ECE 321

Software Requirements Engineering

LAB 5. Software Requirements Specification

IEEE 830 Standard

Lab Instructor: Chaari Ahmed

Department of Electrical and Computer Engineering

University of Alberta

SRS – IEEE Standard 830

1. Introduction

1.1. Purpose

delineate purpose of the SRS and specify the intended audience for the SRS

1.2. Scope

identify products, explain what product will and will not do

1.3. Definitions

all terms, acronyms and abbreviations

1.4. References

all documents referenced elsewhere

1.5. Overview

describe what the rest of document contains

SRS – IEEE Standard 830

2. Overall description

2.1. Product perspective

put product into perspective with other products (block diagram showing major components)

2.1.1. System Interfaces

2.1.2. User interfaces

2.1.3. Hardware interfaces

2.1.4. Software interfaces

2.1.5. Communication interfaces

2.1.6. Memory constrains

2.1.7. Operations

2.1.8. Site adaptation requirements

2.2. Product functions

summary of major functions

2.3. User characteristics

educational level, experience, technical expertise

SRS – IEEE Standard 830

2.4. Constraints

description of relevant regulatory policies, hardware limitations, interfaces to other applications, parallel operation

2.5. Assumptions and dependencies

factors affecting SRS, changes to them will change SRS

2.6. Apportioning of requirements

requirements that may be delayed to future versions

3. Specific requirements

should contain all the software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements

3.1. External interfaces

detailed description of all inputs into and outputs from the software system

3.2. Functions

define fundamental actions (the bulk of the document goes here)

3.3. Performance requirements

static and dynamic numerical requirements

SRS – IEEE Standard 830

3.4. Logical database requirements

logical requirements for any information that is to be placed into a database (frequency of use, accessing capabilities, data entities and their relationships, etc.)

3.5. Design constraints

language, technology, standards

3.6. Software system attributes

reliability, availability, security, maintainability, portability, etc.

4. Supporting information

4.1 Table of Contents and Index

4.2 Appendices

SRS – IEEE Standard 830

3.2 Functional Requirements – LAYOUTS

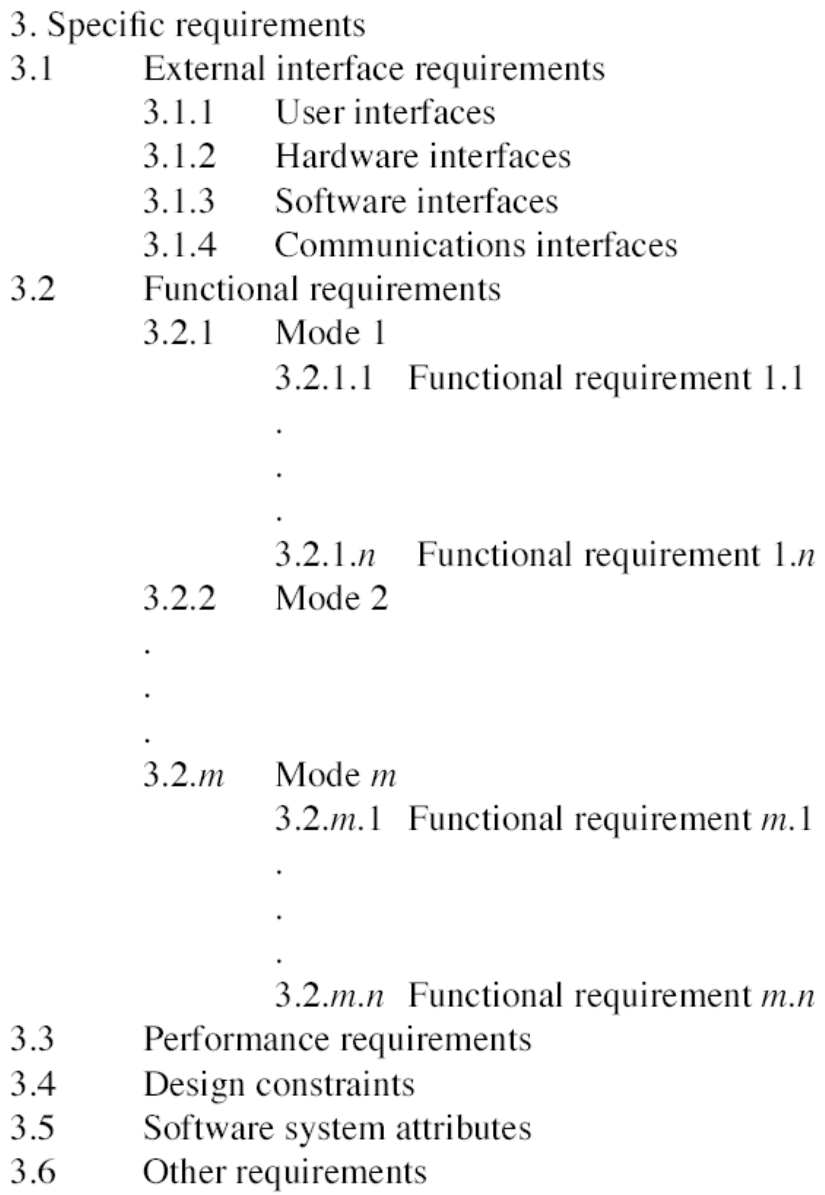
- **system mode** (systems that behave differently depending on a mode of operation)
- **user class** (systems that provide different sets of functions to different classes of users)
- **objects** (systems that are centered on providing functionality for individual objects in the system)
- **features** (systems that are centered on providing features to the user)
- **stimulus** (systems that can be best organized by describing their functions in terms of stimuli (inputs))
- **response** (systems that can be best organized by describing functions in support of generation of a response)

SRS – IEEE Standard 830

A.1 Template of SRS Section 3 organized by mode: Version 1

System mode

- some systems behave quite differently depending on the mode of operation
  - a control system may have different sets of functions depending on its mode: training, normal, or emergency
  - the choice depends on whether interfaces and performance are dependent on mode



