

노트

번다운 차트 추가

Part 2.

애자일 방법론을 이용한 프로젝트 관리

시스템 설계는, 이를 구현하는 팀의 구조와 개발 프로세스를 이해함으로써, 팀에 맞는 시스템 구조를 설계해야 한다.
이번장에서는 애자일 기반의 소프트웨어 개발프로세스를 이해하고, 어떻게 설계와 연결되는지를 이해한다.

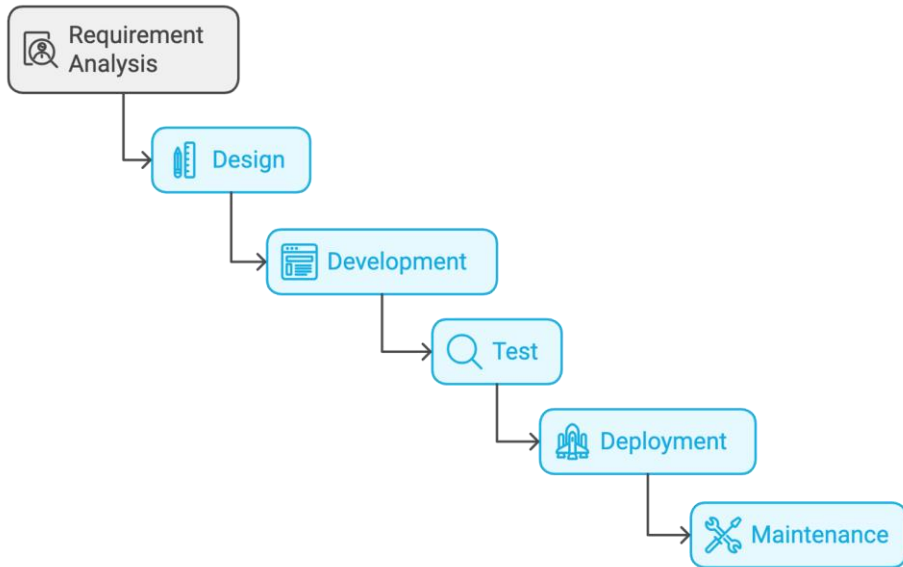
소프트웨어 개발 방법론

소프트웨어 개발 방법론

소프트웨어를 개발하기 위해서 요구사항 분석, 설계, 개발, 테스트 및 배포에 대한 절차와 원칙을 정의한 방법론

기존 개발 방법론

워터폴 모델 : 완벽한 요구 사항 분석, 디자인 후, 구현



- 학문적인 연구에서 출발
- 요구 사항 변화, 디자인 변경에 취약함
- “전단계의 결과가 완벽하다”는 전제에서 시작하지만 사람은 완벽하지 않다
- 빠른 비즈니스 환경 변화에 대응하기 어려움

기타 : CBD, RUP, Spiral model, V Model etc

실용주의 방법론

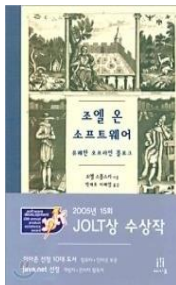
2000년대 초반 실용 주의 방법론

Erich Gamma, Joel Spolsky, Andrew Hunt

(학계가 아니라 실무진으로 부터)

- 사람은 완벽하지 않고 실수를 한다.
- 요구 사항은 항상 변화할 수 있다.

- 점진적, 반복적 (Incremental, Iterative)
- 애자일 기반
- 지속적 통합 (CI)
- 자주 테스트 (Test Automation)
- 협업과 커뮤니케이션 위주



