

StudyRoom

스터디룸 협업 학습 플랫폼

제품 구성 / 배포 / 운영 자료

Version 1.0
2025년 12월

목차

1. 프로젝트 개요
2. 기술 스택
3. 시스템 아키텍처
4. 데이터베이스 스키마
5. API 명세
6. 환경 변수 설정
7. 배포 가이드
8. 주요 기능 상세
9. 운영 및 모니터링
10. 트러블슈팅

1. 프로젝트 개요

1.1 프로젝트 소개

StudyRoom은 효율적인 그룹 학습을 위한 협업 플랫폼입니다. 자료 공유, AI 기반 퀴즈 생성, 그룹 토론, 일정 관리 기능을 통합하여 학습 효율을 극대화합니다.

1.2 핵심 기능

- 스터디룸 생성 및 멤버 관리 (PIN 기반 참여)
- 대용량 파일 업로드 및 자동 벡터화
- GPT-5.1 기반 AI 퀴즈 자동 생성
- RAG 기반 AI 학습 도우미 채팅
- 그룹 학습 및 오답 분석
- 일정 관리 (캘린더)

2. 기술 스택

2.1 Frontend / Backend

구분	기술
Framework	Next.js 15 (App Router, Turbopack)
언어	JavaScript (ES6+)
UI	TailwindCSS 4
상태관리	TanStack Query (React Query)
인증	Supabase Auth + JWT

2.2 Database & Storage

구분	기술
데이터베이스	Supabase (PostgreSQL)
파일 스토리지	Supabase Storage (bucket: room-files)
벡터 DB	Supabase pgvector (임베딩 검색)

2.3 AI & External Services

구분	기술
퀴즈 생성 AI	GPT-5.1 (Thinking, reasoning_effort: medium)
채팅 AI	GPT-5.1 Chat Latest (Instant)
임베딩	OpenAI text-embedding-3-small
OCR	Google Cloud Vision API
임시 파일 저장	Google Cloud Storage

2.4 배포 플랫폼

- 호스팅: Vercel
- **Functions 제한:** maxDuration 60 초, Memory 3008MB
- **Body Size 제한:** 50MB (Server Actions)

3. 시스템 아키텍처

3.1 전체 아키텍처

클라이언트 → Next.js API Routes → Supabase/OpenAI → 응답

파일 업로드 플로우

1. 클라이언트에서 Presigned URL 요청
2. 서버가 Supabase Storage Presigned URL 발급
3. 클라이언트 → Supabase Storage 직접 업로드 (Vercel 4.5MB 제한 우회)
4. 업로드 완료 후 메타데이터 저장 API 호출
5. 서버에서 텍스트 추출 및 벡터화 처리

AI 퀴즈 생성 플로우

1. 사용자가 파일/주제 선택 및 난이도 설정
2. 선택된 파일의 벡터 데이터 조회
3. GPT-5.1 에 컨텍스트와 함께 퀴즈 생성 요청
4. 생성된 퀴즈 DB 저장

AI 채팅 플로우 (RAG)

1. 사용자 질문 임베딩
2. pgvector 로 유사 문서 청크 검색
3. 검색된 컨텍스트 + 질문으로 GPT-5.1 Chat 호출
4. 스트리밍 응답 반환

3.2 프로젝트 파일 구조

studyroom/

├─ app/	# App Router
│ └─ api/	# API Routes
│ └─ auth/	# 인증 (login, register, logout)
│ └─ room/	# 강의실 CRUD
│ └─ quiz/	# 퀴즈 생성/제출
│ └─ chat/	# AI 채팅
│ └─ (auth)/	# 인증 페이지 그룹
│ └─ room/	# 강의실 페이지
│ └─ dashboard/	# 대시보드
├─ components/	# 재사용 컴포넌트
│ └─ layout/	# Sidebar, Header, Footer
│ └─ ui/	# 공통 UI 컴포넌트
├─ lib/	# 유틸리티
│ └─ supabase/	# Supabase 클라이언트
│ └─ auth.js	# 인증 헬퍼

