

Software Development
Project



SanHark Project

요구사항 작성 가이드라인

Fall, 2025

Jang-Eui Hong

Understanding Requirements

What is Requirements?

- Statements or description of what the system must do or what characteristic it must have. -- BABOK
- Conditions or capability needed by a user to solve a problem or achieve an objective. -- IEEE Standards
- Conditions or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. -- wiki.
- What need to be done in order to achieve the need or goal. -- Anonymity



* BABOK : Business Analysis Body of Knowledge

Understanding Requirements

Requirements Classification

- Functional requirements
 - directly related to the biz process that the system has to perform
- Nonfunctional requirements
 - behavioral properties that the system must have, such as performance
- Interface requirements
 - related to the interactions with other external systems
 - related to the interactions with users (i.e., GUI)
- Other requirements
 - related to the required issues (constraints) on the contracts, such as ORACLE
 - Issues to be decided with users, from a developer's point of view



.. Non-functional Requirements : Example

1. Operational Requirements

- 1.1 The system will operate in Windows environment.
- 1.2 The system should be able to connect to printers wirelessly.
- 1.3 The system should automatically back up at the end of each day.

2. Performance Requirements

- 2.1 The system will store a new data in 2 seconds or less.
- 2.2 The system will retrieve the daily schedule in 2 seconds or less.

3. Security Requirements

- 3.1 Only doctors can set their availability.
- 3.2 Only a manager can produce a schedule.

4. Cultural and Political Requirements

- 4.1 Laws, Company policies and Standards
- 4.2 Regional characteristics: phone number format,

Characteristics of Good Requirements (1)

Unitary

- address one and only one thing

Unambiguous

- one and only one interpretation.
- Vague subjects, adjectives, prepositions and verbs are avoided.
- Negative statements are also avoided.

Complete

- fully stated in one place with no missing information

Consistent

- no contractions with the other requirements
- fully consistent with all authoritative external documents as well as internal documents within a project.

Characteristics of Good Requirements (2)

Atomic

- does not contain conjunctions (not compound statement)

Traceable

- meets all of the needs stated by stakeholders
- referenceable for each individual requirement

Verifiable

- Each requirement must be implementable.
- Possible to confirm that each requirement is met
- Each artifact should be able to be checked with basic methods.
 - Artifacts : analysis and design documents, source codes
 - basic methods : inspection, testing, demonstration, simulation.

Where to use Requirements Specification

Communication with users

Inputs to the analysis phase

Inputs for creating test cases (black box testing)

Criteria for acceptance testing



Elements of a requirement statement

- (1) Identifier
- (2) Functional Group
- (3) Requirement Description
 - S+V+D+P+I, Simple Sentence
- (4) Importance Level
- (5) Priority
- (6) Remark (explanatory note)

Requirement Structure in Agile (Scrum)

Theme (주제, Initiative)

- 서로 다른 주체(Actor)별로 구분되어지는 상위 수준 기능
- 특정 액터에 대하여 독립성을 갖는 상위 수준 기능
- 공동의 목표를 추구하는 에픽들의 모임

Epic (구성)

- 주제에 속하는 하위의 기본 기능
- 사용자 독립적으로 활용하는 기본 기능
- 다수의 스프린트에서 구현될 수 있음.

Story (사용자 요구사항)

- 각 에픽에 대한 보다 상세한 세부 기능의 표현
- 각 스토리는 사용자 관점에서 수행하는 (보여지는) 단위 기능

제품 백로그 (Product Backlog) 작성

활동 목적

- 개발 대상 시스템이 제공해야 하는 기능 식별 및 정리
- 사용자 유형별 시스템의 서비스 기능에 대한 명세
- 사용자 요구사항을 지속적으로 유지 관리하기 위한 기본 틀 제공

입력 정보

- 시스템 정의서
- 사용자 인터뷰 결과서 (존재시)
- 기존 유사 시스템 조사 분석서 (시스템 정의서의 일부)
- 시스템 아키텍처

출력 정보

- 제품 백로그 : 엑셀(Excel)을 이용하여 작성
- PB = {사용자 유형 + 주제 ID + 주제명 + 에픽 ID + 에픽명 + 스토리 ID + 스토리 + (중요도) + 우선순위+추정 작업량 + 진도}

제품 백로그 (Product Backlog) 작성

주요 활동

- 입력 정보를 이해하고 분석한다.
- 사용자 유형을 모두 구분하여 식별한다.
 - 일반사용자, 회원사용자, 시스템운영관리자, 비즈니스관리자 등
- 각 사용자별 주제를 정의한다.
- 각 주제별 에픽을 정의한다.
 - 주제별 에픽에 대하여 기능간 실행 순서를 정의 한다.
- 각 에픽별로 사용자 스토리를 도출한다.
 - 스토리는 요구사항 기술 방식 (SVDPI) 을 준수한다.
- 우선순위 결정 : 요구사항들에 대한 우선순위 결정 한다
 - 기능 중요도, 사용자 긴급성, 개발 기술 수준 등을 고려함.
- 스토리 포인트를 산정한다. (Planning Poker)

제품 백로그 (Product Backlog) 작성

(?) 비기능적 요구사항은 어떻게 처리하나요?

- 특정한 에픽 또는 스토리와 직접 관련된 비기능적 요구사항
 - 하나의 고유 식별자를 갖는 하나의 독립된 스토리로 추가한다.
- 특정 주제(다수의 에픽)에 걸친 비기능적 요구사항일 경우
 - 주제의 하위 레벨인 고유의 에픽으로 식별한다.
 - 식별된 에픽은 다시 스토리 레벨로 분리하여 세분화 한다.
(경우에 따라 에픽과 스토리가 반복 기술이 될 수 있음)
- 시스템 수준에서의 비기능적 요구사항일 경우
 - 예를 들면, Reliability, Availability, Maintainability 등의 품질관련 요구사항
 - 별도의 주제로 식별한다. (주제명: Quality, 에픽명: Reliability)
 - = 스토리 : 시스템은 평균적으로 6개월에 1번의 고장만 허용해야 한다.
(i.e., MTBF = 6 month)

과제 안내

과제명 : 사용자 요구사항 정리 작성

과제 제출 기한 : 9월 19일(금) 밤 12시전까지

과제 내용

- 액셀 파일을 이용하여 요구사항을 정의 한다.

제출물 : 제품 백로그

제출장소 : LMS 과제 보드에 업로드

과제 제출 참고사항

- 팀에서 한 사람만 제출할 것
- 파일명 : Team Name_PB-V1 과 같은 순서로 부여할 것



산학 프로젝트 – Product Backlog 작성

질의 및 응답