스크럼 방법론

Input from end-user,
customer, team and other
stakeholders



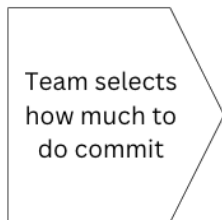
Product Owner



Team



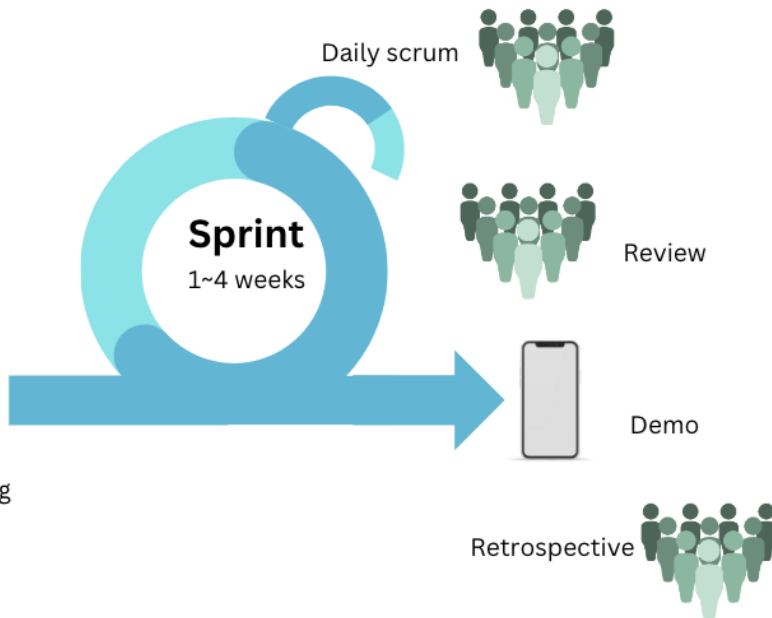
Product backlog



Team selects
how much to
do commit



Sprint backlog



Daily scrum

Review

Demo

Retrospective

스크럼 방법론

Sprint



Restropective



Grooming

Sprint Planning



Story point
voting



Sprint



Restropective



Grooming

스크럼 팀의 구조



스크럼 마스터

- 스크럼 프로세스를 관리하고 수행한다. 데일리 스크럼, 스프린트 리뷰, 스프린트 회고와 같은 각종 이벤트를 진행한다.
- 장애물 제거 (팀이 겪는 문제나 장애를 해결한다)
- 코칭과 교육



프로덕트 오너

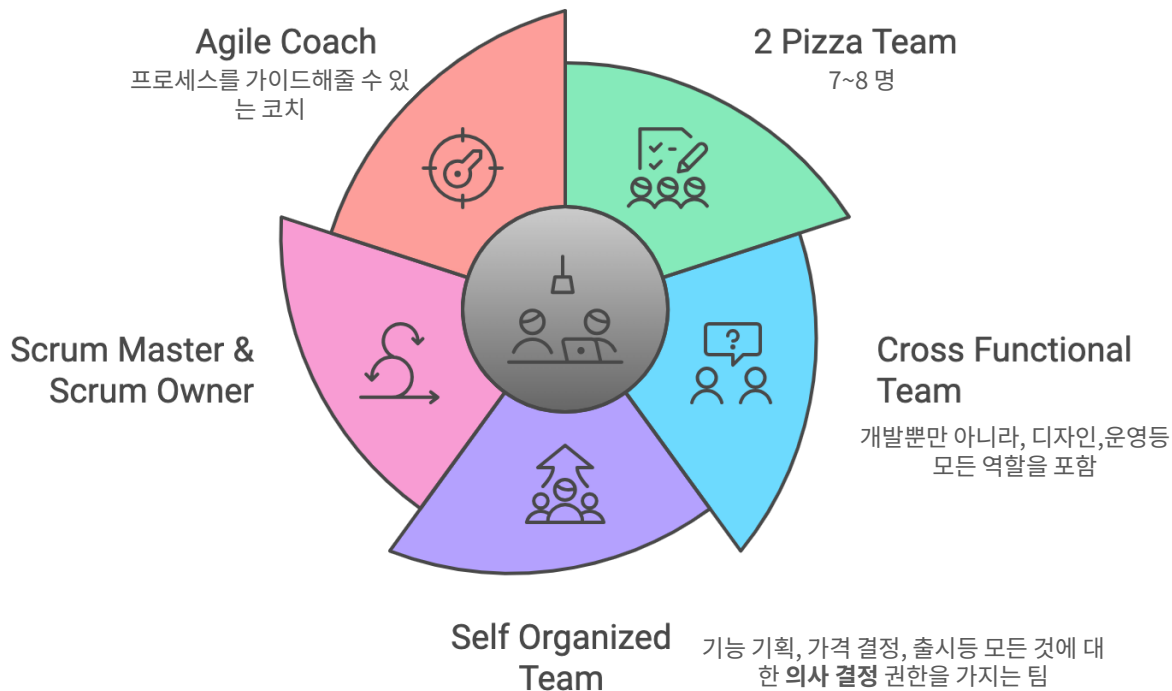
- 개발 기능 목록 (백로그) 관리 및 우선순위 조정
- 이해 관계자 조율 (고객, 사용자, 경영진등의 피드백 적용)
- 비즈니스 가치 극대화



스크럼 코치

- 필수는 아니며, 팀 외부에서 스크럼 프로세스에 대한 교육과 멘토링을 수행한다.
- 협력적이고 민첩한 문화를 구축하도록 돕는다.

스크럼 팀의 구조



JIRA를 이용한 스크럼 운영 실전

1. 그루밍 (Grooming)



스프린트 진행중, PO가 다음 스프린트에 대한 기능을 대략적으로 설명하는 행위

● 왜?

- 다음 스프린트에 대한 준비 (미리 생각할 수 있는 시간)
- 개발 시작전 리뷰를 통해서 개발 가능성 및 기획상의 미비한 부분을 미리 발견

● 어떻게

- 1시간 정도, 사용자 스토리나 UX 프로토타입을 리뷰
- 실제로 작동하는 UX 목업이 좋음 (애니메이션이나 UX의 복잡도등을 미리 예측 가능)

2. 사용자 스토리 작성

사용자 스토리

: 주요 기능을 사용자 관점에서 기술. (서비스 또는 제품의 요구사항 리스트)

“As a {User}, I want to do {Something}”

“나는 {사용자로서},{무엇}을 한다”

