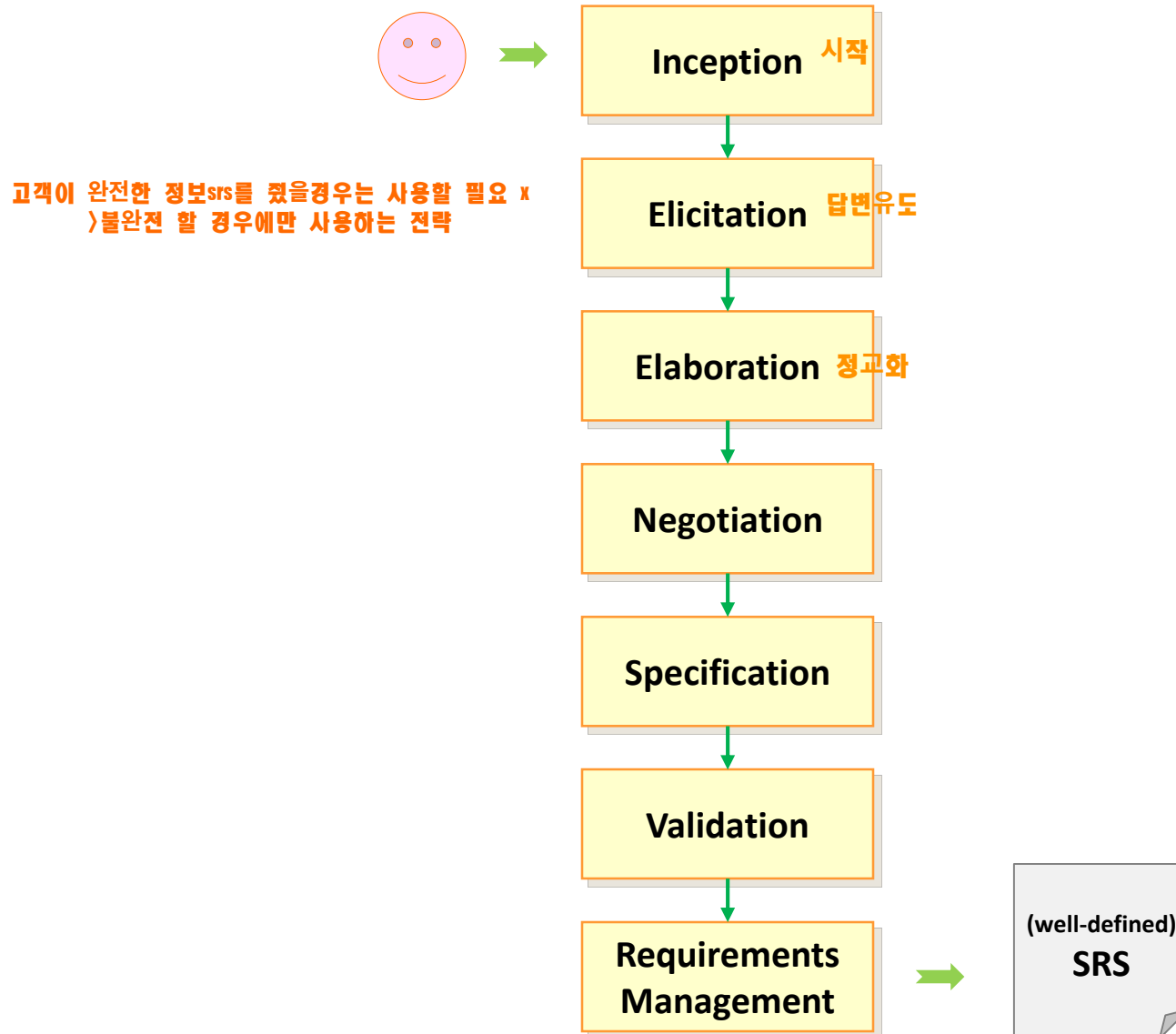


Unit 3.

Requirement Engineering

Tasks in Requirements Engineering



Tasks in Req. Engineering (1)

- **Step 1. Inception**

- Asking questions that establish;
 - Basic understanding of the problem
 - People who want a solution
 - Nature of the solution that is desired, and
 - Effectiveness of preliminary communication and collaboration between the customer and the developer

- **Step 2. Elicitation**

- Eliciting requirements from all stakeholders

Tasks in Req. Engineering (2)

- **Step 3. Elaboration**

- Creating an analysis model that identifies data, function and behavioral requirements

- **Step 4. Negotiation**

- Agreeing on a deliverable system that is realistic for developers and customers

- **Step 5. Specification**

- Representing requirements in a written document, a set of models, a formal mathematical, a collection of user scenarios (use-cases), and/or a prototype

Tasks in Req. Engineering (3)

- **Step 6. Validation**

- A review mechanism that looks for;
 - errors in content or interpretation
 - areas where clarification may be required
 - missing information
 - inconsistencies
 - conflicting or unrealistic requirements.

