

2022학년도 2학기-응용SW실무

# 2주차 : 프로젝트 계획하기

2022. 09. 05 (월)

Prepared by DaeKyeong Kim

Ph.D.

Dept. of CSE  
University of YUHAN



1

1교시: Agile 개발 방법론 종류 및 주요 원리

2

2교시: 이터레이션 제로

3

3교시 : Agile Estimation과 Release Planning

프로젝트 계획하기

# 1교시 : Agile 개발 방법론 종류 및 주요 원리





# 1-1. Agile 방법론의 종류

## 학습목표

- 이 워크샵에서는 Agile 선언문에 대해 알 수 있다.

## 눈높이 체크

- Agile 선언문에 대해 알고 계신가요?

# 1-1. Agile 방법론의 종류

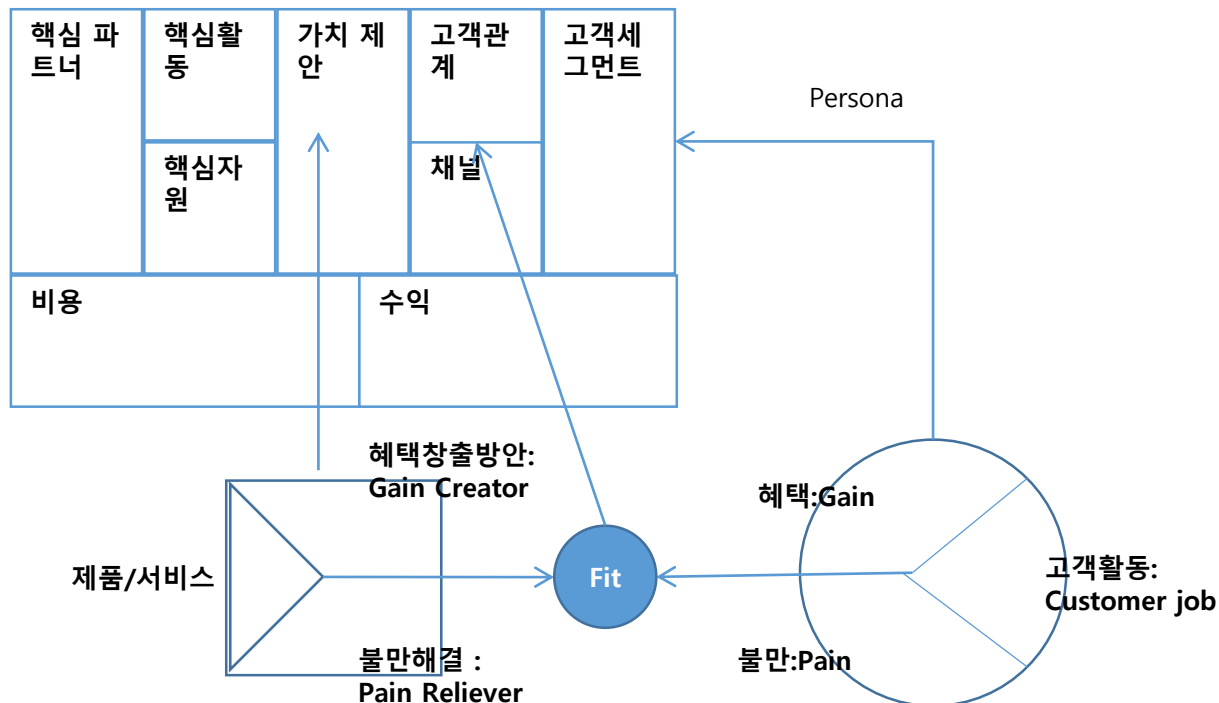
## Why Agile?

- 폭포수모델이라고 불리는 기존의 계획, 디자인, 설계, 코딩, 배포 순으로 이루어지는 개발 프로세스는 클라이언트가 실제 결과물을 확인할 때 까지 시간이 오래걸린다.
- 그리고 구현시 설계와 달라지는 경우가 매우 많아, 그에 대응하기가 어렵다는 단점이 있다.
- 이런 부작용들은 납기지연, 예산초과, 철야근무 등을 초래한다.
- Agile은 이런 단방향적인 개발 모델이 아니라, 짧은 주기를 가지고 눈에 보이는 결과물을 만들어내며 클라이언트와 소통한다.
- 그 다음 주기에서는 수정사항을 반영하여 계획하고, 개발을 수행한다. 이런 주기를 계속 반복하여 최종적으로는 클라이언트가 원하는 제품에 가장 근접하게 개발할 수 있도록 하는 것이 Agile이다.

# 1-1. Agile 방법론의 종류

## 가치 제안

- 방법론은 가치를 기반으로 하는 실천법이다.
- XP Primary Values : Four Values(Communication, Simplicity, Feedback, Courage)
  - Respect [This value lies below the surface of 4 Values]
  - Secondary XP values are SAFTEY, SECURITY, PREDICTIBILTY and QUALTIY OF LIFE



# 1-1. Agile 방법론의 종류

## 가치로서의 Agile 정신과 실천법의 필요

### Agile Manifesto

Individuals and interactions over processes and tools.  
Working software over comprehensive documentation  
Customer collaboration over contract negotiation  
Responding to change over following a plan



누구를 위한...?

Kent Beck  
Mike Beedle  
Arie van Benn  
ekum  
Alistair Cockb  
urn  
Ward Cunnin  
gham  
Martin Fowler

James Gren  
ning  
Jim Highsm  
ith  
Andrew Hu  
nt  
Ron Jeffries  
Jon Kern  
Brian Maric

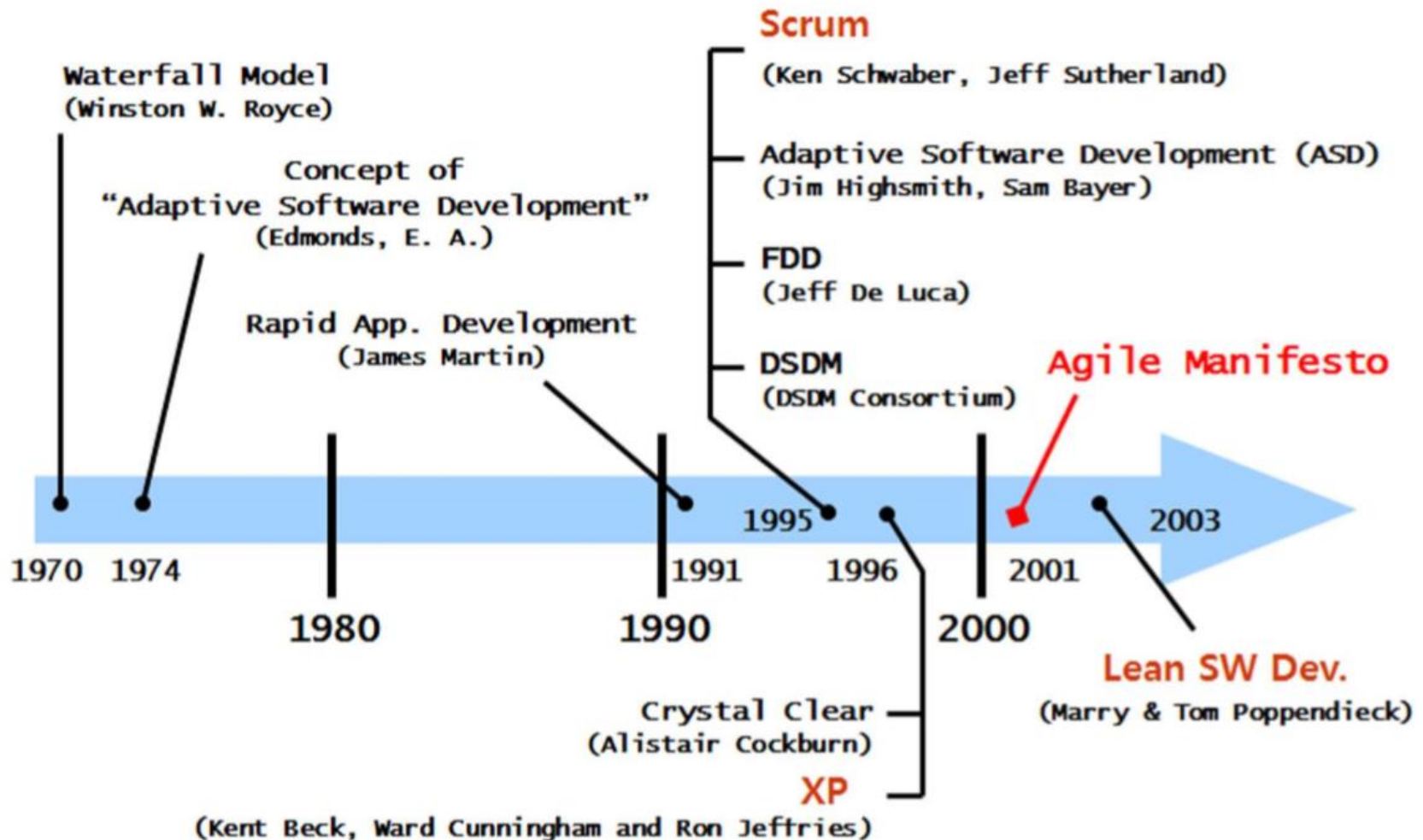
Robert C. Ma  
rtin  
Steve Mellor  
Ken Schwab  
er  
Jeff Sutherla  
nd  
Dave Thoma  
s

공정과 도구보다 개인과 상호작용을  
포괄적인 문서보다 작동하는 소프트웨어를  
계약 협상보다 고객과의 협력을  
계획을 따르기보다 변화에 대응하기를

- 2001년, 2월 Utah의 Wasatch산에 있는 Snowbird 스키 리조트에서 17명의 인원들이 Agile SW 개발 방식에 대한 선언문을 발표
- 경영,마케팅,HR과 리더십,기타 다양한 사업과 조직

# 1-1. Agile 방법론의 종류

## Agile 방법론의 기원





# 1-1. Agile 방법론의 종류

## Agile 방법론의 종류

- Agile 선언문이 나온 이후로 다양한 Agile 방법론이 등장. 하지만 그 기반에는 공통적으로 '신뢰성 높은 소프트웨어를 빨리 개발하자'는 가치를 공유. 다음은 Agile 개발 방법의 이름과 그 방법론을 정의한 사람의 목록.
- ① 동적 시스템 개발 방법론(Dynamic Systems Development Method, DSDM), 데인 포크너 외, 1994
- ② 스크럼(Scrum), 켄 슈와버/제프 서덜랜드, 1995
- ③ 크리스탈 패밀리(Crystal Family), 앨리스테어 코번, 1996
- ④ 기능 주도 개발방법론(Feature Driven Development, FDD), 피터 코드/제프 드루카, 1997
- ⑤ 익스트림 프로그래밍(eXtreme Programing, XP), 켄트 벡/에릭 감마, 1999
- ⑥ 적응형 소프트웨어 개발 방법론(Adaptive Software Development, ASD), 짐 하이스미스, 2000
- ⑦ 린(Lean) 소프트웨어 개발방법론, 메리 포펜딕/톰 포펜딕, 2003
- ⑧ Agile UP(Agile Unified Process, AUP), 스콧 앰블러, 2011
- ⑨ Kanban, 2006



# 1-2. Agile과 Scrum(스크럼)

## 학습목표

- 이 워크샵에서는 Agile 프로세스와 프랙틱스 시나리오에 대해 설명한다.
- Agile 개발 방법론 종류 및 주요 원리, 각 실천법 등을 알 수 있다.
- Agile 프로세스는 Prepare Product Backlog, Release Planning, Sprint Planning, Sprint Tracking, Retrospective 등 SRS로 부터 문제 개선까지의 내용을 다룬다. 프랙틱스 시나리오를 통해 Agile 프로세스를 수행할 수 있다.

## 눈높이 체크

- Agile 프로세스를 알고 계신가요?

# 1-2. Agile과 Scrum(스크럼)

## Agile에서의 일의 단위

- Theme > Epic > Story(기본단위) > Task(티켓 할당 단위)
  1. Theme: 종합 온라인 쇼핑몰 구축
  2. Epic: 고객은 스마트폰이나 데스크탑으로 주문을 할 수 있음.
  3. Story: 고객(누가)은 주문을 하기 위해서(왜) 배송 시간을 고를 수 있어야 한다.(행동)
  4. Task: 배송 시간을 선택할 수 있는 기능을 구현한다. 각 작업의 단위에서 가장 중요하고 기본이 되는 단위는 'Story'. Story는 특이하게도 '누가 왜 그 행동을 하는가'를 말함. 우리가 구축하고자 하는 시스템에서의 고객의 행동을 Story라 함. 고객의 행동 = 스토리. 그런 고객의 스토리들이 모여서 하나의 에픽을 구성. 주문을 하는 행위는 배송시간을 고르는 행위를 포함하고 있다. 'Task'는 고객이 그 행동을 할 수 있도록 실제로 개발해야 하는 단위를 나타냄. 각 Task가 담당 개발자들에게 티켓으로 할당. 이 Task를 좀 더 잘게 나눠서 Sub Task를 구성. Agile 팀에서는 매일 아침 스탠드업 미팅을 함. 그 미팅에서 다루는 일의 단위는 Task나 Sub Task임.

# 1-2. Agile과 Scrum(스크럼)

## Agile과 Scrum(스크럼) 이해하기

- 추정 및 조정 기반의 경험적 관리기법의 대표적인 형태.
- 처음 시작은 노나카 이쿠지로와 타케우지 히로타카가 1986년 1~2월 Harvard Business Review에 올린 "The New New Product Developement Game"라는 기사를 그 기원으로 봄. 이후 1995년에 켄 슈와버(Ken Schwaber )와 제프 서덜랜드(Jeff Sutherland )가 이 방법을 소프트웨어 개발에 소개하면서 스크럼이라 부르게 됨.
- Scrum은 비즈니스 요구를 충족시키는데 초점을 맞추기 위해, 작은 목표를 짧은 주기 (Sprint) 로 점진적이며 경험적으로 제품을 지속적으로 개발(전달)하는 관리 프레임워크(기법)
- 사람들이 효과적으로 성취감을 충족하며 협업할 수 있게 하여, 복잡하고 정교한 제품을 생산할 수 있게 함.