특징

- 직관적으로 이해가 되어야 한다.
- 테스트가 가능해야 함

대분류, 스토리, 디테일 설명형태로 기술하는 것이 좋음

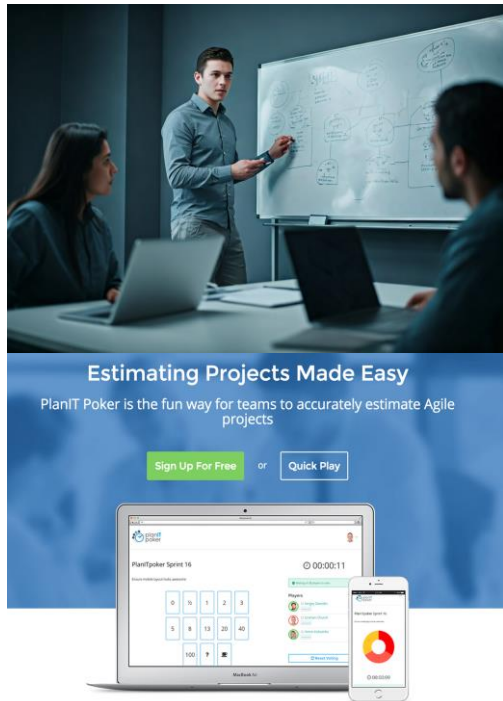
대분류

가능하면 시간 순서로

사용자 스토리

Feature (Lv 1)	Feature (Lv 2)	Details
사용자 가입 및 로그인	사용자는 구글 계정으로 로그인 한다.	사용자는 구글 계정으로 로그인한다.
	사용자는 페이스북 계정으로 로그인한다.	사용자는 페이스북 계정으로 로그인한다
비디오 피드 리스트 보기	사용자는 최근 업데이트된 순서대로 비디오 피드를 본다	사용자는 상하로 스크롤되는 비디오 피드를 볼 수 있다. 비디오 피드는 1초간 스크롤하지 않으면 현재 화면에 있는 비디오가 미리보기로 플레이된다.
비디오 촬영 및 업로드	사용자는 양백등반하는 장면을 비디오로 촬영한다	사용자는 양백 등반하는 비디오를 앱에서 촬영할 수 있다.
	사용자는 촬영한 비디오를 업로드할 수 있다.	사용자는 촬영한 비디오를 촬영이 끝나자마자 업로드하거나 또는 비디오 갤러리에서 이미 촬영된 비디오를 선택하여 업로드한다.
	사용자는 비디오를 업로드할때 해쉬태그를 추가할 수 있다.	사용자는 비디오 업로드시 제목과 설명을 추가하여 업로드한다. 업로드시에 위치 정보를 같이 업로드 한다. GPS 정보를 이용하여 위치를 업로드 하거나 또는 근처에 이미 등록된 양백등반장이 있을 경우, 양백등반장 이름을 목록으로 보여주고 선택할 수 있게 한다.
	사용자는 비디오 등록시 양백등반장 등록할 수 있다.	사용자는 비디오 등록시, 설명 부분에 해쉬태그를 추가할 수 있다.
비디오 보기	사용자는 비디오 등록시 양백등반장 등록할 수 있다.	사용자는 비디오 등록시 양백등반장을 GPS 정보를 이용하여 등록할 수 있다. 양백등반장은 이름과 위치로 등록하며, 이미 다른 사람이 등록했을 경우 유사한 이름의 등반장은 하나로 합친다.
	사용자는 피드에서 비디오를 선택해서 플레이할 수 있다.	사용자가 비디오 피드에서 비디오를 클릭하면 비디오가 재생된다.
	사용자는 비디오를 플레이중에도 비디오에 대한 댓글을 달 수 있다.	사용자는 비디오를 재생하는 화면에서 댓글을 달아서 자신의 의견을 등록할 수 있다.
	사용자는 비디오의 댓글중에 부적절한 댓글을 신고할 수 있다.	사용자는 비디오에 적절하지 않은 댓글이 있을 경우 신고하기 버튼을 이용하여 관리자에게 신고하고 삭제요청할 수 있다. 신고된 댓글은 관리자의 판단이 있을때까지 내용을 표시하지 않고 대신, "신고된 댓글입니다."라고 표시된다.
	사용자는 부적절한 비디오를 신고할 수 있다.	사용자는 비디오 재생하기나 또는 목록보기에서 적절하지 않은 비디오를 신고할 수 있으며, 신고에 대한 내용을 간단한 메모로 작성할 수 있다.
비디오 검색	사용자는 비디오를 추천할 수 있다.	사용자는 비디오를 재생하면서 마음에 드는 비디오에 "추천하기" 버튼을 누를 수 있다.
	사용자는 비디오 검색을 위해서 사진을 촬영할 수 있다.	사용자는 유사한 양백등반 코스를 검색하기 위해서, 실제 양백등반 코스를 사진을 촬영할 수 있다.
	사용자는 촬영한 사진을 이용하여 유사한 양백의 비디오를 검색할 수 있다.	사용자는 이미 갤러리에 저장된 실제 양백등반 코스나 또는 방금 촬영한 양백등반 코스 사진을 이용하여, 유사한 코스의 비디오를 검색할 수 있다.
	사용자는 비디오를 검색할때, 현재 위치와 유사한 위치에 있는 비디오를 검색할 수 있다.	일반 사용자는 일주일당 10개의 비디오를 사진 검색을 통하여 검색할 수 있으며, 유료 사용자는 무제한으로 검색을 할 수 있다.
저장하기	사용자는 비디오를 검색할때, 검색어와 유사한 비디오를 검색할 수 있다.	사용자가 사진 기반으로 검색할때, 유사한 비디오를 사진뿐만 아니라, 사용자가 GPS를 활성화한 경우, GPS 정보를 이용하여, 반경 2KM내에서 촬영한 영상을 우선으로 검색한다.
	사용자는 비디오를 검색할때 검색어를 추가하여 검색할 수 있다.	사용자는 검색시, 필요한 검색 키워드를 추가하여 검색할 수 있다. 이 검색 키워드는 비디오 설명에 있는 텍스트와 해쉬 태그를 검색한다.
	유료 사용자는 자신이 마음에 드는 비디오를 북마크식으로 저장해놓을 수 있다.	유료 사용자는 자신에게 유용한 비디오를 북마크를 통해서 저장할 수 있다.
	유료 사용자는 북마크에 저장된 비디오의 목록을 볼 수 있다.	유료 사용자는 자신이 북마크한 비디오 목록을 볼 수 있다.