- **Step 7. Requirements Management**

- Requirement Evolution within a Release and among Releases

Typical Problems in SRS

소프트웨어 요구사항서에서 발생하는 전형적인 문제들

- **Incompleteness**

클라이언트들은 모든 요구조건을 완벽하게 적지않는다(빠진 부분이 있음)

- **Inconsistency**

불일괄성 > 단어선택이 정확하게 진행되어야 하지만 내용이 긴 경우 실수할때가있음(ex customer, user, member)

- **Incorrectness**

정보는 적혀있지만 잘못된 정보가 적혀있는 경우

- **Ambiguity**

애매모호한 용어를 사용하는 경우(ex 그렇게/모던하고 맨디하게 만들어주세요)

- **Imprecision, Blur**

- **Redundancy**

이러한 문제가 발생 할 경우
결과물의 상태가 매우 좋지 않아지니까
해결해야함

1. ^{시작}Inception (1)

- **Identify stakeholders**
 - “who else do you think I should talk to?”
- **Recognize multiple points of view**
- **Work toward collaboration**
- **Break the ice and initiate the communication**

1. Inception (2)

- **First Questions**

- **To focus on the customer, other stakeholders, the overall goals, and the benefits**
 - Who is behind the request for this work?
 - Who will use the solution?
 - What will be the economic benefit of a successful solution?
 - Is there another source for the solution that you need?

1. Inception (3)

- **Next Questions**

- **To gain a better understanding of the problem**
 - How would you characterize "good" output that would be generated by a successful solution?
 - What problem(s) will this solution address?
 - Can you show me (or describe) the business environment in which the solution will be used?
 - Will special performance issues or constraints affect the way the solution is approached?

1. Inception (4)

- **Final Questions**

- **To focus on the effectiveness of the communication activity itself**
 - Are you the right person to answer these questions? Are your answers "official"?
 - Are my questions relevant to the problem that you have?
 - Am I asking too many questions?
 - Can anyone else provide additional information?
 - Should I be asking you anything else?

^{끌어낸다} *2. Eliciting Requirements (1)*

- **Goals**

- To identify the problem.
- To propose elements of the solution
- To negotiate different approaches, and
- To specify a preliminary set of solution requirements

- **Two Activities**

- Collaborative Requirements Gathering
- Quality Function Deployment

2. Eliciting Requirements (2)

● Collaborative Requirements Gathering

- Meetings are conducted and attended by both software engineers and customers.
- Rules for preparation and participation are established.
 - An *agenda* is suggested.
 - A *facilitator* controls the meeting.
 - A *definition mechanism* is used.
 - Work sheets, Flip charts, Wall stickers, Electronic Bulletin Board, or Chat

● Quality Function Deployment

- To translate the needs of the customer into technical requirements for software
- To emphasize an understanding of what is valuable to the customer and then deploy these values throughout the engineering process through functions, information, and tasks

공들여 만들기

3. *Elaboration*

- Building an Analysis Model
- Elements of Analysis Model
 - Scenario-based elements
 - Class-based elements
 - Implied by scenarios
 - Behavioral elements
 - State diagram
 - Flow-oriented elements
 - Data Flow Diagram

diagram을 그림
>그림을 그리는게 훨씬 더 정확함
>내 이해가 맞는지 확인할 수 있음

4. Negotiation

기존의 이해와 달라졌을 때
> 필요 시간, 예산이 달라질 수 있음
> 이를 협상

- Identify the key stakeholders for Negotiation.
- Determine each of the stakeholders “win conditions”
 - Win conditions are not always obvious.
- Negotiate
 - Work toward a set of requirements that lead to “win-win”.

