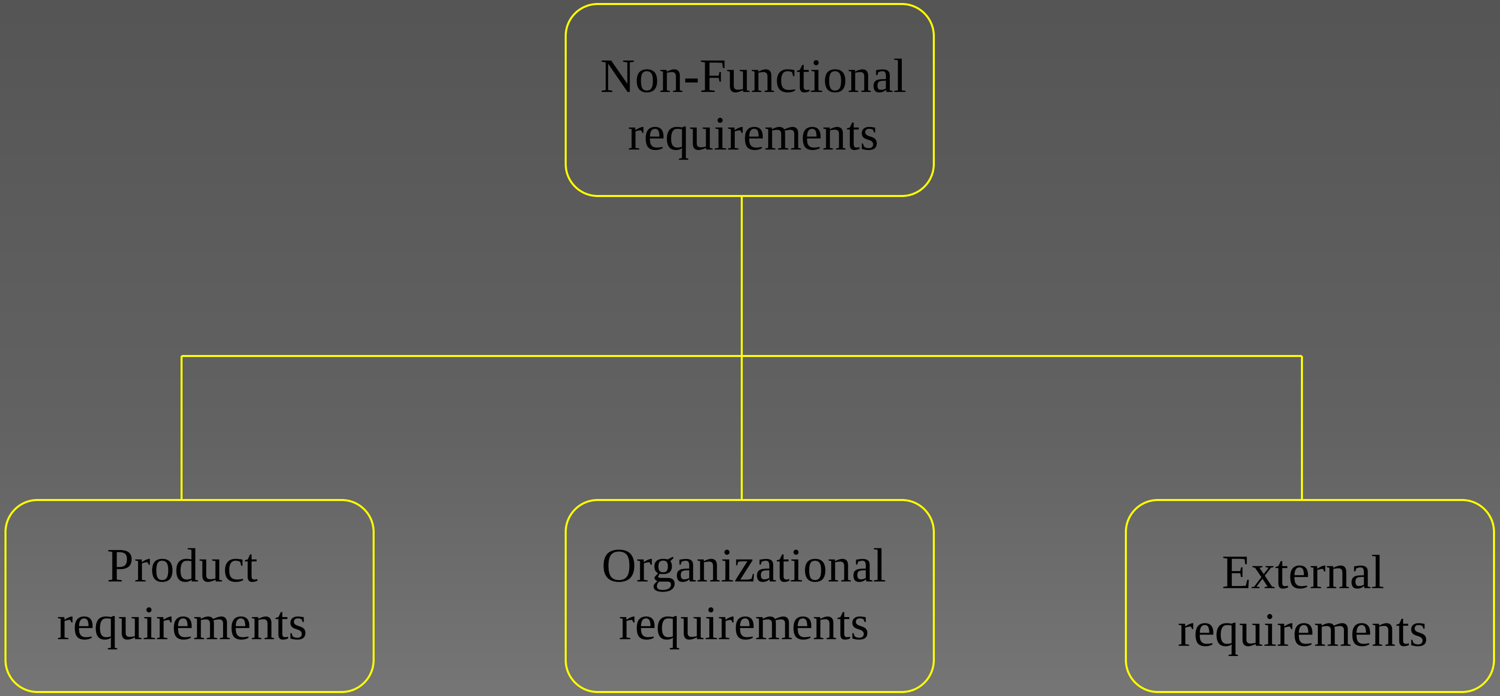
Non-Functional Requirements - 4

- For example, if an aircraft system does not meet reliability requirements, it will not be certified as ‘safe’
- If a real-time control system fails to meet its performance requirements, the control functions will not operate correctly

Non-Functional Requirements - 5

- Non-functional requirements arise through user needs, because of budget constraints, because of organizational policies, because of the need of interoperability with other software and hardware systems, or because of external factors such as safety regulations, privacy legislation, etc.