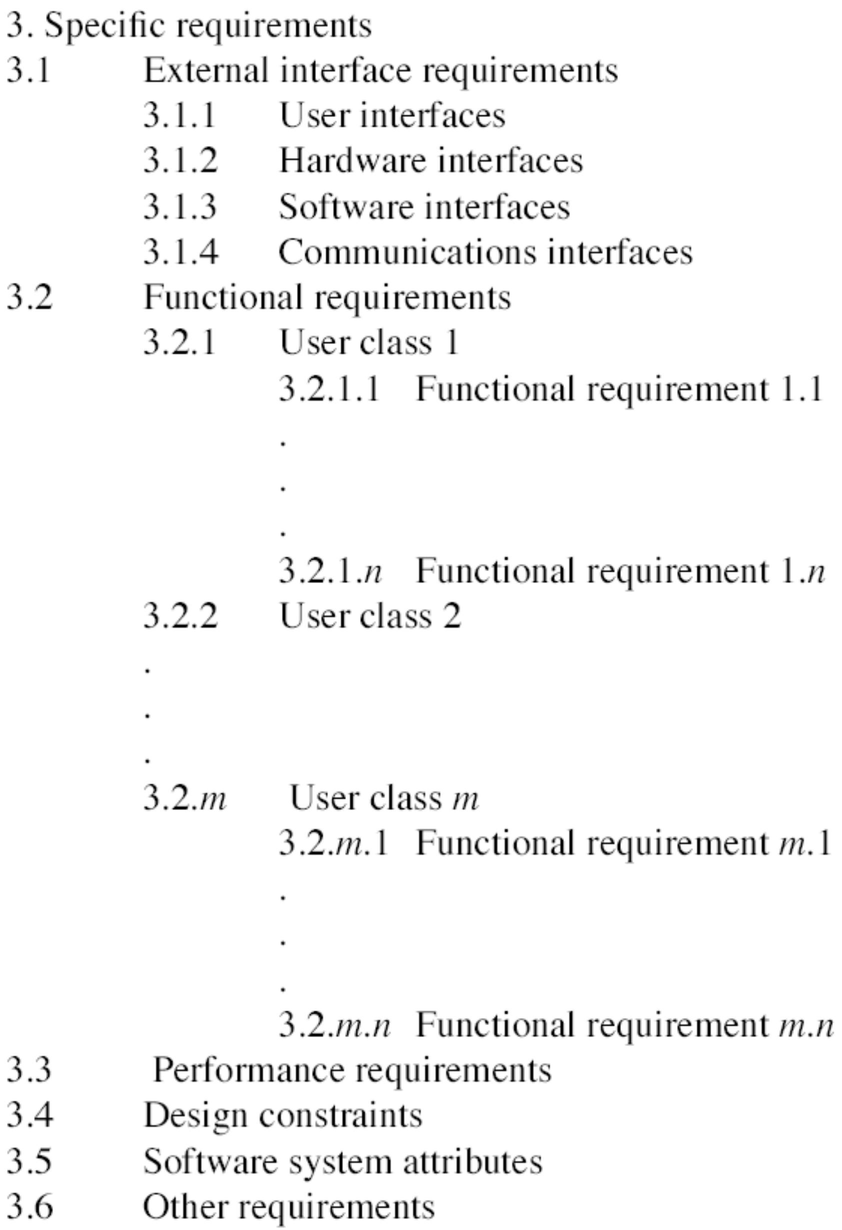
3. Specific requirements
3.1 External interface requirements
    3.1.1 User interfaces
    3.1.2 Hardware interfaces
    3.1.3 Software interfaces
    3.1.4 Communications interfaces
3.2 Functional requirements
    3.2.1 Mode 1
        3.2.1.1 Functional requirement 1.1
        .
        .
        .
        3.2.1.n Functional requirement 1.n
    3.2.2 Mode 2
    .
    .
    .
    3.2.m Mode m
        3.2.m.1 Functional requirement m.1
        .
        .
        .
        3.2.m.n Functional requirement m.n
3.3 Performance requirements
3.4 Design constraints
3.5 Software system attributes
3.6 Other requirements

SRS – IEEE Standard 830

User class

- some systems provide different sets of functions to different classes of users
  - for example, an elevator control system presents different capabilities to passengers, maintenance workers, and fire fighters

