

3M 프로젝트 회고록

필수 질문

- 프로젝트를 소개해 주세요!
 - 에빙 하우스의 망각 곡선에 따르면 사람은 학습한 후 10분 후부터 50%의 내용을 잊습니다.
 - 학습한 내용을 장기 기억으로 저장하기 위해 필요한 것은, 주기적으로 반복 학습하는 것이 필요합니다.
 - 이 프로젝트는 에빙 하우스의 망각 곡선에 기초하여 사용자가 학습한 내용을 장기 기억으로 저장할 수 있도록 돕는 서비스를 만드는 프로젝트입니다.
 - 이 서비스를 이용하면 학습한 내용의 장기 기억을 통해 사용자는 궁극적으로 공부(학업)에 대한 자신감을 얻을 수 있습니다.
- 프로젝트를 하면서 어떤 문제를 겪었나요?
 - 협업이 처음이라 API문서가 무엇을 의미하는지 몰라서 문서를 보지않은채 개발을 진행했습니다. 요청을 보낼 때 이름이 달라 문제가 생겼었습니다.
 - 그 이후 API 문서의 중요성을 깨닫고 문서를 보며 개발하게 되었습니다. API 문서를 프론트와 백엔드가 같이 작성해야 한다는 말을 공감하게 되는 순간이었습니다.
 - 프로젝트 특성 상 로그인에 의존되는 기능들이 많은데 로그인 상태를 유지하는 것이 문제가 생겨서 다른 기능까지 추가하지 못하는 문제가 있었습니다.
 - 데브 기간이 얼마 남지않아서 로그인과 다른 기능들의 의존관계를 끊고 데브를 위한 개발을 하게 되었습니다. 데브는 끝났지만 현재 리팩토링을 진행하면서 해당 오류를 해결하고 있습니다.
- 프로젝트 하기 전 후 달라진 점이 있다면?
 - 3M 팀 모두 이 프로젝트가 첫 프로젝트입니다. 프로젝트를 하기 전에는 소프트웨어나 애플리케이션 개발이 어떠한 과정으로 이루어지는지 전혀 몰랐습니다. 소프트웨어 하나를 만들기 위해서는 요구사항 분석, 기획, 개발, 테스트, 배포, 운영 등 이러한 과정을 모두 거쳐야 하는 것을 알게 되었고 이 서비스를 통해서 사용자는 궁극적으로 어떤 결과를 얻게 될지 생각해 보는 것도 필요함을 알게되었습니다.

- 한 학기 동안 수행하는 프로젝트 만큼 장기적인 기간 동안 지치지 않는 정신력이 필요하다고 생각합니다. 만약에 지칠 경우라도 빠른 시간 안에 다시 회복할 수 있는 능력을 길러야 한다고 생각합니다.
- 3M 팀 모두 이번 프로젝트를 통해 정신력을 기를 수 있었고 기획 단계부터 팀원 모두가 참여하여 마지막까지 포기하지 않고 프로젝트를 끝냈으므로 팀원 모두가 개발자로서 한 단계 성장했다고 생각합니다.
- 프로젝트를 시작하는 팀에게 전해줄 꿀팁을 말해주세요.
 - 팀 목표는 당연하고, 개인적인 목표도 설정하면 좋을 것 같습니다. 저는 새로운 기술을 익히고 특정 기능을 구현하는 목표를 세웠습니다. 확실히 목표를 세우니까 제가 더 프로젝트에 적극적으로 참여하게 되고, 성취감을 느낄수 있었습니다.
 - 프로젝트 기간을 명확히 정하면 좋을거 같습니다. 회의 날짜와 시간을 정해서 회의가 늘어지지 않게 하는 것도 도움이 많이 되었던거 같습니다. 또한, 개발 기간이나 특정 기능을 구현하는 기간 등을 정해 놓으면 조금 더 효율적으로 개발을 할 수 있을거 같습니다.

