Chapter 4
System Requirements

Lesson Objectives

- Differentiate between User requirements and System requirements
- Differentiate between Functional and Non-functional Requirement
- Explain why natural language specification is not desirable
- Understand how requirements may be organised in a software requirements document (SRS)

User Requirements vs. System Requirements

User Requirements vs. System Requirements

Requirements for a software system set out what the system should do and define constraints on its operation and implementation.

Two different types of requirements:

- User requirements
- System requirements

User Requirements

- User requirements are high-level abstract requirements. They are intended for use by people involved in using and procuring the system.

- User requirements are statements in natural language, with easy-to-understand diagrams and tables, of *what* services the system is expected to provide and the *constraints* under which it must operate.

System Requirements

- System requirements are more precise and detail descriptions of the system's functions, services and operational constraints.

- They may be written in structured form of natural language supported by system models and tables

System Requirements

- System requirements are also used by software designers as the starting point for the system design, so they are sometimes called “functional specifications”.

- They may be the basis of a contract between the system developer and customer. Therefore should be complete and consistent specification of the whole system.

User Requirements vs. System Requirements

Readers of User Requirements

Client Managers
Contractor Managers
System End-User
Client Engineers
System Architects

Readers of System Requirements

Software Developers
System End-User
Client Engineers
System Architects

User Requirements vs. System Requirements

- Examples:

User Requirement

1. The software must provide a means of representing and accessing external files created by other tools

System Requirement

1.1 The user should be provided with facilities to define the type of external files

1.2 Each external file type may be represented as a specific icon on the user's display.

1.3 Facilities should be provided for the icon representing an external file type to be define by the user

User Requirements vs. System Requirements

- Examples:

User Requirement

1. LIBSYS shall keep track of all data required by copyright licensing agencies in the UK and elsewhere

System Requirement

1.1 On making a request for a document from LIBSYS, the requestor shall be presented with a form that records details of the user and the request made.

1.2 LIBSYS request forms shall be stored on the system for five years from that data of the request.

1.3 All LIBSYS request forms must be indexed by user, by name of the material requested and by the supplier of the request.

1.4 LIBSYS shall maintain a log of all requests that have been made to the system.

1.5 For material where authors' lending rights apply, loan details shall be sent monthly to copyright licensing agencies that have registered with LIBSYS

User Requirements vs. System Requirements

What is the main difference between user requirement and system requirement?

System requirement is more precise and detail than user requirement as they are intended to communicate information about the system to different types of readers.

Exercise – Past Year Question

Prepare the user requirement and system requirement for the Online Purchase Concert Ticket function for an online system which allows online users to check seat availability, concert details, purchase tickets etc.

Marking Criteria	Marks Allocation
User Requirement	2
System Requirement	7

Exercise – Past Year Question

Online Purchase Concert Ticket

User Requirement

1. The system must provide a facility which allows the online users to check on concert details, seat availability and purchase tickets