A.3 Template of SRS Section 3 organized by user class

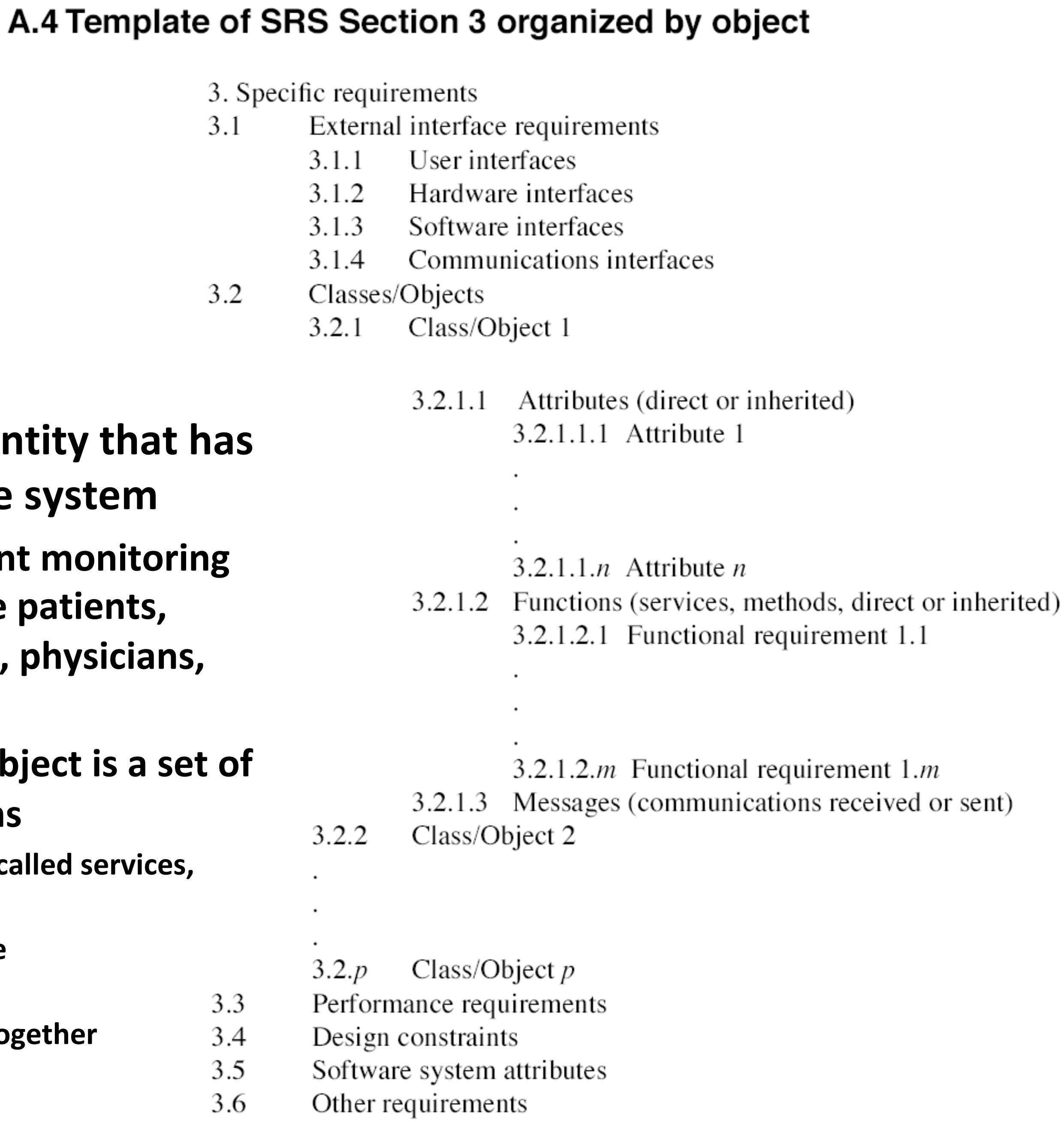


- 3. Specific requirements
- 3.1 External interface requirements
  - 3.1.1 User interfaces
  - 3.1.2 Hardware interfaces
  - 3.1.3 Software interfaces
  - 3.1.4 Communications interfaces
- 3.2 Functional requirements
  - 3.2.1 User class 1
    - 3.2.1.1 Functional requirement 1.1
    - .
    - .
    - .
    - 3.2.1.*n* Functional requirement 1.*n*
  - 3.2.2 User class 2
  - .
  - .
  - .
  - 3.2.*m* User class *m*
    - 3.2.*m*.1 Functional requirement *m*.1
    - .
    - .
    - .
    - 3.2.*m*.*n* Functional requirement *m*.*n*
- 3.3 Performance requirements
- 3.4 Design constraints
- 3.5 Software system attributes
- 3.6 Other requirements

SRS – IEEE Standard 830

Objects

- an **object** is real-world entity that has a counterpart within the system
  - for example, in a patient monitoring system, objects include patients, sensors, nurses, rooms, physicians, etc.
  - associated with each object is a set of attributes and functions
    - these functions are also called services, methods, or processes
    - sets of objects may share attributes and services
      - these are grouped together as classes



A.4 Template of SRS Section 3 organized by object

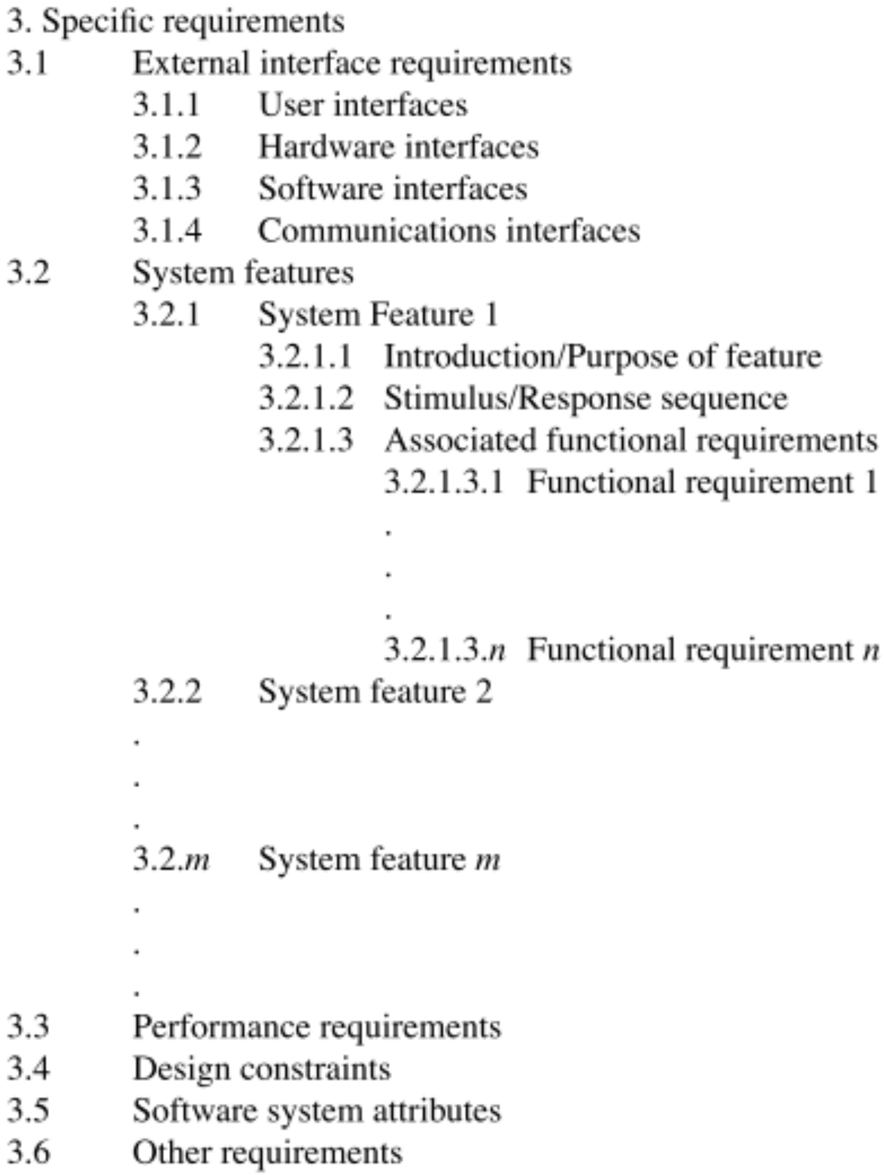
- 3. Specific requirements
- 3.1 External interface requirements
  - 3.1.1 User interfaces
  - 3.1.2 Hardware interfaces
  - 3.1.3 Software interfaces
  - 3.1.4 Communications interfaces
- 3.2 Classes/Objects
  - 3.2.1 Class/Object 1
    - 3.2.1.1 Attributes (direct or inherited)
      - 3.2.1.1.1 Attribute 1
      - .
      - .
      - .
      - 3.2.1.1.*n* Attribute *n*
    - 3.2.1.2 Functions (services, methods, direct or inherited)
      - 3.2.1.2.1 Functional requirement 1.1
      - .
      - .
      - .
      - 3.2.1.2.*m* Functional requirement 1.*m*
    - 3.2.1.3 Messages (communications received or sent)
  - 3.2.2 Class/Object 2
  - .
  - .
  - .
  - 3.2.*p* Class/Object *p*
- 3.3 Performance requirements
- 3.4 Design constraints
- 3.5 Software system attributes
- 3.6 Other requirements