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- 혼다, 캐논, 후지 제록스와 같은 기업이 모든 작업 절차를 단일 과정 내에 수행하는 방식(all-at-once product development)을 통해 팀 단위로 유연한 제품을 개발하는 내용을 소개

The New New Product Development Game  
by Hirotaka Takeuchi and Ikujiro Nonaka  
From the January 1986 Issue

**EXHIBIT 1**  
Sequential (A) vs. overlapping (B and C) phases of development

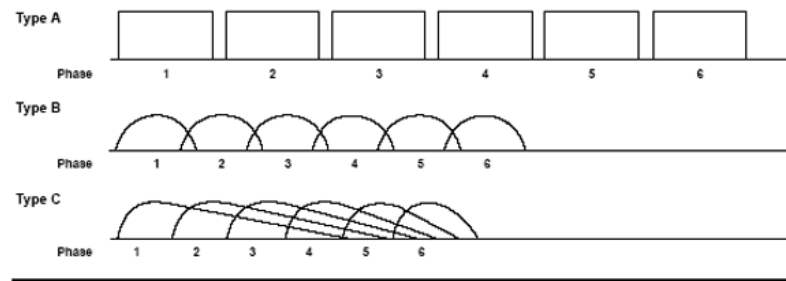


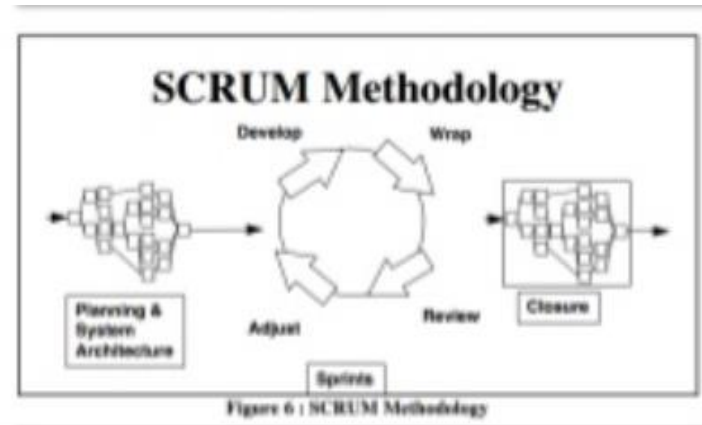
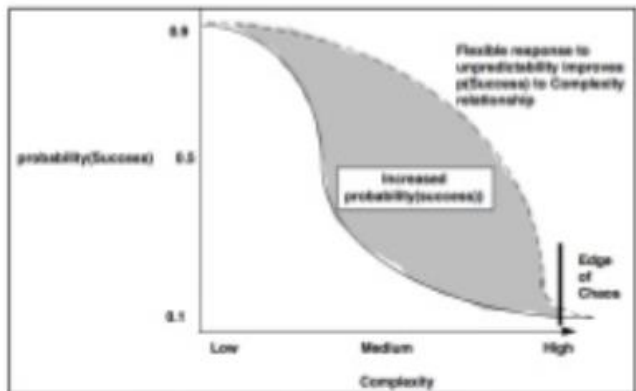
Exhibit 1 Sequential (A) vs. overlapping (B and C) phases of development



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

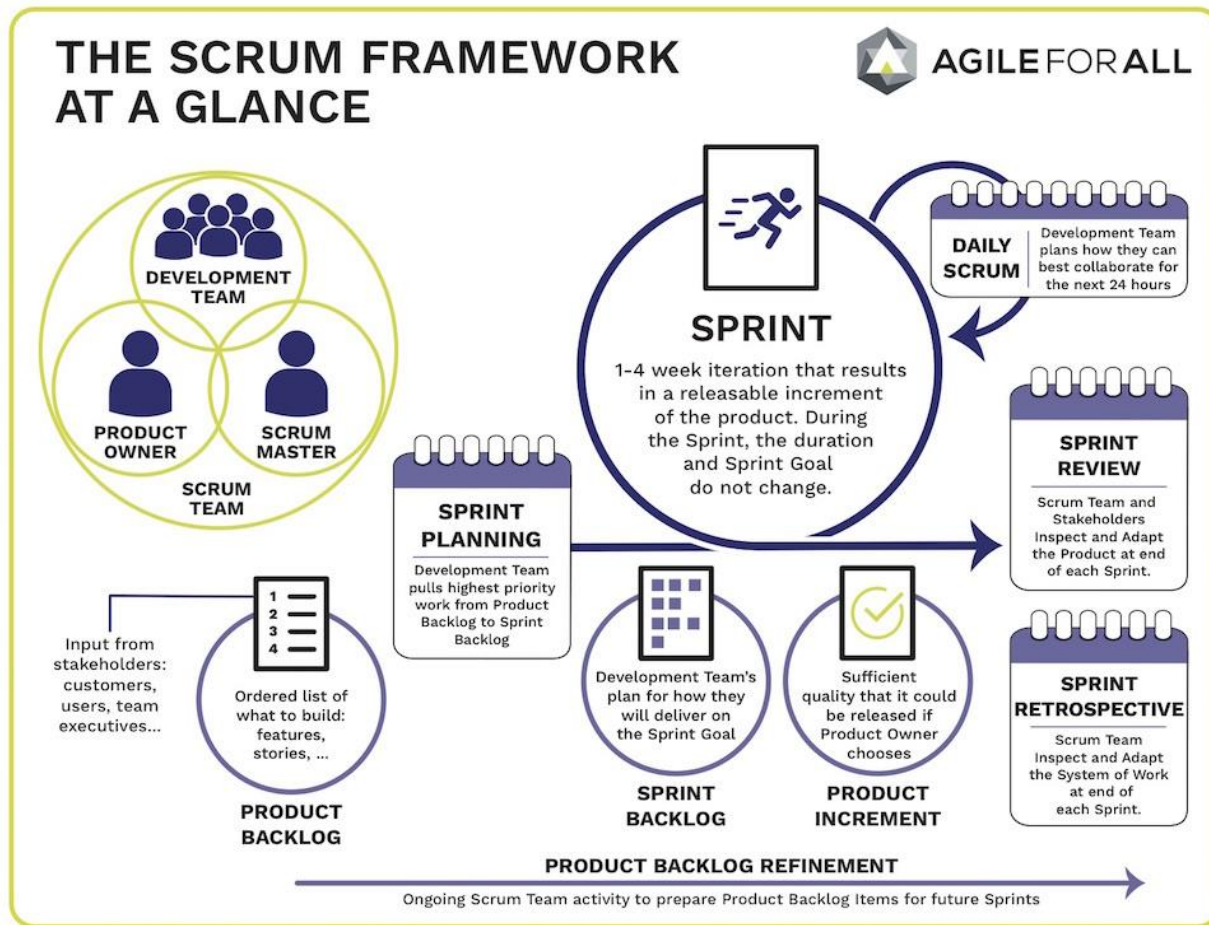
- Scrum의 탄생
  - 1995년 Ken Schwaber가 OOPLSA 1995에 처음으로 'SCRUM Development Process' 논문 게재
  - 복잡하게 얽혀 있는 시스템 개발은 유연성을 극대화시키고 적절한 통제를 필요로 하는 방식 제시



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● 스크럼 프레임워크 개요



출처:

<https://agileforall.com/resources/introduction-to-agile/>

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● 스크럼 진행 순서

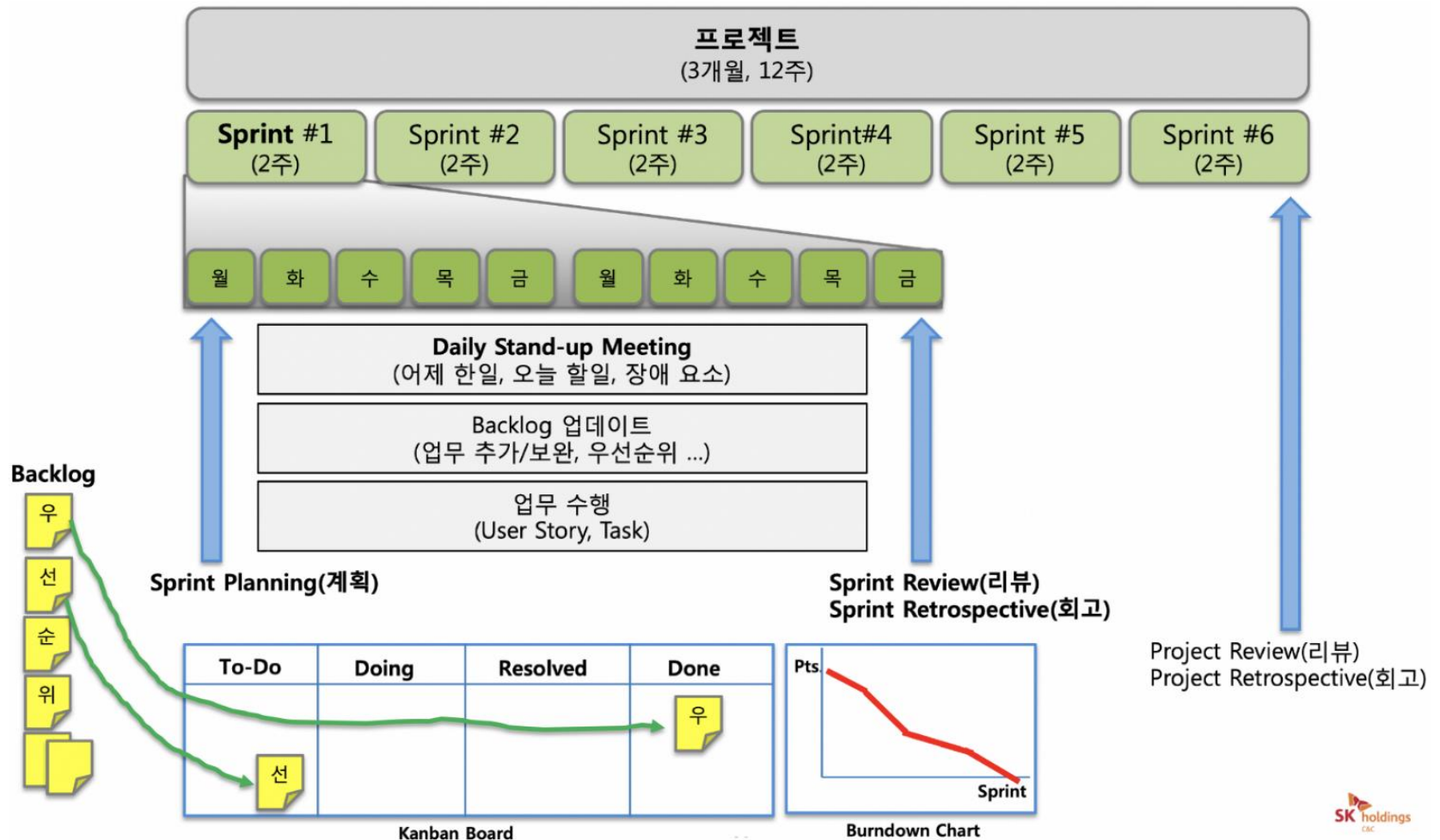
- ① PO는 제품의 요구기능(User Story)과 우선순위를 제품 백로그로 정한다.
- ② PO가 정한 제품의 우선순위에서 어디까지 작업을 할지 팀과 조율 한다.
- ③ 스프린트 목표를 구현 가능 하도록 팀에서 스프린트 백로그를 작성한 뒤 작업을 할당한다.(해설:PO는 기능과 우선순위에 대한 권한이 있고, 개발팀은 Sprint내에 해야 할 업무량을 결정할 권한이 있다. 특히 개발경험 있는 PO가 너무 적은 개발량을 문제제기 하기도 하지만, 실제 개발하는 개발팀원의 개발속도(Velocity)로 예측하며 조정이 중요하다.)
- ④ 스프린트를 진행하는 동안, 매일 정해진 장소와 시간에 모든 개발 팀원이 참여하는 일일 스크럼 회의를 가진다.
- ⑤ 매회의 스프린트가 종료할 때마다, 스프린트 리뷰(Review)를 통해 만들어진 제품을 검토하고 개선사항을 이해 한다.
- ⑥ 제품의 리뷰를 통해 제품의 지속적 개선사항 도출이 끝나면, 스프린트 회고(Retrospective)를 통해 팀의 개발 문화(프로세스)에 대한 개선의 시간을 갖는다.(해설:스프린트 Review는 제품을 개선하는 활동이고, 회고는 우리(프로세스, 문화)를 성장시키는 활동이다.)
- ⑦ 다음 스프린트에서 수행할 백로그를 PO와 필요 인원이 모여 선정하고 계획하는 시간을 갖는다.



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- 3개월 단위의 Agile 프로젝트의 전체 진행 모습



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● 주요 특징

- ① 솔루션에 포함할 기능/개선점에 대한 우선 순위를 부여한다.
- ② 개발 주기는 1~4주 정도로 하고 개발 주기마다 실제 동작할 수 있는 결과를 제공하라.  
(설명:너무 짧으면 개발(분석/설계/개발/테스트) 할 수 있는 시간이 부족하고, 너무 길면 느슨해지고 재작업의 양도 늘어나므로 적용해보면서 필요시 조율 필요)
- ③ 개발 주기마다 적용할 기능이나 개선에 대한 목록을 제공하라.  
(설명:해당 주기의 Goal을 작성하지 않으면 목적을 잃은 기능 목록이 될 수 있음)
- ④ 매일 15분 정도의 Scrum meeting 회의를 가져라.  
(설명:공유이지 보고하는 자리가 아니다. 교과서적으로 Scrum meeting은 개발팀원만 참여해야하고, 팀원이 아닌 사람은 발언기회는 없다고 한다. 개인적인 생각으로는 수평문화가 되어 있는 Agile Culture의팀이라면 PO 및 관리자가 함께 참석하여 공유하면 좋다고 생각한다. 이들도 참석한다면 이 프로젝트와 관련되어 한일/할일/이슈를 공유해야 한다. 안그러면 한팀이 아닌 관리자 모드로 돌아선다.)
- ⑤ 항상 팀을 우선으로 생각하라.  
(설명:자신의 task보다 주변 이슈가 더 급하면 도와줘야 한다. 마치 배에 구멍이 나면 그 문제 해결이 1순위이다.)
- ⑥ 원활한 의사소통을 위하여, 구분 없는 열린 공간과 마음을 유지하라.

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum의 추구 가치



© Scrum.org

출처:

<https://www.scrum.org/resources/blog/5-scrum-values-take-center-stage>

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Scrum의 추구 가치

- ① 용기 : 옳은 일을 할 수 있도록 팀원간 갈등과 도전을 위한 용기를 가져라!  
(설명 : 해당 기능이 이해가 안되거나 문제가 있다면 말할 수 있어야 하고, 더 일을 잘 할 수 있는 환경을 요구하고, 자신의 신념을 설득 시켜야 한다. 또한 도전적으로 시도해보는 용기와 완료 할 수 없는 업무량이라고 모두 말 할 수 있어야 한다.)
- ② 집중 : 할 일을 하라. 모든 노력과 기술은 성공을 위해 집중하고, 그 외는 걱정(신경쓰지) 마라!  
(설명 : 한번에 하나의 일부터 마무리하고, 업무에 집중 할 수 있도록 불필요한 회의 참석은 지양하며, 루틴한 반복 작업은 제거 하거나 자동화해야 한다.)
- ③ 헌신/책임/공약 : 팀의 목표 달성을 위해 개개인이 공약한 목표 달성을 위해 팀에 헌신하며, 이를 달성 위해 필요한 모든 권한을 부여하라!  
(설명 : 개인보다는 팀성과 달성이 우선이고, Value 있는 SW를 개발 할 수 있게 최대한 지원과 권한이 필요하다.)
- ④ 존중 : 경력과 경험이 사람을 만든다. 팀원들을 존중하라!  
(설명 : 개개인별로 장단점이 있고, 그 사람이 그렇게 할 수 밖에 없는 이유가 있을것이다.)
- ⑤ 투명성/개방성 : 프로젝트에 대한 모든 내용을 투명하게 공개하라!  
(설명 : 제품백로그, 스크럼 회의, 스프린트 리뷰를 통해 공유되며, 자신에게 불리해도 숨기지 않고 도움을 요청한다.)

