Software Requirements Specification

- Main aim of requirements specification:
  - systematically organize the requirements arrived during requirements analysis
  - document requirements properly.

Software Requirements Specification

- The SRS document is useful in various contexts:
  - statement of user needs
  - contract document
  - reference document
  - definition for implementation

Software Requirements Specification: A Contract Document

- Requirements document is a reference document.
- SRS document is a contract between the development team and the customer.
  - Once the SRS document is approved by the customer,
    - any subsequent controversies are settled by referring the SRS document.

Software Requirements Specification: A Contract Document

- Once customer agrees to the SRS document:
  - development team starts to develop the product according to the requirements recorded in the SRS document.
- The final product will be acceptable to the customer:
  - as long as it satisfies all the requirements recorded in the SRS document.

SRS Document (CONT.)

- The SRS document is known as black-box specification:
  - the system is considered as a black box whose internal details are not known.
  - only its visible external (i.e. input/output) behavior is documented.

SRS Document (CONT.)

- SRS document concentrates on:
  - what needs to be done
  - carefully avoids the solution ("how to do") aspects.
- The SRS document serves as a contract
  - between development team and the customer.
  - Should be carefully written

SRS Document (CONT.)

- The requirements at this stage:
  - written using end-user terminology.
- If necessary:
  - later a formal requirement specification may be developed from it.

Properties of a good SRS document

- It should be concise
  - and at the same time should not be ambiguous.
- It should specify what the system must do
  - and not say how to do it.
- Easy to change.,
  - i.e. it should be well-structured.
- It should be consistent.
- It should be complete.

Properties of a good SRS document (cont...)

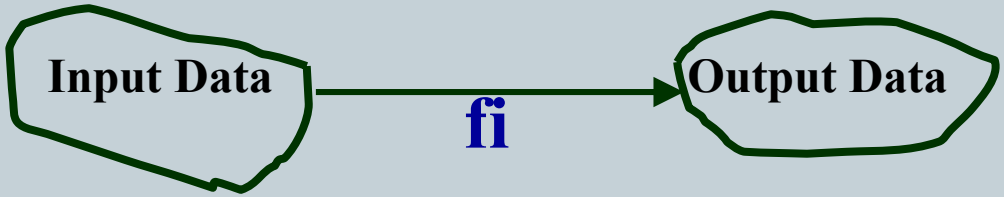
- It should be traceable
  - you should be able to trace which part of the specification corresponds to which part of the design and code, etc and vice versa.
- It should be verifiable

SRS Document (CONT.)

- SRS document, normally contains three important parts:
  - functional requirements,
  - nonfunctional requirements,
  - constraints on the system.

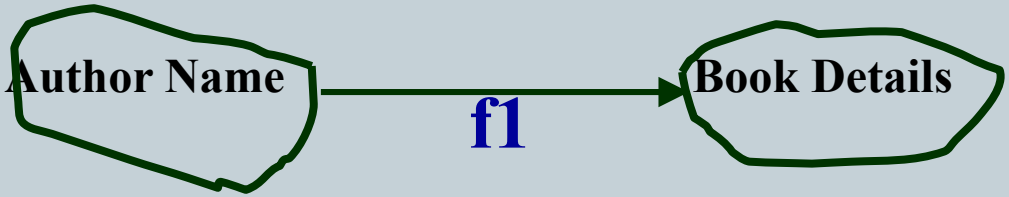
SRS Document (CONT.)

- It is desirable to consider every system:
  - performing a set of functions {fi}.
  - Each function fi considered as:
  - transforming a set of input data to corresponding output data.



Example: Functional Requirement

- F1: Search Book
  - Input:
    - an author's name:
  - Output:
    - details of the author's books and the locations of these books in the library.



Functional Requirements

- Functional requirements describe:
  - A set of high-level requirements
  - Each high-level requirement:
    - takes in some data from the user
    - outputs some data to the user
  - Each high-level requirement:
    - might consist of a set of identifiable functions

Functional Requirements

- For each high-level requirement:
  - every function is described in terms of
    - input data set
    - output data set
    - processing required to obtain the output data set from the input data set

Nonfunctional Requirements

- Characteristics of the system which can not be expressed as functions:
  - maintainability,
  - portability,
  - usability, etc.

Nonfunctional Requirements

- Nonfunctional requirements include:
  - reliability issues,
  - performance issues,
  - human-computer interface issues,
  - Interface with other external systems,
  - security, maintainability, etc.

Constraints

- Constraints describe things that the system should or should not do.
  - For example,
    - how fast the system can produce results
      - so that it does not overload another system to which it supplies data, etc.

Examples of constraints

- Hardware to be used,
- Operating system
  - or DBMS to be used
- Capabilities of I/O devices
- Standards compliance
- Data representations
  - by the interfaced system

Examples of Bad SRS Documents

- Unstructured Specifications:
  - Narrative essay --- one of the worst types of specification document:
    - Difficult to change,
    - difficult to be precise,
    - difficult to be unambiguous,
    - scope for contradictions, etc.

Organization of the SRS Document

- 1. Introduction to the Document
  - 1.1 Purpose of the Product
  - 1.2 Scope of the Product
  - 1.3 Acronyms, Abbreviations, Definitions
  - 1.4 References
  - 1.5 Outline of the rest of the SRS
- 2. General Description of Product
  - 2.1 Context of Product
  - 2.2 Product Functions
  - 2.3 User Characteristics
  - 2.4 Constraints
  - 2.5 Assumptions and Dependencies
- 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

Organization of the SRS Document(contd)

- 3.2 Functional Requirements
  - 3.2.1 Class 1
  - 3.2.2 Class 2
  - …
- 3.3 Performance Requirements
- 3.4 Design Constraints
- 3.5 Quality Requirements
- 3.6 Other Requirements

- 4. Appendices

Thank you