A.5 Template of SRS Section 3 organized by feature

SRS – IEEE Standard 830

Feature

- a **feature** is an externally desired service by the system that may require a sequence of inputs to effect the desired result
  - for example, in a telephone system, features include local call, call forwarding, and conference call
  - each feature is generally described in a sequence of stimulus-response pairs

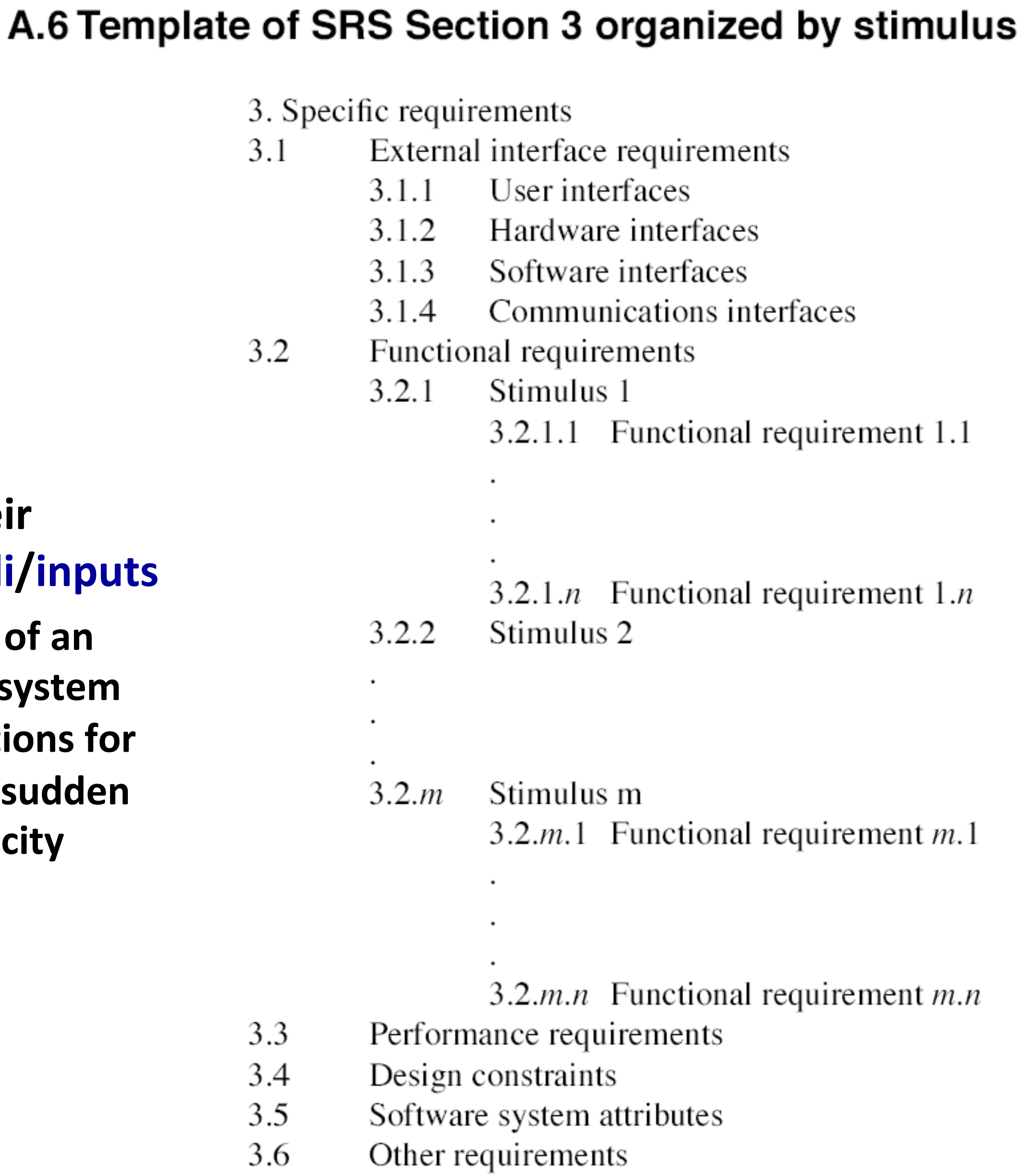


3. Specific requirements
- 3.1 External interface requirements
  - 3.1.1 User interfaces
  - 3.1.2 Hardware interfaces
  - 3.1.3 Software interfaces
  - 3.1.4 Communications interfaces
- 3.2 System features
  - 3.2.1 System Feature 1
    - 3.2.1.1 Introduction/Purpose of feature
    - 3.2.1.2 Stimulus/Response sequence
    - 3.2.1.3 Associated functional requirements
      - 3.2.1.3.1 Functional requirement 1
      - .
      - .
      - .
      - 3.2.1.3.*n* Functional requirement *n*
  - 3.2.2 System feature 2
  - .
  - .
  - .
  - 3.2.*m* System feature *m*
  - .
  - .
  - .
- 3.3 Performance requirements
- 3.4 Design constraints
- 3.5 Software system attributes
- 3.6 Other requirements

SRS – IEEE Standard 830

Stimulus

- some systems can be best organized by describing their functions in terms of stimuli/inputs
  - for example, the functions of an automatic aircraft landing system may be organized into sections for loss of power, wind shear, sudden change in roll, vertical velocity excessive, etc.