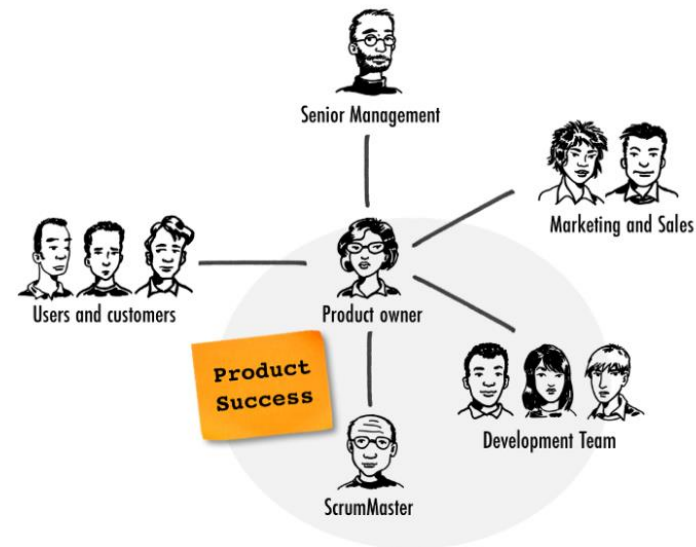
# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Scrum의 주요 역할자

#### • 제품 책임자(Product Owner)

- ① 비즈니스 목표를 충족시키는 제품을 만들기 위해 제품 백 로그를 관리하고 제품을 검토합니다.
- ② 제품 백로그(요구사항) 관리/설명
- ③ 제품 백로그의 우선순위 관리
- ④ 만족스럽게 개발되었는지 확인



출처:

<https://www.romanpichler.com/blog/every-great-product-owner-needs-great-scrummaster/>

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum의 주요 역할자

- 스크럼 마스터(ScrumMaster)

- ① Product Owner와 Development Team이 가치(Value)와 원칙(Principle)으로 성공적인 제품을 만들고, 조직 변화를 촉진하고 민첩한 작업 방식을 수립하여 유지할 수 있도록 책임을 가집니다.
- ② 팀을 보호하고 장애 요소를 해결
- ③ 일일 스크럼 회의를 진행
- ④ 모니터링 및 Tracking

- 개발 팀 (Developer)

- ① 최선의 기술로 백로그를 개발하여 고객을 만족시킵니다.



출처:

<https://www.romanpichler.com/blog/every-great-product-owner-needs-great-scrummaster/>

# 1-2. Agile과 Scrum(스크럼)

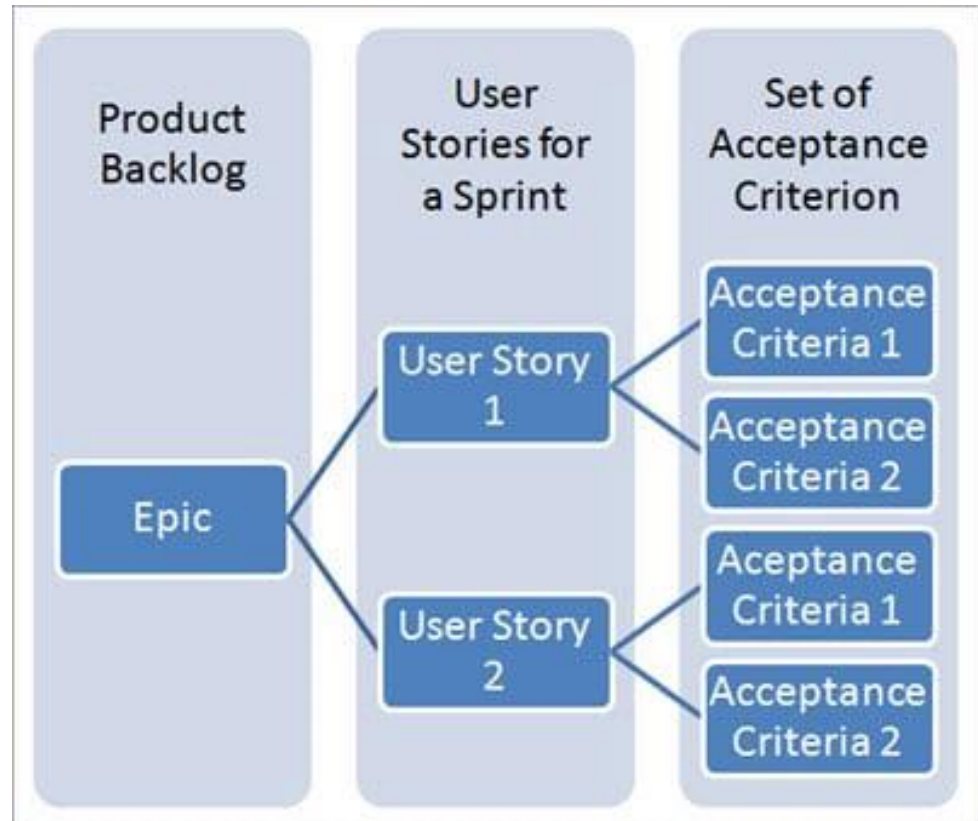
## Agile Scrum(스크럼) 이해하기

### ● Scrum 주요 활동

- 제품 백로그(Product Backlog) : 개발할 제품의 요구사항인 사용자 스토리 집합이며, 우선순위로 관리

ToDo List

ID	Story	Estimation	Priority
7	As an unauthorized User I want to create a new account	3	1
1	As an unauthorized User I want to login	1	2
10	As an authorized User I want to logout	1	3
9	Create script to purge database	1	4
2	As an authorized User I want to see the list of items so that I can select one	2	5
4	As an authorized User I want to add a new item so that it appears in the list	5	6
3	As an authorized User I want to delete the selected item	2	7
5	As an authorized User I want to edit the selected item	5	8
6	As an authorized User I want to set a reminder for a selected item so that I am reminded when item is due	8	9
8	As an administrator I want to see the list of accounts on login	2	10
Total		30	



출처:

<https://www.softwaretestinghelp.com/user-story-acceptance-criteria/>

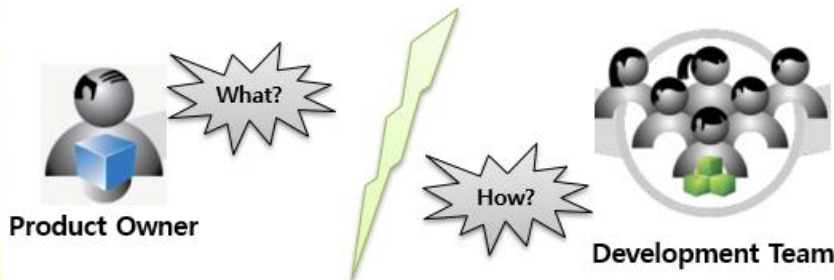


# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Scrum 주요 활동

- 사용자 스토리(User Story) : 과거 요구사항 명세처럼 업무 범위를 구체화하기 위한 개발자 입장이 아닌, User Story는 사용자가 사용하는 관점에서 어떤 가치를 제공할 것인지를 설명  
(해설 : PO는 이 기능이 누구에게 무슨 value를 제공하는지를 설명하고, 향후 개발자는 그 기능의 Value를 제공하기 위한 기술적인 역할과 책임을 가짐)



### User Story Card Examples

**As a Consumer, I want** to be able to see my daily energy usage **so that** I can lower my energy costs and usage

**As an administrator, I can** set the consumer's password security rules **so that** users are required to create and retain secure passwords, keeping the system secure.

**As a utility Marketing Director, I can** present users with new pricing programs **so that** they are more likely to continue purchasing energy from me.

**As a technical support member, I want** the user to receive a consistent and clear message anywhere in the application **so that** they can fix the issue without calling support.

See *Agile Software Requirements* by Dean Leffingwell for the Tendril case study and more examples

출처:

<https://www.softwaretestinghelp.com/user-story-acceptance-criteria/>



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum 주요 활동

- 사용자 스토리(User Story)

완료 기준(Definition of Done), 인수 기준(Acceptance Criteria) : 사용자 스토리를 완료시키기 위한 조건 명세(Given, When, Then)

Example:

```
Feature: User shopping check out
  Scenario: User successful Check out
    Given: user logger in
      And: user has billing and shipping details
5.    And: user on check out page

      When: user confirms on the shipping address
        And: user selects the payment details
        And: user performs the check out
10.   Then: application lands on confirmation page
      And: user should able to print the confirmation
```

출처:

<https://twitter.com/kickstartpros/status/524414603359686657>

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum 주요 활동
  - 사용자 스토리(User Story) 목록

ID	Story	Estimation	Priority
	Authenticated users must be able to send private messages		must
	Architects must track issues' progress		should
	There should be a notification about incoming private message		could
	Multiple message providers could be supported		won't
	Encrypted private messages won't be supported		

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum 주요 활동
  - 사용자 스토리(User Story) 목록

User Story	Tasks	Day 1	Day 2	Day 3	Day 4	Day 5	...
As a member, I can read profiles of other members so that I can find someone to date.	Code the ...	8	4	8	0		
	Design the ...	16	12	10	4		
	Meet with Mary about ...	8	16	16	11		
	Design the UI	12	6	0	0		
	Automate tests ...	4	4	1	0		
	Code the other ...	8	8	8	8		
As a member, I can update my billing information.	Update security tests	6	6	4	0		
	Design a solution to ...	12	6	0	0		
	Write test plan	8	8	4	0		
	Automate tests ...	12	12	10	6		
	Code the ...	8	8	8	4		

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- 스프린트 계획 Sprint Planning[Before Sprint]: : 각 스프린트에 대한 목표를 세우고 제품 백로그로부터 스프린트에서 진행할 항목을 선택. 각 항목에 대한 담당자를 배정하고 태스크 단위로 계획을 수립.
- Scrum 3가지 미팅
  - 일일 스크럼 Daily Scrum Meeting[During Sprint]: 매일 진행하는 15분 간의 프로젝트 진행상황을 공유하는 회의. 매일 어제 한일, 오늘 할일, 해결해야 할 장애/문제 요소를 공유하는 회의(매일 15분 정도 수행)  
(해설:모든 참여자가 보고가 아닌 수평적으로 공유 차원에서 진행해야 하며 팀원이 아니면 발언권은 없다. 개인적인 생각으로는 만약 PO, 관리자 등도 참여한다면 똑같이 3가지를 공유해야 한다. 그렇지 않으면 듣는 자리, 즉 보고 받는 자리로 변질되고, 개발자 관점에서는 관리자가 노는 사람 처럼 보인다. 또한 Sprint Planning에 없는 제3자가 시킨 잡무를 하고 있다는 것도 파악이 가능하다. 장애요소는 화이트보드에 적어놓고 지속적으로 해결한다. 만약 해결보다 쌓이는게 만다면 회사는 팀을 충분히 지원하지 않는 것이며, 일일 스크럼 회의는 생산적이지 않은 낭비로 인식된다.)

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Scrum 3가지 미팅

- 스프린트 리뷰 Sprint Review[After Sprint]: 스프린트 목표를 달성했는지 작업 진행과 결과물을 확인하는 회의. 스프린트 마지막날 개발자가 개발한 내용을 Stakeholder, 고객, 제품 책임자에게 시연하고 검토(4주 스프린트 기준 4시간 정도 수행). 고객이 참여하는 것이 좋다. 이 때, 스크럼 마스터는 스프린트 동안 잘된 점, 아쉬웠던 점, 개선할 사항 등을 찾기 위한 회고를 진행할 수 있다.
- 스프린트 회고 Sprint Retrospective[After Sprint] : 스프린트 마지막날 좋았던 점, 개선할 점을 도출하고 더 나은 방향으로 개선(4주 스프린트 기준 3시간 정도 수행)

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum Checklist(스크럼 체크리스트)

- Burndown Chart

- Burn down chart는 남아 있는 work (backlog) 와 시간 사이의 관계를 그래픽으로 표현한 2차원적인 차트이다. Run chart (런차트)라고 불리며 일반적으로 agile software development methodologies에서 사용된다. (Scrum). 예를 들어보면 약 2정도의 scrum으로 발생하는 모든 work를 나열하고 각 day에서 해결되는 수를 연결한 차트를 그리는 것이다.
- 만약 Actual Task가 더 아래있게 되면 원래 예상했던 것보다 더 적은 일이 남아있는 것을 의미하고 위에 있게되면 그 반대를 의미하게 된다.
- ※ 기대되는 효과
- up-to-date project status가 visible 하게 보이고 모든 팀멤버가 이를 지켜보고 스스로 혹은 서로 이를 해결하기 위해서 노력할 수 있게 된다. 기존의 PM만이 이를 알고 공유를 해야만하고 팀원을 push 해야하는 것에 비해 더 많은 encourage와 참여가 있을 것으로 보인다.

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Scrum 프로세스 산출물

- 스프린트 Sprint : 달력기준 1~4주 단위의 반복개발기간을 가리킵니다.
- 3가지 산출물 :
  - 제품 백로그 Product Backlog : 제품에 담고자 하는 기능의 우선순위를 정리한 목록, 고객을 대표하여 제품 책임자가 주로 우선순위를 결정합니다. 제품 백로그에 정의된 기능을 사용자 스토리라고 부르며 사용자 업무량에 대한 추정은 주로 스토리 포인트라 불리는 기준을 이용합니다.
  - 스프린트 백로그 Sprint Backlog : 하나의 스프린트 동안 개발할 목록으로 사용자 스토리와 이를 완료하기 위한 작업을 태스크로 정의합니다. 각각의 태스크의 크기는 시간 단위로 추정합니다.
  - 제품증가 Increment of Product : 모든 스프린트의 필수 결과. 제품의 통합 버전

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

### ● Agile 성과지표

- 프로젝트가 계획한 대로 잘 진행되고 있는지 파악하려면 프로젝트의 진행 상태를 객관적으로 분석할 수 있어야 한다.