3. 개발 범위 산정 (스토리포커)



- 한 사람이 구현하고자 하는 기능에 대해서 설명한다.
- 참석한 사람(프론트,백엔드, 모바일)은 기능에 대해서 궁금한점이 있으면 질문을 한다.
- 질문이 끝난 후에, 각자 개발에 **전체** 얼마가 걸릴지 투표를 한다. (프론트,모바일,백엔드 모두 포함하여)
- 투표가 끝난 후에, 가장 많은 시간과 가장 낮은 시간이 나온 사람이 이유를 설명한다. (1시간, 2시간,4시간,8시간, 16시간 이런식으로)
- 다시 투표를 한다. (시간이 일치 될때 까지)
- 16시간이 넘는 (2일 또는 3일) 경우에는 기능을 작은 단위로 분리한다.

3. 개발 범위 산정 (스토리포커 예시)

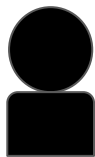


사용자 스토리

사용자는 피드에서 비디오를 선택해서 플레이할 수 있다.

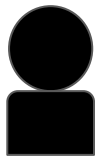
사용자가 비디오 피드에서 비디오를 클릭하면 비디오가 재생된다.

1



“저장된 동영상을 다운로드만 하는 것이기 때문에 하루면 만들 수 있어”

2



“저장된 동영상 다운로드이기는 하지만, CDN을 사용하면 비용과 성능을 높일 수 있기 때문에, CDN 연동 시간이 필요해”

4



“아무 사용자나 동영상을 보여줄 수 없잖아. 최소한 사용자 인증까지 생각하면 2일은 더 필요해”

3. 개발 범위 산정

팁

- 산출된 포인트로 개발일정 지정 (2~4주)
- 버퍼 (회의, 문서 작업, 배포, 오류 수정)등을 고려할것 (1.5~2배)
- 4주가 넘는 일정은 우선순위에 따라 다를 것

4-1. JIRA에서 프로젝트 생성

- JIRA에서 프로젝트를 생성한다.
- JIRA는 소프트웨어 개발뿐만 아니라, 운영, 고객 지원 등 다양한 이슈를 제공하며, 그에 따른 템플릿이 다르다.
- Software development > Scrum 을 선택하여 프로젝트를 생성한다.

X

Project templates

Made for you

Software development

Service management

Work management

Product management

Marketing

Human resources

Finance

Design

Personal

Operations

Legal

Sales

Analytics

IT

Project templates

Software development

Plan, track and release great software. Get up and running quickly with templates that suit the way your team works. Plus, integrations for DevOps teams that want to connect work across their entire toolchain.

[Import data to a new project](#)



Top-level planning RECOMMENDED

Jira PREMIUM

Monitor work from multiple projects, and create an easy-to-share plan for stakeholders.



Kanban

Jira

Visualize and advance your project forward using issues on a powerful board.



Scrum

Jira

Sprint toward your project goals with a board, backlog, and timeline.



4-2. 에픽 등록

JIRA를 Scrum 방법론에 맞춰서 설정할 것이다.

에픽 등록

- 에픽(EPIC)의 사용자 스토리의 집합으로, 기능의 대분류에 해당한다.
- 앞서 엑셀에서 정의한 Feature Lv1을 Epic으로 등록한다.

Feature (Lv 1)	Feature (Lv 1)
사용자 가입 및 로그인	사용자는 구글 계정으로 로그인할 수 있다. 사용자는 페이스북으로 로그인할 수 있다.
비디오 피드 리스트 보기	사용자는 최근 업로드된 비디오 피드를 본다. 사용자는 암벽등반 관련 비디오 피드를 본다.
비디오 촬영 및 업로드	사용자는 촬영한 비디오를 업로드할 수 있다. 사용자는 비디오를 추가할 수 있다. 사용자는 비디오를 삭제할 수 있다.
비디오 보기	사용자는 피드에서 비디오를 본다. 사용자는 비디오를 댓글을 달 수 있다. 사용자는 비디오의 신고할 수 있다. 사용자는 부적절한 비디오를 신고할 수 있다.