A.6 Template of SRS Section 3 organized by stimulus

3. Specific requirements

3.1 External interface requirements

- 3.1.1 User interfaces
- 3.1.2 Hardware interfaces
- 3.1.3 Software interfaces
- 3.1.4 Communications interfaces

3.2 Functional requirements

- 3.2.1 Stimulus 1
  - 3.2.1.1 Functional requirement 1.1
  - .
  - .
  - .
  - 3.2.1.*n* Functional requirement 1.*n*
- 3.2.2 Stimulus 2
- .
- .
- .
- 3.2.*m* Stimulus m
  - 3.2.*m*.1 Functional requirement *m*.1
  - .
  - .
  - .
  - 3.2.*m*.*n* Functional requirement *m*.*n*

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

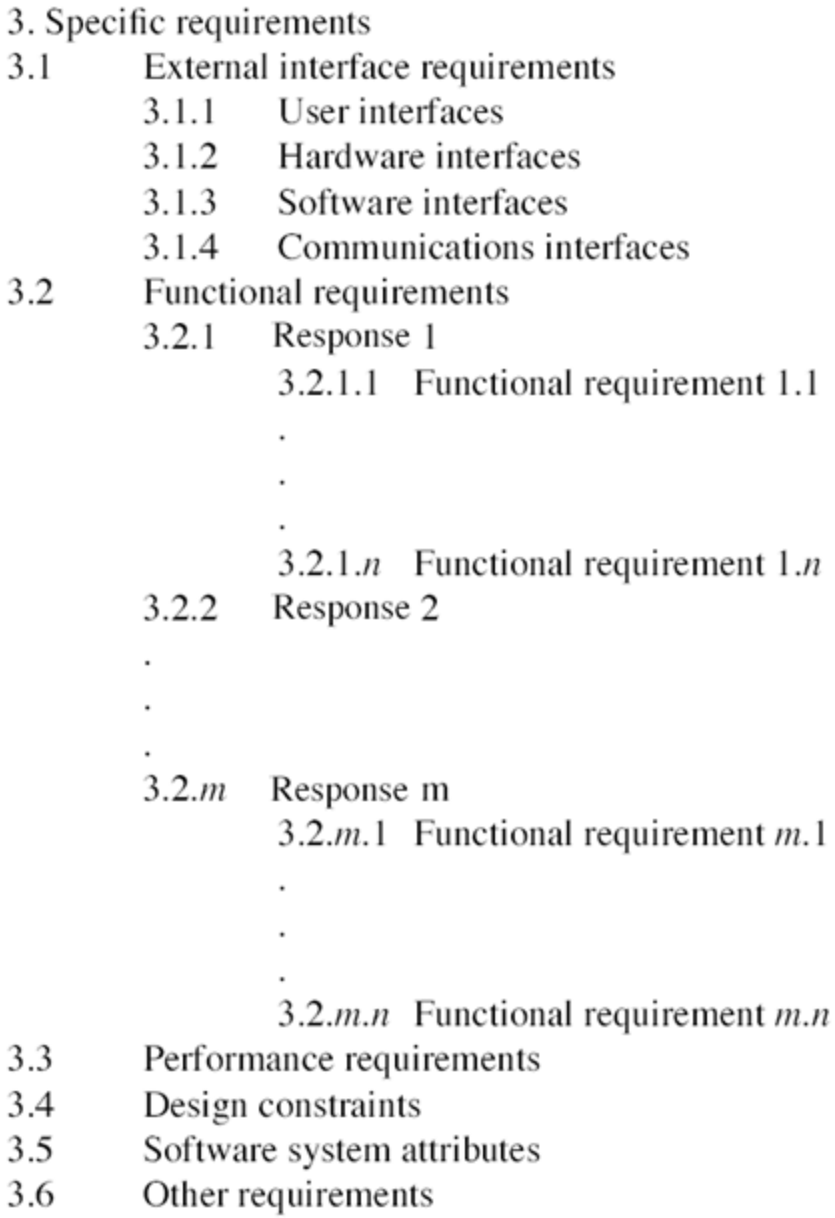
3.6 Other requirements

A.6 Template of SRS Section 3 organized by stimulus

SRS – IEEE Standard 830

Response

- some systems can be best organized by describing all the functions in support of the generation of a response/output
  - for example, the functions of a personnel system may be organized into sections corresponding to all functions associated with generating paychecks, all functions associated with generating a current list of employees, etc

