

청춘장터 프로젝트 디렉토리 구조

전체 프로젝트 구조

| | |
|---|-------------------|
| cheongchun-platform/ | |
| ├ docs/ | # 문서 관리 |
| │ ├── api/ | |
| │ │ └ cheongchun_api_specs.md | # API 명세서 |
| │ ├── architecture/ | |
| │ │ └ ai_backend_separation.md | # AI 백엔드 분리 아키텍처 |
| │ ├── presentation/ | |
| │ │ └ ir_presentation.pdf | # IR 프레젠테이션 |
| │ └ README.md | |
| ├ backend/ | # Spring Boot 백엔드 |
| │ ├── src/ | |
| │ │ ├── main/ | |
| │ │ │ ├── java/ | |
| │ │ │ │ └ com/ | |
| │ │ │ │ │ └ cheongchun/ | |
| │ │ │ │ │ │ ├── config/ | # 설정 클래스 |
| │ │ │ │ │ │ ├── controller/ | # REST API 컨트롤러 |
| │ │ │ │ │ │ │ ├── auth/ | # 인증 관련 |
| │ │ │ │ │ │ │ ├── user/ | # 사용자 관리 |
| │ │ │ │ │ │ │ ├── meeting/ | # 모임 관리 |
| │ │ │ │ │ │ │ ├── chat/ | # 채팅 |
| │ │ │ │ │ │ │ └ notification/ | # 알림 |
| │ │ │ │ │ │ ├── service/ | # 비즈니스 로직 |
| │ │ │ │ │ │ │ ├── auth/ | |
| │ │ │ │ │ │ │ ├── user/ | |
| │ │ │ │ │ │ │ ├── meeting/ | |
| │ │ │ │ │ │ │ ├── chat/ | |
| │ │ │ │ │ │ │ └ notification/ | |
| │ │ │ │ │ │ ├── repository/ | # 데이터 액세스 |
| │ │ │ │ │ │ ├── entity/ | # JPA 엔티티 |
| │ │ │ │ │ │ ├── dto/ | # 데이터 전송 객체 |
| │ │ │ │ │ │ ├── security/ | # 보안 관련 |
| │ │ │ │ │ │ ├── websocket/ | # 웹소켓 설정 |
| │ │ │ │ │ │ └ utils/ | # 유틸리티 |
| │ │ │ │ └ resources/ | |
| │ │ │ │ │ ├── application.yml | |
| │ │ │ │ │ ├── application-dev.yml | |
| │ │ │ │ │ └ application-prod.yml | |
| │ │ │ └ test/ | |
| │ ├── Dockerfile | |
| │ ├── docker-compose.yml | |
| │ ├── pom.xml | |
| │ └ README.md | |
| ├ ai-services/ | # Python AI 서비스 |
| │ ├── meeting-crawling/ | # 모임 크롤링 서비스 |
| │ │ └ src/ | |

```
├── crawlers/
│   ├── seoul_lifelong_learning.py
│   ├── community_center.py
│   └── library_programs.py
├── processors/
│   ├── data_cleaner.py
│   └── category_classifier.py
├── models/
│   └── main.py
├── requirements.txt
├── Dockerfile
├── README.md
├── meeting-ai-core/                                # AI 추천 엔진
│   ├── src/
│   │   ├── recommendation/
│   │   │   ├── collaborative_filtering.py
│   │   │   ├── content_based.py
│   │   │   └── hybrid_recommender.py
│   │   ├── models/
│   │   │   ├── user_profile.py
│   │   │   └── meeting_embeddings.py
│   │   ├── training/
│   │   │   ├── model_trainer.py
│   │   │   └── data_preprocessor.py
│   │   └── main.py
│   ├── requirements.txt
│   ├── Dockerfile
│   └── README.md
├── voice-compare/                                # 음성 및 비교 서비스
│   ├── src/
│   │   ├── voice/
│   │   │   ├── stt_processor.py
│   │   │   ├── nlp_analyzer.py
│   │   │   └── intent_classifier.py
│   │   ├── compare/
│   │   │   ├── gpt_analyzer.py
│   │   │   └── comparison_engine.py
│   │   ├── models/
│   │   └── main.py
│   ├── requirements.txt
│   ├── Dockerfile
│   └── README.md
├── frontend/                                    # React Native 앱
│   ├── src/
│   │   ├── components/
│   │   │   ├── common/
│   │   │   ├── auth/
│   │   │   ├── meeting/
│   │   │   ├── chat/
│   │   │   └── profile/
│   │   ├── screens/
│   │   │   ├── auth/
│   │   │   ├── home/
│   │   │   └── meeting/
```

```
├── chat/
│   └── profile/
├── services/
│   ├── api/
│   ├── auth/
│   └── websocket/
├── utils/
├── constants/
└── App.js
├── android/
├── ios/
├── package.json
└── README.md
├── database/                                # 데이터베이스 관련
│   ├── migrations/
│   ├── seeds/
│   ├── schemas/
│   │   ├── user_schema.sql
│   │   ├── meeting_schema.sql
│   │   └── chat_schema.sql
│   └── init.sql
├── infrastructure/                         # 인프라 설정
│   ├── docker/
│   │   ├── docker-compose.yml
│   │   ├── docker-compose.prod.yml
│   │   └── nginx/
│   │       └── nginx.conf
│   ├── k8s/                               # Kubernetes 설정
│   │   ├── backend/
│   │   ├── ai-services/
│   │   └── ingress/
│   ├── terraform/                         # 인프라 as Code
│   └── scripts/
│       ├── deploy.sh
│       └── backup.sh
├── monitoring/                            # 모니터링 설정
│   ├── prometheus/
│   ├── grafana/
│   └── elasticsearch/
├── tests/                                 # 통합 테스트
│   ├── e2e/
│   ├── integration/
│   └── load/
└── .github/                              # CI/CD 설정
    ├── workflows/
    │   ├── backend-ci.yml
    │   ├── ai-services-ci.yml
    │   └── frontend-ci.yml
```