5. Specification (1)

● IEEE Recommended Practice for Software Requirements Specifications [IEEE Std 830-1998]

Contents

1. Overview	1
1.1 Scope	1
2. References	2
3. Definitions	2
4. Considerations for producing a good SRS	3
4.1 Nature of the SRS	3
4.2 Environment of the SRS	3
4.3 Characteristics of a good SRS	4
4.4 Joint preparation of the SRS	8
4.5 SRS evolution	8
4.6 Prototyping	9
4.7 Embedding design in the SRS	9
4.8 Embedding project requirements in the SRS	10
5. The parts of an SRS	10
5.1 Introduction (Section 1 of the SRS)	11
5.2 Overall description (Section 2 of the SRS)	12
5.3 Specific requirements (Section 3 of the SRS)	15
5.4 Supporting information	19
Annex A (informative) SRS templates	21
Annex B (informative) Guidelines for compliance with IEEE/EIA 12207.1-1997	27

5. Specification (2)

● Template for SRS

Table of Contents

1. Introduction
1.1 Purpose
1.2 Scope
1.3 Definitions, acronyms, and abbreviations
1.4 References
1.5 Overview
2. Overall description
2.1 Product perspective
2.2 Product functions
2.3 User characteristics
2.4 Constraints
2.5 Assumptions and dependencies
3. Specific requirements (See 5.3.1 through 5.3.8 for explanations of possible specific requirements. See also Annex A for several different ways of organizing this section of the SRS.)
Appendixes
Index

3. Specific requirements

3.1	External interface requirements
3.1.1	User interfaces
3.1.2	Hardware interfaces
3.1.3	Software interfaces
3.1.4	Communications interfaces
3.2	Functional requirements
3.2.1	Mode 1
3.2.1.1	Functional requirement 1.1
.	
.	
.	
3.2.1.n	Functional requirement 1.n
3.2.2	Mode 2
.	
.	
.	
3.2.m	Mode m
3.2.m.1	Functional requirement m.1
.	
.	
.	
3.2.m.n	Functional requirement m.n
3.3	Performance requirements
3.4	Design constraints
3.5	Software system attributes
3.6	Other requirements

6. Validation (1)

srs가 제대로 되어있는지
확인하다

- Is each requirement consistent with the overall objective for the system/product?
- Have all requirements been specified at the proper level of abstraction? That is, do some requirements provide a level of technical detail that is inappropriate at this stage?
- Is the requirement really necessary or does it represent an add-on feature that may not be essential to the objective of the system?
- Is each requirement bounded and unambiguous?
- Does each requirement have attribution? That is, is a source (generally, a specific individual) noted for each requirement?
- Do any requirements conflict with other requirements?

6. Validation (2)

- **Is each requirement achievable in the technical environment that will house the system or product?**
- **Is each requirement testable, once implemented?**
- **Does the requirements model properly reflect the information, function and behavior of the system to be built.**
- **Has the requirements model been “partitioned” in a way that exposes progressively more detailed information about the system.**
- **Have requirements patterns been used to simplify the requirements model. Have all patterns been properly validated? Are all patterns consistent with customer requirements?**

7. Requirements Management

버전컨트롤하기

ex) srs 1.0v
 } srs 1.2v

- Why need?
 - Requirements change, and the desire to change requirements persists throughout the life of the system.
- Activities that help the project team identify, control and track requirements evolution.
- Similar to Configuration Management (CM) except the CM Items are on requirement.

Standards for Requirement Engineering

- Systems and software engineering - Life cycle processes – Requirements Engineering [ISO/IEC/IEEE 29148: 201]
- Specifies the required processes

