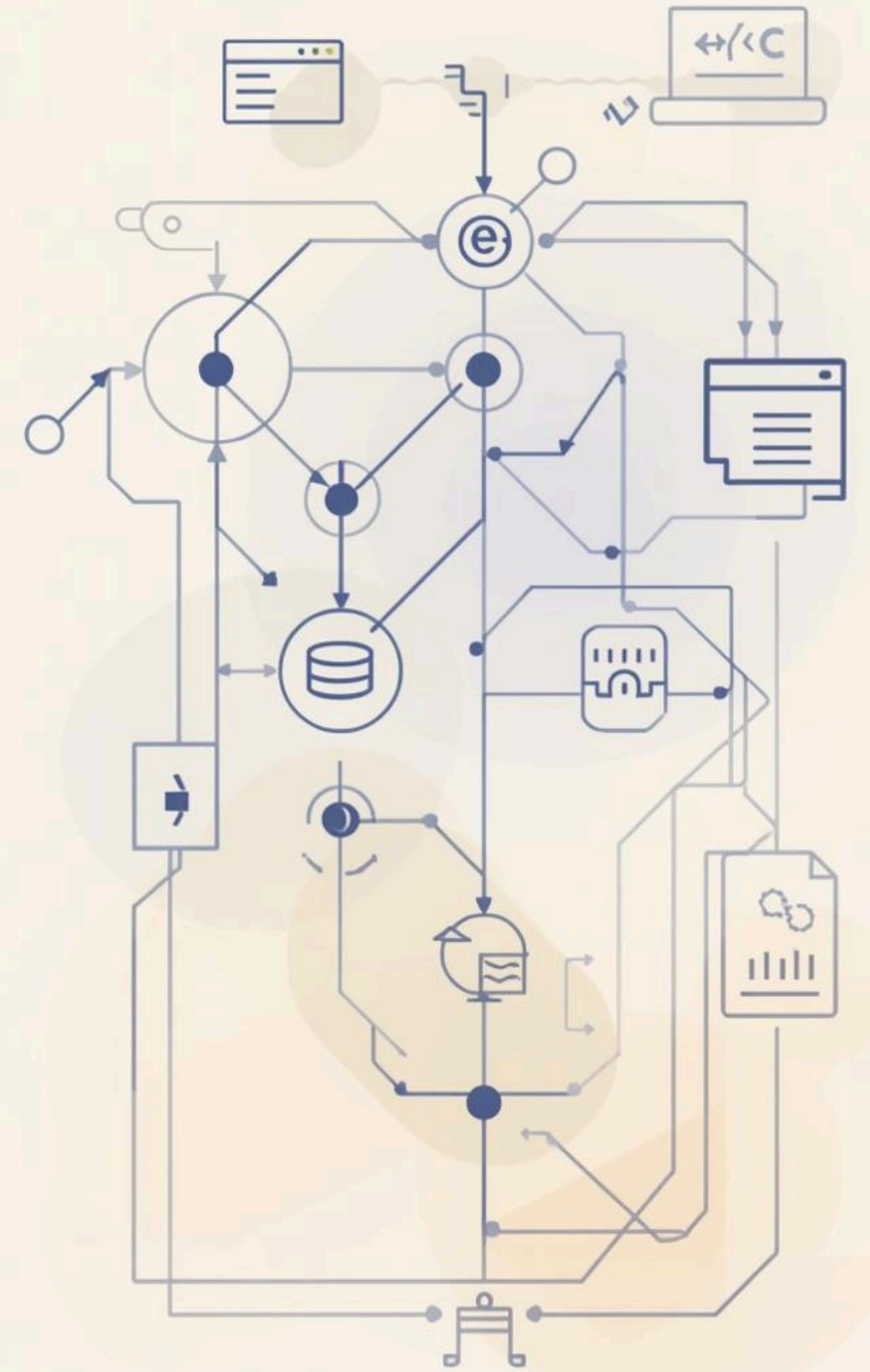


2025-02 산학프로젝트

# LawFriend

박상수 변소윤 양현아



# 개발 현황

<지금까지 구현 완료된 부분>  
백엔드

Django 프로젝트 및 앱 생성

cases 앱: 법령 및 판례 모델(LawArticle, CaseLaw) 생성 완료

DB 마이그레이션 완료 (makemigrations, migrate)