Exercise – Past Year Question

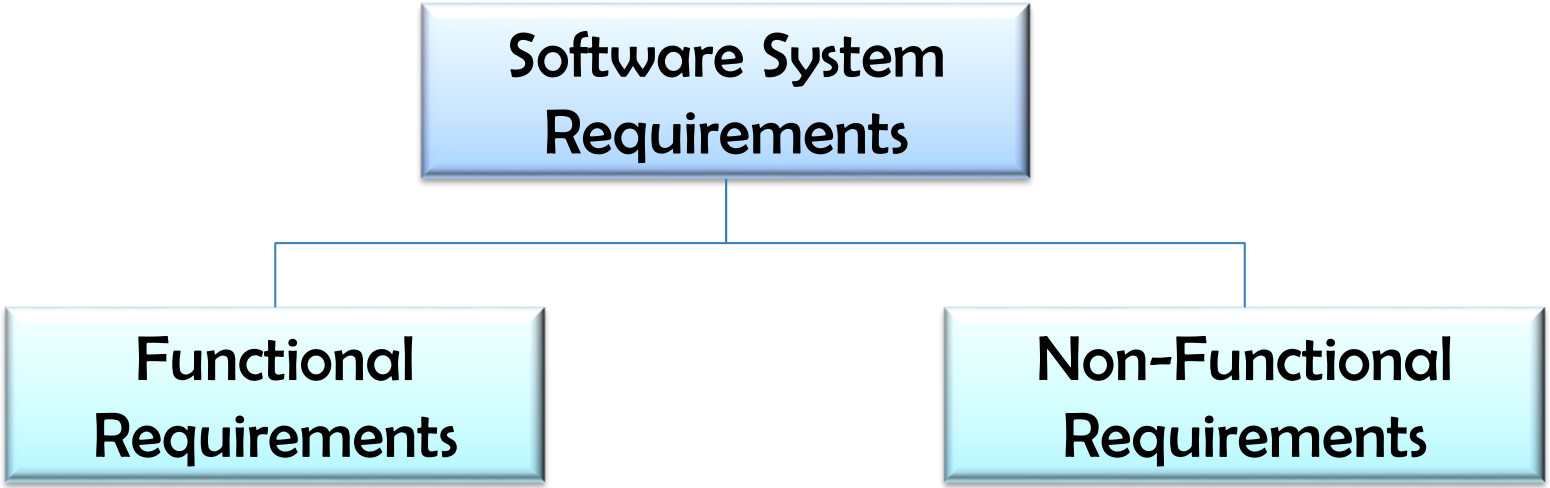
Online Purchase Concert Ticket

System Requirement

1.1 The system allows the online users to view concert's details
1.2 The users can check the seat availability for the concert they are interested in
1.3 The users can select the seating position available and make a purchase for the ticket(s)
1.4 The system will calculate the charges once the user confirmed the transaction
1.5 The system will also check on the payment details such credit card number and etc.
1.6 If invalid payment details, the system will show message
1.7 If valid, the system will proceed transaction and update the seat(s) to not available
1.8 The system will generate on-line receipt

Functional and Non-Functional Requirements

Functional & Non-Functional Requirements



Functional Requirements

- These are statement of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situation. In some cases, the functional requirements may also explicitly state what the system should not do.

Functional Requirements

- **Examples of functional requirements of a university library system called LIBSYS:**

  - The user shall be able to search either all of the initial set of databases or select a subset from it.
  - The system shall provide appropriate viewers for the user to read documents in the document store.

Functional Requirements

- ## HR System – Employee Self Service Sub-system:

1.0 Basic Personal Tasks

- 1.1 Employees can perform routine HR administration tasks such as changing name, address, home / work telephone numbers, emergency contacts, next of kin, marriage, births or adoption details
- 1.2 Employees can upload common information to existing databases eg telephone directories and organisation charts
- 1.3 Employees can view employees own employment history ie promotions, dates, appraisal reviews, training records
- 1.4 Employees can view conditions of employment, HR policies, procedures, information packs, benefits details

Functional Requirements

- ## HR System – Employee Self Service Sub-system:

2.0 Time and Attendance Tasks

2.1 Employees can enter time sheets, plus with workflow, automatically submit request to manager for approval and once approved, automatically posted to payroll.

2.2 Employees can manage absences from work. Enter absence details.

Non Functional Requirements

- **Product Requirements**

These are requirements which result from the need for the delivered product to behave in a particular way.

