Software Engineering - Lecture 5

Ossama Edbali

January 27, 2015

1 System requirements

In the software life cycle activities there are 3 main steps which can be further split:

- Requirements
  - Requirements elicitation/gathering
  - Requirements analysis
- Design
  - System design
  - Detailed design
- Realization
  - Implementation
  - Testing

Requirements are the bridges that connects the abstract ideas gathered from the user world and the software-based system. Though, requirements may range from very abstract to detailed statements.

Therefore requirements engineering is the process of establishing the services that:

- The customer requires from a system
- The constraints under which it operates and is developed

2 Requirements engineering components

- Requirements gathering: direct interaction between software engineer and client; the main issues here are → what is to be accomplished? How the system will fit into the needs of the business and how the system will be used in a day-to-day basis.

- Requirements analysis: refining and detailing formally the gathered requirements

- Requirements specification: documenting in natural language the requirements process. One might think of it as a special contract between the software engineer/company and the customer(s). **See template on slides at page 41**.

Gathering requirements may involve the usage of questionnaires, interviews, task analysis (observing end users in their operational environment) and scenarios. Though, requirements gathering is a difficult task because there may occur problems of scope, understanding and/or volatility.

3 Types of requirements

Functional requirements

Describes the functionality/ability of the system and its subsystems (or services). How the system should react to particular inputs? How the system should behave in particular situations?

Non-functional requirements

These define system properties, behaviour and constraints e.g. reliability, response time and storage requirements. Constraints may be I/O devices, hardware, system representations etc…

Non-functional requirements are grouped in:

- Product requirements (related to the product itself)

- Organisational requirements: related to organisational issues such as registration, company policies etc…

- External requirements

In order to measure the efficiency of the non-functional requirements there are several properties to take into account (see presentation on page 35).

Domain requirements

see lecture notes on page 36.

4 Requirements validation

This part is concerned with validating the requirements and their integrity (we do not accept wrong requirements, i.e. that do not match the wanted system).

- Correctness: The requirements represent the client's view

- Completeness: All possible scenarios are described
- Consistency: There are no requirements that contradict each other
- Clarity: One and only one interpretation
- Realism: Implementation and delivery
- Traceability: Each system behaviour can be traced to a set of functional requirements