



# 2025학년도 모바일 프로그래밍 기말 프로젝트 안내

이번 기말 프로젝트는 단순한 앱 개발을 넘어서,  
프로그래밍 사고력과 협업 중심 개발 경험을 목표로 구성되었습니다.



## 프로젝트의 목적

본 프로젝트는 다음과 같은 교육적 목적을 중심으로 설계되었습니다:

### 1. AI 도구 활용을 전제로 한 사고 중심 프로그래밍

- 단순한 코드 작성은 AI가 도와줄 수 있는 시대입니다.
- 무엇을 만들지, 어떻게 구조화할지, 왜 그렇게 설계할지를 스스로 판단하는 인간 중심의 사고 능력이 더욱 중요해졌습니다.

### 2. 현업에 가까운 전체 개발 사이클 체험

- 기획부터 구현, 테스트, 실행까지의 개발 전 과정을 팀으로 분담하여 수행합니다.
- 단순한 기능 구현을 넘어, 서비스 전체의 구조와 실행 흐름을 설계해보는 데 초점을 둡니다.

### 3. 실제 협업 방식을 경험하는 팀 프로젝트

- GitHub를 활용한 형상관리, 브랜치 전략, 역할 분담, 리뷰 등을 통해 실무 협업의 기본기를 체험하게 됩니다.



## 프로젝트 수행 단계 (총 3단계)

각 팀은 다음 3단계 중 자신 있는 수준까지 도전할 수 있으며,  
완성도에 따라 평가 점수가 달라집니다.

단계	요구 사항 요약	평가 기준
1단계 (기본)	- SQLite를 사용하는 안드로이드 앱 - 로컬 DB 저장/조회 기능	최대 80점
2단계 (심화)	- 개발 모드: SQLite 사용 - 프로덕션 모드: Firebase 연동 - 설정값에 따라 DB 전환 가능	최대 85점 (진척도에 따라 부분 가산점 부여 가능)
3단계 (도전)	- 2단계를 포함 - 별도 서버와 통신 - 서버는 DB와 연결되며 독립 실행 가능	최대 100점 (진척도에 따라 부분 가산점 부여 가능)

**! 1단계는 모든 팀이 반드시 수행해야 합니다.**

2단계 및 3단계는 선택 과제이며, 도전 시 진척도에 따라 추가 점수 또는 가산점이 부여됩니다. 3단계 개발시 서버는 팀에서 선택한 언어와 기술로 자유롭게 개발할 수 있습니다. 또한, RESTful API, gRPC, TCP 소켓 통신 등 원하는 통신 프로토콜을 자유롭게 사용할 수 있습니다. 단, 클라이언트 앱과 명확하게 통신되도록 실행 방식과 인터페이스를 문서화해 주세요. 3단계 개발시 DB는 SQLite, NoSQL, RDB 등등 무엇을 사용해도 관계 없다.

## 협업 및 형상관리 지침

- **GitHub 사용은 필수입니다.**
  - 모든 소스 코드는 GitHub의 **공개 저장소**에 관리해야 합니다.
  - 브랜치 전략 적용 필수 (예: **main**, **dev**, **feature/\*** 등)
- 커밋 및 브랜치 관리
  - 명확한 커밋 메시지 작성과 기능 단위 브랜치 운영을 권장합니다.
- **Issue / PR 활용 (선택 사항)**
  - GitHub Issue를 통한 작업 관리
  - Pull Request 기반 코드 리뷰
  - 위 기능들을 활용한 팀에게는 **협업 완성도에 따라 가산점**이 부여될 수 있습니다.

## 실행 방식 (3단계 해당)

### 안드로이드 앱

- GitHub 저장소에서 코드를 내려받아 **Android Studio**에서 바로 **Run** 가능한 수준으로 구성되어야 합니다.

### 서버

- 서버가 포함된 프로젝트의 경우, **명령어 실행만으로 실행 가능하도록** 구성해야 합니다.
- 어떤 디비를 사용하는지 기술하고, 평가자가 쉽게 디비를 초기화 할수 있도록 한다.

### 예시 실행 방식

환경	명령어 예시
Node.js	<code>npm install → npm start</code>
Python (FastAPI 등)	<code>pip install -r requirements.txt → uvicorn main:app --reload</code>

### README.md 필수 포함 내용

- 서버 실행 방법
- 의존성 설치 명령어
- 포트 및 환경변수 설정 예시
- 기타 실행 관련 주의사항

△ Docker 사용도 가능합니다.

단, image repository는 제공되지 않으므로 **Dockerfile**을 포함하고,  
**docker build**가 실패하지 않도록 주의해주세요.

## 빌드/배포 자동화 제외

- CI/CD, 자동 빌드 및 클라우드 배포는 이번 수업 범위에서 제외합니다.
- 다만, 모든 소스코드는 로컬 환경에서 수동으로 빌드 및 실행 가능해야 합니다.

## 제출 방식

항목	제출 내용
GitHub 링크	소스 코드, README, 문서 포함
발표 자료	PDF 형식 (팀별 발표용)

## 17 주요 일정

- 팀 구성 마감: 10월 17일 (목)
- 중간고사 발표 (주제 제안): 10월 24일 (목)
  - 팀 단위 발표 (전체 성적의 **10%** 반영)
  - 발표 후 주제 변경 또는 범위 조정 가능
  - 발표하지 않을 경우 **0점** 처리
- 최종 프로젝트 발표 및 제출일: 12월 중 (일정은 추후 공지 예정)

## 평가 기준 안내

구체적인 평가 항목 및 세부 배점은 별도 공지를 통해 안내할 예정입니다.

평가 항목에는 다음이 포함될 수 있습니다:

- 단계별 구현 수준
- 협업 방식 및 GitHub 활용도
- 문서화 및 README 구성
- 실행 가능성
- 코드 구조 (클라이언트 앱에 한해 평가)

## 마무리

이번 프로젝트는 단순한 앱 제작 과제가 아닙니다.

개발자로서 설계 능력과 협업 사고를 실습하는 실제 서비스 구축 프로젝트입니다.

- 기술은 도구일 뿐입니다.
- 중요한 것은 설계하고, 구조화하고, 함께 만드는 능력입니다.

여러분의 도전과 창의적인 결과물을 기대합니다.