분류	지표	측정 목적	지표 정의	측정 시기
일정	포인트 진척률(%)	일정 시점에서 프로젝트가 실제달성한 실적을 파악	완료 포인트/전체 포인트	매주 또는 스프린트 단위
	백로그 진척률(%)		완료 백로그/전체 백로그	매주 또는 스프린트 단위
	번다운 차트	단위 기간 동안에 남은 작업량을 파악	시간 경과에 따른 남은 작업량을 그래프로 표시한 도표 스프린트 번다운 차트 릴리스 번다운 차트	매일 또는 스프린트 단위
	번업 차트	단위 기간 동안에 완료된 작업량을 파악	시간 경과에 따른 남은 작업량을 그래프로 표시한 도표 스프린트 번다운 차트 릴리스 번다운 차트	매일 또는 스프린트 단위
생산성	속도(velocity)	단위 기간 동안에 완료된 팀의 생산성 파악	스프린트 기간에 수행된 제품 백로그들의 스토리 포인트 합	스프린트 단위
팀워크	팀 사기(team morale)	개발팀의 사기 파악	팀의 사기와 협업 수준의 설문 측정	월간
품질	결함발생률	시스템의 품질 상태 파악	결함 발견 건수	월간



# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Agile 성과지표
  - 스토리 포인트 진척률과 백로그 진척률 예

스토리 포인트 진척률

스프린트	1	2	3	4	5	6	계
계획	45	60	62	64	68	10	309
실제	40	55	55				150
진척률	88.9%	91.7%	88.7%				48.5%

제품 백로그 진척률

스프린트							
계획							
실제							
진척률							

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Agile 성과지표
- 스프린트 백로그 작업공수 기록표 예

ID	제품 백로그 항목	No.	스프린트 백로그	담당자	공수	2 / 2	2 / 3	2 / 4	2 / 5	2 / 8	2 / 9
1	사용자 로그인	1.1	사용자는 회원 아이디와 비밀번호를 입력하고 로그인할 수 있다 .		2	2					
		1.2	비회원은 회원 가입 없이 1 회 로그인을 할 수 있다 .		1	1					
2	도서 검색 (제목별 , 저자별 )	2.1	사용자는 제목별 , 저자별로 도서를 검색할 수 있다 .		3	2					
		2.2	사용자는 선택한 책의 상세 정보를 볼 수 있다 .		2	2					
계					8	7					



- Agile 성과지표
- 팀 사기 측정 예

[illegible]

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

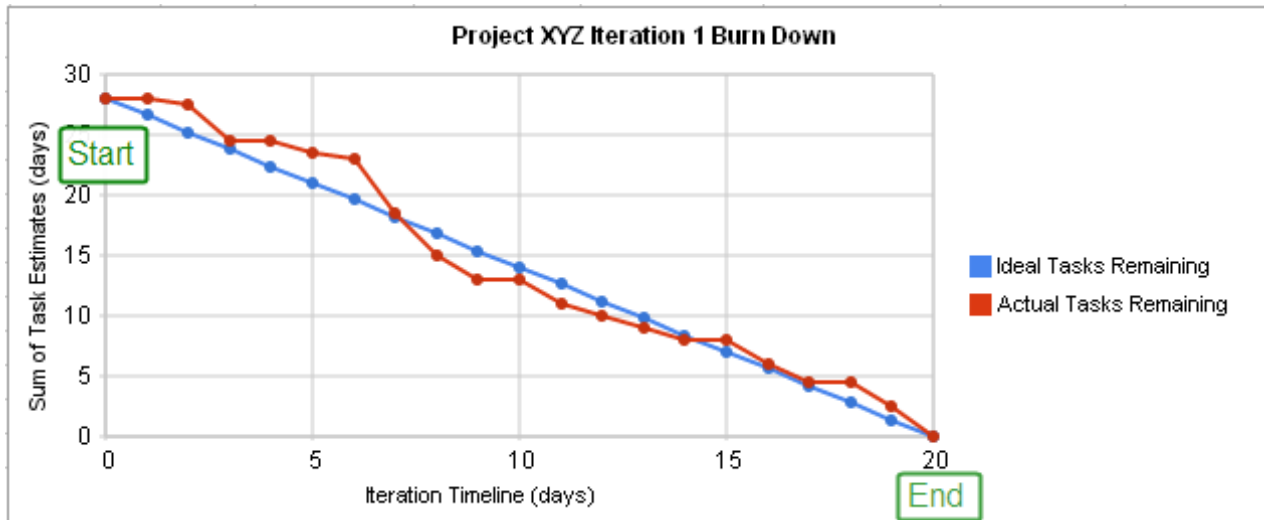
- Agile 성과지표
- 팀 사기 측정 결과지 예

?월 ?일 (금)				
	1 = 전혀 아니다 / 5 = 매우 그렇다	말하는 방향 ->		
질문		Harry	Hermione	Ronald
1	이번주 스프린트를 진행하면서 나는 에너지가 넘쳤다.	3	3	3
2	나는 이번주에 한 일들이 자랑스럽다.	4	4	4
3	이번주에 한 일들은 나에게 도전적인 일들이었다.	4	4	4
4	이번주 스프린트 이슈들이 나에게 적합(fit)하다고 느꼈다.	4	4	3
5	이번주 스프린트에서 난관에 부딪혔을 때 빨리 회복할 수 있었다.	2	3	4
6	다음도 이번주 스프린트 같아도 오랫동안 계속할 수 있다.	3	2	3
	개별 의욕도	3.33	3.33	3.50
	팀전체 평균 의욕도	3.39		
	다짐	책을 더 읽자	리모트 잘 활용하기 바이크 많이 타기	여행가서 리프레쉬 하기

# 1-2. Agile과 Scrum(스크럼)

## Agile Scrum(스크럼) 이해하기

- Scrum Checklist(스크럼 체크리스트)
- Burndown Chart



- ① X 축은 프로젝트 혹은 iteration 의 timeline을 의미한다.
- ② Y 축은 프로젝트에서 완료되어야 할 work의 수를 나타낸다.
- ③ Start point 는 iteration 혹은 프로젝트의 시작점 day 0 을 뜻한다.
- ④ End point는 프로젝트 혹은 반복일의 마지막 날을 뜻한다.
- ⑤ Worker의 수와 효율성의 관계: 예를 들어 2명이 70%의 효율로 일을 한다고 했을 때 계산은 28일을 기준으로  $(28/2)/0.7 = 20$  일로 결과값이 나오게 된다.

출처:

웹

# 프로젝트 계획하기

## 2교시 :

### 이터레이션 제로





# 1-1. 이터레이션 제로 스타트업 미팅

## 학습목표

- Agile 팀에 대해 정의할 수 있다.
- Agile 팀 구성에 대해 알 수 있다.
  - 팀 리더
  - 팀원
  - 전문가
- 현장 고객 On-site Customer에 대해 정의할 수 있다.
- 고객에 대해 정의할 수 있다.
- Agile 소프트웨어 개발자는 문제 분석 활동을 수행할 수 있다.

## 눈높이 체크

- Agile 팀 구성에 대해 알고 계신가요?



# 1-1. 이터레이션 제로 스타트업 미팅

## 이터레이션 제로 스타트업 미팅과 팀원 모집

- 팀원 모집

- 필요한 모든 팀 기술을 결정하고 모집에 합의한다. 이것은 결정된 팀 규모(팀 리더를 포함해 최소 3명, 최대 10 명)와 다양한 기술 또는 전문적 자원의 필요 여부에 기반한다. 팀에서의 역할들을 확인해 명확한 팀 구조를 만든다. 그런 다음 팀원을 내부에서, 외부에서 혹은 둘 다로부터 모집할 지를 결정한다.



# 1-1. 이터레이션 제로 스타트업 미팅

## 이터레이션 제로 스타트업 미팅과 팀원 모집

### ● 팀 구성 절차

- ① 팀 구성원에게 서로 소개하기
- ② 브레인스토밍을 통해서 팀 이름 정하기
- ③ 팀 구성의 목적 공유하기
- ④ 팀 활동을 통해서 습득할 지식 및 능력에 대해 토의하기
- ⑤ 팀 리더 결정
- ⑥ 팀원 임무 규정문 작성하기
- ⑦ 모든 팀 구성원의 연락처 공유하기
- ⑧ 팀 프로젝트 수행을 위한 정기 회의 시간 결정하기
- ⑨ 팀 운영 규칙 작성하기
- ⑩ 팀 구성 보고서 작성하기

# 1-1. 이터레이션 제로 스타트업 미팅

## 이터레이션 제로 스타트업 미팅과 팀원 모집

### ● 팀 구성 보고서

팀 구성 보고서					
일자	__년 __월 __일		팀 리더		기록자
팀 구성 목적					
팀 운영 기간	__년 __월 __일 ~ __년 __월 __일				
팀 이름					
팀 구성원 명단	이름	이메일		휴대폰	

첨부 1: 팀원 임무 규정문

첨부2 : 팀 운영 규칙

# 1-1. 이터레이션 제로 스타트업 미팅

## 이터레이션 제로 스타트업 미팅과 팀원 모집

### ● 팀원 임무 규정문

팀원 임무 규정문 개정안			
팀 이름		일자	_년 _월 _일
팀리더		기록자	
팀 구성원			
임무	담당자	주요 책임	
팀 리더		팀 관리자로서 프로젝트의 전체적인 진행을 관리하고, 팀원들의 의견을 조율하며 작업 내용을 할당 한다.	
총무		회계 업무 및 팀 행사를 담당한다.	
회의 주체자		회의 관련 내용을 준비하고 진행한다.	
회의 기록자		회의록을 작성 배포한다.	
보고서 작성 담당		팀 구성원의 보고서를 통합하여 각종 보고서와 발표 자료를 만든다.	
발표 담당		각종 발표를 수행한다.	

# 1-1. 이터레이션 제로 스타트업 미팅

## 이터레이션 제로 스타트업 미팅과 팀원 모집

### ● 팀 운영 규칙

팀 운영 규칙					
1. 프로젝트 수행을 위한 회의개최는 회의 주체자의 전일(前日) 통보 후 전원의 동의를 필요로 한다.					
2. 회의 시작 시간을 엄수하고 특별한 사유없이 지각하는 경우 음료를, 불참하는 경우 점심을 사기로 한다. 특별한 사유는 천재지변 및 관혼상제에 한한다.					
3. 회의시간은 최대 15분을 넘지 않는다.					
2. 긴급회의는 당일에 개최가 가능하며, 긴급회의 횟수는 주 1회로 제한한다.					
4. 회의록 최초 2회는 팀장이 작성하고 그후에는 팀원이 돌아가면서 작성한다.					
5. 문서 작성후 담당자는 모두의 확인(결재)을 받고 마지막 확인자는 결재 완료 폴더로 이동시킨다.					
5. 발표는 장희성,주효원,양현일 순으로 돌아가며 발표하고 최종 발표는 회의를 통해 담당자를 별도로 결정한다.					
6. 주요 사항의 결정은 팀 구성원 모두가 참석한 회의에서 만장일치로 찬성해야 한다.					
7. 기타 팀 운영 규칙에 명시되지 않은 내용은 팀 회의를 통해서 만장일치제로 결정한다.					
8. 팀 운영 규칙은 구성원 모두의 동의를 받아 개정할 수 있다.					
팀구성원	홍길동 (서명)	홍길동 (서명)	홍길동 (서명)	홍길동 (서명)	홍길동 (서명)



## 1-2. 프로젝트 계획서

### 학습목표

- 프로젝트 계획서는 항상 볼 수 있는 곳에 있어야 하고 종종 그래픽 형식으로 돼 있으며 팀의 사무실 벽에 게시된다. 도표, 마인드맵, 표 및 목록 등 모든 적절한 형식이 사용될 수 있다.

### 눈높이 체크

- 프로젝트 계획서를 알고 계신가요?



# 1-2. 프로젝트 계획서

## 개념

- 프로젝트 계획서는 간결해야 하며 모든 팀원이 볼 수 있어야 하고 명확해야 한다. 마인드맵, 포스터 등은 프로젝트 계획서 문서의 좋은 예이다.

프로젝트 계획서 목적	<p>1 페이지 분량의 프로젝트 목표: 목표, 리스크, 제약 조건, 대상 및 성공 지표. 프로젝트 계획서는 프로젝트의 이유가 한 페이지에 요약돼 있고 다음에 대한 간결한 설명을 포함한다.</p> <ul style="list-style-type: none"><li>• 식별된 목표</li><li>• 프로젝트를 완료하지 못하는 식별된 리스크</li><li>• 프로젝트로부터 기대되는 이익</li><li>• 식별된 제약 조건 (예: 예산 및 납기)</li><li>• 식별된 성공 지표</li></ul>
역할과 책임	<p>고객이 프로젝트 계획서 초안을 작성한다. 팀 리더와 제품 책임자가 이에 동의하면 이것은 방향에 대한 가이드가 된다.</p> <p>프로젝트의 방향이 크게 바뀌면 제품 책임자와 팀 리더가 현장 변경에 합의하고 팀 리더는 변경 사항을 작성해 팀이 새 프로젝트 계획서를 파악하게 한다. 제품 책임자는 먼저 고객이 변경 사항을 알고 있으며 동의하는지 확인해야 한다.</p>
보관 / 게시	<p>포스터, 스크린 세이버, 배너 - 팀 사무실의 벽면에 볼 수 있도록 게시 한다.</p>
활동	<p>프로젝트 계획서 초안은 고객에 의해 프로젝트 시작의 전제 조건으로서 작성되며 이터레이션 제로 동안 합의된다.</p>



# 1-2. 프로젝트 계획서

## 프로젝트 계획서 작성

### ● 화이트보드에 쓰여진 프로젝트 계획서

고객 광고(CEO)

CEO '광고'는 웹사이트가 여름 휴가철의 최대 구매 성수기를 대비해 업데이트되기를 바란다. 그는 마케팅 일정에 맞는 출시를 원한다.

#### ❖ 프로젝트 목표

- ✓ '마이트래블(MyTravel)' 을 가족의 여름철을 위한 휴가 선택 주요 웹사이트로 만들기
- ✓ 휴가철을 위해 온라인 검색과 구매가 가능하도록 하기
- ✓ 고객의 서비스 비용 줄이기

#### ❖ 프로젝트 목표를 달성하지 못할 경우의 리스크

- ✓ 500,000,000원으로 추정되는 매출 손실
- ✓ 경쟁사가 온라인 서비스로 제공하는 250,000,000원으로 추정되는 시장 점유율 손실

#### ❖ 기회

- ✓ 300,000,000원만큼의 매출 증가
- ✓ 1년에 200,000,000원만큼의 비용 감소
- ✓ 12% 수익 상승



# 1-2. 프로젝트 계획서

## 프로젝트 계획서 작성

### ● 화이트보드에 쓰여진 프로젝트 계획서

#### ❖ 제약

- ✓ 제품 책임자인 '숙자'는 '마이샵'(MyShop)과도 작업할 예정이어서 일부 시간만 가능함
- ✓ 작업 예산은 300,000,000원

#### ❖ 완료 조건

- ✓ 12월 1일 전에 웹사이트 완료
- ✓ 마케팅 일정에 맞춰 마케팅이 요청하면 웹사이트가 론칭해야 함

#### ❖ 성공 지표

- ✓ 매출/수익 증가
- ✓ 비용 감소
- ✓ 시장 점유율 유지 또는 증가, 고객으로부터의 좋은 피드백





## 1-2. 프로젝트 계획서

### 프로젝트 계획서 작성

- 포스터로서의 프로젝트 계획서
- 포스터로서의 프로젝트 계획서 '타이거 토이즈'(TigerToys) 웹사이트와 온라인 쇼핑 프로젝트 팀은 팀 룸 벽에 부착할 포스터에 프로젝트 계획서를 만든다.