The screenshot shows the JIRA Backlog interface for the 'Terry-Scrum' project. On the left, a list of epics is shown under 'Issues without epic'. The 'Video Upload' epic is highlighted with a red box. A red arrow points from the '비디오 촬영 및 업로드' row in the table to this epic. On the right, the 'Video Upload' epic is expanded, showing a list of issues: 'User', 'Video feed', 'Video view', 'Search video', and 'Book mark'. The 'Create issue' button is visible at the bottom of the epic list.

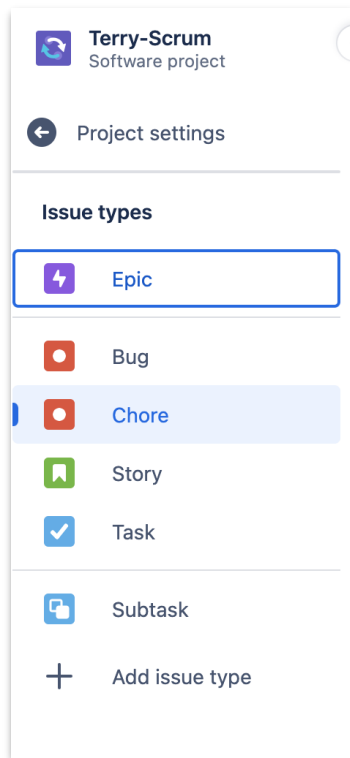
4-3. 이슈 타입 추가

해야하는 일의 타입을 정의할 수 있다.

JIRA에서는 추가로 오른쪽과 같이 이슈의 타입을 추가할 수 있다.

아래는 추천되는 이슈 타입이다.

- **Story** : User Story로, 소프트웨어의 기능에 해당한다.
- **Bug** : 개발중 보고된 버그
- **Task** : 기능이외에 해야할 일로, 인프라 설정, 로그 설정이나 선행 기술 개발등이 해당한다.
- **Sub Task** : 상위 이슈를 완료하기 위해서 해야할 일로, 상위/하위 관계를 갖는다. 예를 들어 “비디오 업로드” 라는 스토리를 구현하기 위해서는 “비디오 업로드 API구현”, “비디오 업로드 UI 구현”, “비디오 업로드 기능 테스트” 등이 이에 해당할 수 있다.



4-3. 이슈 타입 추가

이슈 필드 추가 : 이슈를 추가하는 것에 더해서, 이슈마다 필드를 추가할 수 있다. 일반적으로 권장되는 필드는 다음과 같다.

- **제목** : 이슈 이름
- **상세 설명** : 이슈에 대한 상세 설명으로 외부 문서나 이미지 링크를 포함가능
- **담당자 (Assignee)**
- **컴포넌트** : 마이크로 서비스에서 어느 컴포넌트인지 또는 서버인지, 안드로이드인지 와 같은 구현 컴포넌트를 지정한다.
- **스프린트** : 어떤 스프린트에서 반영할것 인지
- **스토리 포인트** : 해당 이슈를 처리하는데 걸리는 예상 시간

The screenshot displays the 'Project settings / Issue types' configuration page for a project named 'Terry-Scrum'. The 'Task' issue type is selected, and its 'Description fields' are being configured. A red box highlights the 'Component' field configuration, which includes a 'Field Name' (Component), a 'Display description' (Describe how people should use this field), and a list of 'Options' (IOS, Android, Server, Option 4). To the right, a 'Fields' panel shows various field types like Short text, Paragraph, Number, Time stamp, Dropdown, Checkbox, Dependent dropdown, and URL. The bottom of the page shows 'Suggested fields' including 'Due date' and 'Start date'.

5. 사용자 스토리 등록

이슈 타입과 필드에 대한 작업후, 사용자 스토리 등록

- 사용자 스토리는 **백로그에 등록됨**
- 사용자 스토리 등록시 **포인트 포커를 통해서** 스토리 포인트 추가 권장

Projects / DEMO / DEMO-2 / DEMO-9

As a user, I want to upload the video

Attach Add a child issue Link issue ...

Description

Users upload their videos as soon as they are finished shooting or select videos already shot from the video gallery. Users upload videos by adding a title and description. Location information is also uploaded when uploading.

Uploading a location using GPS information or, if there is a rock climbing gym already registered nearby, a list of rock climbing gym names is displayed and the user can select them.

Child issues

Order by ... + Suggest subtasks 0% Done

DEMO-10	video upload API	1	TO DO
DEMO-11	video upload UI	1	TO DO

Confluence content

Product requirements TRY TEMPLATE

Activity

Show: All Comments History

Summarize Newest first



Add a comment...