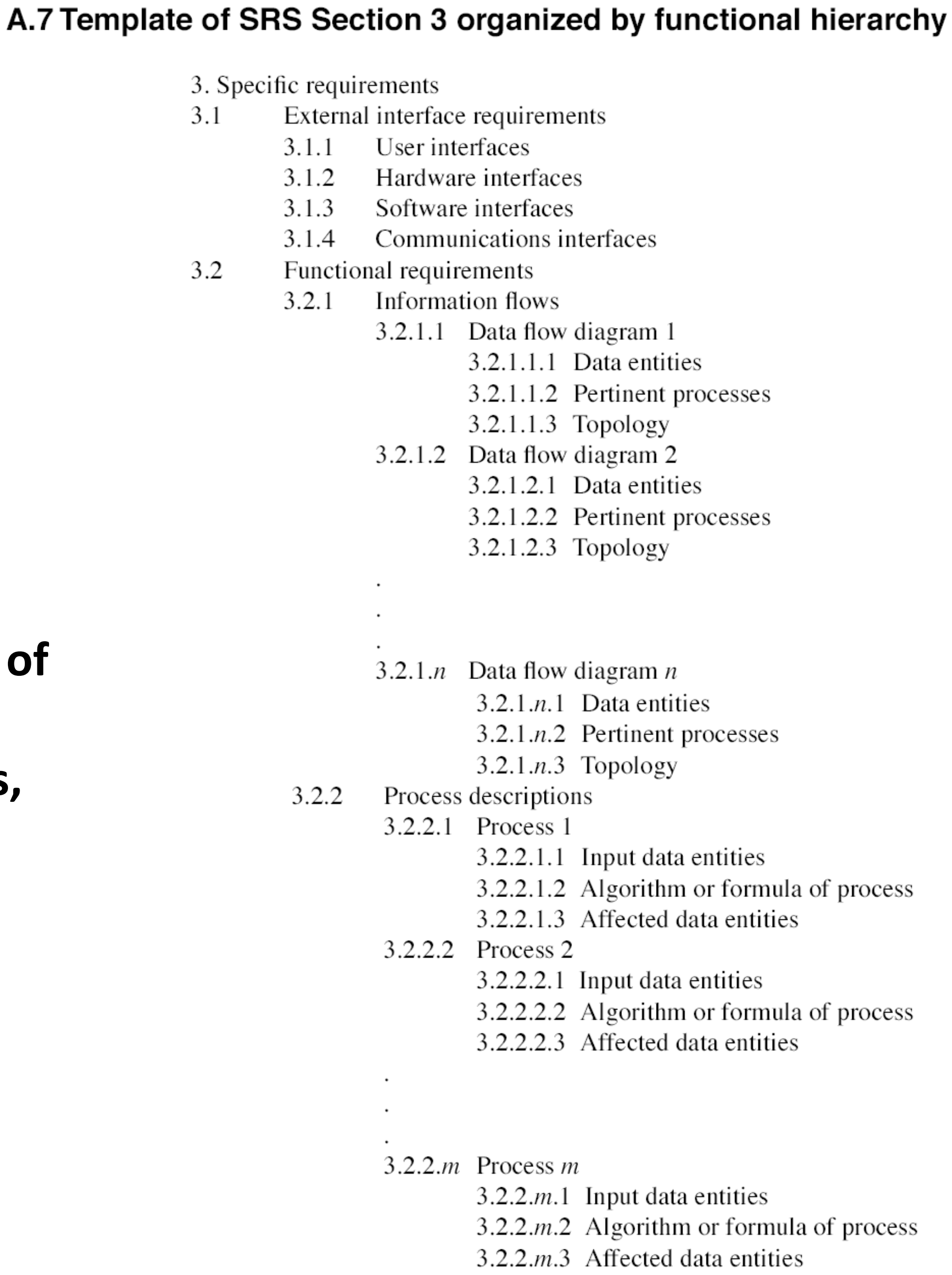


3. Specific requirements
3.1 External interface requirements
    3.1.1 User interfaces
    3.1.2 Hardware interfaces
    3.1.3 Software interfaces
    3.1.4 Communications interfaces
3.2 Functional requirements
    3.2.1 Response 1
        3.2.1.1 Functional requirement 1.1
        .
        .
        .
        3.2.1.*n* Functional requirement 1.*n*
    3.2.2 Response 2
    .
    .
    .
    3.2.*m* Response m
        3.2.*m*.1 Functional requirement *m*.1
        .
        .
        .
        3.2.*m*.*n* Functional requirement *m*.*n*
3.3 Performance requirements
3.4 Design constraints
3.5 Software system attributes
3.6 Other requirements

SRS – IEEE Standard 830

Functional hierarchy

- when none of the above organizational schemes prove helpful, the overall functionality can be organized into a hierarchy of functions organized by either common inputs, common outputs, or common internal data access
  - data flow diagrams and data dictionaries can be used to show the relationships between and among the functions and data



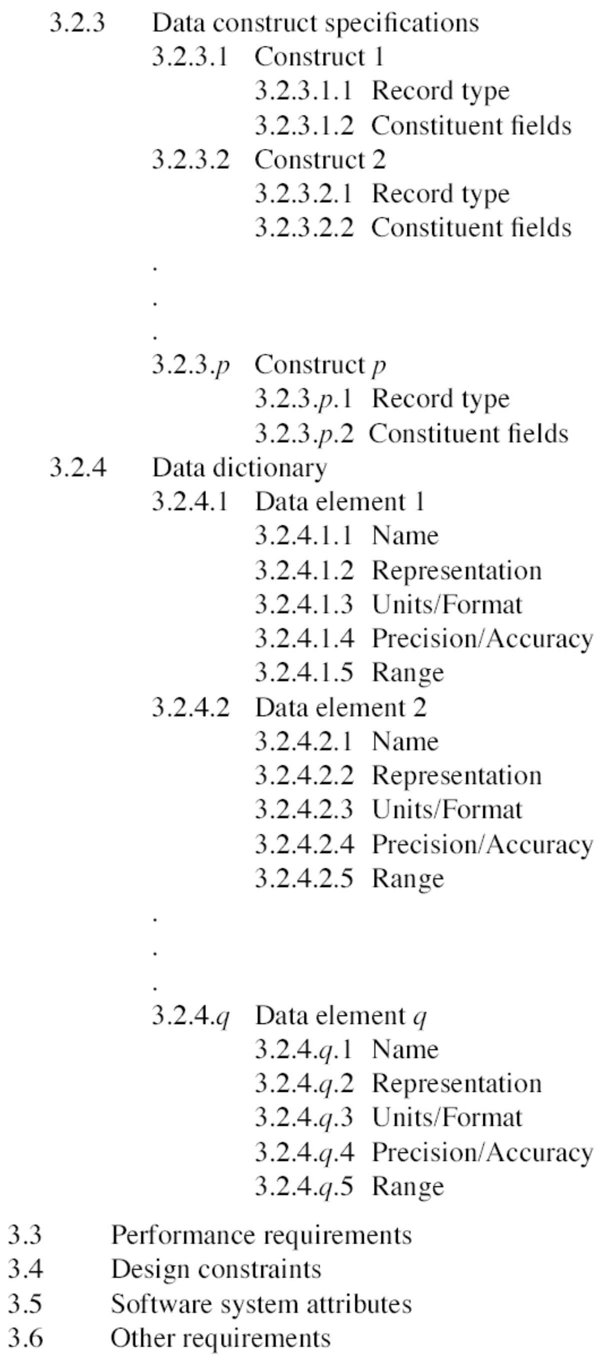
A.7 Template of SRS Section 3 organized by functional hierarchy