Non-Functional Requirements

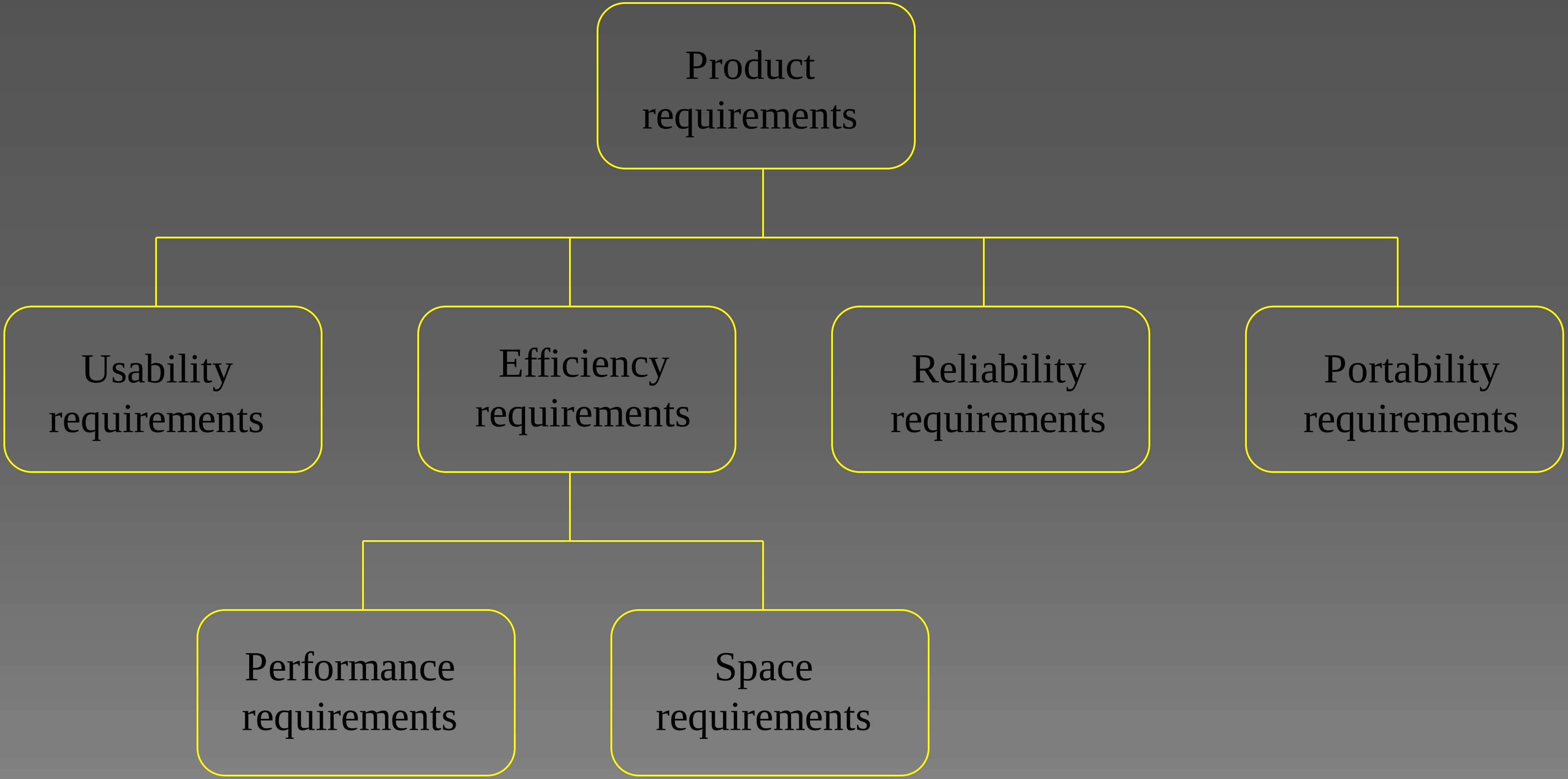

Non-Functional requirements

Product requirements

Organizational requirements

External requirements

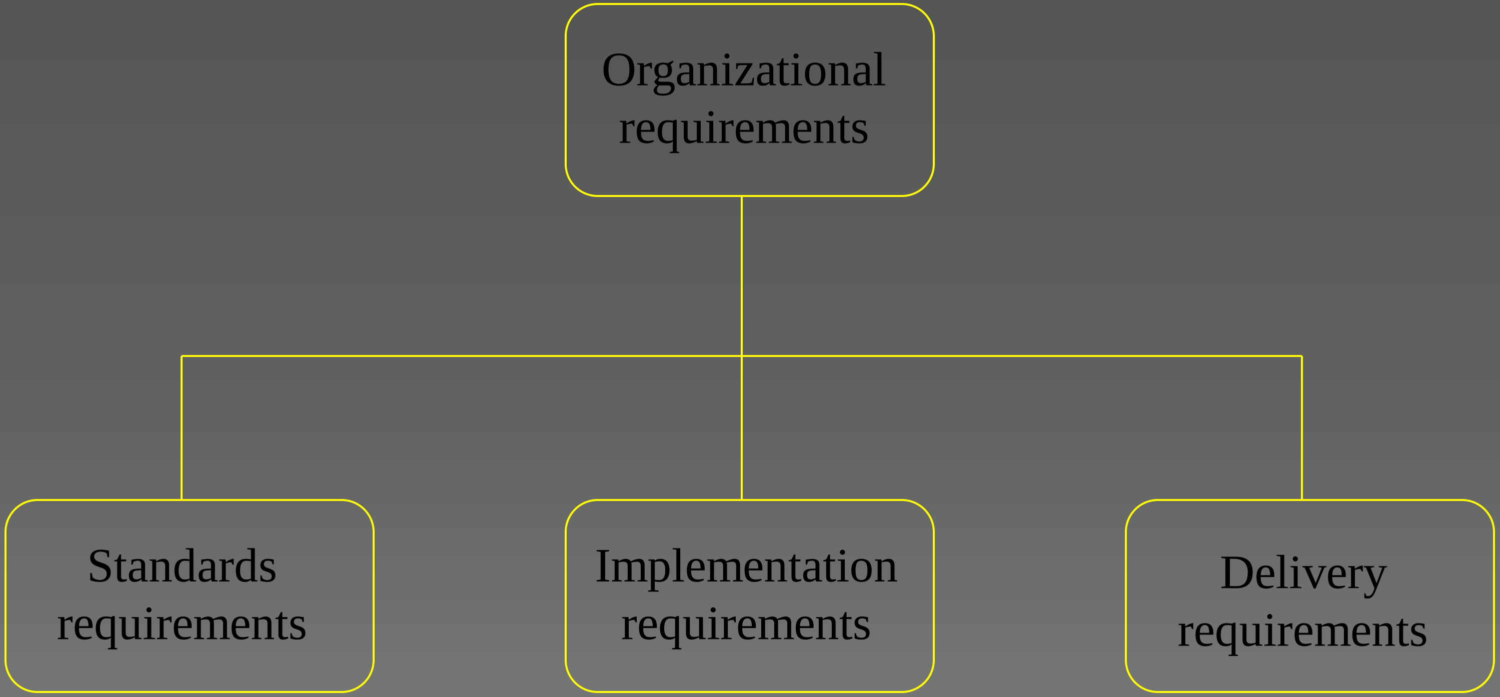
Product Requirements



Product Requirements Examples

- The system shall allow one hundred thousand hits per minute on the website
- The system shall not have down time of more than one second for continuous execution of one thousand hours