Pro tip: press M to comment

1

To Do Actions

Details

Assignee Unassigned Assign to me

Labels None

Parent DEMO-2 Video Upload

Team None

Sprint None

Story point estimate 2

Development NOT INTEGRATED

Branch

Commit

Pull request

Integrate source code

Reporter Terry Cho

Created 12 minutes ago

Updated 11 minutes ago

6. 스프린트 생성

백로그에 이슈 등록후, 스프린트 준비

- 스프린트 생성
- 스프린트 명과 기간 (통상 4주) 정의

☐ Backlog (2 issues)

200

Create sprint

<input checked="" type="checkbox"/> DEMO-8	Video upload API	TO DO ▾	-	
<input checked="" type="checkbox"/> DEMO-9	As a user, I want to upload the video	VIDEO UPLO...	TO DO ▾	2

Edit sprint: DEMO Sprint 1

Required fields are marked with an asterisk *

Sprint name *

DEMO Sprint 1

Duration

4 weeks ▾

Start date

9/16/2024 12:00 AM 🌐

End date

10/14/2024 12:00 AM

Sprint goal

MVP Release

Cancel

Update

6. 스프린트에 스토리 할당

백로그에 이슈 등록후, 스프린트 준비

- 스프린트 생성
- 스프린트 명과 기간 (통상 4주) 정의
- 스토리 포인트의 총합이 자동 계산됨
- 스프린트의 스토리포인트 총합 < 팀의 수용가능한 포인트 (50~70%) : 변경 사항과 일반 업무 부하에 대한 대비
- 스프린트 시작

The screenshot shows the Jira interface with a Sprint and Backlog. The Sprint section is titled 'DEMO Sprint 1' and shows two issues: 'DEMO-9' and 'DEMO-12'. The Backlog section is titled 'Backlog' and shows two issues: 'DEMO-13' and 'DEMO-14'. A red box highlights the 'DEMO Sprint 1' and 'Backlog' sections. A red arrow points from the 'Backlog' section to the 'Sprint' section, labeled 'Drag & drop'. A red box highlights the '2 issues | Estimate: 3' summary in the 'Sprint' section.

스프린트의 스토리 포인트 총합

7. 서브 태스크

스토리 또는 태스크에 서브태스크 등록하여 관리

- 스토리를 완성하기 위한 태스크들
- 예시) 비디오 업로드 스토리
 - 업로드 API
 - 업로드 안드로이드 UI
 - 비디오 인코딩 저장

:

Projects / DEMO / DEMO-2 / DEMO-9

As a user, I want to upload the video

Attach Add a child issue Link issue

Description

Users upload their videos as soon as they are finished shooting or select videos already shot from the video gallery. Users upload videos by adding a title and description. Location information is also uploaded when uploading.

Uploading a location using GPS information or, if there is a rock climbing gym already registered nearby, a list of rock climbing gym names is displayed and the user can select them.

Child issues

Order by ... + Suggest subtasks 0% Done

DEMO-10	video upload API	1	TO DO
DEMO-11	video upload UI	1	TO DO

Confluence content

Product requirements TRY TEMPLATE

Activity

Show: All Comments History

Summarize Newest first



Add a comment...

Pro tip: press M to comment

1

To Do Actions

Details

Assignee	Unassigned Assign to me
Labels	None
Parent	DEMO-2 Video Upload
Team	None
Sprint	None
Story point estimate	2
Development	NOT INTEGRATED Branch Commit Pull request Integrate source code
Reporter	Terry Cho

Created 12 minutes ago
Updated 11 minutes ago

Configure

8. 이슈 업데이트

- 이슈에, 진행상황을 Comment로 업데이트
- 진행 현황을 업데이트
- 리모트 협업시에, 내용을 업데이트해놓고 공유
- 필요한 경우 다른 사람에게 assign
- 또는 다른 이슈를 만들어서 다른 사람에게 assign해놓고 Link
- 대화나 이메일 없이 이슈 노트만 보고 진행 상황을 파악할 수 있도록.

Activity

Show: [All](#) [Comments](#) [History](#)

[✖ Summarize](#)



Add a comment...

Pro tip: press **M** to comment



Terry Cho 10 seconds ago

- Mobile client will send video file with meta data.
- The file needs to be stored in S3.
- After the files is stored, it will trigger the event thru Redis queue to request transcoding

:

Edit · Delete ·

9. 릴리즈 관리



- 1..N 개의 스프린트 후에 제품/서비스 릴리즈
- 릴리즈 버전을 정의할 수 있음
- 릴리즈 시, 이전 릴리즈 부터 현재 릴리즈까지의 이슈 리스트를 릴리즈 노트로 생성 가능

Projects / DEMO
Releases

Unreleased

Create version

Version	Status	Progress	Start date	Release date	Description
V0.1	UNRELEASED	No issues	October 21, 2024	November 18, 2024	
MVP	UNRELEASED	<div></div>	September 16, 2024	November 4, 2024	first release

