ECE 321 Software Requirements Engineering

Lecture 7: The Software Requirements Specification (SRS)

The 3 steps of the requirements development process

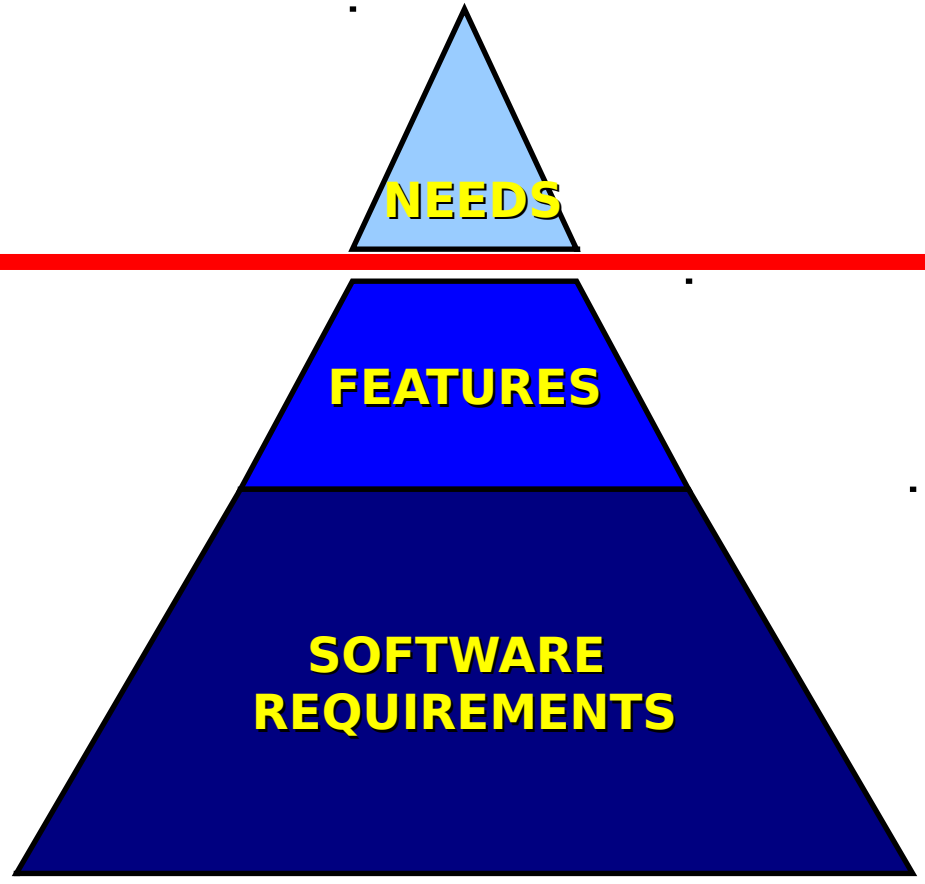
- **Requirement elicitation**
  - Understanding and analyzing the problem
  - Learning and understanding user needs
- **Requirement specification**
  - Developing a vision document
  - Developing requirement specification document
- **Requirement validation and verification**

The 3 steps of the requirements development process

- **Requirement elicitation**
  - Understanding and analyzing the problem
  - Learning and understanding user needs
- **Requirement specification**
  - Developing a vision document
  - Developing requirement specification document
- **Requirement validation and verification**

The software requirements pyramid

- problem domain
- solution domain



Organizing requirements information

- Requirements must be captured and recorded
  - In a document
  - In a database
  - In a model
  - In a tool
  - Or a combination of the above

