RELEASY

ChatGPT를 이용한   
사내 릴리즈 노트 공유 시스템 “Releasy”



|  |  |  |
| --- | --- | --- |
| **TEAM KAKAO99%** | | |
| **학번** | **이름** | **학과** |
| 201835453 | 박도영 | AI·소프트웨어학부(소프트웨어전공) |
| 201734919 | 박재석 | 산업경영공학과 |
| 201735979 | 서강덕 | 컴퓨터공학과 |
| 201835465 | 서지원 | AI·소프트웨어학부(소프트웨어전공) |
| 201835474 | 안해빈 | AI·소프트웨어학부(소프트웨어전공) |



**목차**

1. 보고서 기록 추적
2. 서론
   1. 서비스 개요
   2. 개발 필요성
3. 사전 조사 및 기존 제품 분석
4. WBS 및 스프린트
5. 요구사항 정의서
6. 기술 스택
7. 코드 컨벤션
8. 와이어프레임 및 스토리보드
9. 서비스 실행 화면
10. 개체-관계 다이어그램(ERD)
11. 데이터 흐름도(DFD)
12. 아키텍처 구조도
13. API 및 인터페이스 명세서
14. **보고서 기록 추적**

|  |  |  |
| --- | --- | --- |
| **버전** | **날짜** | **내용** |
| 0.0.1. | 2023. 7. 3. | 보고서 작성 시작 |
| 0.1.0. | 2023. 7. 27. | 프로토타입 보고서 작성 완료 |

**1. 서론**

**1) 서비스 개요**

“Releasy”는 릴리즈 노트 공유를 쉽고 빠르게 작성하고 팀원들과 공유할 수 있으며 프로젝트 중 발생하는 이슈에 대해서 빠르게 대처할 수 있는 사내 릴리즈 노트 공유 웹 서비스이다. 사내 릴리즈 노트 공유 시스템인 만큼 그룹을 생성하고 각 그룹 내에서 프로젝트를 진행할 수 있으며, 프로젝트마다 그룹 내에서 참여하는 멤버가 상이할 수 있다. “Releasy”는 ‘ChatGPT를 이용한 이슈 별 중요도 및 우선순위 추천 서비스’와 ‘드래그 앤 드롭(Drag and Drop)이 가능한 칸반 보드(Kanban Board)를 통한 이슈 관리 기능’을 강점으로 내세운다.

‘ChatGPT를 이용한 이슈 별 중요도 및 우선순위 추천 서비스’는 현재 프로젝트 중 발생한 이슈들의 목록에서 PM이 ChatGPT를 통해 중요도와 우선순위를 추천받고, 이를 적용함으로써 프로젝트의 진행 순서를 명확하고 빠르게 결정하여 이슈와 업무를 분배할 수 있도록 돕는 서비스이다. PM은 직접 중요도를 지정할 수도 있지만, 본 서비스를 이용함으로써 보다 객관적이고 신속한 의사결정이 가능할 것으로 기대된다. 결정된 중요도는 프로젝트 대시보드와 이슈 페이지에 표시되며, 수치와 색상으로 구분한다.

‘드래그 앤 드롭(Drag and Drop)이 가능한 칸반 보드(Kanban Board)를 통한 이슈 관리 기능’은 프로젝트 중 생성된 이슈 또는 업무의 단계와 상태를 시각적으로 명확히 파악할 수 있도록 돕는다. 칸반(Kanban)은 반복적인 프로세스의 단계를 명확히 나누는 프로젝트 관리 방식으로, 발행한 이슈와 업무 목록을 각각 Backlog(진행 전), In Progress(진행 중), Done(완료)의 3가지로 구분하고 각 항목을 드래그 앤 드롭으로 이동시켜 상태를 변경할 수 있으며 이전에서 언급한 중요도와 함께 정보가 함께 표시된다.

또한, 본 목적은 사내 릴리즈 노트의 공유이기 때문에 릴리즈 노트의 작성 및 수정을 직관적이고 간단하게 나타내었으며, 해당 릴리즈 노트와 관련된 이슈들을 직접 추가 및 삭제할 수 있도록 하였다. 실제 해당 버전의 릴리즈 전에 상태를 ‘릴리즈 전’으로 지정해 두고 미리 릴리즈 노트를 작성하여 저장해 둘 수 있으며, 쉽게 상태 및 종속된 이슈, 내용을 변경할 수 있다,

**2) 개발 필요성**

프로젝트 매니저(이하 PM)은 프로젝트 진행 중에 모든 일정과 진행 상황을 통제 및 조율해야 한다. PM이 신경 써야 할 부분이 굉장히 많기 때문에 조금이라도 부담을 덜기 위해서 “Releasy”는 이슈의 우선순위를 결정과 팀원에게 할당 시에 ChatGPT를 통한 추천을 받을 수 있도록 기획하였다. 또한, 이러한 진행 상황을 팀 구성원 모두가 쉽게 파악할 수 있어야 하기 때문에 칸반(Kanban) 보드를 통해 쉽게 파악할 수 있도록 했으며, 드래그 앤 드롭 기능을 통해 자신이 배정받은 이슈의 현재 상태를 쉽고 빠르게 변경하고 팀원들에게 공유할 수 있도록 기획하였다.**2. 사전 조사 및 기존 제품 분석**

릴리즈 노트(release note)는 소프트웨어 또는 서비스가 출시 또는 업데이트 될 때마다 소프트웨어 제품과 함께 배포되는 문서들을 말하며, 제품이 개발 중이거나 테스트 상태(베타 릴리스 등)일 때 추가되기도 한다. 고객이 이미 사용 중인 제품의 경우 릴리스 노트는 업데이트가 출시될 때 고객에게 전달된다. 릴리즈 노트는 종이 매뉴얼, 모바일/웹 애플리케이션, 개발자 사이트 등 다양한 소스에 게시될 수 있다.

* 릴리즈 노트를 통해 소프트웨어의 버전과 릴리즈 정보를 체계적으로 관리 할 수 있다.
* 소프트웨어의 전체 기능, 서비스의 내용, 개선 사항, 버전 정보 등을 사용자와 공유하는 것이다.
* 업데이트 정보를 한데 모아 기록함으로써, 내가 작업하지 않은 업데이트 사항도 빠르게 파악 가능하다.
* 릴리즈 노트를 보면 누구나 제품의 발자취를 한눈에 파악 가능하다.
* 우리 회사의 제품 혹은 서비스가 지속적으로 개선되고 최신의 상태를 유지하고 있음을 안내한다.
* 사용자의 신고와 의견을 반영하고 있다는 인상을 주어 사용자의 적극적인 피드백을 유도한다.

**1) Notion Projects**

문서, 지식, AI와 통합된 프로젝트 관리로 업무를 빠르고 명확하게 만들 수 있도록 지원한다.

* AI 자동 채우기: 요약본, 액션 아이템, 프로젝트 업데이트 등을 자동으로 생성한다.
* 트래킹이 필요한 작업과 프로젝트의 업데이트 알림을 제공한다.
* 스프린트 플래닝: 베타 버전에서 사전 구성된 스프린트 워크플로를 사용할 수 있다.
* 이슈마다 고유 ID 부여: 어느 툴의 이슈와 작업이든 간편하게 연결하고 커뮤니케이션할 수 있다.
* GitHub, Slack 등 API 통합으로 모든 작업 연결: GitHub PR 상태 변경 시 Notion 작업을 자동 업데이트 하는 등의 여러 작업을 지원한다.
* Figma와 Google Drive 파일을 작업에 연결할 수 있도록 지원할 예정이다.
* 변경 사항이 있을 때 수행할 작업을 미리 지정할 수 있도록 할 예정이다.