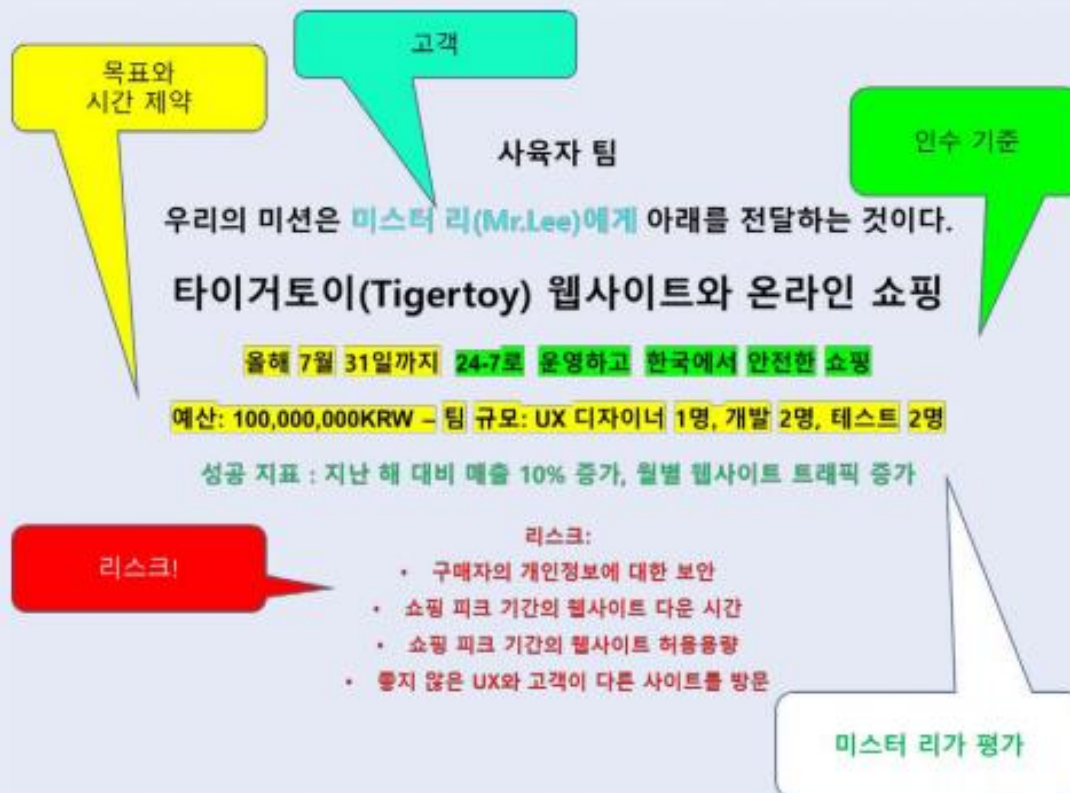


# 1-2. 프로젝트 계획서

## 프로젝트 계획서 작성

### ● 포스터로서의 프로젝트 계획서

예제 37. '타이거 토이즈'(TigerToys) 프로젝트 계획서(포스터)

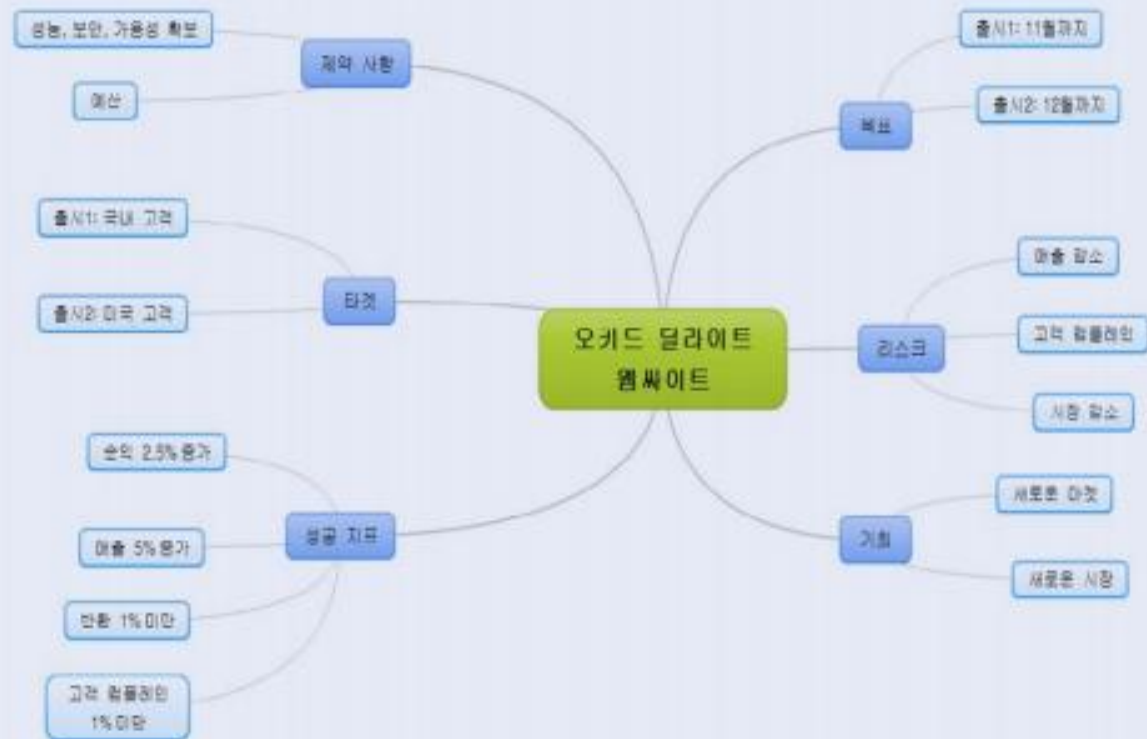


# 1-2. 프로젝트 계획서

## 프로젝트 계획서 작성

### ● 마인드맵

예제 38. '오키드딜라이트' 프로젝트 계획서(마인드맵)





## 1-3. 기타 이터레이션 제로 작업

### 학습목표

- 이 워크샵에서는 자원 계획, 아키텍처 및 기술적 결정 등 기타 이터레이션 제로 작업에 대해 알 수 있다.

### 눈높이 체크

- 이터레이션 제로 작업 산출물을 알고 계신가요?



# 1-3. 기타 이터레이션 제로 작업

## 자원 계획 (Planning resources)

- 최소한의 도구 및 기타 자원으로 시작하고 다른 것이 필요한 경우를 위한 계획을 세워라.
  - 매 이터레이션마다 인수 테스트가 수행될 예정인 경우 이터레이션1부터 인수 테스트 환경이 필요하다.
  - 인수 테스트가 덜 빈번하게 수행될 예정인 경우 인수 테스트가 수행되는 이터레이션까지는 인수 테스트 환경이 필요하지 않다.
- 팀 규모와 전문가의 참여에 따라 다음을 수행한다.
  - 사무실 공간, 데스크, 테이블 및 장비를 구성해 팀이 공동 작업할 수 있게 한다.
  - 필요한 IT 인프라, 도구, 환경을 준비한다.
  - 보안(IT, 장비, 정보 액세스)을 준비한다.
  - 관련이 있는 경우 특수 테스트 장비, 관련 표준, 특수 도구, 접근성 요구 사항 같은 필요한 자원을 식별한다



## 1-3. 기타 이터레이션 제로 작업

### 아키텍처 및 기술적 결정

- 아키텍처와 설계에 사용되는 방법은 조직이 선호하는 소프트웨어 엔지니어링 방법과 직면한 문제에 따라 다르다.
- UML
- 사용자 중심 설계
- 기능 주도 개발방법론(Feature-Driven Development)



# 1-3. 기타 이터레이션 제로 작업

## 초기의 제품 백로그 관리 및 릴리즈 계획

### ● 첫 번째 제품 백로그

#### • 첫 프로젝트인 경우 다음이 권장된다.

- ① 현재의 릴리즈 패턴(존재할 경우)으로부터 개선할 수 있는 릴리즈 계획을 세운다.
- ② 과거에 소프트웨어가 매년 릴리즈되었다면 첫 개선은 분기별로 4회 릴리즈를 수행하는 계획을 세운다.
- ③ 새 프로젝트에 대해 사용자 및 제품 책임자는 매달 인수 테스트 및 배포를 수행한다.
- ④ 제품 백로그 및 릴리즈 계획에 추가하는 작업에 대해서는 사용자와 공동 작업할 수 있도록 합의하라.



## 1-3. 기타 인터레이션 제로 작업

### 초기의 제품 백로그 관리 및 릴리즈 계획

- 제품 책임자는 다음을 수행한다.
  - 고객의 목표와 사용자의 필요를 충족시키는 사용자 요구 사항 목록을 작성한다.
  - 사용자 및 고객에 대한 다른 요구 사항들을 토론에 참고한다.
  - 소프트웨어의 배포 빈도에 관해 사용자 및 고객과 합의한다.
  - 인수 테스트의 빈도에 관해 사용자와 합의한다.
  - 배포의 계획된 소프트웨어와 빈도를 보여주는 릴리즈 계획서를 작성하고 인수 테스트가 수행될 때 이것을 팀 리더와 함께 검토한다.
  - 우선 순위가 가장 높은 요구 사항에 대해 사용자 스토리 카드를 작성한다.





## 1-3. 기타 이터레이션 제로 작업

### 완료 정의 (Definition of Done)

- “완료(Done)”라는 용어는 이터레이션 동안 완료된 작업의 일부, 배포 준비가 된 사용자 스토리의 완성, 또는 배포될 인도 가능한 전체 소프트웨어 세트를 지칭하는 말이다. 고객, 제품 책임자, 팀 리더, 팀 및 사용자 모두가 하나의 공유된 완료 정의를 가지는 것이 중요하다.
- 첫 프로젝트인 경우 다음이 권장된다.
  - 각 활동에 대해 “미니 완료”를 정의하고 이 활동을 작업상황판의 열과 일치시키라. 또한, 사용자 스토리를 한 열에서 다음 열로 이동하는 일은 ‘미니 완료’의 정의를 충족시키는 지에 따른다.
  - 배포가 완료되었음을 의미하는 초기 완료 정의를 하라.



# 1-3. 기타 이터레이션 제로 작업

## 완료 정의 (Definition of Done)

### ● 완료 정의 (Definition of Done) 예

완료된 활동	내용	담당자
설계 완료	설계 검토 및 합의	팀 수석 아키텍트
개발 완료	작성된 코드 정적 분석 통과 100% 문장 커버리지(statement coverage)로 단위 테스트 통과 자동화된 단위 테스트 단위 회귀 테스트 수행 및 통과 자동화된 회귀 스토리 테스트 수행 및 통과	개발자
스토리 테스트 완료	탐색적 테스트 완료 수동 회귀 테스트 수행 및 통과 회귀 테스트 자동화 완료 비기능 테스트 수행 및 통과 결함 해결 (제품 백로그에 대한 수정/재테스트)	테스터



# 1-3. 기타 이터레이션 제로 작업

## 완료 정의 (Definition of Done)

### ● 완료 정의 (Definition of Done) 예

완료된 활동	내용	담당자
인수 테스트 완료	인수 테스트 세트 수행 및 통과 인수 회귀 테스트 수행 및 통과 인수 기준 충족 결함 해결 (제품 백로그에 대한 수정/재테스트)	제품 책임자
배포 완료	데이터 이동 소프트웨어 실행 소프트웨어 사용	팀 리더
완료	설계 완료 개발 완료 스토리 테스트 완료 인수 테스트 완료 배포 완료	제품 책임자 / 팀 리더



# 1-3. 기타 이터레이션 제로 작업

## 완료 정의 (Definition of Done)

### ● 완료 정의 변경

- 프로젝트를 수행한 후 개발 및 테스트 프로세스에 대한 경험 및 개선 사항을 기반으로 향후 프로젝트에 대한 완료
- 정의를 변경하고 싶을 수 있다. 예를 들어 위의 표에서 '개발 완료'에 다음을 추가할 수 있다.
  - ① 사용자 스토리에 대한 단위 테스트 성공적으로 통과 (모두 통과)
  - ② 필요한 경우 코드/설계 리팩터링
  - ③ 동료 검토 통과
  - ④ 새 코드가 빌드에 통합됨
  - ⑤ 작업상황판을 업데이트하고 모든 팀원에게 전달
- 완료 정의가 회귀 테스트의 완료를 포함하고 있음을 항상 확인



# 1-3. 기타 이터레이션 제로 작업

## 추정 준비

### ● 이터레이션 기간 결정

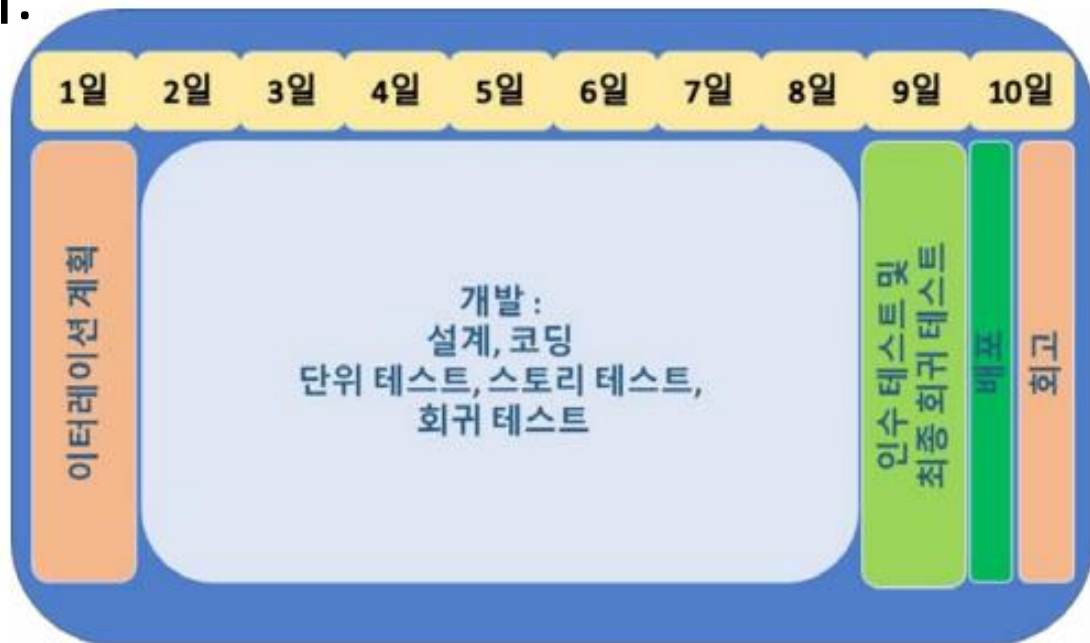
- 하나의 이터레이션은 최대 4주이다. 이터레이션 기간이 프로젝트 동안 변경돼서는 안 된다. 제품 책임자, 사용자 및 고객과 인수 테스트 및 배포 간격에 대해 합의했거나 릴리즈 계획이 있는 경우 이터레이션 기간을 그것에 맞추는 것이 좋다. 마음 편히 인도할 시간을 확보할 수 있도록 이터레이션 기간을 정하라.
- 사용자가 매 이터레이션마다 인수 테스트와 배포를 원하는 경우, 이터레이션의 마지막 주에 인수 테스트, 필요한 수정 및 배포를 수행하는 4주 단위 이터레이션으로 시작하는 것이 합리적일 수 있다. 프로젝트가 작고 리스크가 낮은 경우, 사용자와 팀 사이의 긴밀한 작업 협력 및 의사 소통을 통해 이터레이션의 마지막 날 또는 마지막 2일 동안 인수 테스트와 배포를 수행하는 1~2주 단위의 이터레이션으로 시작하는 것이 현명할 수 있다.



# 1-3. 기타 이터레이션 제로 작업

## 추정 준비

- 사용 가능한 이터레이션 일수 - 이터레이션 제로 동안 평가
- 'PIP' 팀은 2주 이터레이션 동안, 이터레이션 계획에 1일, 회고 미팅에 0.5일, 인수 테스트, 최종 회귀 테스트 및 배포에 1.5일을 할당 받았다. 이것은 설계, 코딩, 단위 테스트, 스토리 테스트 및 회귀 테스트를 포함한 개발 작업을 위해 7일이 할당됨을 의미한다.