Contents	Page		Page
Foreword	vi	7 Information items	41
Introduction	vii	8 Guidelines for information items	42
1 Scope	1	8.1 Requirements information item outlines	42
2 Conformance	1	8.2 Stakeholder requirements specification document	42
2.1 Intended Usage	1	8.2.1 Introduction	42
2.2 Conformance to processes	2	8.2.2 SRS example outline	42
2.3 Conformance to information item content	2	8.3 System requirements specification document	43
2.4 Full conformance	2	8.3.1 Introduction	43
2.5 Tailored conformance	2	8.3.2 SyRS example outline	44
2.5.1 Processes	2	8.4 Software requirements specification document	45
2.5.2 Information items	3	8.4.1 Introduction	45
3 Normative references	3	8.4.2 SRS example outline	45
4 Terms, definitions and abbreviated terms	3	9 Information item content	46
4.1 Terms and definitions	3	9.1 Introduction	46
4.2 Abbreviated terms	8	9.2 General content	46
5 Concepts	8	9.2.1 Identification	46
5.1 Introduction	8	9.2.2 Front matter	46
5.2 Requirements fundamentals	8	9.2.3 Definitions	47
5.2.1 General	8	9.2.4 References	47
5.2.2 Stakeholders	8	9.2.5 Acronyms and abbreviations	47
5.2.3 Transformation of needs into requirements	9	9.3 Stakeholder requirements specification (SRS) document	47
5.2.4 Requirements construct	9	9.3.1 Business purpose	47
5.2.5 Characteristics of individual requirements	11	9.3.2 Business scope	48
5.2.6 Characteristics of a set of requirements	11	9.3.3 Business overview	48
5.2.7 Requirement language criteria	12	9.3.4 Stakeholders	48
5.2.8 Requirements attributes	12	9.3.5 Business environment	48
5.3 Practical considerations	14	9.3.6 Goal and Objective	48
5.3.1 Iteration and recursion of processes	14	9.3.7 Business model	48
5.3.2 Iteration and recursion in requirements engineering	16	9.3.8 Information environment	48
5.4 Requirement information items	17	9.3.9 Business processes	49
6 Processes	18	9.3.10 Business operational policies and rules	49
6.1 Requirement processes	18	9.3.11 Business operational constraints	49
6.1.1 Guidelines for Processes	19	9.3.12 Business operation modes	49
6.2 Stakeholder requirements definition process	19	9.3.13 Business operational quality	49
6.2.1 Purpose	19	9.3.14 Business structure	49
6.2.2 Outcomes	19	9.3.15 User requirements	49
6.2.3 Activities and tasks	20	9.3.16 Operational concept	50
6.3 Requirements analysis process	27	9.3.17 Operational scenarios	50
6.3.1 Purpose	27	9.3.18 Project constraints	50
6.3.2 Outcomes	27	9.4 System requirements specification (SyRS) document	50
6.3.3 Activities and tasks	27	9.4.1 System purpose	50
6.4 Requirements engineering activities in other technical processes	33	9.4.2 System scope	50
6.4.1 Requirements in architectural design	33	9.4.3 System overview	51
6.4.2 Requirements in verification	34	9.4.4 Functional requirements	51
6.4.3 Requirements in validation	36	9.4.5 Usability requirements	51
6.5 Requirements management	37	9.4.6 Performance requirements	51
6.5.1 Management Overview	37	9.4.7 System interfaces	51
6.5.2 Change management	37	9.4.8 System Operations	52
6.5.3 Measurement for requirements	39	9.4.9 System modes and states	52
		9.4.10 Physical characteristics	52
		9.4.11 Environmental conditions	53
		9.4.12 System security	53
		9.4.13 Information management	53
		9.4.14 Policies and regulations	53
		9.4.15 System life cycle sustainment	53
		9.4.16 Packaging, handling, shipping and transportation	53
		9.4.17 Verification	54
		9.4.18 Assumptions and dependencies	54
		9.5 Software requirements specification (SRS) document	54
		9.5.1 Purpose	54
		9.5.2 Scope	54
		9.5.3 Product perspective	54
		9.5.4 Product functions	56
		9.5.5 User characteristics	57
		9.5.6 Limitations	57
		9.5.7 Assumptions and dependencies	57
		9.5.8 Apportioning of requirements	57
		9.5.9 Specific requirements	57
		9.5.10 External interfaces	58
		9.5.11 Functions	58
		9.5.12 Usability requirements	59
		9.5.13 Performance requirements	59
		9.5.14 Logical database requirements	59
		9.5.15 Design constraints	60
		9.5.16 Standards compliance	60
		9.5.17 Software system attributes	60
		9.5.18 Verification	61
		9.5.19 Supporting information	61
		Annex A (normative) System operational concept	62
		A.1 Overview	62
		A.2 Operational concept document (OpsCon)	62
		A.2.1 Scope	63
		A.2.2 Referenced documents	63
		A.2.3 Current system or situation	63
		A.2.4 Justification for and nature of changes	66
		A.2.5 Concepts for the proposed system	68
		A.2.6 Operational scenarios	70
		A.2.7 Summary of impacts	71
		A.2.8 Analysis of the proposed system	72
		A.2.9 Appendices	73
		A.2.10 Glossary	73
		Annex B (informative) Concept of operations	74
		B.1 Overview	74
		B.2 Concept of operation document	74
		B.2.1 Purpose	74
		B.2.2 Scope	74
		B.2.3 Strategic plan	74
		B.2.4 Effectiveness	74
		B.2.5 Overall operation	74
		B.2.6 Governance	75
		Annex C (informative) Process Mapping from ISO/IEC 15288 and ISO/IEC 12207	76
		C.1 Stakeholder requirements definition process	76
		C.2 Requirements analysis Process	77
		C.3 Other technical requirements-related processes	78
		Annex D (normative) Tailoring policies	80
		D.1 Introduction	80
		D.2 Information item tailoring process	80
		D.2.1 Purpose	80
		D.2.2 Outcomes	80
		D.2.3 Activities and tasks	80
		Bibliography	82