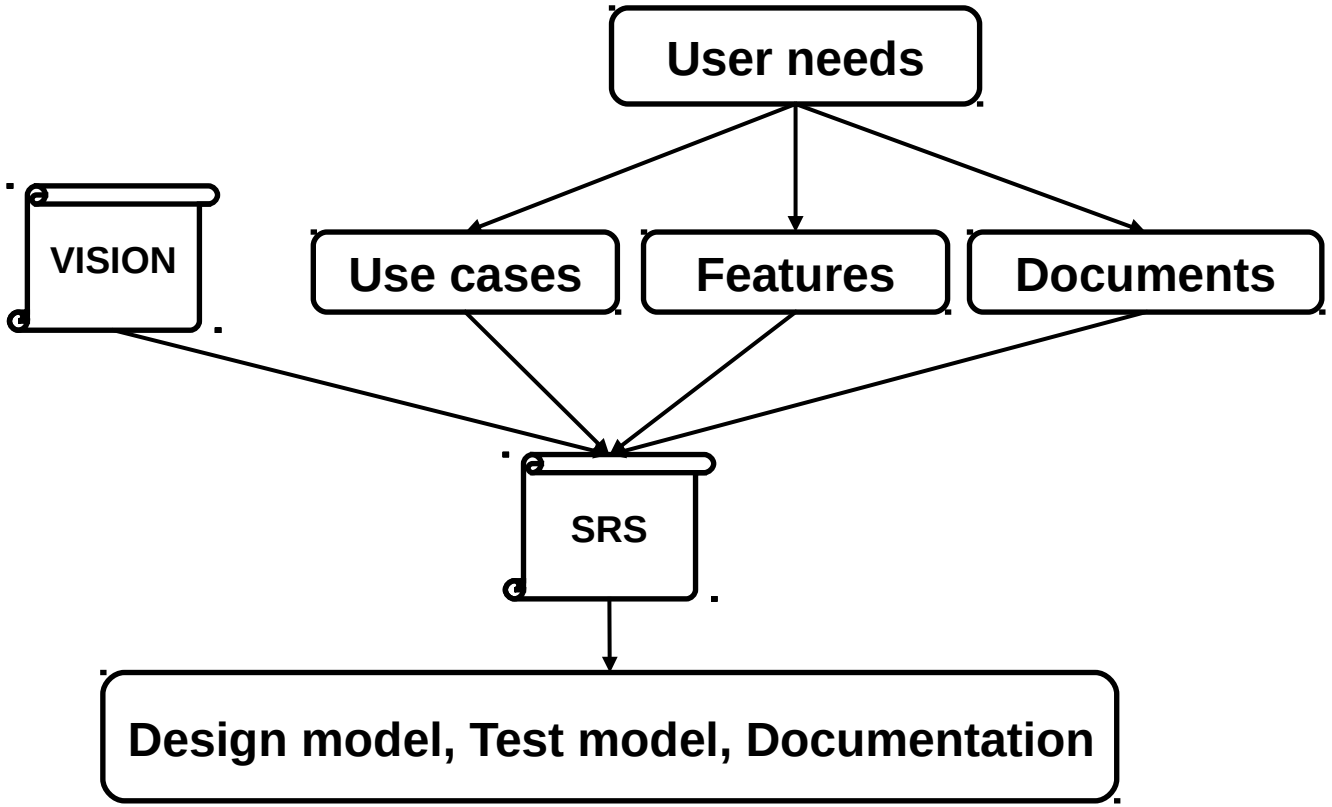
The software requirements specification (SRS)

- A collection of artifacts describing the complete external behaviour of the system
- The SRS has the following inputs:
  - The vision document
  - Use cases
  - Formal models
  - Elicited detailed requirements

A collection of artifacts describing the complete external behaviour of the system

- What, not how!
  - Not a schedule
  - Not a project plan
  - Not a budget
  - Not a test plan
  - Not a design

The inputs of the SRS visualized



SRS vs. Vision document

Vision document
Feature 63: The defect-tracking system will provide trending information to help user access project status.

SRS vs. Vision document

Vision document	SRS
Feature 63: The defect-tracking system will provide trending information to help user access project status.	SR 63.1: Trending information will be provided in a histogram report. SR 63.2: The user can enter the trending period in units of days, weeks, or months. SR63.3: An example trend report is shown in Figure 3.

The SRS is the software manager’s reference

- Represents agreement among various parties
- Basis of communication among those parties
- Serves as input for the design and implementation
- Serves as input to the software testing
- Used to control the evolution of the system throughout the development phase of the project

A template for the SRS 1/2

Revision History

1 Introduction
- 1.1 Purpose
- 1.2 Project scope and product features
- 1.3 References

2 Overall description
- 2.1 Product perspective
- 2.2 User classes and characteristics
- 2.3 Operating environment
- 2.4 Design and implementation constraints
- 2.5 User documentation
- 2.6 Assumptions and dependencies

3 System features
- 3.n Feature n

A template for the SRS 2/2

4 External interface requirements
- 4.1 User interfaces
- 4.2 Hardware interfaces
- 4.3 Software interfaces
- 4.4 Communications interfaces

5 Other nonfunctional requirements
- 5.1 Performance requirements
- 5.2 Safety requirements
- 5.3 Security requirements
- 5.4 Software quality attributes

Appendix A: Data dictionary and data model

Appendix B: Analysis models

An example SRS document (template 1)

