Specification, Part 1

System Specification

Introduction to System Engineering

AARHUS
UNIVERSITET
INGENIØRHØJSKOLEN

Agenda

- System Specification
- Types of requirements
- Specifying requirements
- Finding requirements (Elicitation)
- Good / Bad Requirements
- Traceability

System engineers view of the world

System

Specifies /validates

Track and control

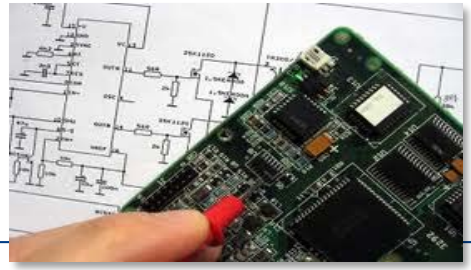
Customer

Uses

Control

Contracts

Development



Constructs

AARHUS UNIVERSITET
INGENIØRHØJSKOLEN

System engineers view of the world

Underspecified

System

Breaks

Delays

Misuses

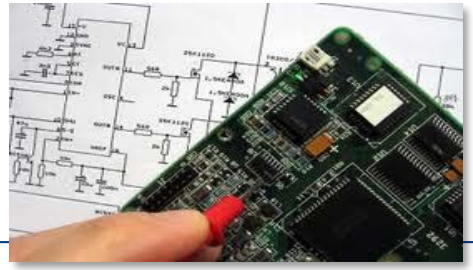
Surprises

Customer

Control

Idealistic expectations

Development



Partially implements

AARHUS UNIVERSITET
INGENIØRHØJSKOLEN

What is a system specification?

- "a statement that identifies a capability, characteristic, or quality factor of a system in order for it to have value by a user or a customer to solve a problem or achieve an objective"

  Ralph Young, *Requirements Engineering Handbook, 2004*

- Do the right thing
- The ***what*** and not the ***how***

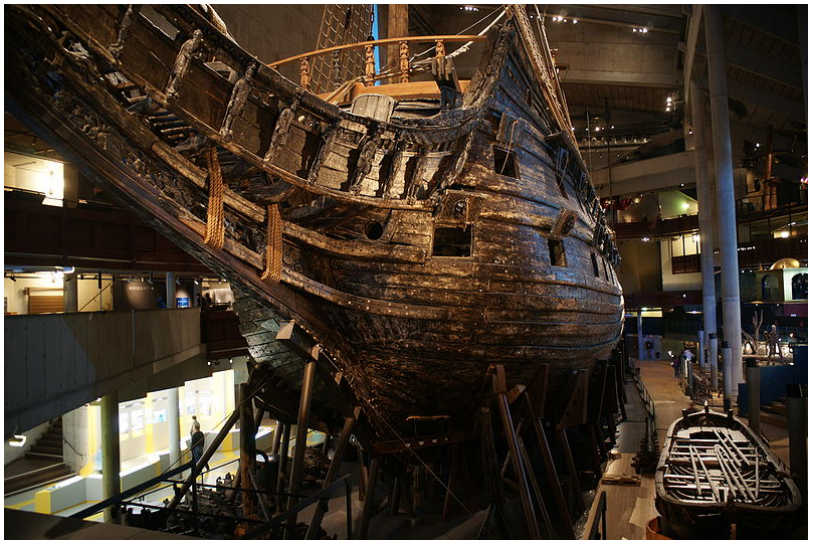
What is a system specification?

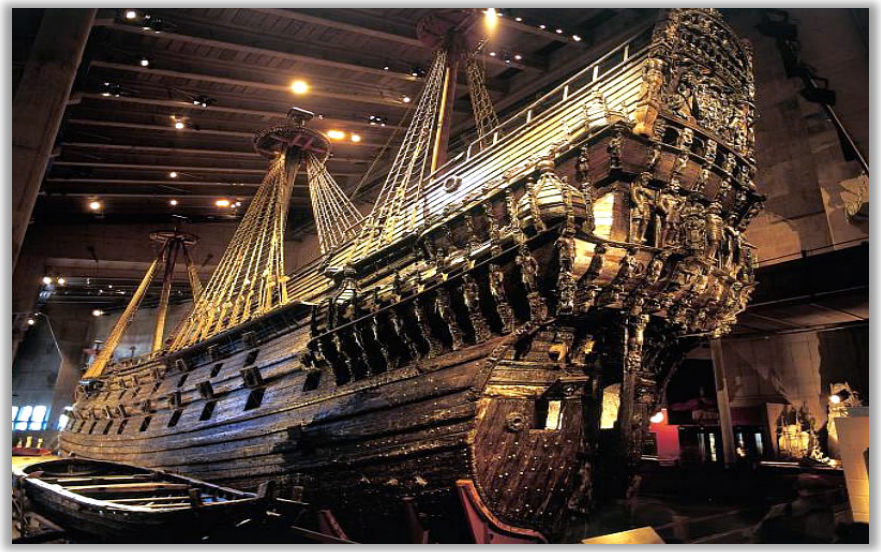
- The Stakeholders description of a desire functionality or behavior for a system to achieve an objective.
- Functional requirements state what the system is required to do.
- The primary input to the design process
- The baseline against which acceptance tests are carried out.

