Week 4 Requirement Engineering

- **Recall: Software Process**
  - Software Specification
  - Software Design and Implementation
  - Software Validation
  - Software Evolution
- **Definition**
  - The process of finding out, analyzing, documenting and checking these needs and constraints is called requirement engineering.
- **Software Requirement (functionality)**
  - Functional Requirement
    - Definition
      - Statements of services the system should provide
    - Completeness
      - All services required by the user should be defined
    - consistency
      - Requirements should not have contradictory definitions
  - Non-functional Requirement
    - Definition
      - Constraints on the services or functions offered by the system
    - Classification
      - Product requirements
        - Specify or constrain the behavior of the software
      - Organizational requirements
        - Broad syste requirements derived from policies and procedures in customers and developer's organization
      - External requirements
        - Covers all requirements derived from factors external ti the system and its development process
    - Testability
      - Speed
      - Size
      - Ease of use
      - Reliability
      - Robustness
      - Portability
    - It's difficult to separate functional and non-functional requirements in the requirements document. You should explicitly highlight requirements that are clearly related to emergent system properties, such as performance and reliability.
- **Software Requirements Specification/Document (SRS)**
  - Definition
    - SRS is an official statement of what the system developers should implement.
  - Include
    - User Requirement

            - Statements in a nature language plus diagrams to describe the services and constraint of a system -- understandable
        - System Requirement
            - Add details and explain how the user requirement should be provided by the system
            - Describe the external behavior of the system and its operational constraints
            - Exclude design information
    - Usage for different positions
        - Refer to the diagram wk4 p23
    - Level of Details
        - Depends on the type of system that is being developed and the development process used
        - Detailed requirements
            - Critical systems
            - System is to be developed by a sperate company
        - Less detailed requirements
            - Inhouse
            - Iterative development process
    - Quality Characteristics
        - Correct
        - Complete
        - Unambiguous
        - Verficable
        - Consistent
        - Ranked for importance and/or stability
        - Modifiable
        - Traceable
    - Tips
        - Specify only the external behavior of the system
        - Should not include details of the system architechture design
- **Requirement Engineering Process**
    - ❶ Feasibility study
        - Accessing if the system is useful to the business
    - ❷ Elicitation and analysis
        - Involve many people
            - End users
            - Engineers
            - Business managers
            - Domain experts
            - Trade union representatives
        - Difficulties
            - Unrealistic demands by stakeholders
            - Personal expression of requirements and implicit knowledge of stakeholders
            - Different stakeholders conflict with each other
            - Political factors
            - Economic and business environment is dynamic -- analysis is difficult
        - Iterative Process (ref to diagram)

- 1. Requirements discovery
  - Definition
    - The process of gathering information about the required system and existsing systems, and distilling the user and system requirements from this information
  - Stakeholders
    - Stakeholders range from end-users of a system through managers to external stakeholders such as regulators
  - Interview
    - Closed interview
      - Stakeholder answers a pre-defined set of questions
    - Open interview
      - No pre-defined agenda. The requirement engineering team explores a range of issues with system stakeholders
    - Benefits
      - Getting overall understanding of what stakeholders do
      - How they might interact with the new system
      - The difficulties that they face with current systems
    - Difficulties
      - Terminology
        - Application specialists use terminology in a precise and subtle way that is easy to misunderstand
      - Stakeholder
        - Stakeholder found difficult to explain or they think some domain knowledge is so fundamental that it isn't worth mentioning
      - Not an effective technique because there are subtle power relationship between different people and organization
    - Interviewers
      - Infomration from them supplements other infomration about existing systems, user observations, etc
  - Scenarios
    - A scenario is the descriptions of example interaction sessions. Usually it starts with an outline of the interaction.
- 2. Requirements classification and organization
  - Takes unstructured collection of requirements, groups related requiremetns, and organizes them into coherent clusters -- grouping -- system architecture
- 3. Requirements prioritization and negotiation
  - Resolving requirements conflicts through negotiation
- 4. Requirements specification
  - The requirements are documented and input into the next round of the spiral

- 3️⃣ Specification
  - Converting these requirements into some standard form
  - Types of Specification Structure
    - Natural language specification
      - Advantages: expressive, intuitive and universal
      - Drawbacks: potentially vague, ambiguous
      - Skills minimizing misunderstandings
        - Standard format

                - Distinguish mandatory and desirable requirements
                - Text highlighting
                - Associate a rationale with each user requirement
        - Structured Specification
            - Advantages: overcome the limitation of natural language using tables or graphical models to show computations, system state changes, system interaction, execution sequences
- ④ Validation
    - Definition
        - Checking that the requirements actually define the system that the customer wants
    - Checks
        - Validity checks
        - Consistency checks
        - Completeness checks
        - Realism checks
        - Verifiability
    - Techniques
        - Requirements reviews
        - Prototyping
            - Demonstrate an excutable model to end-users and customers
        - Test-case generation