

PrayGround

교회 청년부 커뮤니티 플랫폼

Next.js 14와 Supabase를 활용한 현대적인 웹 기반 PWA로, 청년부 소통과 일정 관리를 통합한
올인원 플랫폼입니다.



프로젝트 개요

PrayGround는 교회 청년부를 위한 통합 커뮤니티 플랫폼입니다. 분산된 공지사항, 자료, 소통 채널을 하나로 통합하여 참여를 극대화하고, 교회 인증 기반의 안전한 커뮤니티를 제공합니다.

Repository: https://github.com/Dello96/AI_project.git

Live Demo: <https://ai-project-f45i.vercel.app>

개발일지: <https://velog.io/@dello96>

36

커밋 수

94.4%

TypeScript 비율

2.5%

PLpgSQL 비율

1.5%

CSS 비율



기술 스택

Frontend

Next.js 14 (App Router)

TypeScript (94.4%)

TailwindCSS

Shadcn UI

Framer Motion

Backend

Supabase

PostgreSQL (2.5%)

Row Level Security

Real-time Subscriptions

PLpgSQL

Authentication

Supabase Auth

OAuth (Google, Kakao)

JWT Tokens

Role-based Access

Church Domain Validation

Storage & AI

Supabase Storage
Google Gemini AI
File Upload System
Image Processing
Chat Attachments

Payment

Toss Payments
Secure Transactions
Missionary Support
Payment Verification

Deployment

Vercel
PWA Support
CDN
Auto Scaling
Environment Variables

✨ 주요 기능



통합 게시판

- ✓ 공지사항, 자유게시판, Q&A 카테고리
- ✓ 실명/익명 선택 가능
- ✓ 좋아요 및 댓글 시스템
- ✓ 실시간 알림
- ✓ 파일 첨부 지원

17 일정 관리

- ✓ FullCalendar 기반 캘린더
- ✓ 예배, 모임, 행사, 소그룹 일정
- ✓ 출석 체크 기능
- ✓ 월/주/일 뷰 지원
- ✓ 일정 알림

인증 시스템

- ✓ 교회 이메일 기반 가입
- ✓ 관리자 승인 시스템
- ✓ OAuth 소셜 로그인
- ✓ 역할 기반 권한 관리
- ✓ 보안 강화

AI 챗봇

- ✓ Google Gemini AI 기반
- ✓ 파일 첨부 지원
- ✓ 채팅 히스토리 관리
- ✓ 실시간 대화
- ✓ 커스텀 프롬프트

결제 시스템

- ✓ Toss Payments 연동
- ✓ 선교사 후원 기능
- ✓ 안전한 결제 처리

- ✓ 기부 통계
- ✓ 결제 내역 관리

PWA 지원

- ✓ 모바일 최적화
- ✓ 오프라인 지원
- ✓ 푸시 알림
- ✓ 앱 설치 가능
- ✓ 반응형 디자인

프로젝트 구조

Next.js 14 App Router를 기반으로 한 모듈화된 프로젝트 구조입니다.

```
ai_project/ |— app/ # Next.js App Router | |— layout.tsx # 루트 레이아웃 | |— page.tsx # 홈페이지 | |— globals.css # 전역 스타일 | |— api/ # API 라우트 | |— admin/ # 관리자 페이지 | |— auth/ # 인증 페이지 | |— board/ # 게시판 페이지 | |— calendar/ # 캘린더 페이지 | |— components/ # 재사용 가능한 컴포넌트 | |— ui/ # 기본 UI 컴포넌트 | |— auth/ # 인증 관련 컴포넌트 | |— board/ # 게시판 컴포넌트 | |— calendar/ # 캘린더 컴포넌트 | |— chat/ # AI 챗봇 컴포넌트 | |— payments/ # 결제 컴포넌트 | |— hooks/ # 커스텀 훅 | |— lib/ # 유틸리티 및 설정 | |— types/ # TypeScript 타입 정의 | |— database/ # 데이터베이스 스키마 |— public/ # 정적 자산
```

개발 과정

1단계: 프로젝트 설정 및 인증 시스템

Next.js 14 프로젝트 초기화, Supabase 연동, 사용자 인증 시스템 구축

2단계: 게시판 및 댓글 시스템

CRUD 기능 구현, 좋아요 시스템, 실시간 댓글, RLS 정책 설정

3단계: 캘린더 및 일정 관리

FullCalendar 연동, 이벤트 관리, 출석 체크, 다크 테마 적용

4단계: AI 챗봇 및 파일 업로드

Google Gemini AI 연동, 파일 첨부 기능, 채팅 히스토리 관리

5단계: 결제 시스템 및 UI 개선

Toss Payments 연동, 인터파크 극장 스타일 디자인, PWA 최적화



기술적 도전과 해결



Row Level Security (RLS) 정책 최적화

Supabase RLS를 활용하여 데이터 보안을 강화하고, 복잡한 권한 관리를 구현했습니다.

```
-- 사용자별 데이터 접근 제어 CREATE POLICY "Users can only see their own data" ON posts FOR SELECT USING (auth.uid() = author_id);
```

실시간 기능 구현

Supabase Realtime을 활용하여 댓글, 좋아요, 알림 등의 실시간 업데이트를 구현했습니다.

```
// 실시간 댓글 구독 const { data, error } = await supabase
  .channel('comments') .on('postgres_changes', { event: 'INSERT',
    schema: 'public', table: 'comments' }, (payload) => { // 실시간 댓글 업데이트 }) .subscribe();
```

인터파크 극장 스타일 디자인

사용자 경험을 향상시키기 위해 인터파크 극장의 디자인 시스템을 참고하여 모던하고 세련된 UI를 구현했습니다.

PWA 최적화

Service Worker, Web App Manifest를 활용하여 네이티브 앱과 유사한 사용자 경험을 제공합니다.

성과 및 지표

성능 최적화

- ✓ Lighthouse PWA 점수: 90+
- ✓ First Contentful Paint: < 2초
- ✓ Largest Contentful Paint: < 3초
- ✓ Cumulative Layout Shift: < 0.1

코드 품질

- ✓ TypeScript 100% 적용
- ✓ ESLint 규칙 준수
- ✓ Clean Code 원칙 적용

✓ 컴포넌트 재사용성 극대화

사용자 경험

✓ 반응형 디자인

✓ 직관적인 네비게이션

✓ 빠른 로딩 속도

✓ 접근성 고려



프로젝트 스크린샷

메인 페이지

인터파크 극장 스타일의
메인 페이지 디자인

게시판

다크 테마의
게시판 인터페이스

캘린더

FullCalendar 기반의
일정 관리 시스템

AI 챗봇

파일 첨부 기능이 있는
AI 챗봇 인터페이스

배운 점 및 향후 계획

주요 학습 내용

- **Next.js 14 App Router:** 최신 React 기능과 서버 컴포넌트 활용
- **Supabase:** BaaS를 활용한 빠른 백엔드 개발
- **TypeScript:** 타입 안전성을 통한 개발 생산성 향상
- **PWA:** 웹 기술로 네이티브 앱과 유사한 경험 제공
- **AI Integration:** Google Gemini API를 활용한 챗봇 구현

향후 개선 계획

- 모바일 앱 개발 (React Native)
- 고급 분석 및 대시보드
- 소그룹별 커스텀 기능
- 오프라인 동기화 강화

- 다국어 지원

프로젝트 정보

이 프로젝트는 교회 청년부 커뮤니티의 디지털 전환을 위해 개발된 웹 애플리케이션입니다.

[GitHub Repository](#)

[Live Demo](#)

[개발일지](#)

© 2024 PrayGround Project. All rights reserved.