└─ public/

정적 파일

4. 데이터베이스 스키마

4.1 핵심 테이블

User

컬럼명	타입	설명
UserID	VARCHAR(15)	PK, Supabase Auth ID
UserInputID	VARCHAR(15)	이메일 (로그인 ID)
name	VARCHAR(15)	사용자 이름

Room

컬럼명	타입	설명
RoomID	VARCHAR(20)	PK
RoomName	VARCHAR(15)	강의실 이름
AdminID	VARCHAR(15)	FK → User.UserID
EnterPin	VARCHAR(15)	6자리 참여 PIN
RoomPassword	VARCHAR(20)	선택적 비밀번호

기타 주요 테이블

- **RoomMember:** RoomID, UserID, Role (owner/editor/viewer)
- **File:** FileID, RoomID, FileName, FileUrl, UserID
- **FileChunk:** ChunkID, FileID, ChunkText, Embedding (vector)
- **Quiz:** QuizID, QuizTitle, QuizRoomID
- **Question:** QuestionID, QuizID, 문제, optionA~D, correctAnswer, explanation
- **Answer:** AnswerID, QuestionID, userAnswer, isCorrect
- **QuizAttempt:** AttemptID, UserID, QuizID, Score, CompletedAt
- **Schedule:** EventID, EventTitle, EventDate, RoomID
- **chat_sessions:** SessionID, UserID, QuestionID, started_at
- **chat_messages:** AI_chatID, sender, message, created_at, SessionID
- **comment:** CommentId, Comment, TypeTime, QuestionID, UserID

5. API 명세

5.1 인증 API

Method	Endpoint	설명
POST	/api/auth/register	회원가입
POST	/api/auth/login	로그인
POST	/api/auth/logout	로그아웃
GET	/api/auth/me	현재 사용자 조회

5.2 강의실 API

Method	Endpoint	설명
POST	/api/room	강의실 생성
POST	/api/room/join	강의실 참여 (PIN)
GET	/api/room/[roomId]/members	멤버 목록 조회
PATCH	/api/room/[roomId]/members	멤버 권한 수정

5.3 파일 API

Method	Endpoint	설명
POST	/api/room/[roomId]/file/presigned-url	Presigned URL 발급
POST	/api/room/[roomId]/file	파일 메타데이터 저장
GET	/api/room/[roomId]/file	파일 목록 조회
GET	/api/room/[roomId]/file/[fileId]/download	파일 다운로드
DELETE	/api/room/[roomId]/file/[fileId]	파일 삭제

5.4 퀴즈/채팅 API

Method	Endpoint	설명
POST	/api/quiz/generate	AI 퀴즈 생성
GET	/api/quiz/[roomId]/[quizId]	퀴즈 상세 조회
POST	/api/quiz/[quizId]/answer	퀴즈 답변 제출
POST	/api/chat/session	채팅 세션 생성
POST	/api/chat/message	메시지 전송 (스트리밍)

6. 환경 변수 설정

6.1 필수 환경 변수

.env.local 파일에 설정:

Supabase

```
NEXT_PUBLIC_SUPABASE_URL=https://xxx.supabase.co
NEXT_PUBLIC_SUPABASE_ANON_KEY=eyJ...
SUPABASE_SERVICE_ROLE_KEY=eyJ...
SUPABASE_ROOM_FILES_BUCKET=room-files
```

OpenAI

```
OPENAI_API_KEY=sk-...
```

Google Cloud

```
GOOGLE_APPLICATION_CREDENTIALS={...} # JSON 문자열
GOOGLE_API_KEY=Aiza...
GOOGLE_OCR_GCS_BUCKET=bucket-name
```

6.2 Vercel 환경 변수 설정

1. Vercel 대시보드 → 프로젝트 선택
2. Settings → Environment Variables
3. 위의 환경 변수들을 Production/Preview/Development 환경에 추가
4. 민감한 키는 Sensitive 옵션 활성화