# 1-3. 기타 이터레이션 제로 작업

## 팀 작업실 설치

- 좋은 팀 작업실 조건
  - 편안한 토론 및 미팅을 위한 장소
  - 일반적인 작업을 위한 장소
  - 개인적 미팅이나 집중적인 작업을 위한 조용한 장소
  - 작업상황판, 화이트보드, 프로젝트 계획서, 번-업 차트 같은 게시물이 있음
  - 보관 장소
  - 팀이 필요로 하는 장비가 구비됨



# 1-3. 기타 이터레이션 제로 작업

## 프로젝트 시작 미팅

- 우선 모든 사람(고객, 사용자, 제품 책임자, 팀 리더, 전문가 및 팀원)이 참석하며 제품 책임자가 회의를 주제한다.
- 릴리즈 계획과 이터레이션 기간을 소개하면서 모든 사람들이 파악하고 있는지 확인한다.
- 소개 - 모두가 서로를 알고 있는지 확인한다.
- 모든 사람을 환영하며 감사를 표한다.
- 모든 사람이 프로젝트 계획서를 파악하고 있는지 확인한다. 즉 프로젝트의 목표(목표, 제약 조건, 대상, 성공지표)가 명확해야 한다.
- 역할과 책임 - 모든 사람이 서로가 수행하는 작업, 보유한 기술, 담당할 부분을 알고 있는지 확인한다.
- 모든 사람이 정보가 있는 곳과 합의된 의사 소통 수단을 알고 있는지 확인한다.
- 고객 및 사용자에게 감사하고 미팅의 첫 번째 부분을 마칩니다.





# 1-3. 기타 이터레이션 제로 작업

## 프로젝트 시작 미팅

- 팀과 제품 책임자 참여
- 팀과 제품 책임자가 미팅을 계속 진행한다. 고객 및 사용자는 반드시 필요하지는 않다.
  - 상세한 작업 방법에 대해 합의한다.
  - 이터레이션1의 준비 여부를 확인한다.
  - 프로젝트에 전념할 것을 약속한다.



## 1-3. 기타 이터레이션 제로 작업

### 작업 산출물

- 이터레이션 제로 동안 다음의 작업 산출물을 작성한다.
  - 프로젝트 계획서
  - 최초 릴리즈 계획
  - 초기의 제품 백로그
  - 초기의 시스템 아키텍처



# 1-3. 기타 이터레이션 제로 작업

## 활동

- 작업 산출물을 작성하기 위해 이터레이션 제로 활동에는 다음의 작업들이 포함된다.
- 이터레이션 제로 스타트업 미팅: 고객과 IT 의사 결정자가 만나서 프로젝트가 수행될 것에 합의한다. 이들은 제품 책임자와 팀 리더를 지명한다.
- 프로젝트 계획서: 고객, 제품 책임자 및 팀 리더가 만나서 프로젝트 계획서에 대해 합의한다.
- 모집(Recruitment): 팀 리더는 팀원 및 모든 전문가를 확인해 요구하고 지명한다. 제품 책임자는 사용자를 확인한다.
- 자원 확보(Resourcing): 팀 리더는 필요할 인프라와 자원을 식별해 요구하며 팀은 이것들이 사용될 준비가 되었는지 확인하고 테스트한다.
  - ① 팀 작업 공간, 데스크 및 장비
  - ② IT 인프라, 도구, 환경
  - ③ 보안 (IT, 장비, 정보 액세스)
  - ④ 기타 필요한 자원



# 1-3. 기타 이터레이션 제로 작업

## 활동

- 시스템 아키텍처: 팀 리더와 팀(필요한 경우 전문가)은 초기의 시스템 아키텍처를 설계하고 검토한다.
- 제품 백로그 관리 및 릴리즈 계획: 제품 책임자는 팀 리더의 도움을 받아 초기의 제품 백로그 및 최초 릴리즈 계획을 작성한다.
- 초기의 프로세스 정의 및 완료 정의: 팀 리더, 팀원 및 제품 책임자는 프로세스와 완료 정의에 대해 합의한다.
- 이터레이션 계획 동안의 추정을 위한 준비: 프로젝트와 팀을 위한 스토리 포인트의 정의 및 이터레이션 동안의 예상 작업 속도의 초기 평가
- 프로젝트 시작 미팅: 이터레이션 제로 팀 전체가 만나서 프로젝트 이터레이션 1의 시작 준비에 동의한다. 모든 사람이 목표에 대해 약속한다. 고객은 시작 준비를 승인한다.

프로젝트 계획하기

3교시 :

# Agile Estimation과 Release Planning





# 1-1. 이터레이션 제로 스타트업 미팅

## 학습목표

- 이 워크샵에서는 제품 백로그 도출 및 관리 를 알 수 있다.

## 눈높이 체크

- 제품백로그를 알고 계신가요?



# 1-1. 제품 백로그 도출 및 관리

## 개요

- 제품백로그는 사용자 스토리 형태로 작성되며 카드 형태로 보관되거나 Excel 이나 구글닥스 같은 도구에 기록될 수 있다. 제품백로그는 제품 책임자가 담당한다.

제품 백로그 목적	제품 백로그에는 수행될 이터레이션에 아직 할당되지 않은 모든 작업 항목이 있다. 사용자의 요구 사항과 필요 사항 팀으로부터의 기술적 요구 사항과 필요 사항 해결해야 할 결함
역할과 책임	제품 책임자가 제품 백로그를 관리한다. 제품 책임자는 제품 백로그에 작업 항목을 추가하고, 제거하고, 우선 순위를 변경할 수 있다.
보관 / 게시	제품 백로그는 사용자 스토리 카드 세트로 유지되어야 하며 제품 백로그가 제품 책임자에 의해 업데이트되지 않을 때에는 팀의 사무실에 있는 박스에 보관된다.
활동	제품 백로그는 이터레이션 제로에서 처음 작성되며 이터레이션 계획 미팅들 사이에 제품 책임자에 의해 계속 업데이트된다. 다음 이터레이션에서 수행될 <b>수</b> 있도록 제품 백로그를 확인해야 한다. 제품 백로그의 크기를 최소한으로 유지한다.



# 1-1. 제품 백로그 도출 및 관리

## 제품 책임자 책임

- 제품 책임자는 이터레이션 계획이 작성될 때 가장 중요한 요구 사항이 제품 백로그에 사용자 스토리로 표현되도록 할 책임이 있다.
- 스토리에는 일반적으로 다음 네 가지의 주요 유형이 있음.
  - 사용자 요구 사항을 표현하는 스토리,
  - 다른 스토리들이 의존하기 때문에 구현돼야 하는 스토리,
  - 보다 효율적인 작성을 위해 팀이 기술적 작업을 수행해야 할 스토리
  - 해결돼야 할 결함(defects)를 표현하는 스토리.
- 스토리들의 우선 순위를 정하는 동안 제품 백로그에 있는 스토리들 중 일부가 시간이 지남에 따라 대체되었거나 이제는 진부하거나 단순히 잘못돼 있음을 발견하게 될 것이다. 이들을 제거해 향후에 백로그 관리가 수월해지도록 해야 한다.





# 1-1. 제품 백로그 도출 및 관리

## 사용자 스토리 우선 순위 정하기

- 요구 사항이나 결함의 긴급함을 문서화한 예
  - != 응급 상황!!!
  - A = 다음 인터레이션이 끝나기 전에 반드시 완료돼야 한다.
  - B = 다음 인터레이션이 끝나기 전에 완료되기 바랍니다.
  - C = 다음 3회의 인터레이션 동안 완료해 주세요.
  - D = 나중에 수행해도 된다.



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그 작성 순서

우선 해당 제품을 사용하게 될 고객의 분류를 판단해 정의합니다.

예) 상담원, 관리자, 지원부서기술자 등.



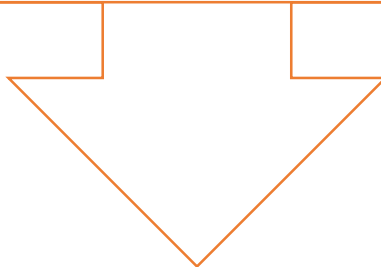
요구사항을 정제하여 짧은 문장으로 기능을 표현합니다. 이때, “누가”, “무엇을”, “어떤 이유로” 원하는 가에 대해 정확히 기술해야 합니다.

예)

**As a Role :** “고객분석담당자는”

**So that :** “현재 가입된 고객의 남녀 성비와 사는 지역의 정보를 파악하기 위해”

**I want :** “전체 고객 명단의 목록을 성별과 지역으로 정렬해서 볼 수 있다”

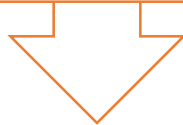




# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그 작성 순서

일반적으로 '스토리 카드'라는 이름의 인덱스 카드나 종이를 사용해서 하나의 기능을 한 장의 카드에 적을 것을 권장합니다. 스토리 카드에 적어야 할 내용이 많은 경우, Agile 관리 툴을 쓰는 것이 효과적일 수 도 있습니다.



이 사용자 스토리에 우선 순위를 정하게 되면 그에 따라 개발 분량과 일정을 정합니다.



# 1-1. 제품 백로그 도출 및 관리

## 팀 요구 사항

- 팀으로부터의 기술적 스토리 요구 사항은 두 가지 방식으로 발생할 가능성이 크다. 이터레이션 계획 중에 사용자 스토리들 중 하나에 추가적인 기술 작업이 필요하다는 것을 팀이 인식하거나 회고 동안에 도구 세트를 개선할 필요를 인식한다. 이러한 요구 사항을 평가할 때 팀의 보고를 신중하게 경청하고 팀 리더와 협력해 대안이 있는지 결정하며 단기 및 장기의 사용자에게 무엇이 최상의 가치를 제공할 것인지에 초점을 맞춘다.



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그 미팅

- 제품 백로그 미팅은 다음 이터레이션 계획 미팅을 준비하기 위해 이터레이션별로 열리거나 보다 비정기적으로 수 행될 수 있다. 이상적으로는 사용자와 팀 모두를 불러서 제품 백로그에 있는 내용을 파악하도록 하고 새 요구 사항을 추가하고, 더 이상 필요하지 않은 요구 사항을 제거하고, 다음 몇 번의 이터레이션을 위한 프로젝트 방향을 파악하는 것이다.



# 1-1. 제품 백로그 도출 및 관리

## 사용자 스토리 작성하기

- 요구공학(Requirements Engineering)이란 요구사항을 정의하고, 문서화하고, 관리하는 프로세스를 의미하며, 다음의 과정을 수행내용으로 한다.
  - 1) 수행 내용 1 / 현행 시스템 분석서 작성 및 검토하기
  - 2) 수행 내용 2 / 개발 기술 환경 요구사항 파악하기
  - 3) 수행 내용 3 / 요구사항 정의서 작성
  - 4) 수행 내용 4 / User Story 작성
  - 5) 수행 내용 5 / 사용자케이스(User Case) 작성하기
  - 6) 수행 내용 6 / 요구사항 검증하기
  - 7) 수행 내용 7 / 분석모델 검증하기



# 1-1. 제품 백로그 도출 및 관리



## 제품 백로그(Product Backlog) 작성하기

- **작업 현황판**
  - 이터레이션 계획 회의(팀 활동 계획) 중 두 번째로, 팀은 방금 이터레이션 백로그 내에 포함시키기로 합의한 스토리들을 어떻게 전달할지를 정한다.



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그(Product Backlog) 작성하기

### ● 제품 백로그 예

업무 구분	상위 기능(에픽)	ID	스토리	스토리 점수
인터넷 서점 v1.0	사용자 로그인	A1	사용자는 회원 아이디와 비밀번호를 입력하고 로그인할 수 있다.	2
		A2	비회원은 회원 가입 없이 1회 로그인을 할 수 있다.	1
	도서 검색 (제목별, 저자별)	A3	사용자는 제목별, 저자별로 도서를 검색할 수 있다.	3
		A4	사용자는 선택한 책의 상세 정보를 볼 수 있다.	2
	도서주문	A5	사용자는 배송지 주소를 입력하여 책을 구입할 수 있다.	2
		A6	사용자는 검색한 도서 목록에서 원하는 도서를 장바구니에 담을 수 있다.	3
		A7	사용자는 자신의 장바구니를 조회하고 수량을 변경/삭제할 수 있다.	2



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그(Product Backlog) 작성하기

### ● 스프린트 백로그 예

ID	사용자 스토리	완료조건	SP	스프린트 백로그			
				ID	작업	지원자	MD
1	사용자는 회원 아이디와 비밀번호를 입력하고 로그인할 수 있다 .	회원 아이디와 비밀번호가 불일치할 때는 메시지를 표시하여 다시 입력하게 한다.	2	1.1	웹페이지 설계		2
				1.2	사용자 암호 정책 설정과 구현		3
				1.3	패스워드 정책 검증 구현		2
				1.4	계정 잠금 처리 구현		3
				1.5	단위 테스트		0.5
2	사용자는 자신의 장바구니를 조회하고 수량을 변경 /삭제할 수 있다 .	특정 책의 수량 조절이 가능해야 한다.	2	2.1	웹페이지 설계		2
				2.2	장바구니 조회 구현		2
				2.3	장바구니 변경 구현		3
				2.4	단위 테스트		0.5
A				A.1	○○보고서 연동 API 개발	홍길동	3
				A.2	○○매뉴얼 개선 작업	박문수	3
				A.3	○○업무 개선 회의	팀전체	3
				A.4	코드 리뷰	팀전체	5

# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그(Product Backlog) 작성하기

### ● 스프린트 백로그 예

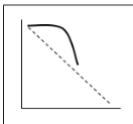
ID	사용자 스토리	완료조건	SP	스프린트 백로그			
				ID	작업	지원자	MD
1	사용자는 회원 아이디와 비밀번호를 입력하고 로그인할 수 있다 .	회원 아이디와 비밀번호가 불일치할 때는 메시지를 표시하여 다시 입력하게 한다.	2	1.1	웹페이지 설계		2
				1.2	사용자 암호 정책 설정과 구현		3
				1.3	패스워드 정책 검증 구현		2
				1.4	계정 잠금 처리 구현		3
				1.5	단위 테스트		0.5
2	사용자는 자신의 장바구니를 조회하고 수량을 변경 /삭제할 수 있다 .	특정 책의 수량 조절이 가능해야 한다.	2	2.1	웹페이지 설계		2
				2.2	장바구니 조회 구현		2
				2.3	장바구니 변경 구현		3
				2.4	단위 테스트		0.5
A				A.1	○○보고서 연동 API 개발	홍길동	3
				A.2	○○매뉴얼 개선 작업	박문수	3
				A.3	○○업무 개선 회의	팀전체	3
				A.4	코드 리뷰	팀전체	5



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그(Product Backlog) 작성하기

### ● Scrum Board - 제품 백로그(Product Backlog)

Product Backlog				SprintBacklog	To-do	In Progress	Done	Sprint Objective
ThemeA	Epic1	Story1	Task1	Story1		Task1		  Sprint BurnDownChart
			...					
		Story2	Task1	Story2	Task1			
			...					
	Epic2	Story1	Task1					
			...					
		Story2	Task1					
			...					
ThemeB	Epic1	Story1	Task1	Story1			Task1	
			...					
		Story2	Task1	Story2	Task1			
			...					
	Epic2	Story1	Task1					
			...					
		Story2	Task1					
			...					