**2) Github Release**

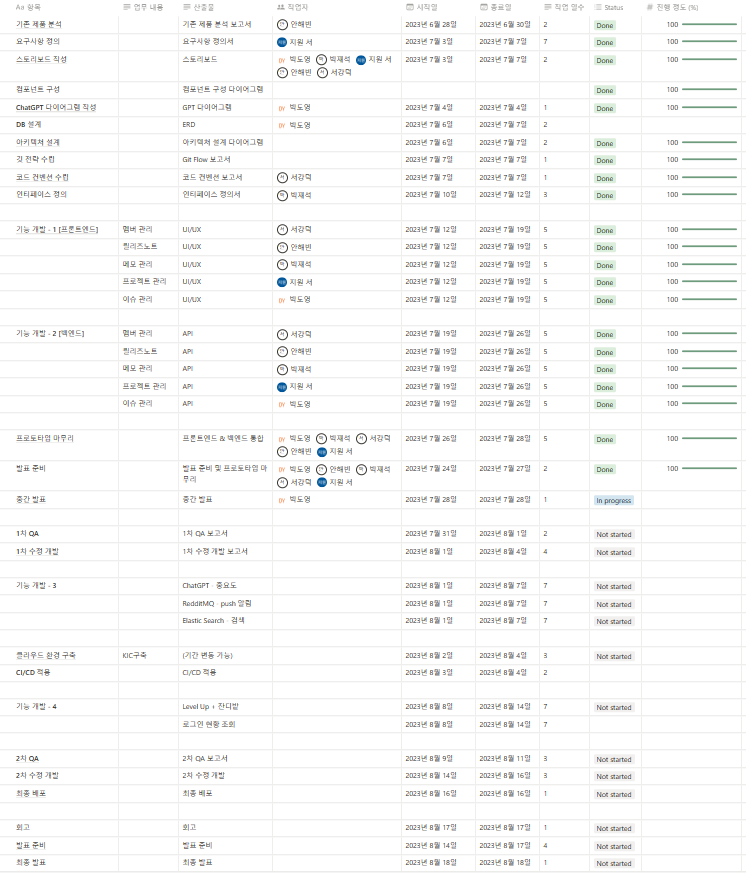
GitHub에서는 소프트웨어의 소스코드 뿐만 아니라 소프트웨어를 빌드한 결과물을 배포할 수 있는 기능까지 제공하고 있다. ‘릴리즈(Release)’ 라는 기능이다.

* Master branch의 태그와 함께 릴리즈 제목, 릴리즈 노트 및 소스코드를 빌드한 바이너리를 첨부할 수 있다.
* 릴리즈 정보 자동 생성: Repository의 코드와 파일들을 바탕으로 릴리즈 정보를 수동으로 작성하는 작업을 자동화하는 기능을 제공한다.
* GitHub-release: GitHub에 공개되어 있는 ‘aktau/github-release’는 CLI 환경을 이용해 깃허브 릴리즈를 쉽게 할 수 있도록 해준다. CLI 환경에서 사용 가능하므로 자동화된 스크립트를 만들기도 좋다. ‘github-release’ 명령을 통해서 깃허브 릴리즈 페이지에 파일을 업로드하고, 릴리즈 노트를 수정할 수도 있다.

**3) Slack**

Slack은 팀과 업무를 하나로 이어주는 단일 플랫폼이다. 클라우드 기반의 팀 협업 도구로, 실시간 메시지, 파일 공유, 작업 관리 등 많은 기능들을 제공하여 사람들이 한 곳에서 통합된 하나의 팀을 운영하여 조직이 커뮤니케이션하는 방법을 개선할 수 있다. 업무에 관련된 모든 커뮤니케이션이 한 곳으로 집중되어 빠른 업무 확인 및 의사 결정이 가능하여 이메일을 대체할 수 있는 커뮤니케이션 도구이다. 다음과 같은 장점들을 가지고 있다.

* 실시간 커뮤니케이션: 채팅 기능을 통해 실시간으로 팀원들과 대화할 수 있다.
* 팀 협업과 프로젝트 관리: 채널을 생성하여 특정 팀, 프로젝트 또는 주제 별로 그룹을 형성할 수 있다. 그룹 내에서 여러 팀원들이 동시에 파일 공유와 협업 기능을 활용하여 문서 공유, 편집, 리뷰 등을 효율적으로 수행할 수 있다.
* 이력 및 검색 기능: 과거의 대화 내용을 쉽게 찾아보고, 필요한 정보다 파일을 빠르게 검색할 수 있다.
* 화상회의 기능, 허들(Huddle): 채널의 인원들과 음성 또는 화상으로 즉시 연결할 수 있다. 화상통화 중 공유한 모든 링크, 문서, 메시지는 자동으로 저장되기 때문에 허들이 종료되어도 쉽게 다시 참조할 수 있다.
* 워크플로: 코드 없이 자동화된 작업을 만들게 도와준다. 누군가 채널에 들어왔을 때나, 채널에 속한 누군가가 특정 반응을 추가했을 때 등에 설정해 놓은 단계별 프로세스가 자동으로 실행되는 기능이다. 노 코드(No-Code) 방식이기 때문에 누구나 쉽게 워크플로를 생성하고 사용할 수 있다..
* 다양한 앱과 통합: 2600개 이상의 앱과의 통합을 지원하여 업무 생산성을 높일 수 있다.