7. 배포 가이드

7.1 Vercel 배포

초기 설정

```
# 1. GitHub 저장소 연결
vercel link

# 2. 환경 변수 설정 (Vercel 대시보드에서)

# 3. 배포
vercel --prod
```

vercel.json 설정

```
{
  "functions": {
    "app/api/room/[roomId]/file/route.js": {
      "maxDuration": 60,
      "memory": 3008
    }
  }
}
```

next.config.mjs 설정

```
const nextConfig = {
  experimental: {
    serverActions: {
      bodySizeLimit: '50mb',
    },
  },
};
```

7.2 Supabase 설정

- **Storage Bucket 생성:** room-files 버킷 생성
- **RLS 정책:** 각 테이블에 Row Level Security 정책 설정 권장
- **pgvector 확장:** Extensions → vector 확장 활성화

8. 주요 기능 상세

8.1 파일 관리

대용량 파일 업로드 (Presigned URL)

- Vercel 의 4.5MB 제한을 우회하기 위해 클라이언트에서 Supabase Storage 로 직접 업로드
- Presigned URL 을 통한 보안 업로드
- 업로드 후 자동으로 텍스트 추출 및 벡터화

8.2 AI 퀴즈 생성

- **모델:** GPT-5.1 (reasoning_effort: medium)
- **입력:** 파일 내용 기반 또는 주제 기반
- **난이도:** 쉬움 / 보통 / 어려움 선택 가능
- **출력:** 객관식 4 지선다 문제 + 해설

8.3 AI 채팅 도우미 (RAG)

- **모델:** GPT-5.1 Chat Latest
- **RAG 파이프라인:** 질문 임베딩 → pgvector 유사도 검색 → 컨텍스트 주입 → 응답 생성
- **응답 방식:** 스트리밍

8.4 그룹 학습

- 오답률 높은 문제 분석
- 문제별 댓글/토론 기능
- 재풀이 모드

8.5 일정 관리

- 캘린더 뷰
- 일정 CRUD
- D-Day 표시

9. 운영 및 모니터링

9.1 주의사항

Supabase RLS

- **현재 상태:** RLS 비활성화 (경고 발생 중)
- **영향:** Service Role Key 사용으로 현재는 정상 작동
- **권장:** 프로덕션 환경에서는 RLS 정책 설정 필요 (보안 강화)

파일 업로드

- **로컬 환경:** FormData 방식도 작동
- **Vercel 배포:** Presigned URL 방식 필수 (4.5MB 제한)

AI 비용

- GPT-5.1 사용으로 GPT-4o-mini 대비 약 2-3 배 비용 증가
- 대량 사용 시 비용 모니터링 필요

9.2 모니터링 포인트

- **Vercel Analytics:** 요청 수, 에러율, 응답 시간
- **Supabase Dashboard:** DB 사용량, Storage 용량, API 호출
- **OpenAI Usage:** 토큰 사용량, 비용
- **Google Cloud Console:** Vision API 호출, Storage 사용량

10. 트러블슈팅

10.1 자주 발생하는 문제

문제	해결 방법
파일 업로드 실패 (4.5MB 초과)	Presigned URL 방식으로 클라이언트 직접 업로드 사용
인증 에러	JWT 토큰 만료 확인, 로그아웃 후 재로그인
퀴즈 생성 느림	GPT-5.1 Thinking 모델 특성, 정상 동작
벡터 검색 결과 없음	파일 벡터화 완료 확인, FileChunk 테이블 확인
로그아웃 후 캐시 문제	TanStack Query 캐시 초기화 + localStorage 정리

10.2 최근 수정 사항 (2025.12.04)

- 클라이언트 직접 업로드 구현 (Vercel 제한 우회)
- AI 모델 업그레이드: GPT-4o-mini → GPT-5.1
- SQL 쿼리 버그 수정 (User 테이블 컬럼명)
- 인증 속도 개선
- 캐시 관리 개선 (로그아웃 시 정리)

— End of Document —