Release notes - DEMO - MVP

Story

[DEMO-9](#) As a user, I want to upload the video

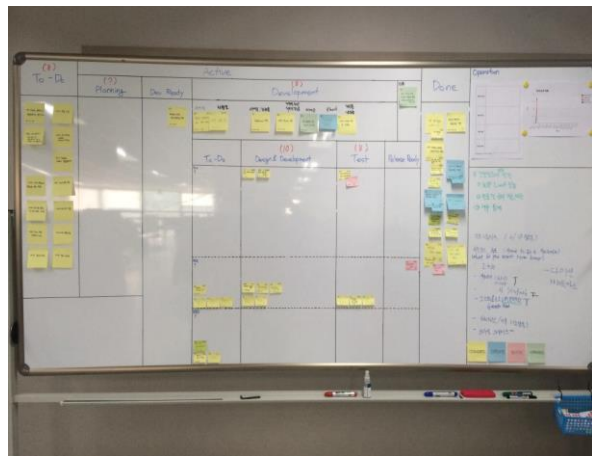
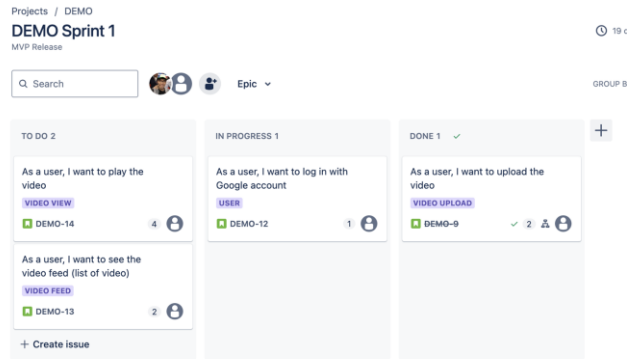
[DEMO-12](#) As a user, I want to log in with Google account

[DEMO-13](#) As a user, I want to see the video feed (list of video)

10. 칸반 보드

칸반 보드

- To do / In Progress / Closed
- 이슈들의 진행상황을 한눈에 확인
- 포스트잇 오프라인 칸반보드 (온라인이 있더라도 병행 운용)
- 팀의 칸반 보드는 지속적으로 발전 (별표, 붉은 마크 등)



11. 일일 스크럼 미팅

15분 정도 진행 상황을 공유하고 문제를 조기에 발견

- 어제 무엇을 했는가?
- 오늘 무엇을 할 것인가?
- 작업을 진행하는데 장애물은 있는가?

팀장에게 보고하는 자리가 아닌 공유하는 자리로
. 가능하면 질문 금지

추가 토론은 스크럼 미팅후 따로 진행



12. 데일리 커밋

매일 작업한 내용을 코드 저장소에 커밋

- 지속적 통합 : 작은 단위로 통합을 통해서 다른 코드와 충돌을 방지
- 문제 조기 발견 : 오류나 코드 충돌을 조기에 감지
- 진행 상황 추적 : 프로젝트의 진전을 보면서, 프로젝트의 진행 상황을 실시간으로 파악

이슈 트래킹 시스템 (JIRA)의 이슈 번호를 넣어서
기능 스펙과 코드간에 추적성 부여

13 . 데모



PO나 이해관계자에게 기능을 시연하는 자리

- 완성된 기능 시연
- 피드백 수집
- 향후 계획 논의

가시성과 피드백이 핵심으로, 팀의 성과를 인증
받기 위함

14. 회고

스프린트가 끝난 후에, 과정과 결과를 돌아보면서 개선할 점을 찾는 시간

- 무엇이 잘되었는가?
- 무엇이 아쉬웠는가?
- 다음 스프린트에서는 무엇을 바꿀 수 있는가?

효과적인 회고를 위해서는

- 자아비판, 마녀 사냥이 되지 않게
- 개선에 대한 목표와 담당자 지정
- KPI 방식 보다는 OKR 방식 (목표를 정해놓고 방식을 고민)

팀 운영 모델

팀모델 - 기능 중심팀

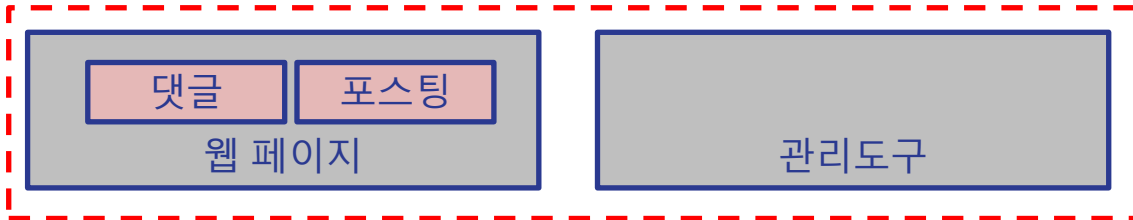
기술 분류에 따른 팀 (Functional Team)

- 프론트,백엔드,앱 개발,운영, 테스트

앱개발팀



프론트 개발팀



백엔드 개발팀

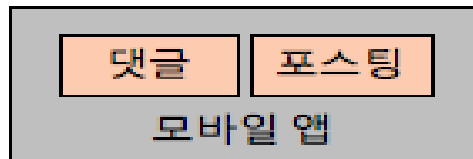


팀모델 - 제품 중심팀

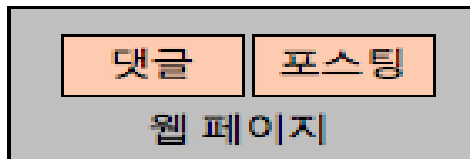
Cross functional team (제품 중심)