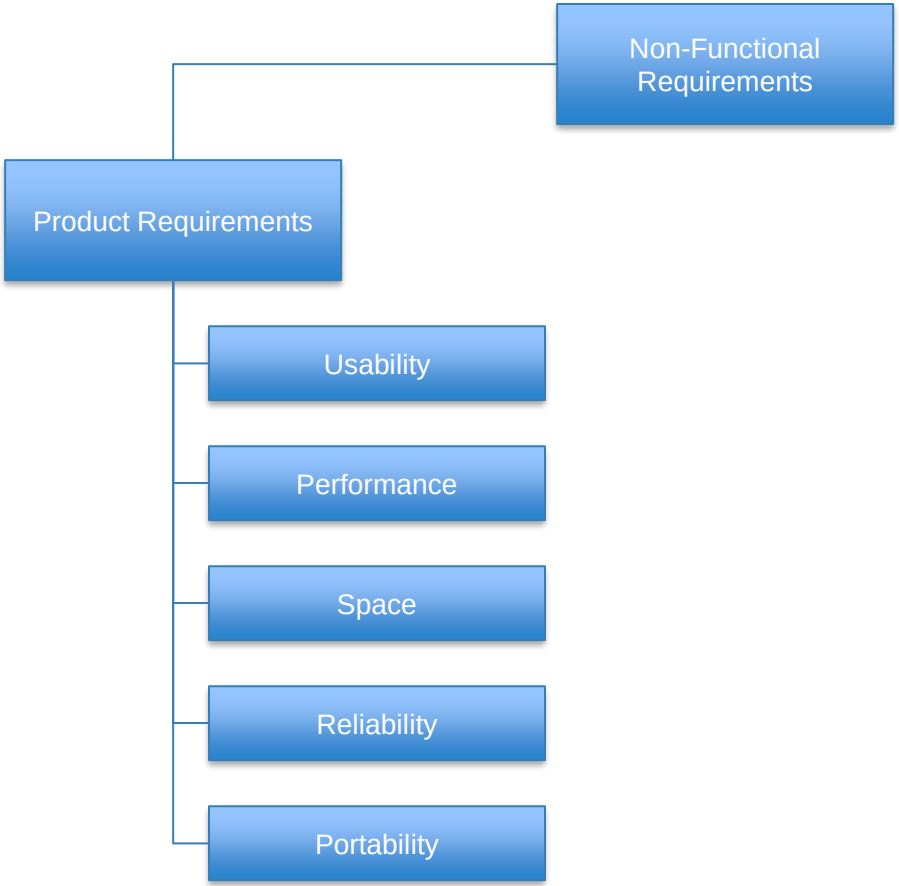
- **Organisational Requirements**

These are requirements which are a consequence of organisational *policies* and *procedures*.

