

## [ 2월 2주차 프로젝트 수행일지 ]

프로젝트 주제	Pooli (폴리) - 실시간 가족 데이터 통합 관리 및 스마트 제약 시스템		
프로젝트 팀명	1조 : 무너팜	프로젝트 팀원	백엔드 조장: 이승현 프론트엔드 조장 : 이해니 조원: 김미수, 김민수, 김예린, 김준하, 박재현, 현홍석

※ 프로젝트 수행일지는 매주 금요일 17:00 까지 구글 드라이브에 업로드

### 1. 프로젝트 수행 계획 및 현황

조원	금주 프로젝트 수행 내용	차주 프로젝트 수행 계획
김미수	인프라 및 개발 환경 구축 프로젝트 구조 설계 정책 관련 API 개발	정책 관련 API 개발 완료 통합 테스트 Traffic Generator 개발
김민수	인프라 및 개발 환경 구축 프로젝트 구조 설계 파일 업로드, 문의/답변 관련 API 개발	문의/답변 관련 API 개발 어플리케이션 API 개발 통합 테스트
김예린	프론트 개발 세팅 설정, 알람, 구성원 개인 데이터, 구성원 별 애플리케이션 페이지 등 UI 개발 진행	UI 개발 마무리(백오피스 페이지, css 수정) API 연결 및 테스트 프론트 페이지 통합 및 1차 배포
김준하	인프라 및 개발 환경 구축 프로젝트 구조 설계 정책 관련 API 개발	인프라 및 개발 환경 구축 프로젝트 구조 설계 Traffic Generator 개발
박재현	인프라 및 개발 환경 구축 프로젝트 구조 설계 Spring Security 기반 인증,인가 구현 가족/회선 도메인 관련 API 개발	인프라 및 개발 환경 구축 프로젝트 구조 설계 Traffic Generator 개발
이승현	인프라 및 개발 환경 구축 프로젝트 구조 설계 권한 관련 API 개발	알람 관련 API 개발 권한 관련 API 개발 통합 테스트
이해니	메인 페이지, 그래프 UI 개발 tanstack query 세팅	UI 개발 마무리(구성원별 정책 페이지, 백오피스 페이지)

	정책 페이지 UI 개발	API 연결 및 테스트 프론트 페이지 통합 및 1차 배포
현황석	인프라 및 개발 환경 구축 프로젝트 구조 설계 Spring Security 기반 인증,인가 구현 가족/회선 도메인 관련 API 개발	가족 / 회선 도메인 관련 API

## 2. 이번주에 받은 피드백 내용 정리

내용	백엔드 질문 사항
	<ul style="list-style-type: none"> <li>- Traffic generator를 포함한 전체적인 로직과 ERD 설계가 잘 되어있는지</li> <li>- 프로젝트에 모니터링 용으로 AIOps를 추가적으로 도입하려고 하는데 우리 팀의 주제와 어울리는지, 너무 AI 쪽으로 중심이 쏠리진 않을지</li> </ul>
	<p>백엔드 피드백</p> <ul style="list-style-type: none"> <li>- 데이터 사용 일자를 Integer로 지정해놨는데 실제 내용과 데이터 형식을 일치하게 만드는 게 좋다. Integer를 사용해야 하는 정말 피치 못할 사정이 없다면 DATE 타입으로 수정해야 한다.</li> <li>- 현업에서는 전체적인 ERD 테이블에 등록, 수정, 삭제 일자를 포함해서 수정자, 삭제자 정보도 컬럼에 포함한다. 우리의 ERD에도 이런 정보를 넣어주면 좋을 것 같다.</li> <li>- 데이터를 실제로 모니터링한다면 그라파나, 프로메테우스 같은 툴을 사용하고 여기에 ai를 붙이는 게 나을 것 같다. 처리하고자 하는 주제(문제)에 비해서는 AI 관련 기능(몸집)이 커진 느낌이 든다. 현재 상황은 어떨고, 몇 시에 문제 발생했고, 몇 시에 해결됐다고와 같은 내용을 보여주는 기능이 좀 더 들어가면 이런 문제를 보완할 수 있을 것 같다.</li> </ul> <p>백엔드 깨달은 점</p> <ul style="list-style-type: none"> <li>- 데이터 사용 일자를 계산하기 쉽게 하려고 현실과는 다른 데이터 타입을 선정했는데 좋지 않다는 사실을 깨달아 멘토님의 피드백대로 수정했다.</li> <li>- AI의 추가적인 기능에 대해 팀원 간의 논의가 추가로 필요해보인다. 모니터링 AI 도입을 추상적으로만 생각했는데 각 기능에 대한 구체화를 통해 기존의 프로젝트 주제와 균형을 잡아 작업해야 할 것 같다.</li> </ul>

## 프론트 피드백

- 멘토링 시간에는 **UX** 측면에서의 완성도 개선 방향에 대한 조언을 구하는 것이 좋을 것 같다. 특히 사용 흐름, 상태 변화에 따른 피드백 방식, 정책 설정 **UI**의 직관성 등에 대한 점검이 필요할 것으로 보인다.
- 협업 과정에서의 **Jira** 이슈 관리 및 업무 분배 방식이 효율적인지에 대해 멘토님께 구체적으로 질문해보는 것이 좋겠다는 조언을 받았다. 기술적인 완성도뿐만 아니라 협업 구조 역시 중요한 평가 요소라고 피드백 해주셨다.
- 발표 및 시연 영상 제작까지 고려했을 때 개발 일정에 여유를 두는 것이 중요하다는 점이 언급되었다. 기능 구현이 완료되더라도 시연 환경 점검, 데이터 연결 테스트, 예외 상황 대응 등을 준비해야 하므로 일정 관리에 신경 써야 할 필요가 있다.
- 전체적으로 프론트엔드 개발은 큰 문제 없이 안정적으로 진행되고 있으며, 남은 기간 동안은 **UX** 디테일 보완과 시연 준비에 집중하는 것이 적절하다는 피드백을 받았다.

## 프론트 깨달은 점

- **TanStack Query**를 활용하여 서버 상태를 관리하는 구조가 단순한 데이터 요청을 넘어 캐싱, 재요청 제어, 로딩/에러 처리까지 포함한다는 점에서 실무적인 장점이 크다는 것을 알게 되었고 프로젝트에 적극적으로 적용해야 할 것 같다.
- 정책 변경 **UI**와 관련해서는 한 번에 저장하는 방식과 즉시 반영 방식 사이의 **UX** 선택 기준을 고민하게 되었고, 최근에는 즉시 반영 패턴이 많이 사용된다는 점을 알게 되었다. 단순히 기능을 구현하는 것을 넘어서 사용자 경험과 정책의 중요도를 함께 고려해야 한다는 점을 배웠다.
- **API** 에러 처리, 로딩 상태, 부분 실패 처리와 같은 예외 상황에 대한 대응이 프로젝트 완성도를 좌우한다는 점을 인지하게 되었고, 테스트 전략과 **QA** 기간을 충분히 확보하는 것이 중요하다는 것을 느꼈다.