SENG 471 Notes

Brian Pho

April 18, 2018

Contents

Chapter 1

Introduction

- Requirements: goals, functionalities, qualities and constraints met by the system
- Requirements Engineering (RE) deals with the elicitation, analysis, and evolution of the requirements that need to be achieved by a software system within an organizational (or physical) environment.
- Requirements Elicitation: is a set of activities to identify and communicate the purpose of a software system and the contexts in which it will be used.
- RE Process: *Information Sources* → **Elicitation** → *Elicitation Notes and Constraints* → **Analysis** → *Requirements Document* (+ *Elicitation Notes and Constraints*) → **Specification** → *Specification Document*
- RE Questions
  - Which problem needs to be solved?
  - Where is the problem?
  - Whose problem is it?
  - Why does it need solving?
  - How might a software system help?
  - When does it need solving?
  - What might prevent from solving it?
- Stakeholder: an individual, group of people, organization or other entity that has a direct or indirect interest in a system
- Quality: fitness for purpose or conformance to requirements
- Why RE? Cost, time, resources, meeting expectations

Chapter 2

Requirements

- Enterprise Requirements → Why?
- Functional Requirements → What?
- Non-Functional Requirements → How?
- User Requirements → Who?
- Requirement vs Specification: requirements are what your program should do, the specifications are how you plan to do it. (what vs how)
- WRSPM Model: specifications bridge the application domain and the machine domain.
- Application Domain vs Machine Domain
- Application Domain (things the machine cannot observe)

  D = Domain Properties

  R = Requirements
- Machine Domain (things private to the machine)

  C = Computers

  P = Programs
- Both (shared things)

  S = Specification
- Verification (Are we building the system right?)

  S and D entail R?

  P on C satisfies S?
- Validation (Are we building the right system?)

  Got all important requirements?

  Got all the relevant domain properties?
- Link for Comparison

Chapter 3

Requirements Specification

- Software Requirements Specification (SRS): document used to communicate to a stakeholder
- Considerations
  - Valid
  - Unambiguous
  - Complete
  - Understandable
  - Consistent
  - Verifiable
  - Modifiable
  - Traceable
- Contents
  - Functionality: what
  - Attributes: considerations
  - Constraints: assumptions/standards
  - External Interfaces: users, SW, HW
  - Performance: criteria
- Should Not Include
  - Development plans
  - Assurance plans
  - Designs
- Typical Mistakes
  - Noise: irrelevant information
  - Silence nondescript feature

  - Over-specification: solutions rather than the problem
  - Contradiction: incompatible definitions
  - Ambiguity: multiple interpretations
  - Forward reference: term used prior to its definition
  - Wishful thinking: features cannot be validated
  - Jigsaw puzzles: info scattered

- Audience
  - Client/user
  - You/Analysts/SRE
  - You/Developer/Manager/QA
  - Lawyers/Court

Chapter 4

Formal Inspection

- Inspection Process

  1. **Planning**: choose reviewers and their roles (leader, recorder, reader, inspector, etc.)
  2. **Overview**: inspection objectives such as scope and criteria
  3. **Preparation**: individual inspection, typo log, focus on criteria
  4. **Inspection Meeting**: identify/record problems, meeting structure (checklist, walk through, round robin, speed review)
  5. **Rework**: address all problems
  6. **Follow-up**: corrections

- Constraints

  - Size
  - Duration
  - Outputs
  - Scope
  - Timing
  - Purpose

Chapter 5

Feasibility Study

- Look at alternatives
- Content
  - Organization of a system
  - Problems with the present system
  - Goals for the new system
  - Constraints
  - Possible alternatives
  - Things to conclude
- Types
  - Technical (possible with current tech, compatible, practical)
  - Operational (urgency, ethics, issues, PIECES)
  - Schedule (time frame, deadlines)
  - Economic (Resources, benefits, costs, developmental costs, operational costs)
- Cost-Benefit Analysis: Net Present Value, Return on Investment, Break Even Point
- Comparing Alternatives: candidate systems matrix

Chapter 6

Stakeholder's Goals

- Part of requirements elicitation
- Identify stakeholders
  - Organization chart
  - Authority level
- Focus on "why?" to derive goals
- "Why" → higher goals
- "How" → lower goals
- "How else" → alternatives
- Scenarios → Specific sequence of interactions
- Goal Modeling
  - Hard goals → Must be carried out
  - Soft goals → Difficult to be fully satisfied
- Use multiple sources if possible
- Associate stakeholders with each goal
- Use scenarios
- Explicit consideration of obstacles

Chapter 7

Requirements Elicitation

- Elicit information relevant to developing a system
- Elicitation Techniques
  - Existing documents/data
  - Interviews/questionnaires
  - Group techniques/observation
- Sources of Information
  - People
  - Existing systems/products
  - Documentation
- Challenges
  - Thin spread of domain knowledge
  - Difficult to verbalize knowledge
  - Limited observability
  - Bias
- Techniques
  - Traditional
  - Collaborative
  - Contextual
  - Cognitive
- Traditional
  - Introspection: imagines what kind of system is required
  - Background reading: read reports, charts, manuals, documentation
  - Analyze hard data: look at numbers

  - Interviews: asking for opinions, structured vs open-ended
  - Questionnaires: collect data from a large number of people
- Collaborative
  - Brainstorming
  - Joint/Rapid Application Development (JAD/RAD): WYSIWYG documentation
- Contextual
  - Participant observation
- Cognitive

Chapter 8

Scoping the Problem

- How to stop looking for bigger problems to solve?
- How to stop from computerizing everything?
- Analysis
  - What are the problems?
  - What are the alternatives?
  - What are the selection criteria?
  - What recommendation would you make?

Chapter 9

Modeling Requirements

- Model: an abstraction of something for the purpose of understanding it before building it
- Model: a number of views that can be static/dynamic
- Modeling Purpose
  - Guide elicitation
  - Provide a measure
  - Uncover problems
  - Check understanding
- Choice of Notation
  - Natural language
  - Semi-formal
  - Formal
- Desired characteristics of Modeling
  - Implementation independent
  - Abstraction
  - Formality
  - Constructability
  - Ease of Analysis
  - Traceability
  - Execution
- Techniques
  - Modeling enterprises
  - Modeling information + behavior
  - Modeling system qualities

Chapter 10

Modeling Enterprises

- Business Processes and Analysis
  - What are business processes?
  - Why model business processes?
  - Examine current process and identify what needs to be changed
  - Compare analyses using table of cost, risk, potential
- Business Process Modeling Notation (BPMN) diagram: activity diagram from UML
- Diagram Guidelines
  - Have initial and final event
  - Read from top-left to bottom-right
  - Each activity has at least one transition into it and at least one transition out of it
  - Is decidable
  - Each fork has a corresponding join

Chapter 11

Midterm

- No multiple choice
- 4 Questions
- Up to today's (Monday, Feb 26) lecture material
- Better bring calculators
- Not going to deduct marks for cardinalities

Chapter 12

Modelling Functions - Relationships

Chapter 13

- State Machine diagram: square brackets are guards, normal transitions are actions
-