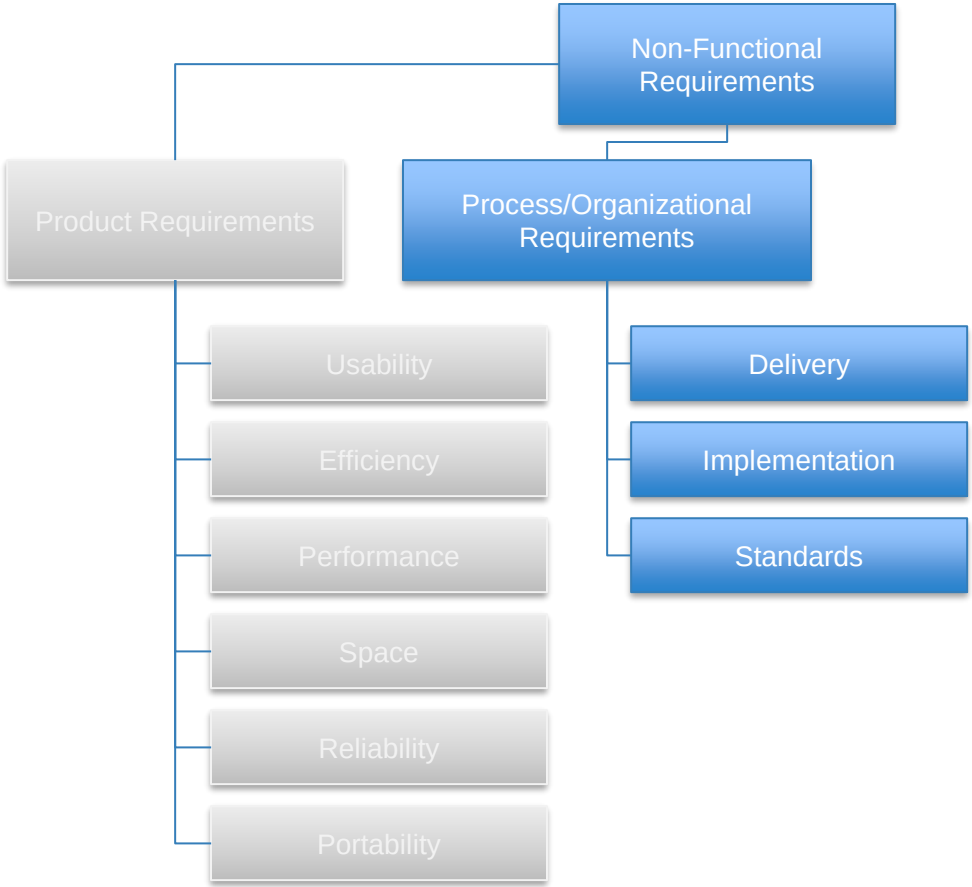
- **External Requirements**

This broad heading covers all requirements which arise from factors external to the system and its development process.