기본 데이터 모델 정의

법령: law\_name, article\_num, article\_title, article\_content, hseq

판례: case\_title, court\_name, judgment\_date, case\_number, content\_summary, full\_text, references

크롤링 스크립트 준비

scrape\_law.py, scrape\_law\_selenium.py 등 스켈레톤 작성

Selenium을 이용한 로그인 + 동적 페이지 접근 시도 기록

현재까지 로그인 없는 요청이나 requests로는 법령 데이터 접근 실패 확인

## 2. 프론트엔드

cases/templates/cases/search.html 존재

챗봇 페이지 뼈대 구성 가능

React 또는 Django 템플릿 기반으로 챗봇 UI 설계 가능

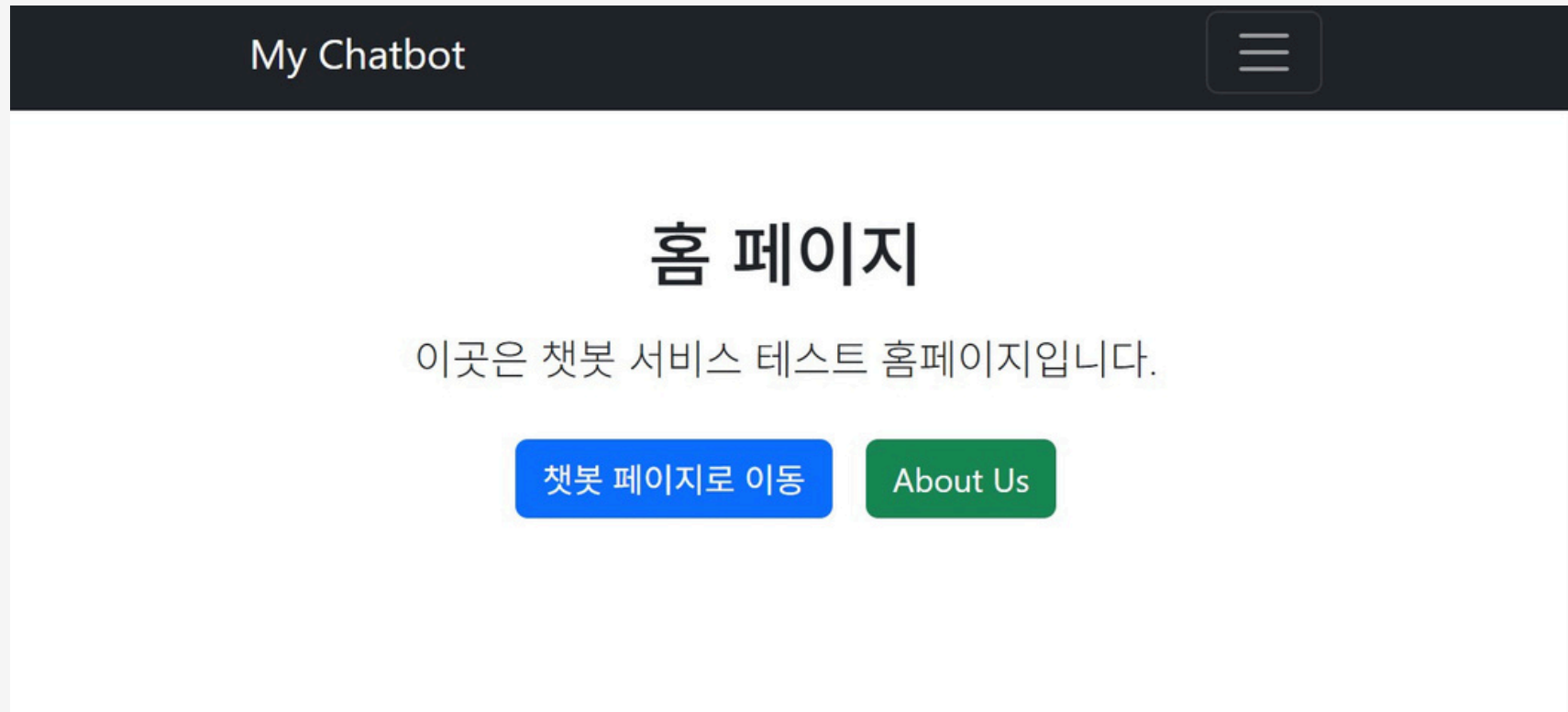
기타

프로젝트 디렉토리 구조 확인

Python virtual environment 사용 중

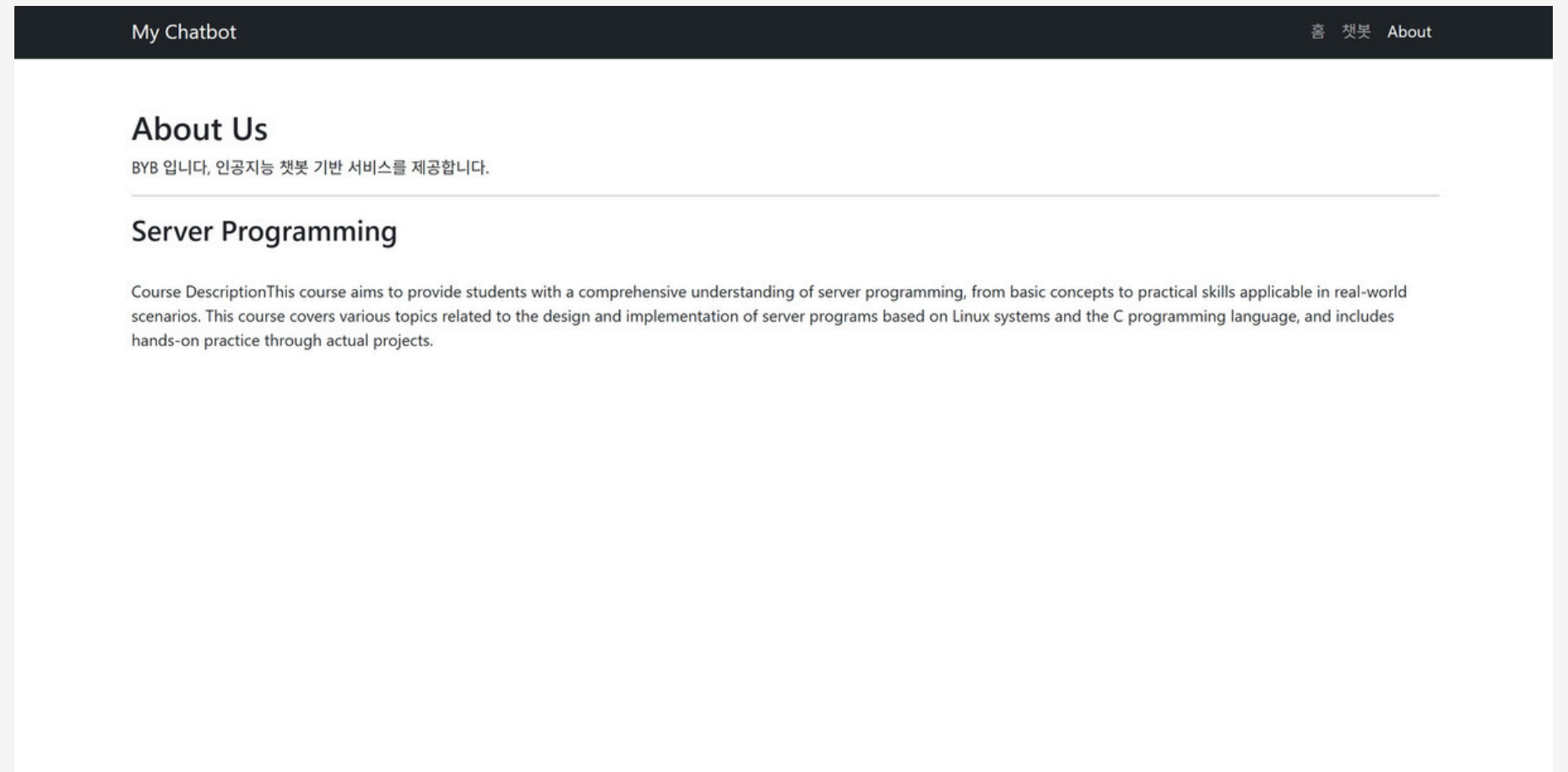
ChromeDriver/Selenium 환경 테스트 중

# 주요 기능 스크린샷



프로젝트 실행

About 페이지 이동 화면



# 주요 기능 스크린샷

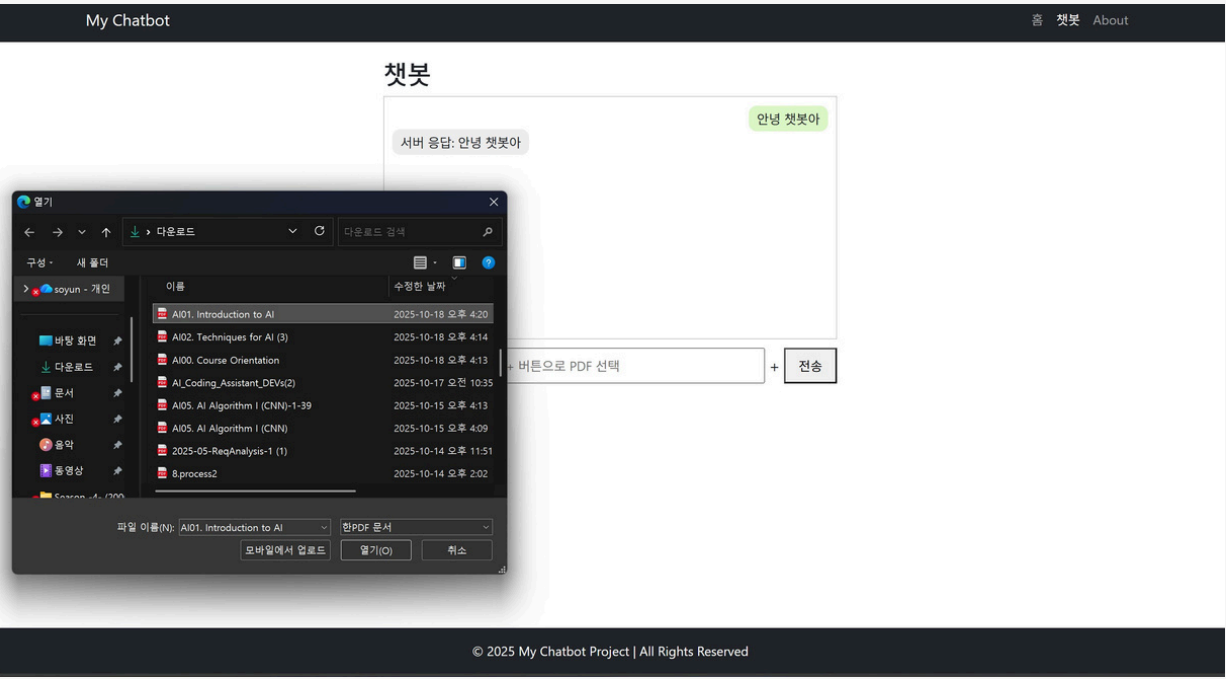


챗봇페이지



사용자의 입력 전송  
서버에서 입력 받고 응답까지 해줌