- 한팀에 프론트,백엔드,앱 운영 QA를 모두 넣음
- MSA에 많이 사용

앱개발팀



웹 개발팀

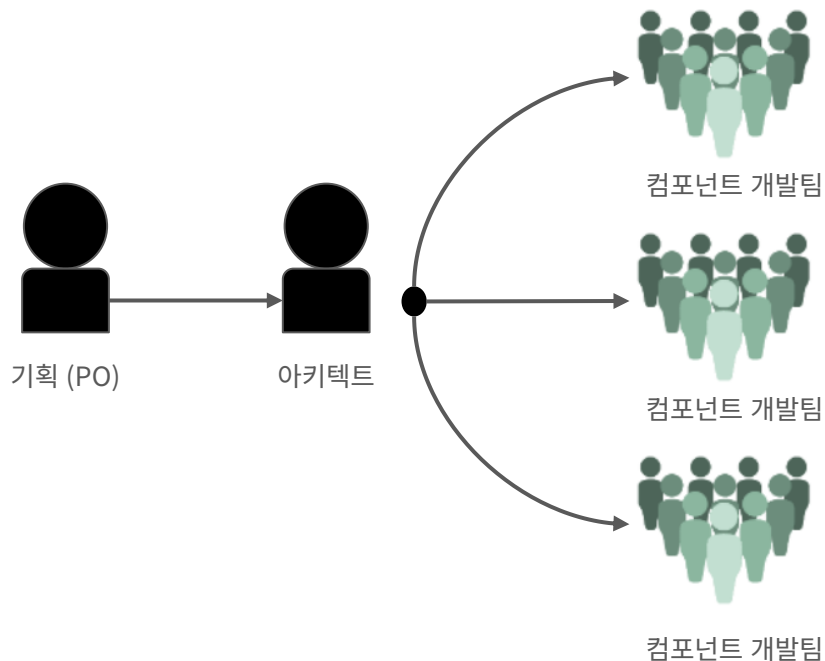


관리도구 개발팀



백엔드

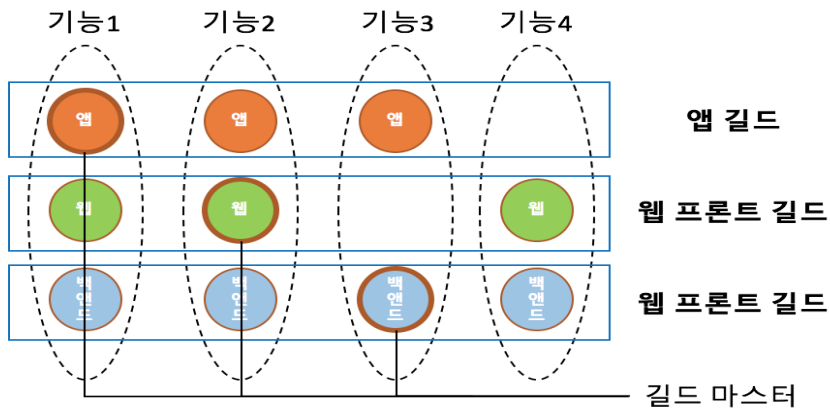
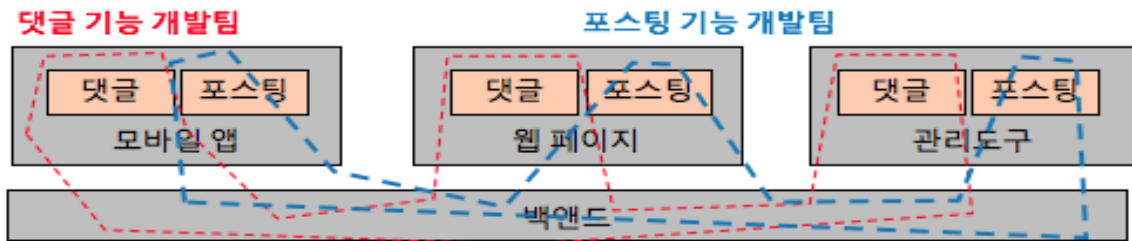
팀 모델 - 제품 중심팀



- 기능이 여러 제품에 걸쳐 있을때, 제품팀간의 조율이 필요함 예) 비디오 업로드 : 웹,안드로이드,IOS,관리자
- 중앙 통제 조직이 필요함 (기획, 프로젝트 관리자, 수석 아키텍트)
- 커뮤니케이션 비용으로, 전체 개발 속도가 늦어질 수 있음

팀 모델 - 기능 중심

Cross functional team (기능 중심)



- 스포티파이가 사용한 모델
- 기능별로 팀을 분류
- 하나의 코드베이스를 공유할 수 있기 때문에 플랫폼화 필수
- 기능에 따른 기획, 개발, 배포이기 때문에 팀간의 별도의 커뮤니케이션이 없어지고, 독립적인 의사 결정 및 빠른 개발 가능

감사합니다.