Week One

- Architecture, Design, and Requirements
- RE Fundamentals
  - System-as-is, and the system-to-be
  - Who, what, why, how
  - Roles, processes, constraints, assumptions
  - Statements:
    • Requirements: prescriptive statements
    • Domain properties: descriptive statements
    • Assumptions/Expectations: expected behavior
Chapter Two

- Domain understanding
- Requirement elicitation
- Techniques
  - Artifact driven
    - Data collection, questionnaires, storyboards, knowledge reuse
  - Stakeholder driven
    - Interviews, ethnographics studies, group meetings

Requirements Evaluation

- Inconsistencies
  - Terminology; strong and weak conflicts
- Risks
  - Risk trees
  - Product-related risks; process-related risks
Sample Risk Tree

4: Requirements Specification

- Natural language
  - Decision tables
  - Templates
- Diagrammatic techniques
  - SADT and dataflow diagrams
  - Context diagrams
  - Use case diagram
  - State transition diagrams
  - Event sequence diagrams
5: Requirements QA

- Requirements inspections and reviews
  - Checklists: omissions, contradictions, inadequacies, ambiguities, unmeasurability, noise, over-specification, infeasibility
- Queries over a requirements database
- Specification animation
- Formal specs enable formal checks
  - (state space analysis/model checking)

6: Requirements Evolution

- Distinction between revisions and variants
- Traceability
  - Derivative requirements
  - Solution artifacts
- Change control
7: Goal Orientation

- Goals: prescriptive statements of intent that the system should satisfy through the cooperation of its agents
- Requirements; Expectations
- Descriptive statements: domain properties or hypotheses
- Behavioral goals; soft goals
- Why goals? Pp. 272-275

8: Goal Modeling

- Goal models: and/or refinement graphs
- Goal details as model annotations
  - Section 8.1
- Conflicts; Heuristics
Requirements in Practice

- Who pays for development?
- The Sales Cycle
- Technology adoption
- Customization and deployment
- The customer-specific dilemma
- Why architecture matters
- Start-ups, revenue and expenditures