Organizational Requirements



Organizational requirements

Standards requirements

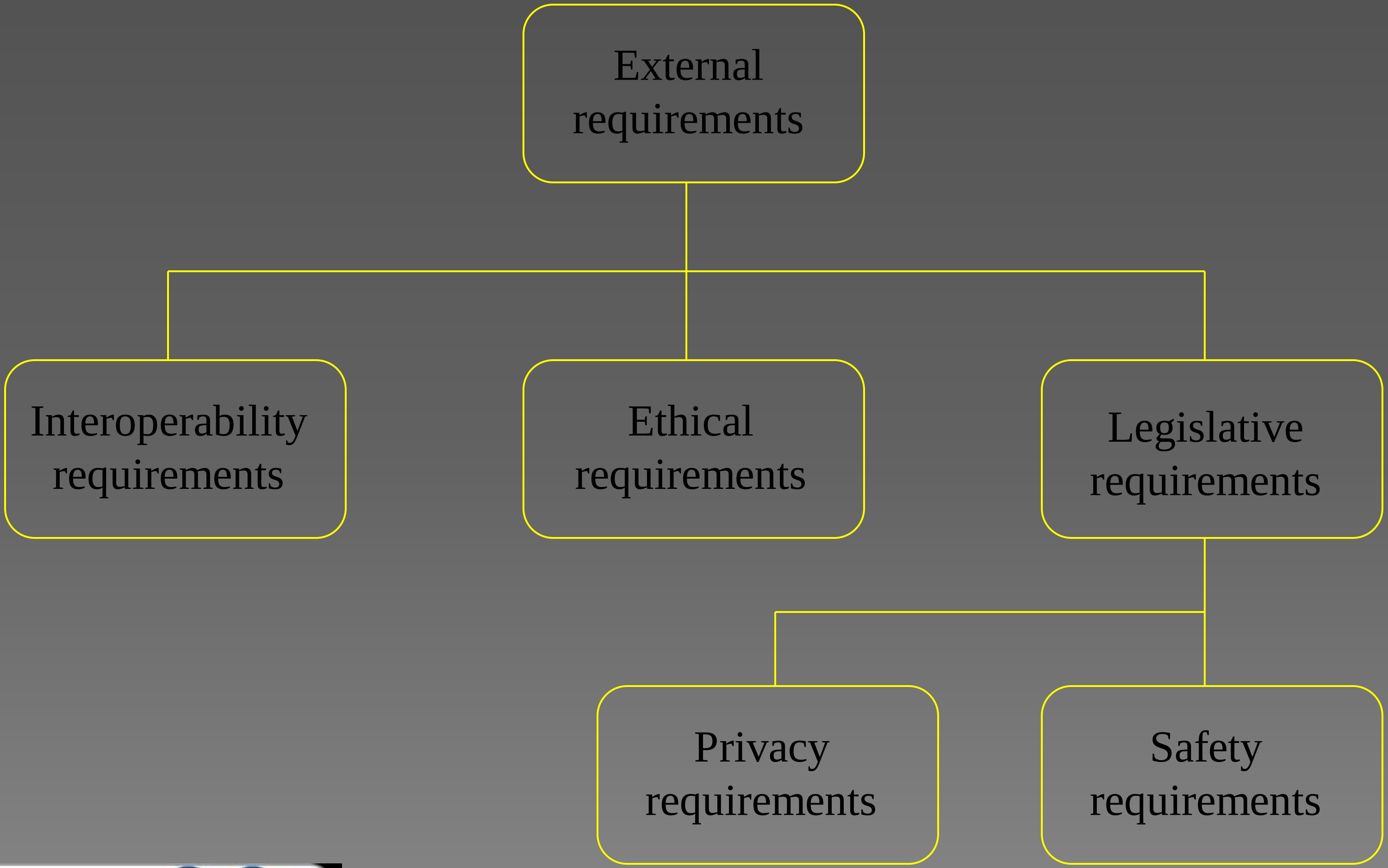
Implementation requirements

Delivery requirements

Organizational Requirements Examples

- The system development process and deliverable documents shall conform to the MIL-STD-2167A
- Any development work sub-contracted by the development organization shall be carried out in accordance with Capability Maturity Model

External Requirements



External requirements

Interoperability requirements

Ethical requirements

Legislative requirements

Privacy requirements

Safety requirements

External Requirements Examples

- The system shall not disclose any personal information about members of the library system to other members except system administrators
- The system shall comply with the local and national laws regarding the use of software tools

Observations on Non-Functional Requirements - 1

- Non-functional requirements can be written to reflect general goals for the system. Examples include:
  - **Ease of use**
  - **Recovery from failure**
  - **Rapid user response**

Observations on Non-Functional Requirements - 2

- Goals are open to misinterpretation
- Objective verification is difficult
- Distinction between functional and non-functional is not always very clear

Observations on Non-Functional Requirements - 3

- Non-functional requirements should be written in a quantitative manner as much as possible, which is not always easy for customers
- For some goals, there are no quantitative measures, e.g., maintainability

Observations on Non-Functional Requirements - 4

- Goals can be useful to designers and developers, as they give clues to them about priorities of the customers

Observations on Non-Functional Requirements - 5

- Chances of conflicts within non-functional requirements are fairly high, because information is coming from different stakeholders. For example, different stakeholders can give different response times or failure tolerance levels, etc.

Observations on Non-Functional Requirements - 6

- Some negotiations must be done among different stakeholders, to achieve an agreement in these situations

Observations on Non-Functional Requirements - 7

- Non-functional requirements should be highlighted in the requirements document, so that they can be used to build the architecture of the software product

Summary

- Discussed different aspects of the non-functional requirements
- Non-functional requirements capture very important emergent behavior of the automated system
- Due importance, time, and resources should be given to non-functional requirements

References

- ‘Requirements Engineering: Processes and Techniques’ by G. Kotonya and I. Sommerville, John Wiley & Sons, 1998
- Software Requirements: Objects, Functions, and States by A. Davis, PH, 1993
- Software Engineering 6th Edition, by I. Sommerville, 2000
- Software Engineering 5th Edition, by R. Pressman