주요 디렉토리 설명

1. Backend (Spring Boot)

- **User Service:** 회원가입, 로그인, 소셜 로그인, 프로필 관리
- **Meeting Service:** 모임 CRUD, 카테고리 관리, 찜 기능, 비교 기능
- **Chat Service:** 실시간 채팅, WebSocket 연결, 알림 시스템

2. AI Services (Python)

- **Meeting Crawling:** 모임 데이터 수집, 실시간 인기도 추적
- **Meeting AI Core:** 개인화 추천, Like/Dislike 학습, 벡터 유사도
- **Voice & Compare:** 음성 인터페이스, GPT 비교 분석

3. Frontend (React Native)

- **Cross-platform:** iOS/Android 지원
- **실시간 기능:** WebSocket 연동, 푸시 알림

4. Infrastructure

- **Docker:** 컨테이너화 배포
- **Kubernetes:** 오케스트레이션
- **Monitoring:** Prometheus, Grafana, ELK Stack

개발 환경별 설정

Local Development

- Backend: `http://localhost:8080`
- AI Services: `http://localhost:8001, 8002, 8003`
- Frontend: `http://localhost:3000`
- Database: PostgreSQL (`localhost:5432`)

Production

- Backend: `https://api.cheongchun.com`
- AI Services: `https://ai.cheongchun.com`
- Frontend: 모바일 앱 스토어 배포
- Database: 클라우드 PostgreSQL

배포 구조

Phase 1: 핵심 기능 (6주)

1. 사용자 관리 시스템
2. 모임 기본 기능
3. AI 추천 시스템
4. 채팅 시스템
5. 음성 인터페이스

6. 통합 및 최적화

Phase 2: 고도화 (4주)

1. AI 성능 개선
2. 사용자 경험 개선
3. 관리 기능 강화
4. 모니터링 시스템