Why have focus on specification?

- Time, Effort, Price
- Correcting or changing functionality following specification
  - 3 x as much during the design phase
  - 5-10 x as much during implementation
  - 10-100 x as much after release

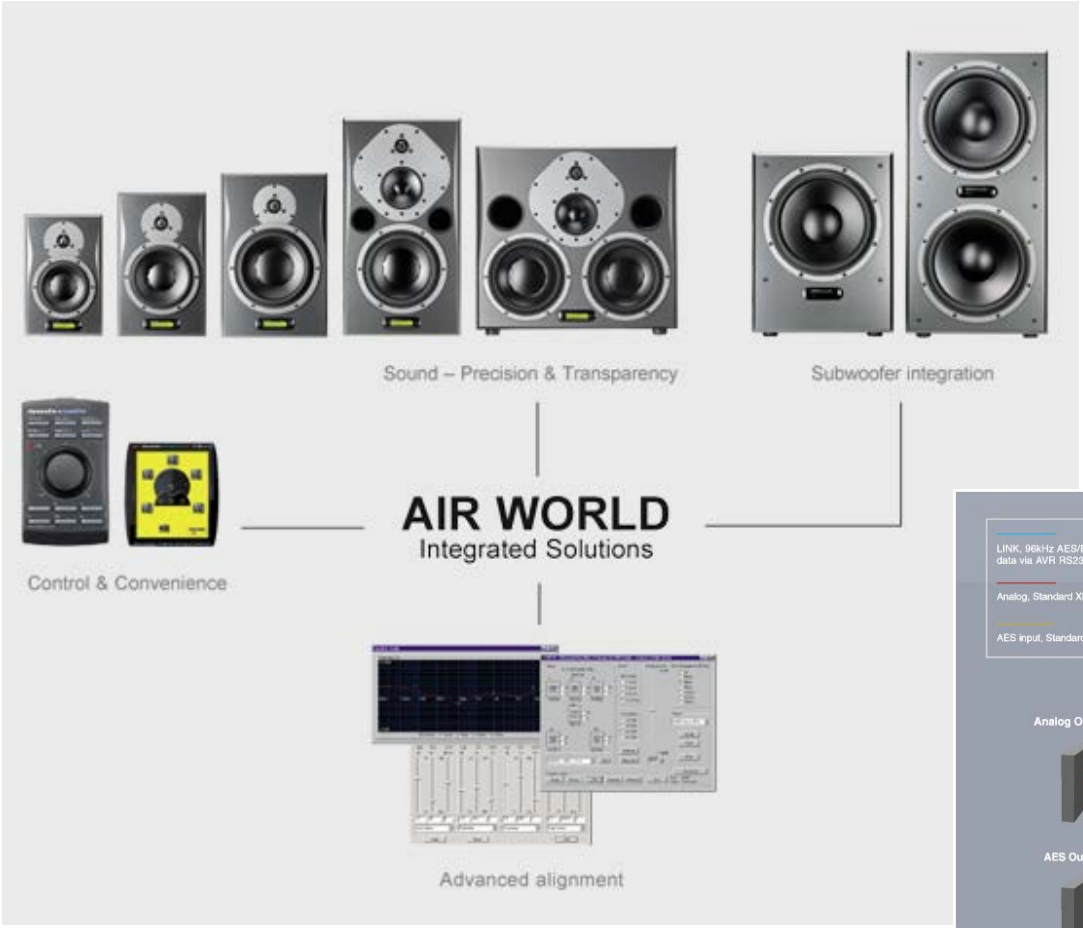
- Vasa, *10. aug. 1628 – †10. aug. 1628
- L: 69m H: 52m B: 11,7m D: 5m
- Requirements creep
- Inexperience
- Bad test conditions
- Schedule Pressure