Software Requirement Specification Document
for
Cafeteria Ordering System (COS)

ver. 1.0

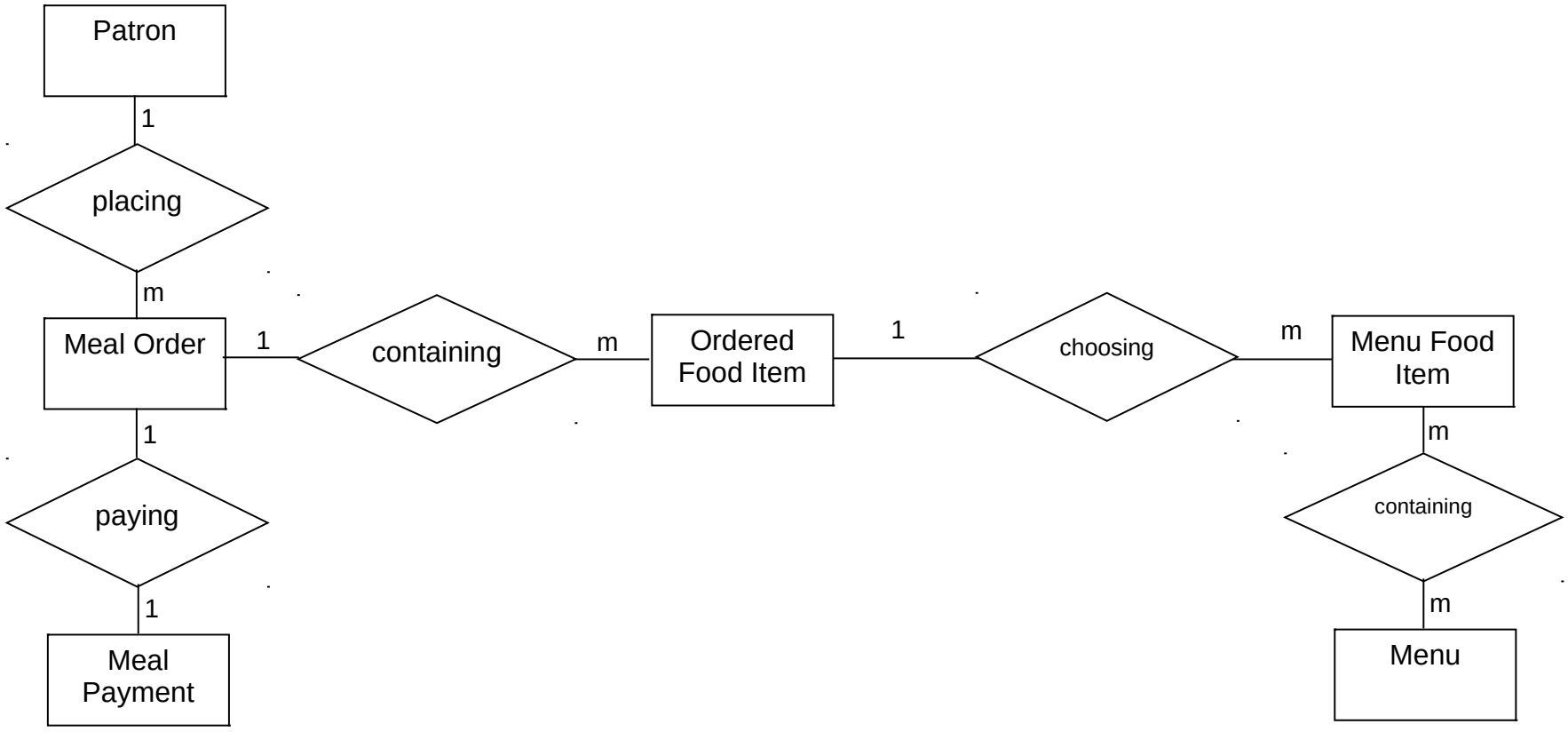
prepared by Karl Wiegers
Process Impact
November 4, 2002

Let’s study the SRS document in a bit more interactive way...

Let’s study the SRS document in a more interactive way...

- An angry employee has been messing in our SRS document :(
- It looks like the employee has added things, removed things, and changed things...
- Can you help identify what the employee changed?
  - This will teach you how to critically look at requirement specifications... what a lucky coincidence!

Entity-Relationship (ER) diagram



Reminder

- Assignment 2 will be posted this afternoon
- Due Oct 11 11:00 am