**3. WBS 및 스프린트 (7/27 기준)**

**4. 요구사항 정의서**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| REQ-ID | 페이지 | 서비스 구분 | 사용자 구분 | 요구사항 | 내용 | 구분 |
| REQ-1 | 모든 페이지 | 성능 | 모든 사용자 | 시스템 사용자 수  처리 | 시스템은 성능 저하 없이 필요한 수의 사용자를 처리할 수 있어야 한다. | 비기능 |
| REQ-2 | 모든 페이지 | 보안 | 모든 사용자 | 시스템 보안 | 시스템은 무단 액세스로부터 안전해야 한다. | 비기능 |
| REQ-3 | 모든 페이지 | 성능 | 모든 사용자 | 페이지 응답 시간 | 웹사이트의 응답시간은 3초 이내여야 한다. | 비기능 |
| REQ-4 | 모든 페이지 | 접근성 | 모든 사용자 | 사용 친화적 화면 | 웹사이트의 모든 화면은 명확하고 사용 친화적이며 누구나 기능을 이해하고 사용할 수 있어야 한다. | 비기능 |
| REQ-5 | 홈 | 메인 랜딩 | 모든 사용자 | 홈페이지 제공 | 사진, 설명, 로그인 버튼, 회원가입 버튼이 정상적으로 노출되어야 한다. | 기능 |
| REQ-6 | 로그인 | 로그인/보안 | 모든 사용자 | 소셜 로그인 지원 | 카카오, 구글, 네이버의 소셜 미디어 계정으로 로그인 할 수 있어야 한다. | 기능 |
| REQ-7 | 로그인 | 로그인/보안 | 모든 사용자 | 이메일 로그인 지원 | 이메일과 비밀번호를 이용하여 로그인할 수 있어야 한다. | 기능 |
| REQ-8 | 로그인 | 로그인/보안 | 모든 사용자 | 로딩 속도 | 로그인 시 3초 이내에 완료되어야 한다. | 비기능 |
| REQ-9 | 로그인 | 로그인/보안 | 모든 사용자 | 입력 횟수 | 비밀번호를 5번 이상 틀린 경우 일정 시간 동안 로그인 시도를 제한해야 한다. | 기능 |
| REQ-10 | 로그인 | 로그인/검증 | 모든 사용자 | 비밀번호 검사 | 사용자가 입력한 비밀번호와 암호화 된 비밀번호가 일치하는지 확인해야 한다. | 기능 |
| REQ-11 | 회원가입 | 회원가입 | 모든 사용자 | 유효성 검사 | 회원가입 양식에 유효성 검사가 적용되어야 한다. | 기능 |
| REQ-12 | 회원가입 | 회원가입 | 모든 사용자 | 중복 방지 | 동일한 email 정보로 가입하는 걸 방지해야 한다. | 기능 |
| REQ-13 | 회원가입 | 비밀번호 | 모든 사용자 | 암호화 | 회원 가입 시 입력하는 비밀번호는 암호화 하여 안전하게 보관해야 한다. | 비기능 |
| REQ-14 | 회원가입 | 비밀번호 | 모든 사용자 | 암호화 | 회원 가입 시 입력하는 비밀번호가 동일해도 암호화 결과는 달라야 한다. | 비기능 |
| REQ-15 | 회원가입 | 그룹 생성 | 모든 사용자 | 유효성 검사 | 생성하고자 하는 그룹명의 중복 여부를 검사해야 한다. | 기능 |
| REQ-16 | 회원가입 | 그룹 참여 | 모든 사용자 | 유효성 검사 | 입력한 그룹 코드가 존재하는지 검사해야 한다. | 기능 |
| REQ-17 | 로그인/회원가입 | 관리자 도구 | 시스템 관리자 | 로그인 설정 및 확인 | 이메일 및 소셜 로그인에 대해 설정하고 정보를 확인할 수 있어야 한다. | 기능 |
| REQ-18 | 로그인/회원가입 | 입력 | 모든 사용자 | 유효성 검사 | 빈 입력 란이 존재하거나, 입력 형식에 맞지 않는 란이 존재하면 양식을 제출할 수 없도록 해야 한다. | 기능 |
| REQ-19 | 로그인/회원가입 | JWT토큰 | 모든 사용자 | JWT 토큰 저장 방식 | JWT토큰은 쿠키로 저장되어 사용된다. | 비기능 |
| REQ-20 | 프로젝트 관리 페이지 | 프로젝트 관리 | 그룹 멤버 | 프로젝트 및 그룹 구분 | 그룹 참여자는 참여 중인 프로젝트와 그룹의 다른 프로젝트를 볼 수 있다. | 기능 |
| REQ-21 | 프로젝트 관리 | 페이지 이동 | 그룹 멤버 | 프로젝트 목록 확인 및 이동 | 그룹 참여자는 참여 중인 프로젝트의 목록을 확인하고 프로젝트 페이지로 이동할 수 있다. | 기능 |
| REQ-22 | 프로젝트 관리 페이지 | 프로젝트 조회 | 그룹 멤버 | 릴리즈 노트 확인 | -그룹 참여자는 그룹의 다른 프로젝트들의 릴리즈 노트를 조회할 수 있다.  -프로젝트에 속해있지 않은 그룹의 프로젝트 대시보드로는 이동할 수 없다. | 기능 |
| REQ-23 | 프로젝트 관리 | 프로젝트 수정 | GM, PM | 프로젝트 수정 및 삭제 기능 | GM 또는 각 프로젝트의 PM은 프로젝트의 이름, 설명, 상태를 수정할 수 있다. | 기능 |
| REQ-24 | 프로젝트 관리 | 프로젝트 삭제 | GM, PM | 프로젝트 삭제 기능 | GM 또는 각 프로젝트의 PM은 프로젝트를 삭제할 수 있다. | 기능 |
| REQ-25 | 프로젝트 생성 | 프로젝트 생성 | 그룹 멤버 | 프로젝트 생성 기능 | 그룹 멤버들은 프로젝트 생성 시 프로젝트 이름, 상태, 설명을 작성해서 프로젝트를 생성할 수 있다. | 기능 |
| REQ-26 | 내 정보 | 정보 | 그룹 멤버 | 정보 조회 기능 | 그룹 멤버는 나에 대한 정보(이름, 핸드폰번호, 이메일, 직책)와 다른 그룹 멤버들의 정보(이름, 직책, 입사일, 이메일)을 확인할 수 있다. | 기능 |
| REQ-27 | 내 정보 | 정보 | 그룹 멤버 | 정보 검색 기능 | 그룹 멤버는 이름을 통해서 검색이 가    능하다. | 기능 |
| REQ-28 | 내 정보 | 그룹 멤버 | GM | 그룹 멤버 삭제 | 그룹 매니저는 현재 내 그룹의 멤버를 삭제할 수 있다. | 기능 |
| REQ-29 | 사이드바 | 프로젝트 페이지 사이드바 | 프로젝트 멤버 | 프로젝트 하위 페이지로의 이동 | 해당 프로젝트의 하위 기능 페이지로 선택하여 이동할 수 있도록 한다. | 기능 |
| REQ-30 | 페이지 설정 | 프로젝트 페이지 설정 | 프로젝트 멤버 | 프로젝트 페이지 UI 수정 | -UI 설정 메뉴에서 색을 눌러서 메뉴 바 색상을 변경할 수 있다. - 메뉴바 타입을 어둡게, 투명하게, 밝게 중에서 선택해서 변경할 수 있다. -메뉴 바 고정을 토글로 설정할 수 있다.  -토글을 활용해서 다크모드로 전환할 수 있다. | 기능 |
| REQ-31 | 대시보드 | 프로젝트 대시보드 | 프로젝트 멤버 | 각 프로젝트 요약 정보 확인 | 해당 프로젝트의 요약된 정보를 그래프와 표로 한 눈에 확인할 수 있어야 한다. (월별 배포한 릴리즈 노트 수, 월별 해결한 작업 수, 현재 처리 중인 이슈 목록, 시간 순 알림) | 기능 |
| REQ-32 | 대시보드 | 시각화(그래프) | 프로젝트 멤버 | 그래프 표현 | 월별 배포한 릴리즈 노트 수, 월별 해결한 작업 수를 그래프로 나타낼 때 대시보드 페이지가 표시되고 2초 안에 그래프 애니메이션이 완료되고 표시되어야 한다. | 비기능 |
| REQ-33 | 대시보드 | 이슈 목록 | 프로젝트 멤버 | 이슈 정렬 | 중요도 순, 기한 순으로 각각 정렬할 수 있어야 하며 나에게 배정된 이슈만 확인할 수도 있어야 한다. | 기능 |
| REQ-34 | 대시보드 | 이슈 목록 | 프로젝트 멤버 | 중요도 | 중요도는 게이지 바 형태로 표시되며, 중요도(정수)에 따라 색을 다르게 해서 구분해야 한다. | 기능 |
| REQ-35 | 대시보드 | 이슈 중요도 지정 | PM | 수동 또는 GPT를 이용한 이슈 중요도 지정 | - PM은 직접 이슈에 대한 각각, 또는 전체의 중요도를 지정할 수 있어야 한다.  - GPT를 이용한 중요도 지정 기능을 이용하면 자동으로 판단해서 중요도를 지정해 주며, 적용 또는 취소 버튼을 눌러 진행할 수 있다. | 기능 |
| REQ-36 | 대시보드 | 알림 | 프로젝트 멤버 | 주요 변경정보 알림 기능 | 자신이 속한 프로젝트의 주요 변경 사항(이슈 생성, 이슈 담당자 지정, 이슈 수정, 릴리즈노트 생성 등)이 생길 때 그 정보를 기록해두고 전 프로젝트 멤버가 그 내용을 알림으로 확인할 수 있어야 한다. | 기능 |
