Requirements Engineering

LECTURE 4

Agenda

1. Introduction.
2. What is a Requirement?
3. Types of Requirements.
4. Adapting RE to Web Applications.
5. The Requirements Collection Process.

1- Introduction

Requirements Engineering (RE) – the principles, methods, & tools for eliciting (استباط أو استخراج), describing, validating, and managing project goals and needs.

Given the complexity of Web apps, RE is a critical initial stage, but often poorly executed.

What are the consequences?

- Inadequate software architectures
- "Unforeseen" problems
  - Budget overruns
  - Production delays
  - "That's not what I asked for"
- Low user acceptance

Agenda

1. Introduction.
2. What is a Requirement?
3. Types of Requirements.
4. Adapting RE to Web Applications.
5. The Requirements Collection Process.

2. What is a Requirement?

A requirement describes a property to be met or a service to be provided by a system.

IEEE 601.12 definition of *requirement:*

1. Condition needed to solve a user's problem
2. Condition to be met or possessed by the system to satisfy a formal agreement
3. Documented representation of conditions as in 1 and 2

Key players in the game

- Contract
- Customer
- Supplier
- User

Agenda

1. Introduction.
2. What is a Requirement?
3. Types of Requirements.
4. Adapting RE to Web Applications.
5. The Requirements Collection Process.

3. Types of Requirements

Many taxonomies exist to describe requirements, but most divide them into two groups:

- **Functional – describes the capability's purpose**
  - e.g., the ability to transfer money between user accounts
- **Non-functional – describes the capability's properties**
  - e.g., the Home Page must load within 5 seconds on a dial-up connection

3.1 Functional Requirement Types

Data Requirements

- How information is stored and managed

Interface Requirements

- How the user is going to interact with the application

Navigational Requirements

- How the user is going to navigate through the application

Personalization Requirements

- How the application adapt itself according to user or environment profile

Transactional Requirements

- How the application behave internally

3.2 Non-Functional Requirement Types

Content

Quality

- Functionality, Usability, Portability, Scalability
- Reliability, Efficiency, Security, Maintainability

System Environment

User Interface

- Self-explanatory & intuitive
- Usage-centered design

Evolution

Project Constraints

Agenda

1. Introduction.
2. What is a Requirement?
3. Types of Requirements.
4. Adapting RE to Web Applications.
5. The Requirements Collection Process.

Specifics in Web Engineering

Is RE for the Web really that different than RE for conventional software?

Top 6 distinguishing characteristics

1) Multidisciplinary teams
2) Unavailability of stakeholders
3) Rapidly changing requirements & constraints
4) Unpredictable operational environment
5) No manual for the user interface
6) Content Management

Adapting RE to Web Applications

There isn't **one single "right way"** to do RE among the many methods, techniques, tools, etc. available.

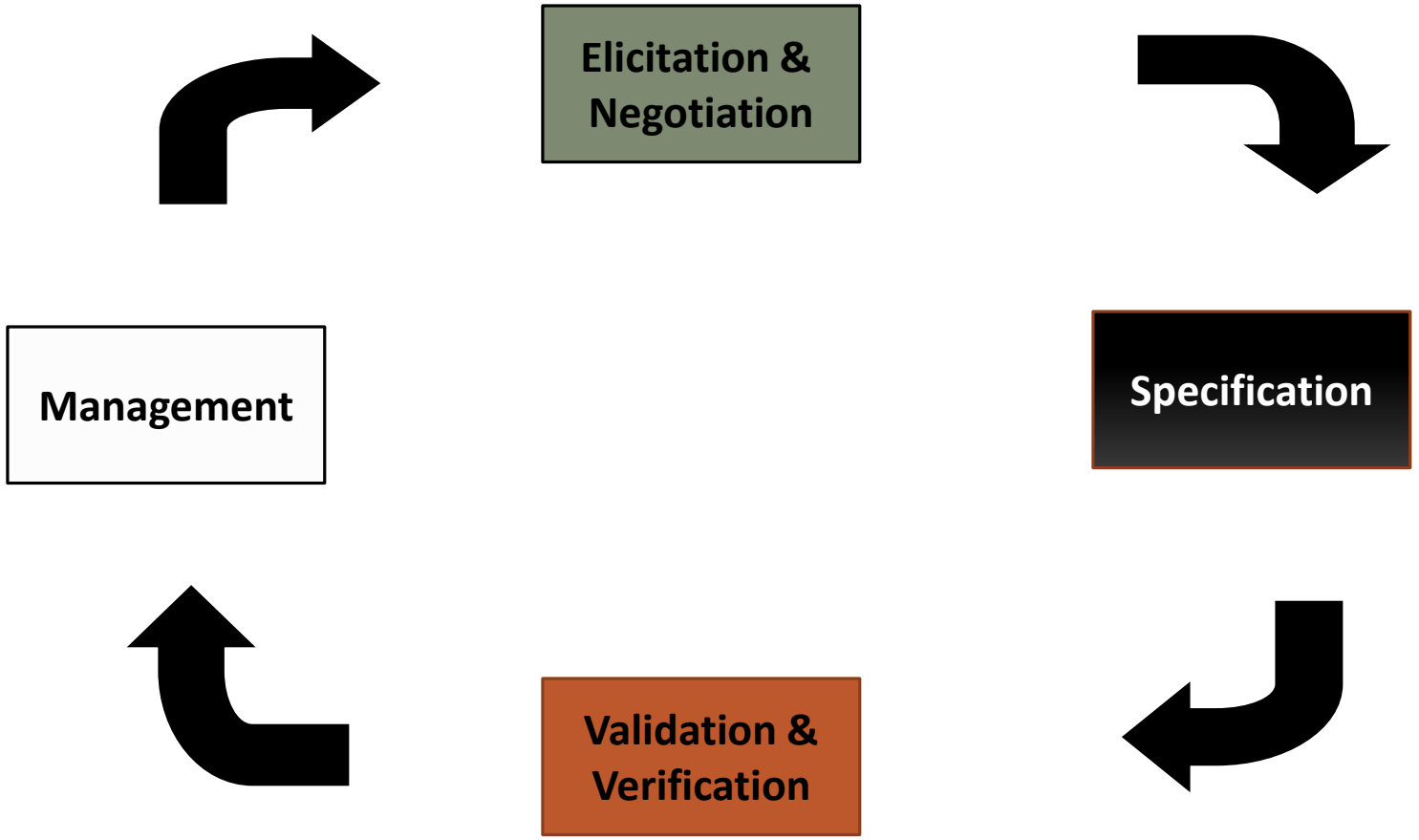
For your Web application project, ask the following questions:

- What are the critical requirements?
- How should requirements be documented?
- What tools should be used, if any?

Agenda

1. Introduction.
2. What is a Requirement?
3. Types of Requirements.
4. Adapting RE to Web Applications.
5. The Requirements Collection Process.

The Requirements Collection Process



Elicitation & Negotiation

HOW TO INTERACT WITH STAKEHOLDERS

Elicitation & Negotiation

Identify and involve (if possible) the *stakeholders*

- Those that directly influence the requirements
- Customers, users, developers

What are their expectations?

- May be misaligned or in conflict.
- May be too narrowly focused or unrealistic.

Why is the web application being developed in the first place?

Techniques for Elicitation & Negotiation

- Interviewing
- Joint Application Design
- Brainstorming
- Concept Mapping
- Storyboard
- Use Case Modeling
- Questionnaires

Challenges with Stakeholders

McConnell (1996)

- Users don’t know what they want.
- Lack of commitment.
- Ever-expanding requirements.
- Communication delays.
- Users don’t take part in reviews.
- Users don’t understand the technology.
- Users don’t understand the process.

Challenges with Developers

Users and engineers/developers speak different “languages”.

The tendency to “shoe-horn” the requirements into an existing model

- Saves time for developers, but results may not meet user’s needs.

Engineers & developers are also asked to do RE, but sometimes lack negotiating skills and domain knowledge.

Specification

HOW TO “FORMALIZE” RECEIVED INPUTS

Specification – Traditional RE

4 main categories of notation

- Stories – Plain-language scenarios; understandable to non-technical persons.
- Itemized Requirements – Plain-language lists of requirements
- Formatted Requirements – Accurately-defined, but allow for plain-language descriptions
  - Ex. Use case scenarios in UML
- Formal Specifications – Expressed in formal syntax & semantics; rarely used in Web applications.

Specification – RE for Web Apps

So, what's best for a Web development project?

- Formatted requirements (i.e. use cases) and stories are heavily used.
- Low to medium accuracy is suitable for Web apps; formal specifications very rarely required.
- Keep effort for eliciting and managing requirements low.
- Scalability is (most likely) important.

VALIDATION AND MANAGEMENT

Validation

This step is essential to verify that requirements specification corresponds to user's needs and customer's requirements

Iterative feedback from stakeholders is essential

- Is the requirement feasible?
- Do the results meet stakeholders' expectations?

Validation Techniques

Review or walk-through

- Reading and correcting the requirements definition documentation and models

Audit

- Partial check of the results presented in the review documentation

Traceability Matrix

- Comparison of the application objectives with the requirements of the system

Prototyping for Validation

- Implement a partial set of functional requirements but provide a global vision of the user interface

Summary

THAT'S ALMOST ALL FOR DAY…

Things to keep in mind

Know your Audience & Objectives

- Balancing stakeholder interests
- Focus on high-level requirements first.

Elicitation & Negotiation is a learning process

RE requires flexibility

- Iterative changes should be expected
- Be sure stakeholders understand this!

Clear documentation is critical

Assignment 2

Use the Template of RE or any template from google search engine that was adapted for web-based application to cover:

Functional and nonfunctional requirements with all its types.

Questions?