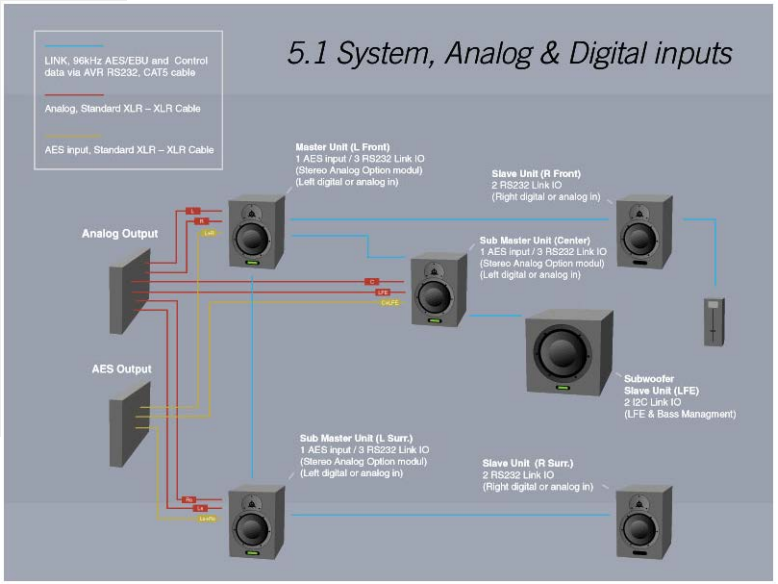


[Why the Vasa Sank: 10 Problems and Some Antidotes for Software Projects, *IEEE Software*, Fairley03]

Dynaudio AIR Series – Studio monitors



- Lack of specification
- Stereo and volume
- Multichannel 7.3



http://dynaudioprofessional.com/air-series/

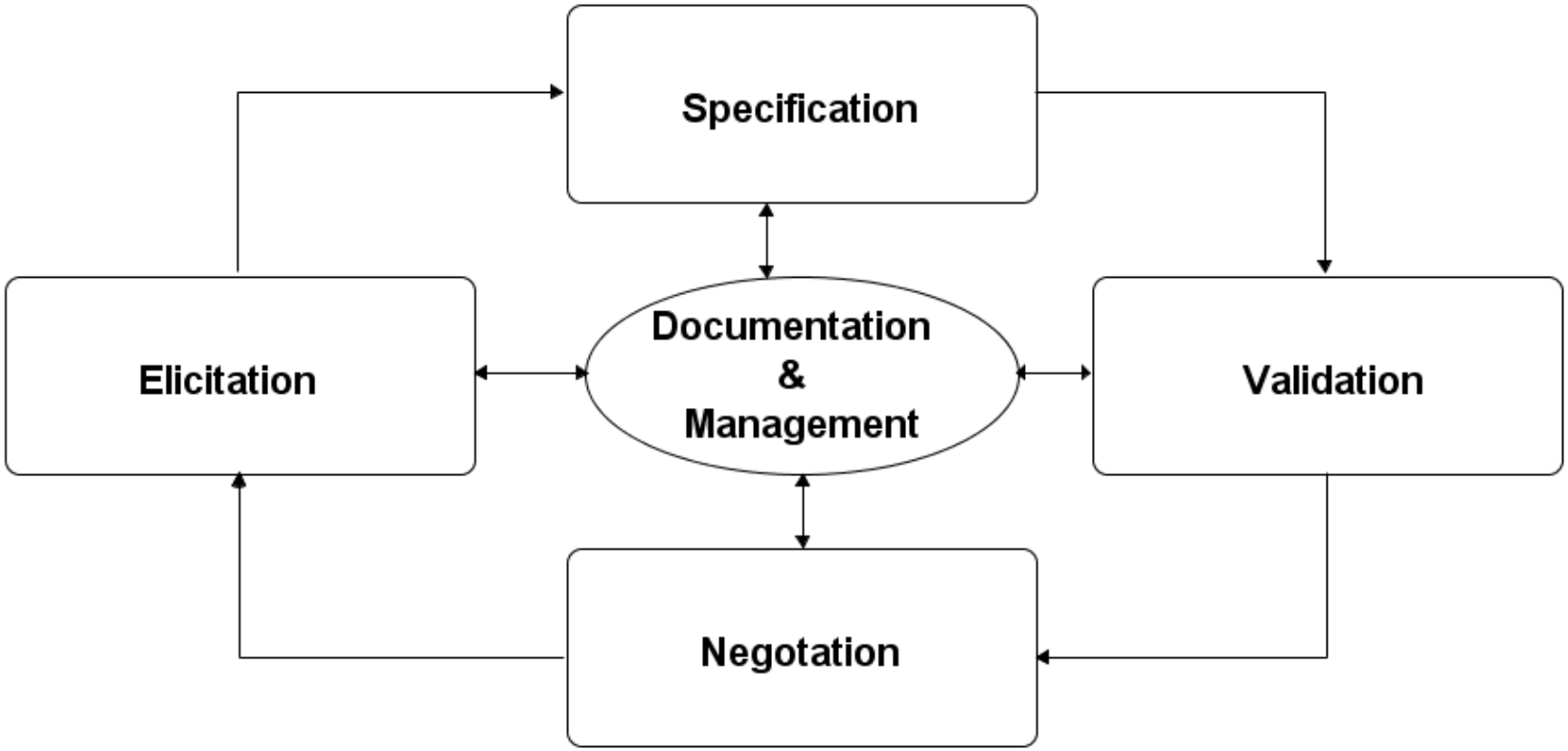
What to derive from these cases?