| REQ-37 | 멤버정보 | 멤버정보 | 프로젝트 멤버 | 멤버 정보 조회 | 멤버 이름, 역할(직급), 상태(온/오프라인),입사일을 표 형식으로 확인 할 수 있어야 한다. 이름을 클릭할 경우 해당 멤버의 프로필 페이지로 이동할 수 있어야 한다. | 기능 |
| REQ-38 | 멤버정보 | 멤버정보 | 프로젝트 멤버 | 멤버 정보 필터링 및 색검색 | 멤버 직급, 직책으로 필터링 할 수 있어야하며, 이름으로 검색하여 멤버를 조회할 수 있어야 한다. | 기능 |
| REQ-39 | 멤버들의 프로젝트 | 멤버들의 프로젝트 | 프로젝트 멤버 | 프로젝트 멤버의 프로젝트 조회 | 화면 좌측에 멤버 목록이 표시 되어야 하며, 멤버를 클릭할 경우 화면 우측에서 해당 멤버의 프로젝트 이름 ,상태(진행중,완료됨,취소됨), 생성일을 조회할 수 있어야 한다. | 기능 |
| REQ-40 | 멤버들의 프로젝트 | 멤버들의 프로젝트 | 프로젝트 멤버 | 프로젝트 멤버 필터링 및 검색 | 멤버 직급, 직책으로 필터링 할 수 있어야하며, 이름으로 검색하여 멤버를 조회하여야 한다. | 기능 |
| REQ-41 | 이슈검색 | 이슈 조회 | 프로젝트 멤버 | 검색 | 검색창에서 이슈 제목으로 이슈를 검색할 수 있어야 한다. | 기능 |
| REQ-42 | 이슈검색 | 이슈 조회 | 프로젝트 멤버 | 필터링 | - 이슈의 상태(백로그, 진행중, 완료),이슈의 타임(bug, story,task) 로 구분해서 이슈를 필터링할 수 있어야 한다.  - 이슈의 담당자로 이슈를 필터링해서 보여줘야 한다. | 기능 |
| REQ-43 | 이슈검색 | 이슈 조회 | 프로젝트 멤버 | 성능 | 빠른 응답 시간과 처리량을 제공할 수 있어야능 한다. 이슈를 신속하게 제출하고 조회 할 수 있어야 한다. | 비기능 |
| REQ-44 | 이슈검색 | 이슈 조회 | 프로젝트 멤버 | 사용자 경험 | - 사용자가 쉽게 이해하고 사용할 수 있도록 직관적이고 편리한 인터페이스를 제공해야 한다. | 비기능 |
| REQ-45 | 이슈검색 | 세부내용 조회 | 프로젝트 멤버 | 이슈 세부 내용 조회 | - 이슈 타입, 이슈 제목, 설명이 보여진다.  - 하부의 활동란에는 댓글과 기록이 보여진다.  - 추가된 댓글은 기록에서 보여진다.  - 이슈의 담당자와 보고자가 보여진다. | 기능 |
| REQ-46 | 이슈검색 | 세부내용 수정 | PM, 이슈 보고자 | 이슈 세부 내용 수정 | - 사진 첨부는 1장까지 가능하다.  - 댓글을 달 수 있다.  - 하위 이슈 추가버튼을 눌러서 하위 이슈를 추가할 수 있다. 추가한 하위 이슈도 모든 이슈 목록에 포함된다.  - 다른 이슈와 연결할 수 있다. 이슈 연결 버튼을 누르면 이전 이슈의 목록이 보여진다. | 기능 |
| REQ-47 | 이슈검색 | 세부내용 수정 | PM, 이슈 보고자 | 이슈 삭제 | 이슈를 삭제하면 이슈 목록에서 사라진다. | 기능 |
| REQ-48 | 이슈관리 | 전체 이슈 목록 | 프로젝트 멤버 | 전체 이슈 조회 | - 프로젝트 내의 모든 이슈가 상태(백로그/진행중/완료)에 따라 구분되어 보여진다.  - 이슈 제목으로 검색할 수 있다.  - 상태, 담당자로 필터링해서 볼 수 있다. | 기능 |
| REQ-49 | 이슈관리 | 이슈 칸반 보드 | 프로젝트 멤버 | 이슈 드롭 다운 | -프로젝트의 이슈들은 드롭다운 기능을 활용해서 상태(백로그/진행중/완료)를 변경할 수 있다.  -칸반보드에서 이슈상태를 변경할 때마다 DB에 바뀐 결과가 적용된다.  - 이슈관리 페이지는 2초마다 칸반보드를 업데이트해서 결과가 바뀌면 적용되게 한다. | 기능 |
| REQ-50 | 이슈관리 | 이슈 생성 | 프로젝트 멤버 | 이슈 생성 | - 새로운 이슈를 생성할 수 있다.  - 이슈 생성 버튼을 누르면 생성을 위한 모달창이 보여진다.  - 사진 첨부는 1장만 가능하다.  - 하위 이슈를 추가할 수 있다.  - 설명과 댓글, 활동 표시를 할 수 있다.  - 상태, 담당자, 분류, 수정 버전, 보고자를 지정할 수 있다.  - 시간은 현재 시간으로 저장된다. | 기능 |
| REQ-51 | 이슈 관리 | 세부내용 수정 | PM, 이슈 보고자 | 첨부파일 용량 | 첨부파일의 크기는 1MB보다 작아야한다. | 비기능 |
| REQ-52 | 댓글검색 | 댓글 조회 | 프로젝트 멤버 | 댓글 조회 | -특정 이슈에 작성된 모든 댓글을 조회할 수 있어야 한다.  -댓글이 많은 경우, 페이지네이션을 지원하여 사용자가 원하는 범위의 댓글을 확인할 수 있어야 한다.  - 특정 기준(작성자, 날짜) 에 따라 댓글을 필터링 할링 수 있어야 한다.능 | 기능 |
| REQ-53 | 댓글검색 | 댓글 조회 | 프로젝트 멤버 | 신속한 응답시간 | 댓글은 실시간으로 동작하여 사용자들이 빠르게 댓글을 작성하고, 이에 대한 응답을 받을 수 있어야 한다. |  |
| REQ-54 | 댓글생성 | 댓글 생성 | 프로젝트 멤버 | 댓글 생성 | 사용자가 댓글을 작성할 수 있어야 한다. | 기능 |
| REQ-55 | 댓글 수정 | 댓글 수정 | 프로젝트 멤버 | 댓글 수정 | -사용자는 자신이 작성한 댓글을 수정할 수 있어야 한다. 수정된 댓글은 즉시 반영되어야 한다.  - 댓글 수정이 완료된 경우, 수정 이력이 표시되어야 한다.  - 댓글 수정에 대하여 해당 댓글 작성자에 한하여 가능하게 해야한다.  댓글 수정 버튼은 댓글 작성자와 관리자에 한하여 볼 수 있어야 한다. | 기능 |
| REQ-56 | 댓글 삭제 | 댓글 삭제 | 프로젝트 멤버 | 댓글 삭제 | 사용자는 자신이 작성한 댓글을 작성 할 수 있어야 한다 삭제된 댓글을 즉시 사라져야 한다.  댓글 삭제에 대하여  댓글 작성자와 관리자에 한하여 가능하게 해야 한다.  댓글 삭제 버튼은 댓글 작성자와 관리자에 한하여 볼 수 있어야 한다. | 기능 |
| REQ-57 | 프로필 | 내 정보 관리 | 모든 사용자 | 내 정보 조회 | 이름, 팀, 전화번호, 이메일, 직책과 내가 참여 했던 프로젝트 목록을 확인할 수 있어야 한다. | 기능 |
| REQ-58 | 프로필 | 내 정보 관리 | 모든 사용자 | 내 정보 수정 | 사용자의 이름, 팀, 전화번호, 이메일, 직책을 수정할 수 있어야 한다. | 기능 |
| REQ-59 | 프로필 | 내 프로젝트 목록 | 모든 사용자 | 프로젝트 목록 조회 | 사용자가 참여 했던 프로젝트  목록을 조회할 수 있어야 한다. | 기능 |
| REQ-60 | 프로필 | 내 정보 관리 | 모든 사용자 | 내 정보 접근 | 권한이 없는 사용자가 정보를 수정, 삭제할 경우 접근 차단 해야 한다. | 비기능 |
| REQ-61 | 프로필 | 내 프로젝트 목록 | 모든 사용자 | 프로젝트 상세 보기 | 프로젝트 목록중에 참여한 프로젝트의 릴리즈 노트를 확인할 수 있어야한다. | 기능 |
| REQ-62 | 릴리즈 노트 | 릴리즈 노트  목록 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | 버전, 상태, 진행률, 생성일자, 릴리즈 일자, 해당 릴리즈노트 설명이 표 형식으로 노출 되어야한다. | 기능 |
| REQ-63 | 릴리즈 노트 | 릴리즈 노트 목록 조회 | 프로젝트 멤버 | 릴리즈 노트 상태, 진행률 변경 | 릴리즈 노트의 상태(릴리즈됨, 릴리즈 안됨)가 표시되어야 하며, 작성자와 작성 일자, 버전이 노출 되어야 한다. | 기능 |
| REQ-64 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | 릴리즈 노트에 대한 설명, 관련 작업 첨부, 관련 이슈, 릴리즈 일자, 담당자, 진행률을 조회 할 수 있어야 한다. | 기능 |
| REQ-65 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | 페이지 로드 시에 해당 릴리즈 노트의 정보를 HTTP GET으로 가져와서 올바르게 표시해야 한다. | 기능 |
| REQ-66 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | 페이지 로드 시에 해당 릴리즈 노트에 등록되어 있는 이슈 목록을 HTTP GET으로 가져와서 올바르게 표시해야 한다. | 기능 |