# 1-1. 제품 백로그 도출 및 관리

## 제품 백로그(Product Backlog) 작성하기

### ● 칸반 看板Kanban Board - 제품 백로그(Product Backlog)

Product Backlog				To-do	Development		Testing		Deployment	Done
					Ongoing	Done	Ongoing	Done		
ThemeA	Epic1	Story1	Task1			Task1				
			...							
		Story2	Task1	Task1	Task1					
			...							
	Epic2	Story1	Task1							
			...							
		Story2	Task1							
			...							
ThemeB	Epic1	Story1	Task1							
			...							
		Story2	Task1	Task1	Task1					
			...							
	Epic2	Story1	Task1							
			...							
		Story2	Task1							
			...							



## 1-2. Agile Estimation

### 학습목표

- 이 워크샵에서는 Agile Estimation에 대해 알 수 있다.
- 개발 규모 추정과 스토리 점수를 알 수 있다.
- 가치 점수와 요구사항 우선순위를 알 수 있다.

### 눈높이 체크

- 스토리 포인트를 알고 계신가요?

# 1-2. Agile Estimation

## 개발 규모 추정과 스토리 점수

### ● 스토리 포인트

- 소프트웨어의 규모를 측정하는 단위로는 소프트웨어 특성에 따라 LOC, FP, 단위 화면, 웹페이지, 클래스 등이 사용되어 왔다. 하지만, 이 단위들은 불명확한 프로젝트 초기에는 사용하기 쉽지 않다. 따라서 많은 개발팀에서는 투입 공수를 기반으로 일정과 비용을 추정했다.
- Agile 개발에서는 이러한 추정 문제를 해결하고자 story point를 도입한다.

### ● 초기의 카드 분류





# 1-2. Agile Estimation

## 개발 규모 추정과 스토리 점수

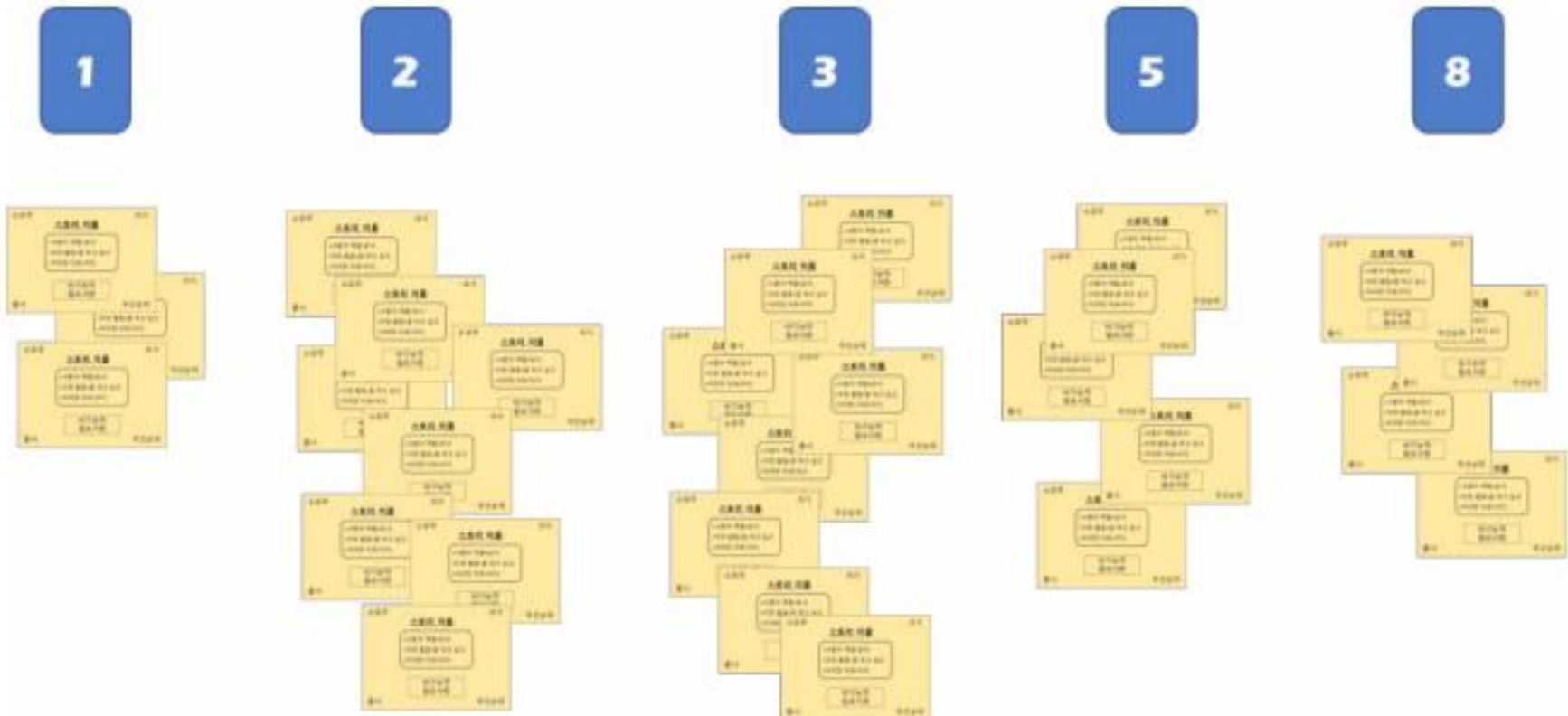
### ● 이터레이션 제로 동안 평가

- 팀은 각각의 스토리를 차례로 가져와서 그것의 설계, 코딩, 스토리 테스트 및 회귀 테스트에 소요될 시간을 대략 적으로 추정하고 합의를 도출한다. 스토리들은 5 그룹으로 분류되고 스토리들의 상대적 크기가 가장 작은 것에서 가장 큰 것까지 표시된다. 잠시 토론을 거친 후 카드들이 다음처럼 보이도록 한다.
- 그 그룹에 새로운 스토리를 추가하는 동안 어떤 스토리가 잘못된 그룹에 있다고 판단될 경우 그 스토리를 올바르게 합의된 그룹으로 옮긴다. 가장 큰 스토리도 구현하는 데 5일 이상 걸리지 않아야 한다. 너무 커서 5일 이상 걸릴 경우 보다 작은 스토리들로 나누어야 한다. 스토리 포인트는 피보나치 수열(Fibonacci sequence)의 처음 5개 숫자, 즉, 1, 2, 3, 5, 8에 기반한다. 다음으로 5개의 그룹에 라벨을 추가한다. (가장 작은 스토리를 위한 1부터 가장 큰 스토리를 위한 8까지) - 이 숫자들은 스토리의 상대적 크기를 나타낸다.

# 1-2. Agile Estimation

## 개발 규모 추정과 스토리 점수

- 스토리 포인트에 라벨 추가하기







# 1-2. Agile Estimation

## 개발 규모 추정과 스토리 점수

- 스토리 포인트에 라벨 추가하기
  - 이제 라벨에 있는 상대적 크기를 인식하면서 카드 그룹을 다시 검토해 일부 카드를 다른 카드 그룹으로 옮길 지 결정할 수 있다. 작업을 마무리하기 전에 예를 들어, '2'로 표시된 그룹의 스토리는 '1'로 표시된 그룹의 스토리를 구현하는 시간의 약 두 배 정도 걸려야 하며, '8'로 표시된 그룹에 있는 스토리들보다는 훨씬 적은 시간이 걸려야 한다는 것에 합의해야 한다. 하지만 팀원은 정확히 추정할 수 없을 수도 있다.



# 1-2. Agile Estimation

## 가치 점수와 요구사항 우선순위

- 가치 점수와 요구사항 우선순위를 정하는 방법으로는, 로직 트리 (logic tree), 델파이(delphi) 기법, BCG(boston consulting group) 매트릭스 기법, 가치점수, MoSCoW 등이 있는 데 이 중 MoSCoW 에 대해 살펴보자.

Product Backlog - MoSCoW

**M o S C o W**



**MUST**

**SHOULD**

**COULD**

**WOULD (NOT)**



# 1-2. Agile Estimation

## 가치 점수와 요구사항 우선순위

### ● MoSCoW

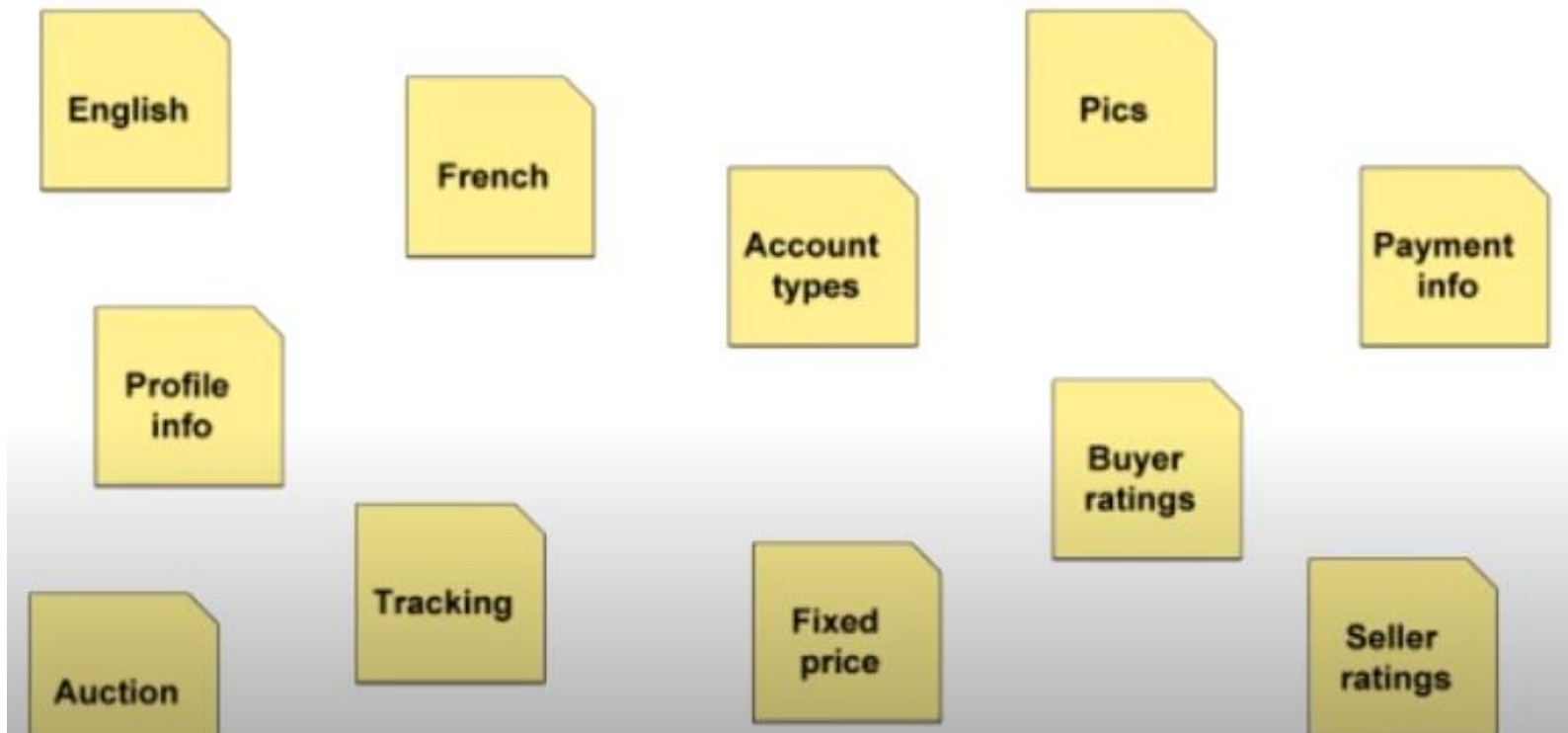
- **Must-have** : 필수, 시스템에 반드시 필요하다.
- **Should-have** : 희망, 중요하지만 단기적으로는 차선택이 있다.
- **Could-have** : 선택, 시간이 부족하다면 다음 릴리즈로 넘겨도 크게 문제가 없다.
- **Won't-have** : 보류, 좋기는 하지만 미뤄도 전혀 문제가 없다.

# 1-2. Agile Estimation

## 가치 점수와 요구사항 우선순위

- MoSCoW

- 스텝1 : 모든 요구 사항 정리





- **MoSCoW**

- | MUST        |               | SHOULD       |                | COULD   |               | WOULD (NOT) |              |
|-------------|---------------|--------------|----------------|---------|---------------|-------------|--------------|
| English     | Account types | Profile info | Seller ratings | French  | Buyer ratings | Tracking    | Payment info |
| Fixed price |               | Pics         |                | Auction |               |             |              |

# 1-2. Agile Estimation

## 가치 점수와 요구사항 우선순위

- MoSCoW

- 스텝3: 카테고리 내에서 순위 지정





# 1-3. Agile Release Planning

## 학습목표

- 이 워크샵에서는 Release Planning에 대해 알 수 있다.

## 눈높이 체크

- 릴리즈 번-업 차트를 알고 계신가요?



# 1-3. Agile Release Planning

## 릴리즈 목표

- 릴리즈 계획은 고객과 함께 도출한 사용자 스토리를 기반으로 팀 전체가 업무량을 산정하고 팀 역량에 맞추어 스프린트 별로 사용자 스토리를 할당하며 프로젝트 전체적인 개발계획을 수립하는 실천법.
- 역할자
  - 진행자: 릴리즈 계획 진행 시 퍼실리테이터(회의나 교육 등의 진행을 원활하게 돕는 역할자) 역할을 수행합니다.
  - 고객: 사용자 스토리에 대한 궁극적 책임을 지며, 사용자 스토리 내용에 대해 참여자에게 설명하고 여러 사용자 스토리의 우선순위를 결정합니다.
  - 팀원: 모든 팀원이 참석하는 것이 가장 바람직 합니다. 팀 규모가 크고, 고객 숫자가 한정되어 있는 경우 적절한 규모의 분할된 팀 간의 릴리즈 계획이 어려울 수도 있습니다. 이런 경우 분석가 역할을 하는 인력이나 업무 도메인을 잘 이해하는 인력이 대표로 참여하여 개발 계획을 수립할 수도 있습니다.