- A specification is more than functional requirements
- The specification should attempt to address:
  - Requirements creep / Requirements Change
  - Test conditions
  - Training
  - Realistic Schedule
  - External Constraints (physical, legal)
  - Ownership and Stakeholders
  - Existing systems/hardware

Who uses a specification

Stakeholder	Use of specification
Customer	Fullfillment of business goal Contract
Manager	Scheduling Progress measuring
System Engineer	Design Functionality Constraints
Test Engineer and QA personnel	Test planning Verification Validation

Specification Process



Requirements Specification

- Documentation of the specification
- Principally the outcome of elicitation
- Complete description of the behavior and constraints of the system to be developed
- Baseline for communication between stakeholders
- Often has high demands for versioning and traceability

Types of requirements

- ***Functional***
  - What the system should do (behaviours)
- ***Non-functional (Quality-demands)***
  - Qualities or criteria of the system, rather than specific behaviour
- Other categorizations exists:
  - Customer Requirements
  - Domain Requirements (Business Rules)

Prioritisation of Requirements

- MoSCoW Method (prioritisation technique)

  M - MUST (skal) have this
  S - SHOULD (bør) have this if possible
  C - COULD (kan) have this if it does not affect anything
  W - WON'T (vil ikke) have this time, but WOULD like in the future

- The system ***must*** be able to make a cup of coffee
- The system ***should*** be able to indicate need of service
- The system ***could*** be able to add milk
- The system ***won't*** be able to automatically refill water

Functional Requirements

- Functional
  - Describes system services
  - Requirement for each input and output
  - Behavioural requirements
  - Use Cases

Exercise 1

- Who would the stakeholders be?
- Write 5-10 functional requirements for the reverse vending machine (Use MoSCoW)

Quality Demands/Non-functional Requirements

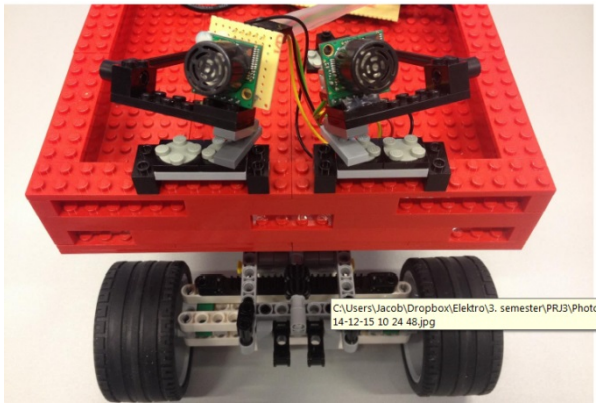
- Qualities or Constraints on the services or functions offered by the system