| REQ-67 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | 표시해야 하는 모든 정보를 HTTP GET으로 가져와서 렌더링 할 정보와 함께 로딩이 완료되는 데에 3초를 초과하지 않아야 한다. | 비기능 |
| REQ-68 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 조회 | - 진행률을 나타내는 Input 칸에는 0부터 100까지의 수만 입력될 수 있으며, 0 이하의 수가 입력되면 0이, 100 이상이 입력되면 100이, 숫자가 아닌 문자가 입력되면 무시해야 한다.  - 진행률을 나타내는 Input 칸에 입력되는 수는 입력과 동시에 칸 하단의 진행 바에 실시간으로 반영되어야 하며, 진행률에 따라서 색깔이 변경되어야 한다. | 기능 |
| REQ-69 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 이슈 조회 | - 릴리즈 노트에 관련되어 있는 이슈 목록이 표시되며, 담당자 필터와 상태(Backlog, In Progress, Done) 필터를 적용시켜 간편하게 조회할 수 있다.  - 필터 초기화 버튼을 통해 필터를 초기화하고 릴리즈 노트에 관련되어 있는 모든 이슈를 확인할 수 있다. | 기능 |
| REQ-70 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 이슈 추가/삭제 | - 이슈 표에서 추가하기 버튼을 눌러 해당 릴리즈 노트에 이슈를 추가할 수 있으며, 추가 후에는 추가하기 모달 창에서는 해당 이슈가 보이지 않고 관련된 이슈 창에서는 추가되어야 한다.  - 이슈 표에서 이슈 클릭 후 제거하기 버튼을 누르면 해당 릴리즈 노트에서 이슈가 제거되며, 관련된 이슈 창에서 해당 이슈가 나타나지 않아야 한다. 이슈 추가하기 모달 창에서 해당 이슈를 다시 확인하고 추가할 수 있어야 한다. | 기능 |
| REQ-71 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 이슈 추가/삭제 | - 릴리즈 노트 생성/수정 페이지에서 이슈가 추가되면 그 즉시 이슈 표에서 확인할 수 있어야 하고, 이슈 추가하기 모달 창에서는 사라져야 한다.  - 릴리즈 노트 생성/수정 페이지에서 이슈가 삭제되면 그 즉시 이슈 표에서 사라져야 하고, 이슈 추가하기 모달 창에서 나타나야 한다. | 비기능 |
| REQ-72 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 수정 | 릴리즈 노트 수정 후 ‘수정’ 버튼을 클릭하면 릴리즈 노트의 정보들과 관련된 이슈 목록을 HTTP PUT 요청을 전달한 후 릴리즈 노트 목록 페이지로 이동한다. | 기능 |
| REQ-73 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 수정 | ‘수정’ 버튼을 눌러 HTTP PUT 요청을 제출하고 릴리즈 노트 목록 페이지로 이동 완료하는 데에 3초를 초과하지 않아야 한다. | 비기능 |
| REQ-74 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 생성 | 내용 입력 후 ‘저장’ 버튼을 누르면 HTTP POST 요청을 통해 정보를 업로드 하고 릴리즈 노트 목록 페이지로 이동한다. | 기능 |
| REQ-75 | 릴리즈 노트 | 릴리즈 노트 내용 조회 | 프로젝트 멤버 | 릴리즈 노트 생성 | ‘저장’ 버튼을 눌러 HTTP PUT 요청을 제출하고 릴리즈 노트 목록 페이지로 이동 완료하는 데에 3초를 초과하지 않아야 한다. | 비기능 |
| REQ-76 | 릴리즈 노트 | 릴리즈 노트 내에서 관련 이슈 조회 | 모든 사용자 | 릴리즈 노트 내에서 관련 이슈 조회 | 릴리즈 노트 조회시 관련 이슈를 클릭하면 , 이슈 관련 세부정보를 조회 할 수 있어야 한다. | 기능 |
| REQ-77 | 릴리즈 노트 | 릴리즈 노트  생성(추가) | PM | 릴리즈 노트  생성 | 릴리즈 노트에 대한 설명, 관련 작업 첨부, 관련 이슈를 생성할 수 있어야 한다. | 기능 |
| REQ-78 | 릴리즈 노트 | 릴리즈 노트 | PM | 릴리즈 노트  수정 | 릴리즈 노트에 대한 설명, 관련 작업 첨부, 관련 이슈를 수정할 수 있어야 한다. | 기능 |
| REQ-79 | 릴리즈 노트 | 릴리즈 노트 | PM | 릴리즈 노트  삭제 | 릴리즈 노트 목록에서 해당 릴리즈 노트를 삭제할 수 있어야 한다. 삭제 요청 시 정말 삭제할 것인지 재확인할 수 있도록 한다. | 기능 |
| REQ-80 | 프로젝트 관리 | 프로젝트 관리 | PM | 프로젝트 정보 조회 | 프로젝트 이름, 설명, 그룹, 상태, 생성일을 조회할 수 있다. | 기능 |
| REQ-81 | 프로젝트 관리 | 프로젝트 관리 | PM | 프로젝트 멤버 관리 | 프로젝트에 참여할 멤버를 추가, 삭제할 수 있다. | 기능 |
| REQ-82 | 프로젝트 관리 | 프로젝트 관리 | PM | 프로젝트 멤버 추가 | 프로젝트 멤버를 추가할 때 그룹 안에 있는 멤버를 검색할 수 있어야 한다. | 기능 |
| REQ-83 | 프로젝트 관리 | 프로젝트 관리 | PM | PM 권한 부여 | 프로젝트 멤버 중 PM 권한을 양도할 수 있다. | 기능 |
| REQ-84 | 프로젝트 관리 | 프로젝트 관리 | PM | 프로젝트 수정 | 프로젝트 이름, 설명, 그룹, 상태, 생성일을 수정할 수 있다. | 기능 |
| REQ-85 | 프로젝트 관리 | 프로젝트 관리 | PM | 프로젝트 멤버 조회 | 프로젝트에 참여하고 있는 멤버의 리스트를 조회할 수 있다. | 기능 |
| REQ-86 | 로그아웃 | 로그아웃 | 그룹 멤버 | 로그아 | - 로그아웃 시 Access\_Token을 삭제 한다.  - 로그인 화면에서 재 로그인 시 다시 서비스를 이용할 수 있다. | 기능 |
| REQ-87 | 웹서버 | NginX |  | 정적 파일 서빙 기능 제공 | Nginx를 사용하여 정적 파일 서빙을 지원해야 한다. | 기능 |
| REQ-88 | 웹서버 | NginX |  | 웹 애플리케이션 서버와 연결 | 웹 애플리케이션 서버(Tomcat)와의 연결을 설정해야 한다. | 기능 |
| REQ-89 | 웹서버 | NginX |  | HTTPS 지원 | SSL/TLS 설정을 포함하여 HTTPS를 지원해야 한다. | 기능 |
| REQ-90 | 웹서버 | NginX |  | 도메인 및 서브도메인 설정 | 다중 도메인 호스팅이 가능하도록 도메인 및 서브도메인 설정을 지원한다. | 기능 |
| REQ-91 | 웹 애플리케이션 서버 | Tomcat |  | Java 웹 애플리케이션 호스팅 | Java 웹 애플리케이션을 호스팅하기 위해 Tomcat을 사용한다. | 기능 |
| REQ-92 | 웹 애플리케이션 서버 | Tomcat |  | 데이터베이스 연동 | JDBC 드라이버 설정으로 데이터베이스와 연동한다. | 기능 |
| REQ-93 | 웹 애플리케이션 서버 | Tomcat |  | 고급 기능 지원 | 세션 관리, 보안 설정, 클러스터링 등 고급 기능이 필요하다. | 기능 |
| REQ-94 | 웹서버, 웹 애플리케이션 서버 | Nginx, Tomcat |  | 프록시 설정 | Nginx를 통해 웹 애플리케이션 서버로 요청을 전달하는 프록시를 설정한다. | 기능 |
| REQ-95 | 데이터베이스 | ElasticSearch, MySQL |  | 데이터베이스 연결 | -MySQL을 활용하여 정형화된 데이터 저장 및 관리한다.  -엘라스틱서치를 활용하여 복잡한 검색 기능을 구현한다. | 기능 |
| REQ-96 | 제품 | 운영체제 |  | 리눅스 사용 | 서버 운영체제는 리눅스 기반 시스템을 사용해야 한다. | 비기능 |
| REQ-97 | 조직 | 구현 방법론 |  | React | React를 기반으로하여 웹페이지를 구현한다. | 기능 |
| REQ-98 | 제품 | 이식성 |  | 브라우저 | 브라우저는 크롬을 기준으로 한다. | 비기능 |
| REQ-99 | 조직 | 프로그래밍 언어 |  | JavaScript Java 사용 | 프로그래밍 언어는 FrontEnd에서는 JavaScript, Backend에서는 Java를 활용한다. | 기능 |