3. Specific requirements
3.1 External interface requirements
3.1.1 User interfaces
3.1.2 Hardware interfaces
3.1.3 Software interfaces
3.1.4 Communications interfaces
3.2 Functional requirements
3.2.1 Information flows
3.2.1.1 Data flow diagram 1
3.2.1.1.1 Data entities
3.2.1.1.2 Pertinent processes
3.2.1.1.3 Topology
3.2.1.2 Data flow diagram 2
3.2.1.2.1 Data entities
3.2.1.2.2 Pertinent processes
3.2.1.2.3 Topology
.
.
.
3.2.1.*n* Data flow diagram *n*
3.2.1.*n*.1 Data entities
3.2.1.*n*.2 Pertinent processes
3.2.1.*n*.3 Topology
3.2.2 Process descriptions
3.2.2.1 Process 1
3.2.2.1.1 Input data entities
3.2.2.1.2 Algorithm or formula of process
3.2.2.1.3 Affected data entities
3.2.2.2 Process 2
3.2.2.2.1 Input data entities
3.2.2.2.2 Algorithm or formula of process
3.2.2.2.3 Affected data entities
.
.
.
3.2.2.*m* Process *m*
3.2.2.*m*.1 Input data entities
3.2.2.*m*.2 Algorithm or formula of process
3.2.2.*m*.3 Affected data entities

SRS – IEEE Standard 830

Functional hierarchy

continued…



3.2.3 Data construct specifications

3.2.3.1 Construct 1

3.2.3.1.1 Record type

3.2.3.1.2 Constituent fields

3.2.3.2 Construct 2

3.2.3.2.1 Record type

3.2.3.2.2 Constituent fields

.

.

.

3.2.3.*p* Construct *p*

3.2.3.*p*.1 Record type

3.2.3.*p*.2 Constituent fields

3.2.4 Data dictionary

3.2.4.1 Data element 1

3.2.4.1.1 Name

3.2.4.1.2 Representation

3.2.4.1.3 Units/Format

3.2.4.1.4 Precision/Accuracy

3.2.4.1.5 Range

3.2.4.2 Data element 2

3.2.4.2.1 Name

3.2.4.2.2 Representation

3.2.4.2.3 Units/Format

3.2.4.2.4 Precision/Accuracy

3.2.4.2.5 Range

.

.

.