# 1-3. Agile Release Planning

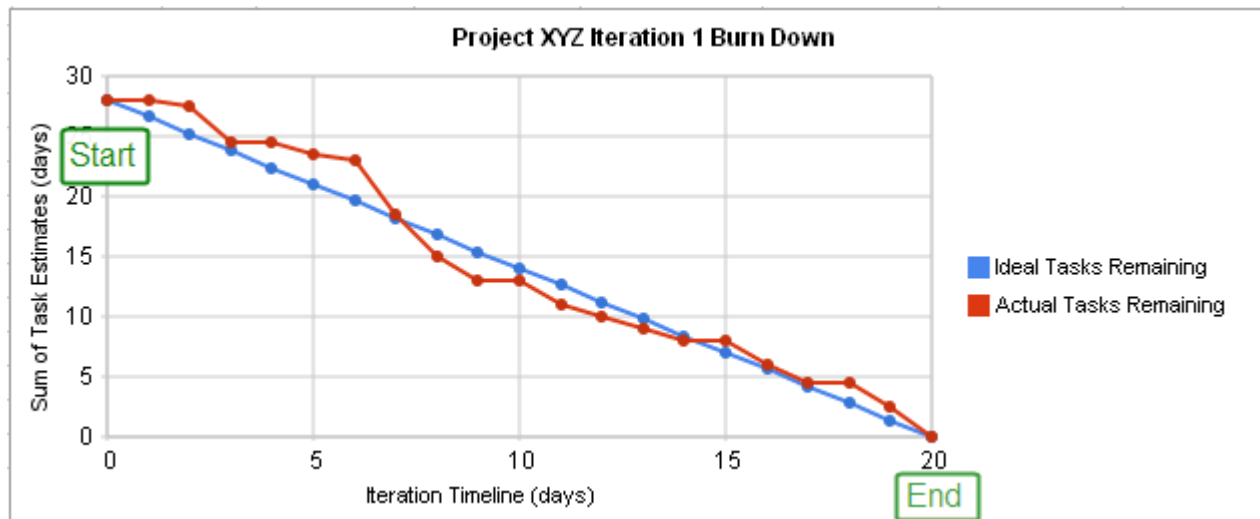
## 릴리즈 목표

- 릴리즈 계획을 시작할 때 목표 및 비전을 세워라. 고객과 제품 책임자는 고객의 비전 및 그것이 어떻게 프로젝트 목표에 부합할지 논하라.
- 제품 책임자는 동의된 사항을 기록하며 이를 모두가 볼 수 있는 곳에 전시한다. 팀 리더는 작성된 스토리들을 아래와 같은 릴리즈 상황판을 만들어 팀원들이 볼 수 있도록 벽면에 부착하거나 도구(Excel, Redmine 등)에 저장한다. 각각의 이터레이션에는 팀이 소화할 수 있는 양만큼의 스토리들을 포함시키며 이터레이션 초기에는 팀의 평균 속도보다 적은 양을 할당하고 이후 늘려간다. 마지막 이터레이션은 보통 통합테스트에 할당하거나 버퍼로 확보한다. 팀 리더는 첫 번째 릴리즈에 해당하는 릴리즈 번-업 차트를 작성하며 이것을 공개적으로 전시한다.

# 1-3. Agile Release Planning

## 릴리즈 계획 작업 산출물

- 릴리즈 계획에서는 다음의 작업 산출물이 작성된다.
- 인수 테스트가 수행될 시점을 포함해 릴리즈로 이어지는 각 이터레이션에서의 작업을 개략적으로 설명하는 릴리즈 계획서
- 팀 리더와 팀이 릴리즈를 위한 진행 상황을 식별하기 위해 사용하는 릴리즈 번-업 차트





# 1-4. Sprint Planning/ Sprint backlog

## 학습목표

- 이 워크샵에서는 Sprint Planning/ Sprint backlog에 대해 알 수 있다.

## 눈높이 체크

- Sprint Planning/ Sprint backlog 를 알고 계신가요?

# 1-4. Sprint Planning/ Sprint backlog

## | Sprint

- 개발팀은 스프린트라고 불리는 한정된 기간(Timebox)동안 계획한 일을 수행. 프로젝트 상황에 따라 다르지만 주로 1주 ~ 4주 정도를 스프린트 기간으로 정함. 이를 다른 용어로 이터레이션(Iteration)이라고도 부름.
- 팀의 역량에 따라 (속도(Velocity)에 근거하여) 스프린트 기간 동안 완료하기로 약속한 업무를 지속 가능한 리듬(Sustainable Pace)으로 작업을 수행. 팀은 스프린트 기간 동안 스프린트 백로그(Sprint Backlog)에 스스로 계획한 목표를 달성하기 위하여, 분석/설계/개발/테스트를 통해 실행 가능한 소프트웨어를 개발.



# 1-4. Sprint Planning/ Sprint backlog

## Sprint

### ● 역할자

- 스크럼 마스터: 스프린트 동안에 팀 외부로부터의 업무나 간섭 등으로부터 팀을 보호.
- 팀원: 스스로 계획한 일을 스프린트 종료일까지 완료하기 위해 노력.
- 제품 책임자: 스프린트 목표를 정하고 명확한 요구사항을 전달하도록 노력.

### ● 적용상황

- 고객이 자신이 투자한 비용으로 얼마나, 어떻게 구현되었는지 빨리 확인하고 싶어할 때
- 많은 변경이 예상되어, 먼저 동작하는 제품에 대해 고객의 확인을 받고 싶을 때

# 1-4. Sprint Planning/ Sprint backlog

## | Sprint

- 더 잘 하려면..?
  - 스프린트 계획을 통하여, 팀원 동의 하에 스스로 수행할 수 있는 정도의 작업량을 정하고 스프린트를 수행.
  - 팀원들이 최대한 스프린트 기간 동안에 목표량에 집중하여 일할 수 있는 환경이 제공.
- 어떤 것이 해결되는가?
  - 분석/설계 단계에는 상대적으로 업무 강도가 낮고 개발/테스트 단계에는 상대적으로 강도가 높던 전통적인 방식에서 벗어나, 프로젝트 전체 기간 동안의 업무 강도가 평준화 됨.
  - 실패하더라도, 빨리 적게 실패하여 프로젝트 전체에 부정적인 영향도를 줄일 수 있음.
  - 다음 스프린트에서 더 나은 방법으로 스프린트를 수행할 수 있음.



# 1-4. Sprint Planning/ Sprint backlog

## Sprint Planning

- 스프린트 동안 수행할 스프린트 목표와 이를 이루기 위한 작업 상세 내역. 작업 상세 내역은 스프린트 동안 수행할 작업을 목록화한 스프린트 백로그로 정리. 스프린트 내 완료기준을 정하고 팀원들과 공유
- 역할자
  - 스크럼 마스터: 스프린트 계획의 퍼실리테이터 역할을 수행.
  - 제품 책임자: 스프린트 목표를 정하고, 스프린트 백로그 내 스토리의 내용에 대해 팀원들에게 설명.
  - 팀원: 스스로 작업할 스토리와 작업량을 결정하고, 능동적으로 제품 책임자에게 구현 내용에 대해 질문하며, 요구사항을 상세화.



# 1-4. Sprint Planning/ Sprint backlog

## Sprint Planning

- 스프린트 동안 수행할 스프린트 목표와 이를 이루기 위한 작업 상세 내역. 작업 상세 내역은 스프린트 동안 수행할 작업을 목록화한 스프린트 백로그로 정리. 스프린트 내 완료기준을 정하고 팀원들과 공유
- 역할자
  - 스크럼 마스터: 스프린트 계획의 퍼실리테이터 역할을 수행.
  - 제품 책임자: 스프린트 목표를 정하고, 스프린트 백로그 내 스토리의 내용에 대해 팀원들에게 설명.
  - 팀원: 스스로 작업할 스토리와 작업량을 결정하고, 능동적으로 제품 책임자에게 구현 내용에 대해 질문하며, 요구사항을 상세화.



# 1-4. Sprint Planning/ Sprint backlog

## Sprint Planning

### ● 적용상황

- 스프린트 내 목표가 있고 팀이 이를 구체화 할 필요성이 있을 때
- 작업자를 명확히 결정할 필요가 있을 때
- 팀별 작업 내용에 대해 공유할 필요가 있을 때

### ● 더 잘 하려면..?

- 변경사항이 발생한 경우, 제품 백로그의 내용 및 우선순위를 업데이트 후 시작
- 제품 백로그에 있는 사용자 스토리 중 완료된 것과 미완료된 것은 명확하게 구분할 수 있어야 함.
- 제품 책임자는 스프린트 계획 대상 업무에 대해 팀원들에게 상세히 설명할 수 있을 만큼 충분히 인지하고 있어야 함.
- 이전 스프린트 수행 결과인 소멸 차트를 업데이트하고, 팀 속도를 확인
- 가능한 한 팀원 모두가 참여하여, 스프린트 동안의 수행해야 할 작업을 공유

# 1-4. Sprint Planning/ Sprint backlog

## | Sprint Planning

- 어떤 것이 해결되는가?
  - 팀원 모두가 팀이 결정한 완료기준을 재차 명확히 인지.
  - 팀원 모두가 스프린트 내 성취해야 할 명확한 목표를 설정
  - 팀원들 스스로 자신이 작업할 내용을 결정하게 되므로 작업완료에 대해 책임감을 가지게 됨
  - 타 팀원의 작업 내용을 알게 되므로, 협업이 필요한 작업을 식별
  - 팀원 모두가 자신이 해야 할 일을 보다 상세하게 인지.
  - 작업량이 많은 팀원과 적은 팀원을 인지하여, 서로 돕거나 배려.



# 1-4. Sprint Planning/ Sprint backlog

## Sprint Planning

### ● 순서

- ① 스프린트 내 수행해야 할 목표를 제품 책임자가 팀원들과 공유합니다. 이 목표는 주로 스프린트 시작 시 제품 백로그 상의 가장 우선순위가 높은 내용들을 말합니다.
- ② 스프린트 내 완료기준을 스크럼 마스터가 팀원들과 공유합니다. 혹시, 완료기준이 변경되어 수정해야 한다면, 제품 책임자를 포함한 팀원 모두의 합의하에 수정합니다.
- ③ 팀원들은 정해진 스프린트 기간 동안 수행할 사용자 스토리를 제품 백 로그에서 선택합니다. 짝 프로그래밍을 한다면, 두 명이 함께 선택할 수도 있습니다.
- ④ 작업하기로 한 스토리 포인트가 개인별로 지나치게 차이가 많이 난다면, 합의 하에 조정합니다.
- ⑤ 팀 전체가 선택한 스토리 포인트의 합을 구하고, 그 합이 이전 스프린트 들의 팀 속도보다 지나치게 높거나 낮을 경우 조정합니다.
- ⑥ 특정 사용자 스토리에 대해 스토리를 완료하기 위한 상세 태스크를 도출합니다.
- ⑦ 도출된 태스크에 담당자와 예상 작업시간을 적습니다
- ⑧ 스크럼 마스터는 태스크에 적힌 개인별 시간의 합을 구하고, 스프린트 내 작업일(Work Day)에 비추어 적절한지 판단하고 조정합니다.
- ⑨ 제품 책임자와 스크럼 마스터는 팀원들이 세운 계획을 리뷰하고, 태스 크에 식별되지 않은 작업이 있다면 추가합니다.
- ⑩ 스프린트 동안 작업할 계획에 대해 요약하여 팀원들과 공유합니다.
- ⑪ 팀원들 모두 정해진 스프린트 기간 동안 계획한 것을 완료하기로 서약 (Commitment)합니다.



# 1-5. Agile 일정 작성

## 학습목표

- 프로젝트 일정 관리의 기준이 되는 릴리스 계획을 알 수 있다.
- 일정 계획 관련 전체 일정과 주기적인 상세 일정을 수립하는 Agile 릴리스 계획을 수립할 수 있다.
- 릴리스 계획을 위한 스프린트 계획을 수립할 수 있다.

## 눈높이 체크

- 릴리스 계획 수립을 알고 계신가요?

# 1-5. Agile 일정 작성

## 릴리스 계획 수립

- 평균 개발 속도 추정

- 개발 속도는 개발팀에서 단위 스프린트에 완료할 수 있는 스토리 점수의 합
- 스프린트 1 예

- ✓ A스토리 5점 + B스토리 13점 + C스토리 3점=속도 21점

- Agile을 처음 적용하는 팀이라면 팀의 개발 속도를 알 수 없기 때문에 팀의 가용 공수를 추정하고 이를 스토리 점수로 환산

예

- ✓ 팀원 6명

- ✓ 2주일 동안 가용할 평균 투입 공수 50MD

- ✓ 1점을 수행하는 데 필요한 평균 투입 공수가 2MD라고 하면 개발 속도는

- ✓ 25점

# 1-5. Agile 일정 작성

## 릴리스 계획 수립

- 전체 일정 추정

- 제품 백로그의 전체 스토리 점수와 개발 속도를 고려하여 제품 개발에 적절한 전체 일정 계획을 수립한다.

예

- ✓ 제품 백로그의 전체 스토리 점수 = 200
- ✓ 2주일 짜리 스프린트의 평균 개발 속도가 25점
- ✓ 프로젝트의 최소 필요 일정은 전체 스토리 점수 합에 팀의 속도를 나눈 값
- ✓ 스프린트 :  $200/25=8$
- ✓ 프로젝트 일정 :  $8 * 2주=16주$ , 곧 4개월이 필요

# 1-5. Agile 일정 작성

## 릴리스 계획 수립

### ● 릴리스 계획 예

업무 구분	상위 기능	기능과 활동	점수	스프린트0	스프린트1	스프린트2	추정
인터넷 서점 v1.0	사용자 로그인	사용자는 회원 아이디와 비밀번호를 입력하고 로그인할 수 있다.	2				3
		비회원은 회원 가입 없이 1 회 로그인을 할 수 있다.	1				
	도서 검색 (제목별, 저자별)	사용자는 제목별, 저자별로 도서를 검색할 수 있다.	3				5
		사용자는 선택한 책의 상세 정보를 볼 수 있다.	2				
	도서주문	사용자는 배송지 주소를 입력하여 책을 구입할 수 있다.	2				7
		사용자는 검색한 도서 목록에서 원하는 도서를 장바구니에 담을 수 있다.	3				
		사용자는 자신의 장바구니를 조회하고 수량을 변경 /삭제할 수 있다.	2				

Unit A

# 참고자료





## 문헌

1. <http://www.ncs.go.kr>
2. AgileSW개발101,nipa
3. 방법론 가이드,SW중소기업을 위한 경량 개발 방법론, 과학기술정보통신부, nipa
4. 김치수(2015). 『쉽게 배우는 소프트웨어 공학』. 한빛아카데미(주)
5. 김향곤, 소프트웨어개발 방법론, 대구 카톨릭대학 • 기술표준원(2013. 11)
6. 프로젝트 관리기준(ISO 21500) 이행가이드
7. 과학기술정보통신부 보도자료(2016. 9. 6)
8. 상용 소프트웨어 현재와 미래 조망
9. 보건복지부(2013).
10. 지식경제부(2010.2.26.). 소프트웨어 사업대가의 기준.
11. 정보통신기술진흥센터(2016. 9. 6). 글로벌 상용 소프트웨어 백서.
12. 특허청(2014). 소프트웨어개발 방법론.
13. 한국소프트웨어산업협회(2016). 소프트웨어 사업 대가 산정 가이드.
14. CMMI (CMMI(Capability Maturity Model Integration) 모델
15. ISO/ IEC 12207 모델 (소프트웨어 개발 생명주기 프로세스)
16. ISO/ IEC 14598 품질 평가 프로세스 • SPICE (ISO 15504) 모델 (프로세스 평가 표준)
17. 린 소프트웨어 개발의 적용
18. 속도 경쟁에서 승리하기
19. 메리 포펜딕, 톰 포펜딕 공저 / 엄위상, 심우곤, 한주영 공역 | 위키북스 | 2007년 08월 31일 | 원제 : Implementing Lean Software Development: From Concept To Cash
20. T 아카데미
21. <https://www.lesstif.com/jira/>
22. 제프랭어, 앤디 헌트, 데이브 토마스, 유동환 역,자바와 JUnit을 활용한 실용주의 단위 테스트,길벗

감사합니다.

❖ Mobile: 010-9591-1401

❖ E-mail: [onlooker2zip@naver.com](mailto:onlooker2zip@naver.com) / [dkkim@lklab.org](mailto:dkkim@lklab.org)