> Qualities are properties or characteristics of the system that its stakeholders care about and hence will affect their degree of satisfaction with the system.
> *[Defining Non-Functional Requirements,* Malan01]

- Quality demands/NFRs should satisfy two attributes
  - **Must be verifiable (measurable metrics)**
  - **Should be objective**

Example - Treasure Robot (3. Semester project)

- Driving robot
- Obstacle sensor
- Metal detector
- GPS



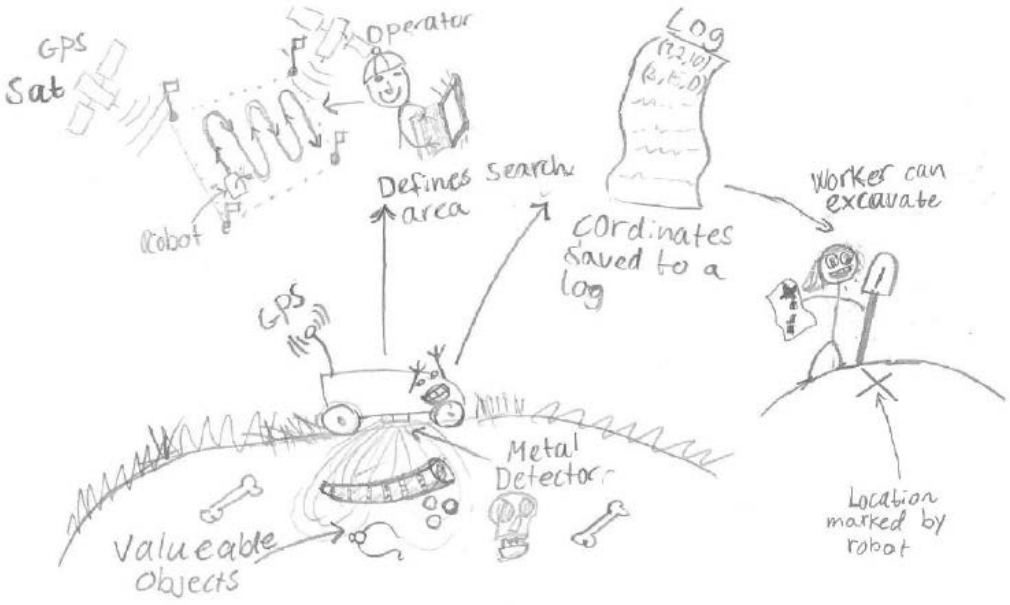
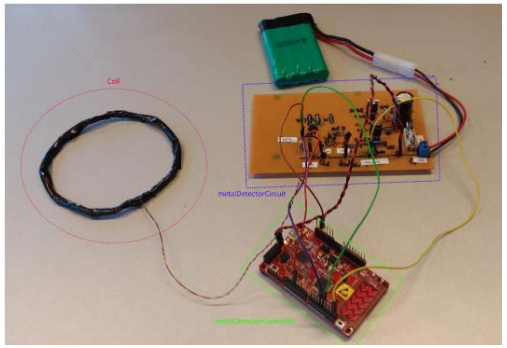


Figure 1 TreasureBot rich picture



Treasure Robot - Non-functional Requirements

3.1 Battery

3.1.1. Battery life **should** be minimum of 20 min when the car is in running mode
3.1.2. Battery life **should** be minimum of 1 hours when the car is idle

3.2 Robot

3.2.1. The robot **must** not exceed the dimension of 40 cm long, 25 cm wide and 15 cm tall
3.2.2. The robot **must** have enough storage capacity to be able to drive and record data for 20 min.
3.2.3. Should be able to save GPS-location every 5 sec. +/- ½ sec.

3.3 Metal detection

3.3.1. Must be able to detect metal to a depth of min. 5 cm from sensor, when driving on dirt or grass
3.3.2. Must be able to detect metal to a depth of min. 3 cm from sensor, when driving on gravel

3.4 Obstacle sensor

3.4.1. Must be able to detect a black box with dimensions 10x10x10 cm, from a distance of 1 m

3.5 GPS

3.3.1. Should have an accuracy of minimum 3 m radius on a clear day
3.3.2. Should have an accuracy of minimum 5 m radius on a cloudy day