**5. 기술 스택 및 버전**

**1) 개발 환경**

Kakao I Cloud

**2) 개발 도구**

intelliJ (Spring boot)

Visual Studio Code (React)

1. **개발 언어**

Front-end : Javascript, CSS

Back-end : Java (JDK Version : open-jdk-17)

1. **개발 프레임 워크**

Web server : Nginx

Web Application Server : Spring boot

Database : Mysql , Redis

infra : Docker

1. **형상 관리 시스템**

CI/CD : Jenkins

협업 Tool

GitHub

1. **주요 Library**

Front-end: react, mui, bootstrap

Back-end: JsonWebToken

**6. 코드 컨벤션**

**1) Directory Format**

- 도메인 별로 구분

- 각 도메인 별로 controller, service, repository, DTO를 분리해서 사용

**2) Naming**

- 패키지 이름은 소문자 사용



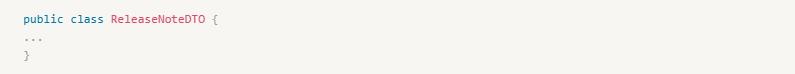
- 클래스/인터페이스는 명사, UpperCamelCase 사용



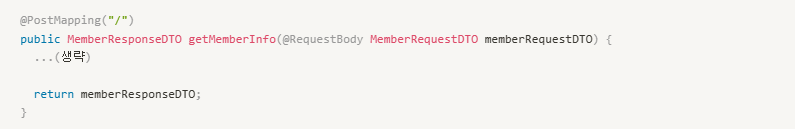
- 상수(Enum)은 CONSTANT\_CASE 사용



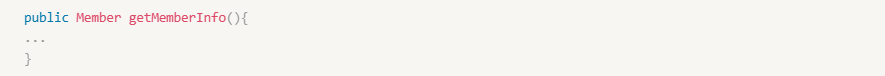
- dto 객체는 이름 뒤에 대문자 DTO 사용



- dto 객체가 Reqeust Body에 사용되면 RequestDTO, Resposne Body에 담아서 응답하면 ResponseDTO로 사용



- 메서드 이름에 동사, lowerCamelCase 사용

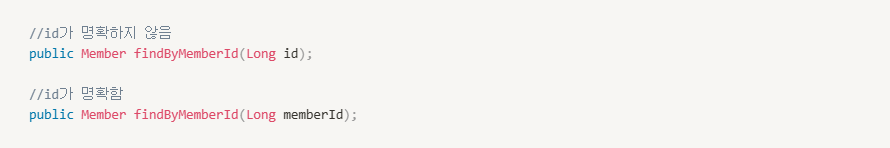


- JPA, queryDSL 메소드 이름을 find이름By이름..로 사용한다.



- JPA, queryDSL을 제외한 다른 클래스의 메소드 이름에서 find로 시작하는 이름을 쓰지 않는다.

- 파라미터 이름을 사용할 때 두글자 이상 사용해야 하며 파라미터의 의미가 명확해야 한다.



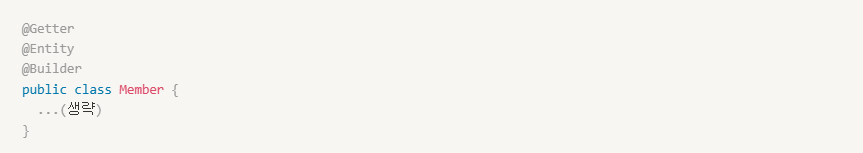
- Test 코드는 클래스명 뒤에 Test를 붙여 사용한다.

**3) Naming**

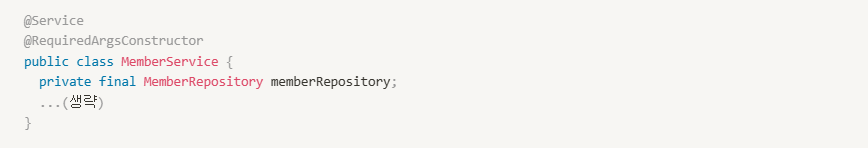
- Entity에 Setter를 사용하지 않는다.



- 클래스 생성자는 Builder pattern을 사용한다.



- DI 주입은 생성자 주입으로 사용하되, @RequiredArgsConstructor 를 사용한다.



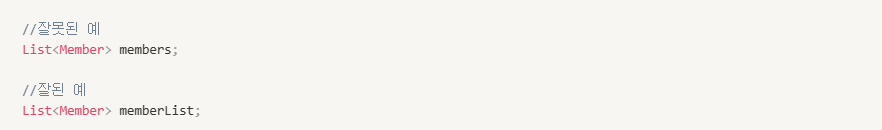
- Controller에 Service를 의존성 주입하고 Service에 Repository를 주입한다.



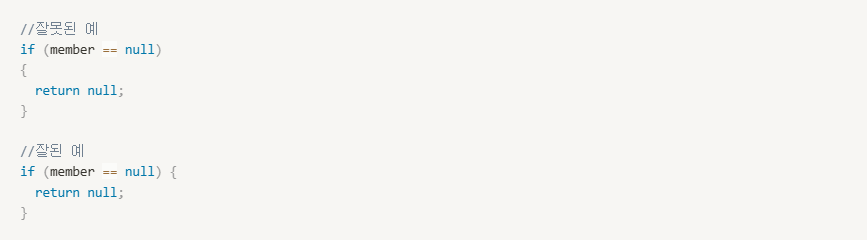
- 예외 처리는 Service에서 진행하고 Service에는 항상 DTO만 반환하며 Controller에서 ResponseEntity를 생성하여 사용한다.



- 컬렉션 인터페이스는 변수명 뒤에 컬렉션 종류를 붙여서 선언한다.



- 클래스 선언, 메서드 선언, 조건/반복문 등의 코드 블럭을 감싸는 중괄호는 K&R 스타일을 사용한다.



- 중괄호 사용시 한줄 띄어서 작성한다.

- 닫는 중괄호와 같은 줄에 else, catch, finally, while 선언한다.

- 조건/반복문에 중괄호 반드시 사용한다.

- 조건/반복문 뒤에 오는 괄호()는 공백 후 사용한다.

- 클래스를 import할 때 와일드카드(\*)를 사용하지 않는다.

- 로그는 System.out.println() 대신 Slf4j를 사용한다.

- 필드 값 주석은 변수 옆에 쓰고, 상세 설명 같은 주석은 바로 위에 작성한다.