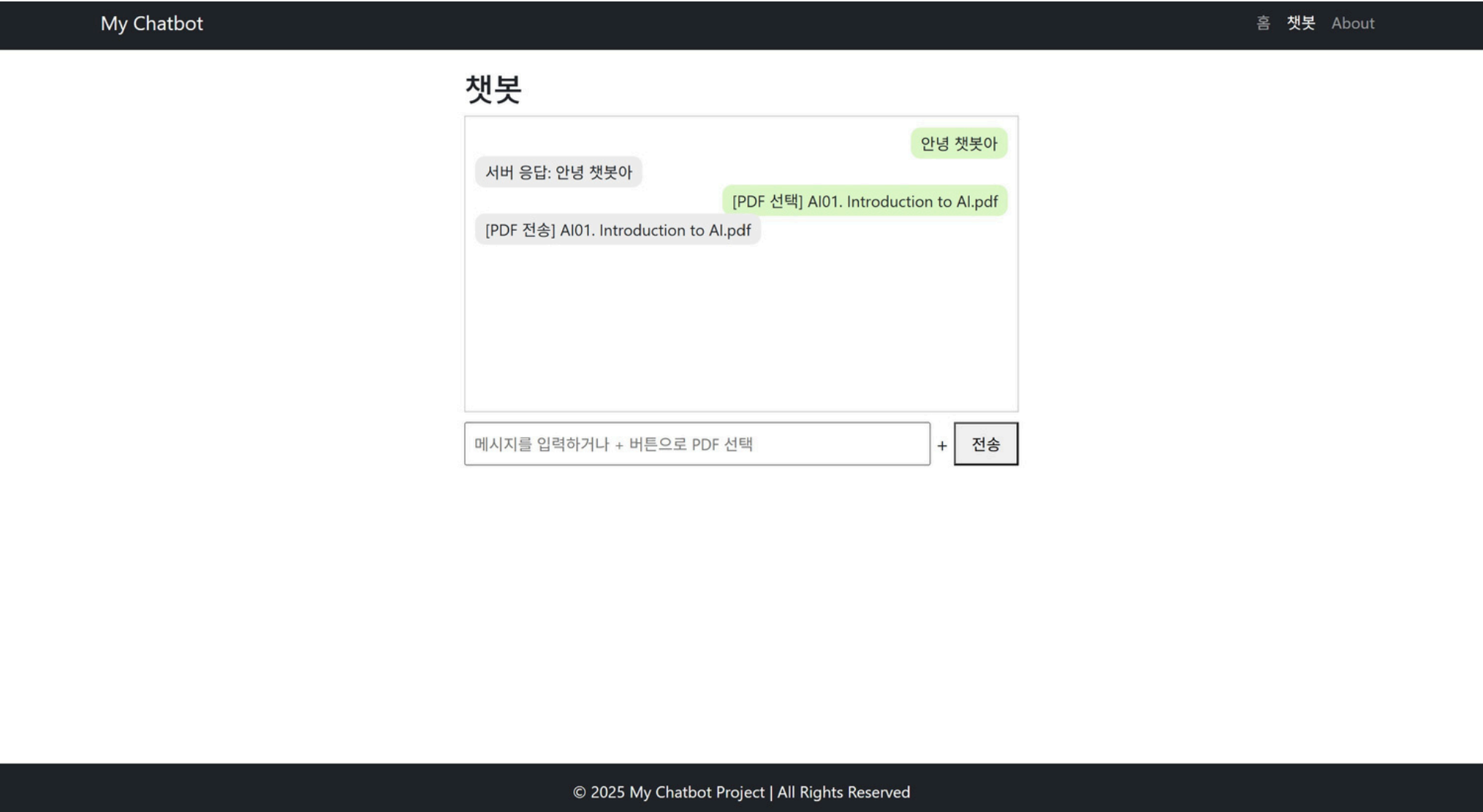
# 주요 기능 스크린샷



+버튼을 눌러서 파일을 추가하는 기능



선택한 파일이 아래 선택됨을 알 수 있음



pdf 전송 기능 구현  
이또한 사용자 입력과 마찬가지로  
서버와 통신이 됨을 확인

# 추후 개발 스케줄

- 데이터 크롤링 스크립트 작성 => 공개 법령 접근
- 데이터로부터 텍스트 데이터 추출 / Langchain 라이브러리 통해 chunking
- sentencetransformer, faiss-cpu 활용하여 embedding 진행
- vLLM 통해 모델 돌릴 예정
- 1차적으로 검색 기능 구현 후 형량 추론 및 변호사 추천 기능 구현 예정
- UI 구현 및 테스트는 마지막에 구현

# 추후 개발 스케줄

## 단계별 구현 계획 (Step-by-Step)

- 공개 법령 크롤러 완성 1~2일
- 판례 크롤러 + 로그인 자동화(Selenium) 2~3일
- DB 저장 및 중복 체크 통합 1일
- Django REST API 구축 (검색/조회) 1~2일
- 챗봇 프론트엔드 구현 (UI + AJAX/WS) 2~3일
- 통합 테스트 (크롤링 → DB → 챗봇) 1~2일
- 배포 및 최종 점검 1~2일

## 핵심 우선순위

- 공개법령 데이터 확보 → requests + BS4
- 판례 데이터 확보 → Selenium 로그인 자동화
- DB 저장/조회 가능 → 챗봇 테스트 가능 상태 확보
- UI/UX → 결과 확인, 사용자 편의 개선