Types of requirements

- FURPS+ (Robert Grady, Hewlett-Packard)
  - **F**unctionality
  - **U**sability
  - **R**eliability
  - **P**erformance
  - **S**upportability
  - \+ (Design and Physical constraints, Interfaces, Legal, Test, Reuse, Economic constraints, Aesthetics, Comprehensibility, Technology tradeoffs)

Usability

- Characteristics
  - Operability
  - Accessibility
  - User Interface (If any)
  - Documentation
- Metrics
  - ~~The system should be easy to use~~
  - Max. number of errors made by users for a specific task over a time period.
  - Time to learn a certain functionality

Reliability (1/2)

- Characteristics
  - Reliability
    - is the probability of a system to perform a required function under stated conditions
  - Availability
    - is a function of how often failures occur, repair time and maintenance interval
  - Maintainability
    - is the ability of a system to restore to a specified condition

Reliability (2/2)

- Metrics
  - Reliability
    - Mean time between failure (MTBF)
  - Availability

$$\frac{MTBF}{MTBF+MTTR} == \frac{UPTIME}{UPTIME+DOWNTIME}$$

  - Maintainability
    - Mean time to restore (MTTR)

Performance

- Characteristics
  - Throughput
  - Response time
  - Start-up time
  - Capacity & Efficiency Constraints
- Metrics
  - Time
  - Specifics is out of scope for this course
  - *95% of the transactions shall be processed in less than 1 second at 80 % load*

Supportability

- Characteristics
  - Compatibility,
  - Installability,
  - Localizability
  - Maintainability,
- Metrics
  - Same as with Usability
  - Measure of success under specified scenarios

+

- Legal
  - Data Protection Act, Health and Safety act,
- Technology trade off
  - Balance between two incompatible features
- Test
  - Conditions, Environment, Access
- Reuse
  - Existing systems, parts, modules
- Environmental
  - RoHS, WEEE, EMC, etc.

Exercise 2

- Find 5-10 NFR / Quality requirements for this coffee machine using (F)URPS+

Usability

Reliability

Performance

Supportability

Challenges in Elicitation

- The larger the project, the more difficult it is to grasp
- Users' expectations are often unrealistic, and compromises must be made between economy and features

Requirements Elicitation Techniques

- Techniques
  - Interview
  - Stakeholder analysis
  - Brainstorm
  - Prototype
  - (Requirements Workshop)
  - (Task Demonstration)
  - (Role Playing)

Interview

- Simple direct technique
- Context-free questions can help achieve bias-free interviews
- Convergence on some common needs will initiate a “requirements repository” for use during the project.
- A questionnaire is no substitute for an interview

Stakeholder analysis

- Who are the stakeholders?
- What are their goals?
- Which risks and costs do they see?

Brainstorm

- Brainstorming involves both idea generation and idea reduction
- The most creative, innovative ideas often result from combining, seemingly unrelated ideas
- Various voting techniques may be used to prioritize the ideas created
- Open to all suggestions
- Suggestions often spawn new ideas

Prototype session

- Preliminary model built for demonstration purposes
- The customer may be more likely to view the prototype and react to it, than to read the specification and react to it
- The prototype provides quick feedback
- Prototype displays unanticipated aspects of the systems behavior

Good Requirements

- Correct – specifying something actually needed
- Unambiguous – only one interpretation
- Complete – includes all significant requirements
- Consistent – no requirements conflict
- Verifiable – all requirements can be proven by test
- Modifiable – changes can easily be made to the requirements
- Traceable – the origin of each requirement is clear

Requirements Traceability Matrix

- Determine the two-way mapping between Requirements and Features/Test
- Are all features mapped to a requirement? And are each requirements fulfilled by a feature?
- Requirements and Test/Verification, or Requirements and Features

Requirements	REQ 1	REQ 2	REQ 3	REQ 4	REQ 5
Test Cases					
1.1	X	X			
1.2			X		
1.3			X		
2.1			X		
2.2	X	X		X	X
2.3	X	X			
3.1	X	X	X	X	X

A specification that will not fit on one page of 8.5x11 inch paper cannot be understood.

Mark Ardis

Professor
Rochester Institute of Technology

Exercise 3 - BeoSound F (pp.13-15)

- Write 10-20 good requirements
- (F)URPS+
  - Usability
  - Reliability: MTBF and availability
  - Performance
  - Supportability
- MoSCoW
  - Must (skal)
  - Should (bør)
  - Could (kunne)

Questions