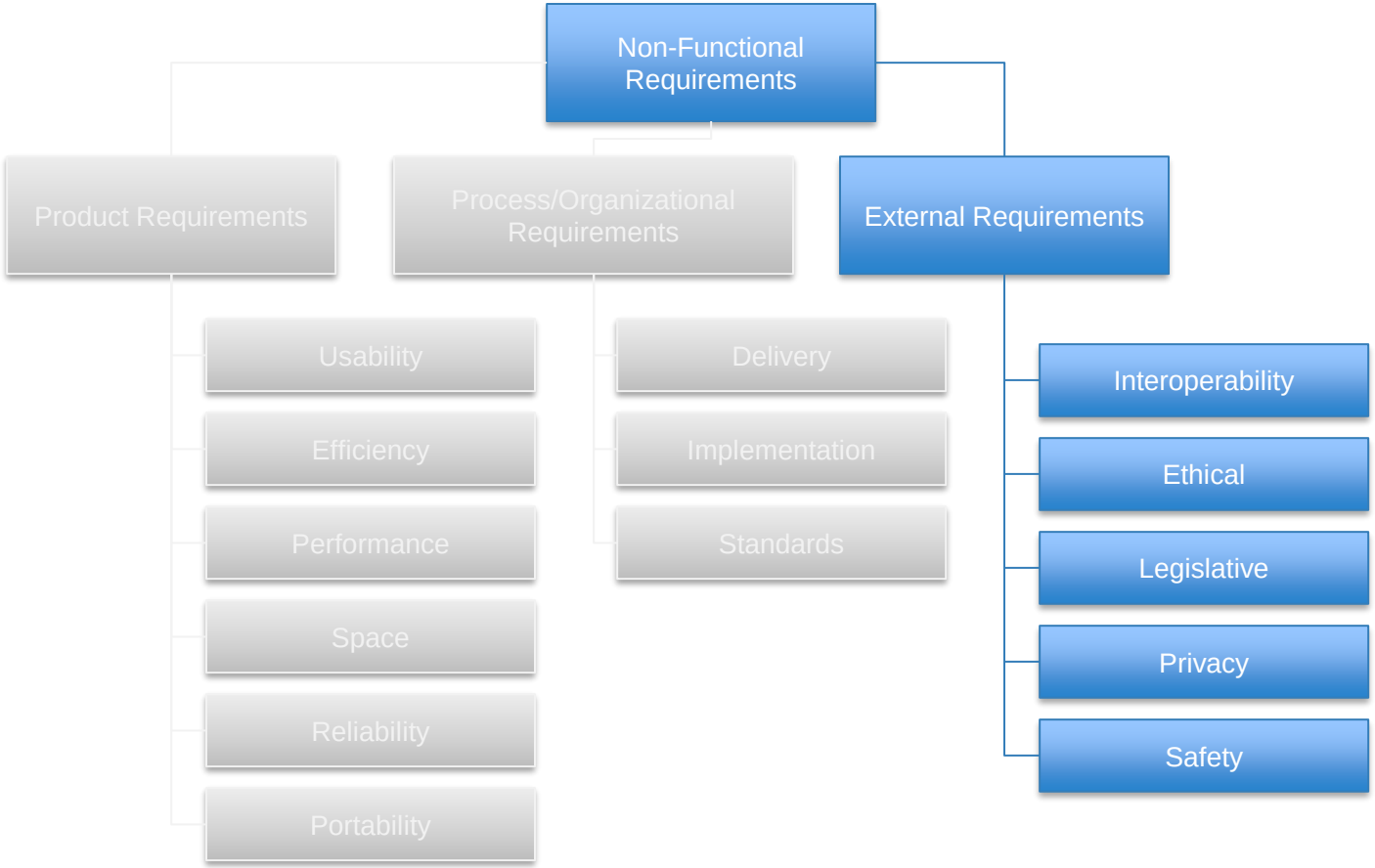
Non Functional Requirements



Non Functional Requirements



Non Functional Requirements



Non Functional Requirements

- Example

Product requirement

*8.1 The **user interface** for LIBSYS shall be implemented as simple **HTML** without frames or Java applets.*

Organizational requirement

*9.3.2 The system development process and deliverable documents shall **conform** to the process and deliverables defined in XYZ-ST95.*

Non Functional Requirements

- Example

External requirement

*10.6 The system **shall not disclose any personal information** about system users apart from their name and library reference number to library staff who use the system.*

Exercise – Past Year Question

Q1: Prepare the user requirement and system requirement for the Online Purchase Concert Ticket function for an online system which allows online users to check seat availability, concert details, purchase tickets etc.

Q2: Give 2 Functional and 2 Non-Functional Requirement for the above online system

Problems with Natural Language

Natural Language Problems

- 3 types of major problem with user requirements written in natural language: –

  i. Lack of clarity : difficult to use language in a precise & unambiguous way without making the document wordy & difficult to read

  ii. Requirements confusion : functional, non-functional requirements, system goals, design info may not be clearly distinguished

  iii. Requirements amalgamation : several requirements expressed as a single requirement

Natural Language Problems - Example

A user requirement for accounting system in LIBSYS.

4.5 LIBSYS shall provide a financial accounting system that maintains records of all payments made by users of the system. System managers may configure this system so that regular users may receive discounted rates.

Discount rate configuration

Payment Record Maintenance: Add, Edit, Delete

Natural Language Problems

Reasons why natural language system requirements specification is not good:

- Natural language understanding relies on the specification readers and writers using the same words for the same concept.
- A natural language requirements specification is over-flexible
- Requirements are not partitioned effectively by the language itself

Notations for System Requirements Specification (SRS)

Solution?

- Structured natural language – Decision tables, template/ table to specify system input, process, output, etc. (like a data dictionary)
- Design description language – uses a language like programming language E.g. PSL/PSA, RSL with special terminologies like programming language
- Graphical notations – DFD, flowchart, use-case diagrams, sequence diagrams, activity diagrams etc.
- Mathematical specifications – based on mathematical concepts such as finite-state machines or sets. Formal specification like Z-specification

Software Requirements Document

Software Requirements Document

- System requirements are expressed in a software requirements document.
- The software requirements document (sometimes known as SRS – software requirements specification) is the official statement of what is required of the system developer.
- It should include both:
  - the user requirements for a system and
  - a detailed specification of the system requirements

Software Requirements Document

- This document is not a design document. It should set out what the system should do without specifying how it should be done.
- In principle, the requirements set out in this document ought to be complete and consistent.
- All system functions should be specified and requirements should not conflict.