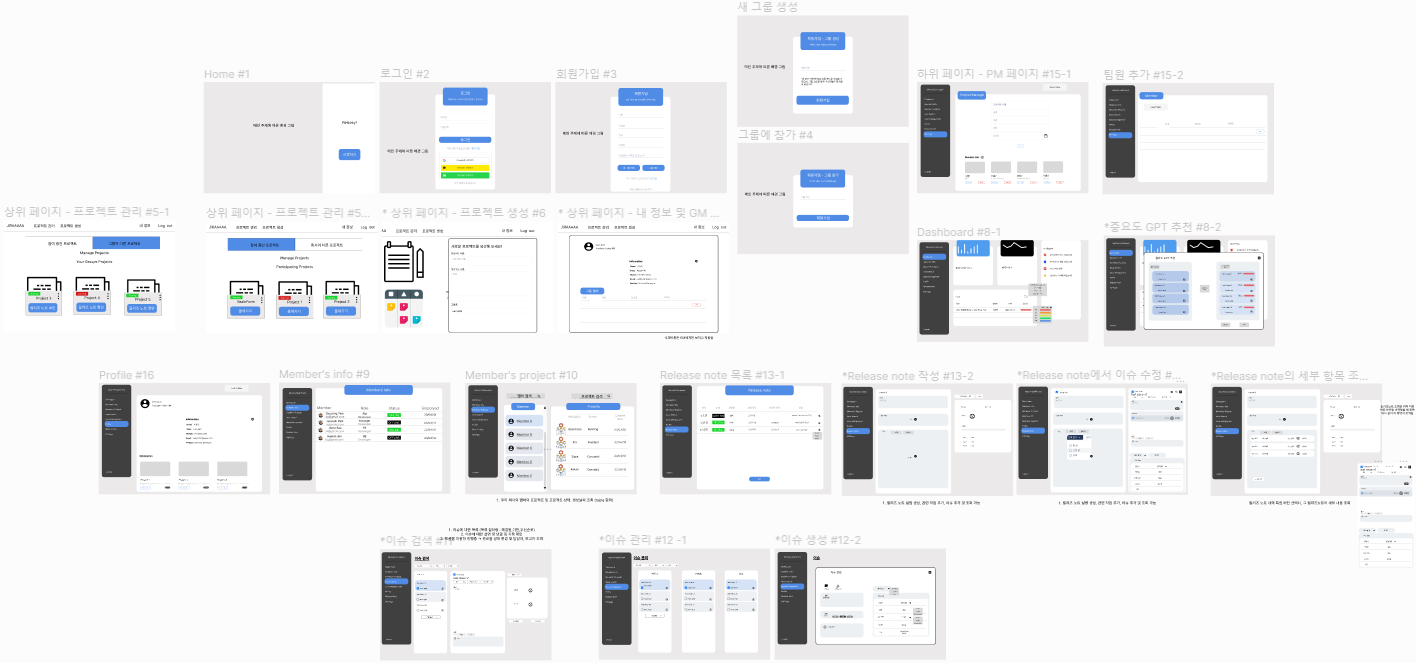
- return 위에 한 줄을 띄어서 작성한다.

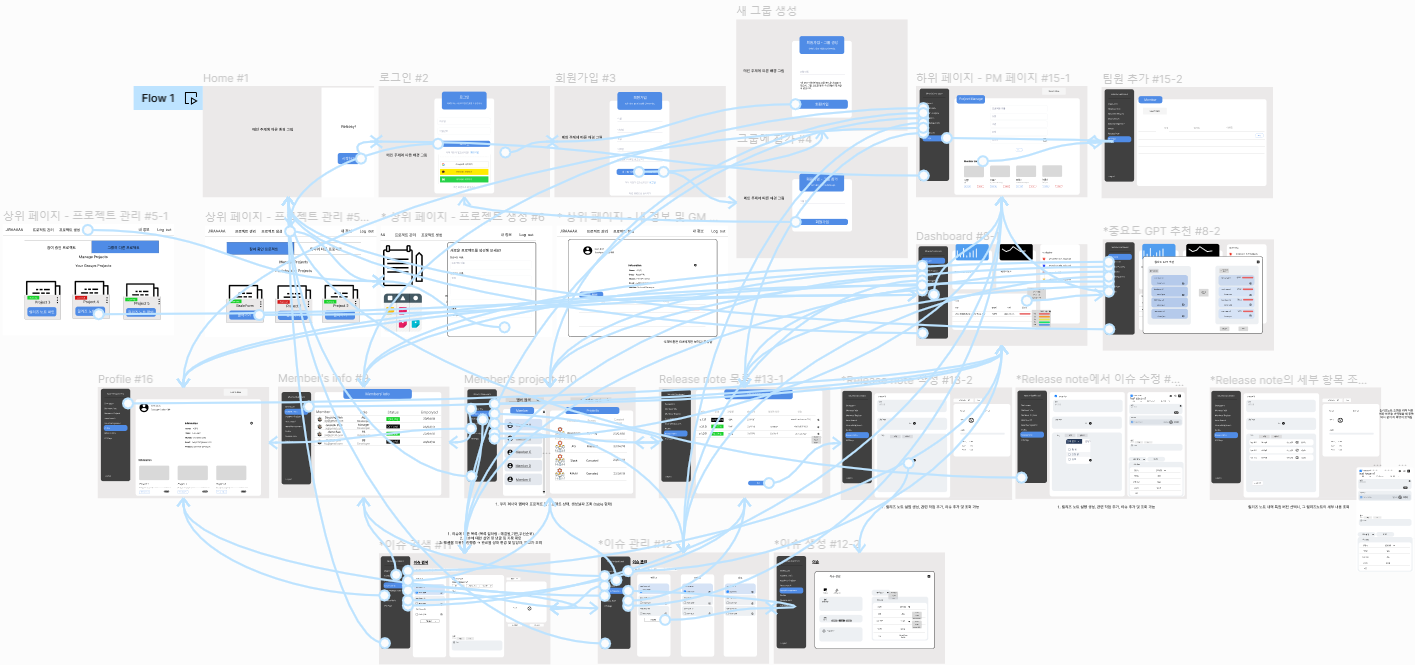
- Test 코드에는 when, given, then 에 맞춰 작성한다.

**4) Gradle**

주석으로 dependency에 대한 설명을 작성한다.