3.2.4.*q* Data element *q*

3.2.4.*q*.1 Name

3.2.4.*q*.2 Representation

3.2.4.*q*.3 Units/Format

3.2.4.*q*.4 Precision/Accuracy

3.2.4.*q*.5 Range

3.3 Performance requirements

3.4 Design constraints

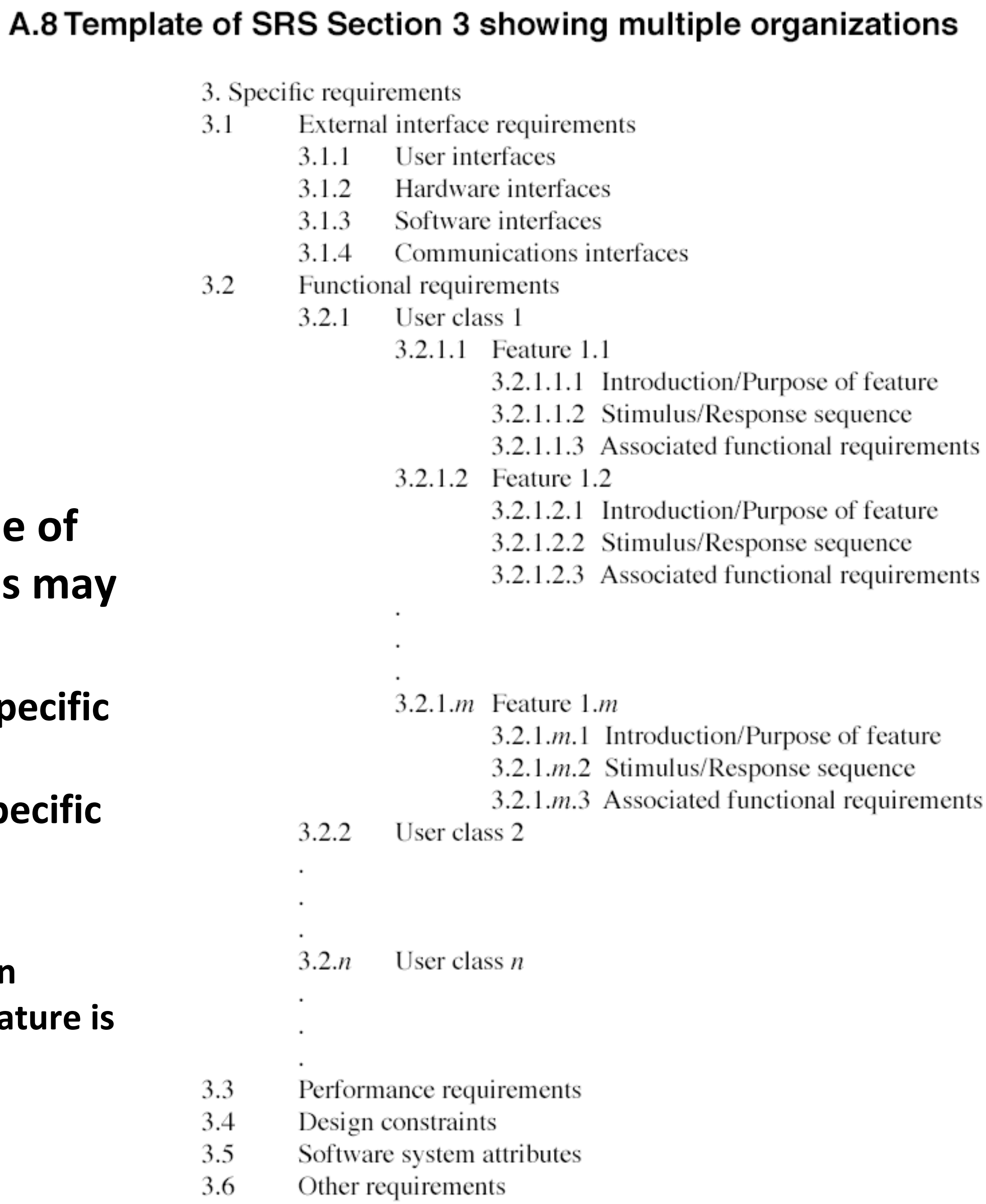
3.5 Software system attributes

3.6 Other requirements

SRS – IEEE Standard 830

Mixed mode

- whenever a new SRS is contemplated, more than one of the organizational techniques may be appropriate
  - in such cases, organize the specific requirements for multiple hierarchies tailored to the specific needs of the system under specification
    - for example, an organization combining user class and feature is shown on the right



A.8 Template of SRS Section 3 showing multiple organizations

3. Specific requirements
- 3.1 External interface requirements
  - 3.1.1 User interfaces
  - 3.1.2 Hardware interfaces
  - 3.1.3 Software interfaces
  - 3.1.4 Communications interfaces
- 3.2 Functional requirements
  - 3.2.1 User class 1
    - 3.2.1.1 Feature 1.1
      - 3.2.1.1.1 Introduction/Purpose of feature
      - 3.2.1.1.2 Stimulus/Response sequence
      - 3.2.1.1.3 Associated functional requirements
    - 3.2.1.2 Feature 1.2
      - 3.2.1.2.1 Introduction/Purpose of feature
      - 3.2.1.2.2 Stimulus/Response sequence
      - 3.2.1.2.3 Associated functional requirements

      .
      .
      .
    - 3.2.1.*m* Feature 1.*m*
      - 3.2.1.*m*.1 Introduction/Purpose of feature
      - 3.2.1.*m*.2 Stimulus/Response sequence
      - 3.2.1.*m*.3 Associated functional requirements
  - 3.2.2 User class 2

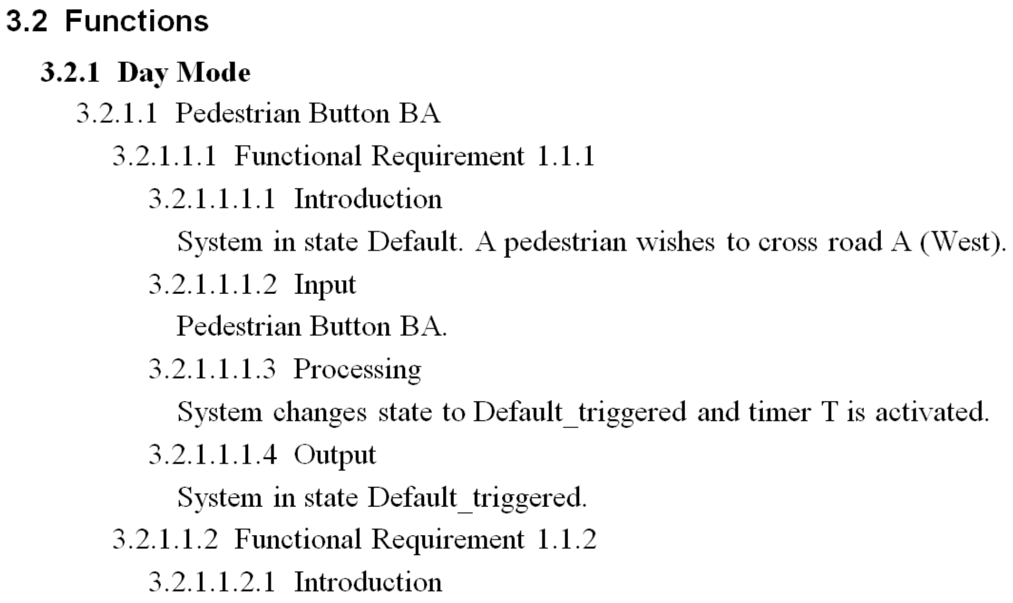
    .
    .
    .
  - 3.2.*n* User class *n*

    .
    .
    .
- 3.3 Performance requirements
- 3.4 Design constraints
- 3.5 Software system attributes
- 3.6 Other requirements

SRS – IEEE Standard 830

Mixed mode

- for the project you are required to use mixed mode that lists system mode first and stimulus second



3.2 Functions

3.2.1 Day Mode

3.2.1.1 Pedestrian Button BA

3.2.1.1.1 Functional Requirement 1.1.1

3.2.1.1.1.1 Introduction

System in state Default. A pedestrian wishes to cross road A (West).

3.2.1.1.1.2 Input

Pedestrian Button BA.

3.2.1.1.1.3 Processing

System changes state to Default_triggered and timer T is activated.

3.2.1.1.1.4 Output

System in state Default_triggered.

3.2.1.1.2 Functional Requirement 1.1.2

3.2.1.1.2.1 Introduction

Project – SRS Document

Each group must submit one document

- **15% of total grade**
- **Correct and complete information from Lab2-4**
- **Use the IEEE 830 SRS standard**
  - Consistency of format, expression and abbreviations
  - Include tables/graph if necessary
- **Single-spaced, using 12 font size, and with standard margins**
- **Only hardcopies will be accepted**
- **Due on Wednesday, December 12th, 9am MST**