Software Requirements Document

The requirements document has a diverse set of users:

i) System customers

Specify the requirements and read them to check that they meet their needs. They specify changes to the requirements

ii) (Project) Managers

Use the requirements document to plan a bid for the system and to plan the system development process

Software Requirements Document

The requirements document has a diverse set of users:

iii) System engineers

Use the requirements to understand what system is to be developed

iv) System test engineers

Use the requirements to develop validation tests for the system

v) System maintenance engineers

Use the requirements to help understand the system and the relationships between its parts

Software Requirements Document

The structure of SRS:

Preface

Introduction

- ✓Purpose
- ✓Scope
- ✓Overview
- ✓Business Context

Software Requirements Document

The structure of SRS:

Preface

Introduction

Glossary

✓Definition of terms

Software Requirements Document

The structure of SRS:

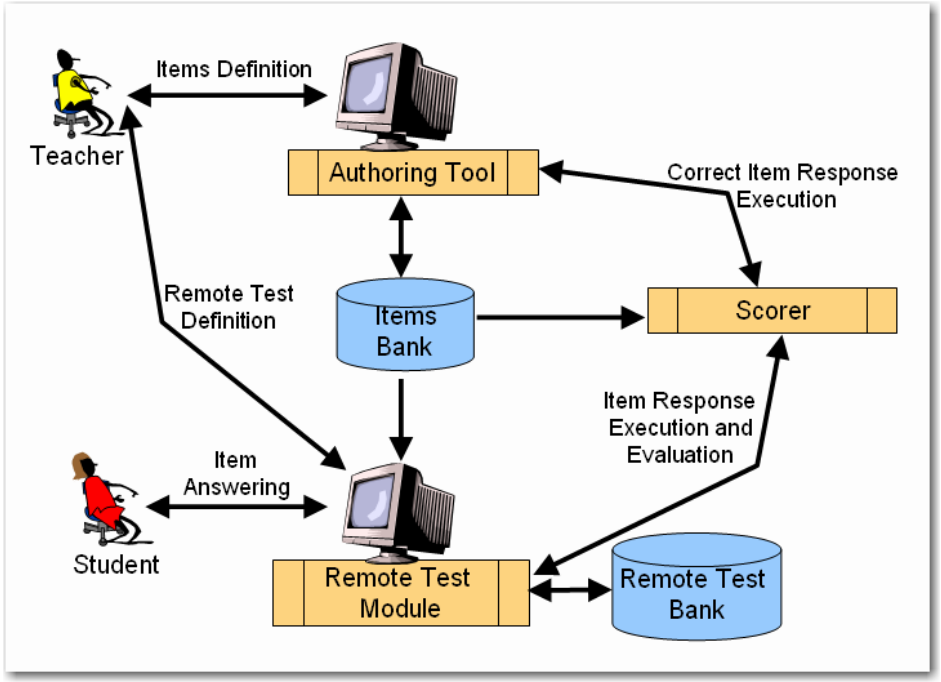
Preface
Introduction
Glossary
User requirements definition

✓General

Software Requirements Document

The structure of SRS:

Preface
Introduction
Glossary
User requirements definition
System architecture



Software Requirements Document

The structure of SRS:

Preface
Introduction
Glossary
User requirements definition
System architecture
System requirements specification

- ✓Functional
- ✓Non-Functional
- ✓Interface
- ✓Performance

Software Requirements Document

The structure of SRS:

Preface
Introduction
Glossary
User requirements definition
System architecture
System requirements specification
System models

✓Chapter 6

Software Requirements Document

The structure of SRS:

Preface
Introduction
Glossary
User requirements definition
System architecture
System requirements specification
System models
System evolution

✓Version

Software Requirements Document

The structure of SRS:

Preface
Introduction
Glossary
User requirements definition
System architecture
System requirements specification
System models
System evolution
Appendices

✓References (supporting documents)

Software Requirements Document

The structure of S

Preface
Introduction
Glossary
User requirements d
System architecture
System requirement
System models
System evolution
Appendices
Index

Index

Software Requirements Document

The structure of SRS:

Separate chapter or appendices:

- Hardware
- Database requirements