**7. 와이어프레임 및 스토리보드**

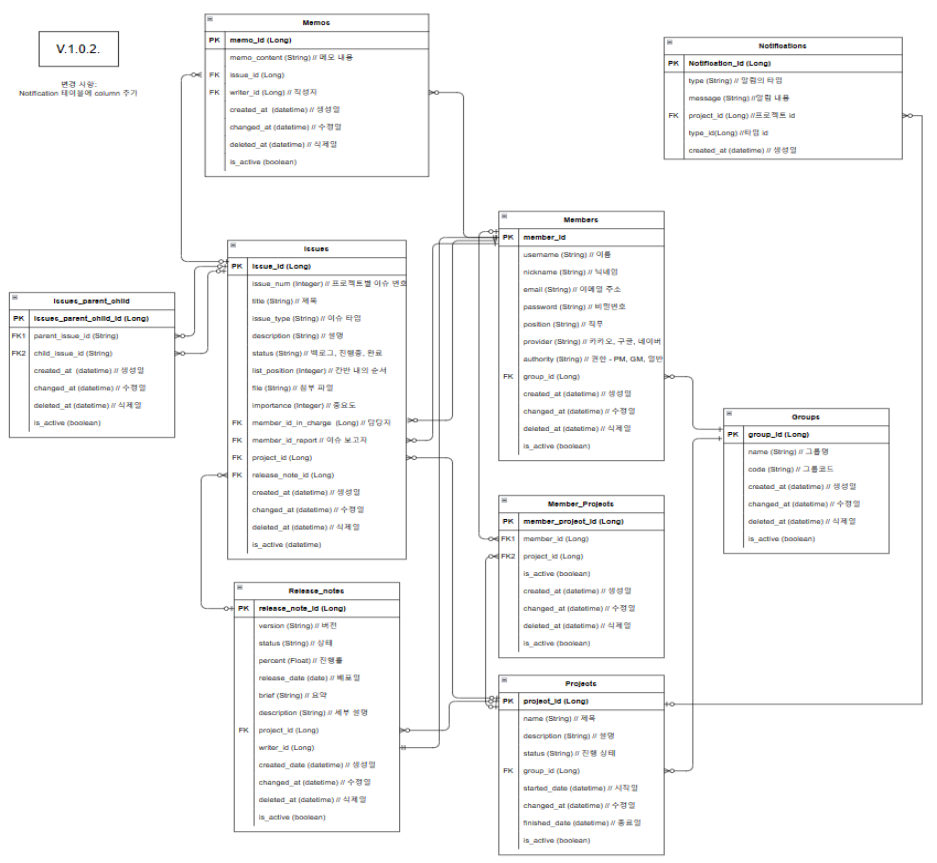
****

****

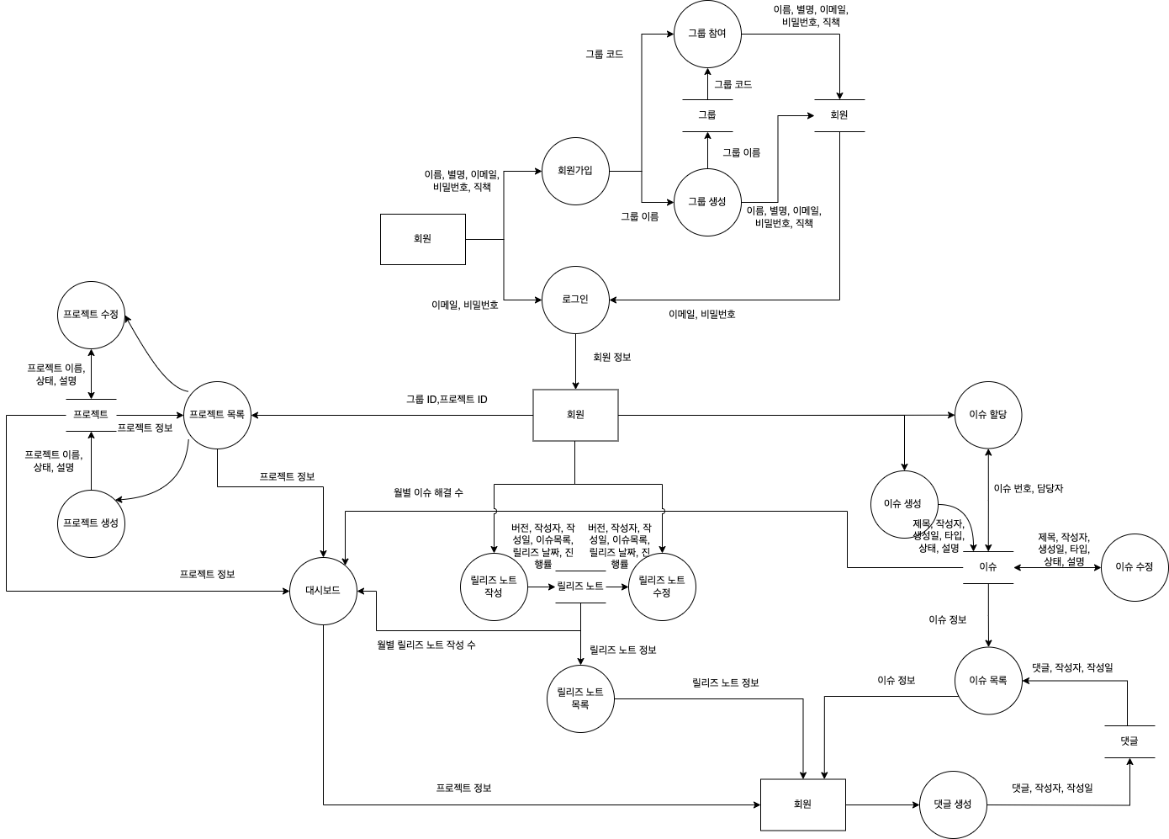
**8. 서비스 실행 화면**

추가 예정

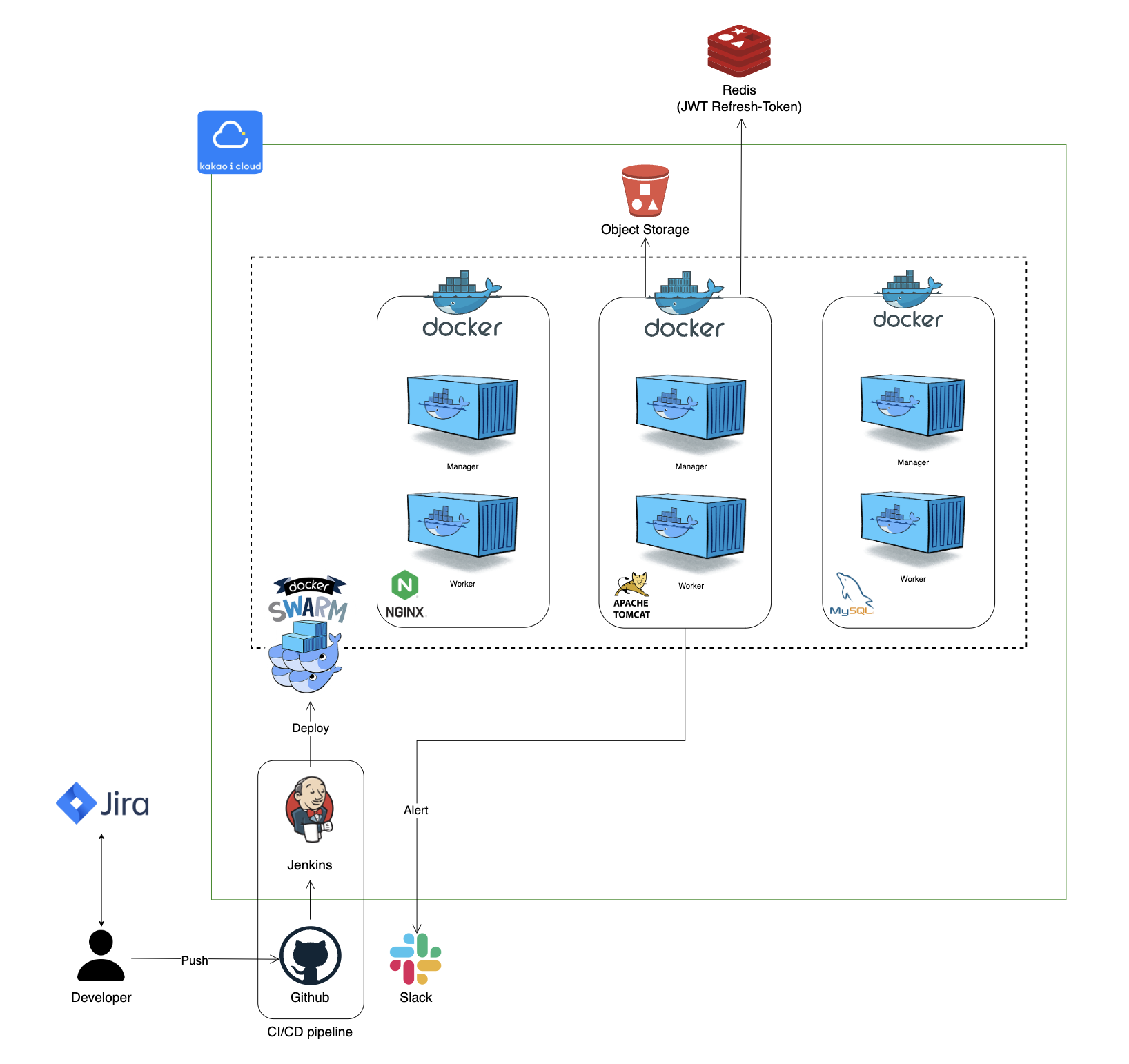
**9. 개체-관계 다이어그램(ERD)**



**10. 데이터 흐름도(DFD)**



**11. 아키텍처 구조도**



**12. API 및 인터페이스 명세서**

양이 많아 별첨하였음.