개발자 질문

- 민준 : 팀 내 협업에서 문제는 없었는지 궁금합니다!
 - 3M 팀은 처음부터 명확한 목표를 정하고, 일정 주기마다 회의를 통해 프로젝트 진행 상황을 공유했습니다. 이를 통해 각자의 역할과 책임을 명확히 했습니다.
 - 또한 팀원들은 서로의 의견을 존중하고 개방적으로 의사소통을 이루었습니다. 프로젝트 진행 중 발생한 이슈나 개선점에 대해서는 정기적인 피드백 시간을 가지고 논의하였고, 이를 통해 문제를 해결하고 협업 능력을 향상시켰습니다.
 - 뿐만 아니라 개인의 역량 강화에도 노력을 기울였습니다. 팀원들은 새로운 기술과 도구에 대한 학습을 꾸준히 진행하고, 서로의 지식을 나누어 팀 전체의 역량을 향상시켰습니다.
 - 이를 통해 팀원들은 개인적인 성장을 이루고 동시에 팀의 전체적인 역량을 향상시키는 데에 기여했습니다.
 - 3M 팀은 위와 같은 노력들을 통해 개발동아리에서 협업에 큰 문제 없이 활동할 수 있었습니다.
- 명현 : 개발자로서의 역량 향상을 위해 어떤 노력을 기울였으며, 이 프로젝트를 통해 어떤 기술적 성장을 이루었나요?

- 저는 프로젝트를 통해서 Git을 이용한 협업 능력을 향상 시키고 싶었고, 백엔드 개발자로서 기본기를 강화하고 트렌디한 기술을 학습하고 싶었습니다.
- 프로젝트를 하는동안 Git을 효과적으로 활용하여 협업과 버전 관리를 원활하게 진행했습니다. Git 명령어를 익히고, Git-flow 라는 브랜치 전략과 커밋 메시지 규칙을 통해 협업 프로세스를 개선하고 코드베이스를 보다 깔끔하게 유지할 수 있었습니다. 이를 통해 다른 개발자들과의 협업을 효율적으로 할 수 있었습니다.
- 백엔드 개발을 하면서 스프링부트의 원리를 더 이해할 수 있었습니다. 또한, JPA와 하이버네이트 기술을 공부해서 객체와 관계형 데이터베이스 간의 매핑 및 조작에 대한 이해도를 높일 수 있었습니다. 그리고, 프로젝트에서 다양한 요구사항에 맞추어 RestAPI를 설계하고 구현하는 경험을 통해 RestAPI에 대한 실전적인 능력을 키웠습니다. 이를 통해 클라이언트와 서버 간의 효율적인 통신을 구현할 수 있었습니다.
- **종민 :** 프로젝트 개발 중 어려움을 겪은 경험이 있나요? 어떻게 해결했으며 그 과정에서 얻은 교훈은 무엇인가요?
 - 프로젝트 초기에 서비스 기획(기능 명세, API 작성)이 끝나고 DB 설계를 진행했는데, 이 과정에서 기능 구현에 만 급급한 DB 설계를 경험했습니다.
 - 프로젝트 기간 중 초기 3주 정도를 DB 설계에만 소모하여 점점 지치고, 프로젝트 진전이 되지 않았습니다. 가장 큰 원인은 우리가 원하는 기능을 DB로 어떻게 구현하지? 라는 생각 때문이었습니다.
 - 첫 프로젝트이다 보니 기능 구현에만 급급했던 것 같습니다. 또한 너무 열정적이어서 기능 구현을 DB로만 했었던 것 같습니다. 하지만, 멘토링을 통해 생각의 변화를 겪었습니다. DB는 바로 데이터를 저장하기 위한 수단일 뿐 기능 구현 용도가 아니라는 것입니다. DB 계층에서 기능을 구현하는 생각을 버리고 서비스 계층에서 기능을 구현해도 된다는 생각을 가지니 어느 정도 DB 설계가 마무리 되었습니다.
 - 만약 기능 구현에만 급급하여 DB 설계를 마쳤다면 기능을 추가하고 싶을 때 테이블 구조를 전부다 알아야 할 것 입니다. 물론 기능 구현도 중요하긴 하지만, DB 설계 단계에서는 어느 정도 서비스의 확장성을 고려하여 DB 설계를 해야됨을 깨닫게 되었습니다.