
오픈소스소프트웨어개발

최종 보고서

팀 11	소프트웨어학부	이동호
	수학과	이준석
백색 태양의 일격	소프트웨어학부	장기현
	소프트웨어학부	최진우
	수학과	황정묵

프로젝트 개요

▶ 프로젝트 소개

구매대행

구매, 대량으로 같이해서 행복해요

▶ 프로젝트 목적

편의점 1+1, 2+1 행사 상품들을 공동 구매하여 원하는 양만, 값싸게 살 수 있도록 한다.

▶ 소프트웨어의 기능

회원가입 및 로그인 기능

회원 가입과 로그인을 통해 소프트웨어에 접근.

주변 편의점 확인

지도 API와 현재 위치, 반경을 통해 주변 편의점을 확인.

행사 물품 목록 확인

크롤링을 통해 편의점의 행사 물품 목록(1+1, 2+1) 확인.

챗봇 시스템

행사 목록을 기반으로 저녁 메뉴, 꿀조합 추천 등이 가능.

포스트 기능

챗봇을 통한 꿀조합이 아닌 사용자의 꿀조합 추천이 가능.

회원 이름으로 QR 생성

공동 구매 등 회원 간 만남 시 QR 확인으로 식별 가능.

사용자 간 채팅

로그인한 사용자 간의 채팅으로 공동 구매 추진 가능.

▶ 설치 방법

1. Python 환경 설치
2. 실행에 필요한 모듈 설치

```
pip install -r requirements.txt
```

채팅을 위해 docker 설치, 그 후 docker 실행

```
docker run -p 6379:6379 -d redis:5
```

3. crawling 폴더에서 크롤링 실시

```
cd ./crawling # crawling 폴더로 이동
python crawling_CU.py
python crawling_GS.py
python crawling_SE.py
```

모두 실행되면 3개의 CSV 파일이 생성된다. 이를 토대로 이미지를 다운받는다.

```
python CU_Images.py
python GS_Images.py
python SE_Images.py
```

4. 다시 project 폴더로 이동 후 서버 실행

```
cd .. # project 폴더로 이동
python manage.py runserver
```

5. 실행 시 나타나는 주소로 이동

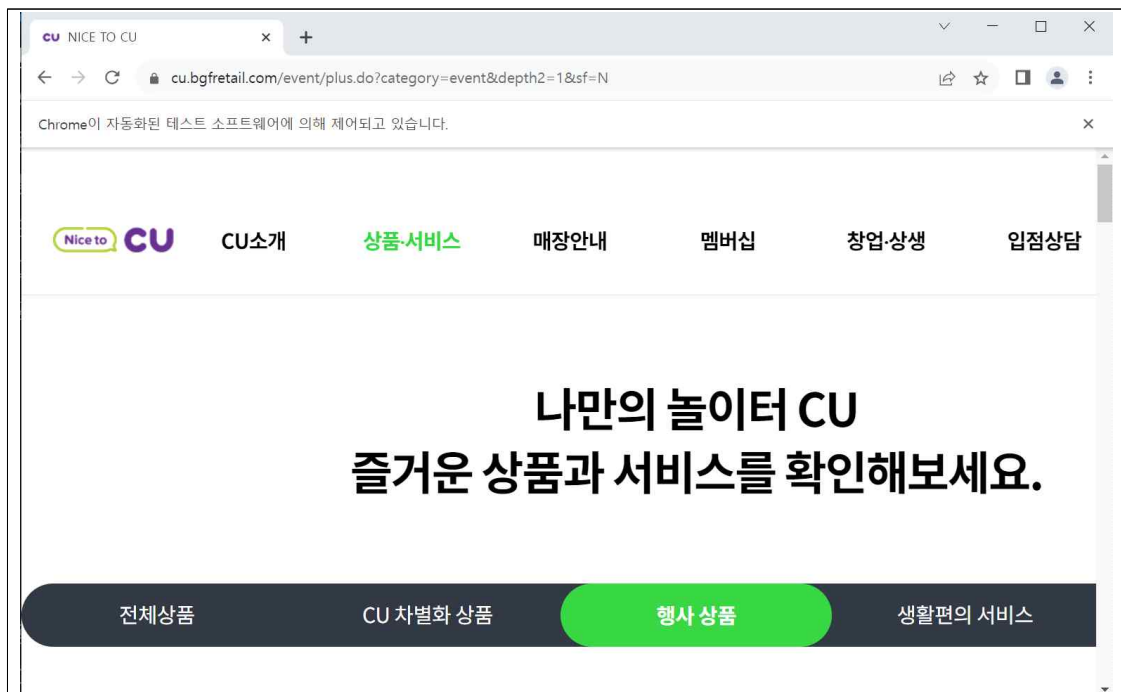
<http://127.0.0.1:8000/>

▶ 사용방법 및 예시

[사전작업_크롤링]

크롤링 예시로 CU 행사 상품 목록을 크롤링한다고 하자.

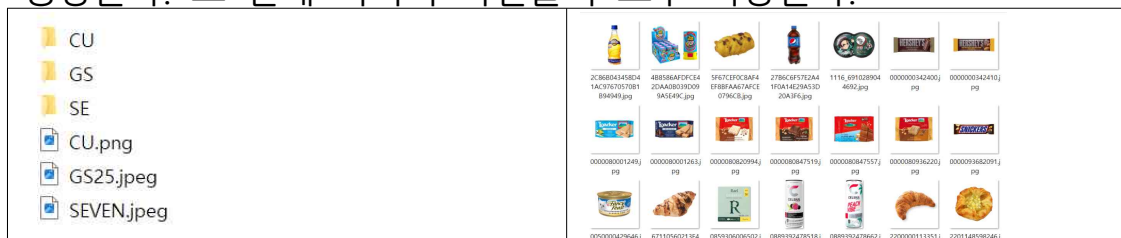
```
python crawling_CU.py
```



사진과 같이 chrome 창이 뜨면서 크롤링이 시작된다.
해당 창이 종료되면 크롤링이 끝났다는 사실을 알 수 있다.
그 후에는 이미지 파일을 다운로드 받는다.

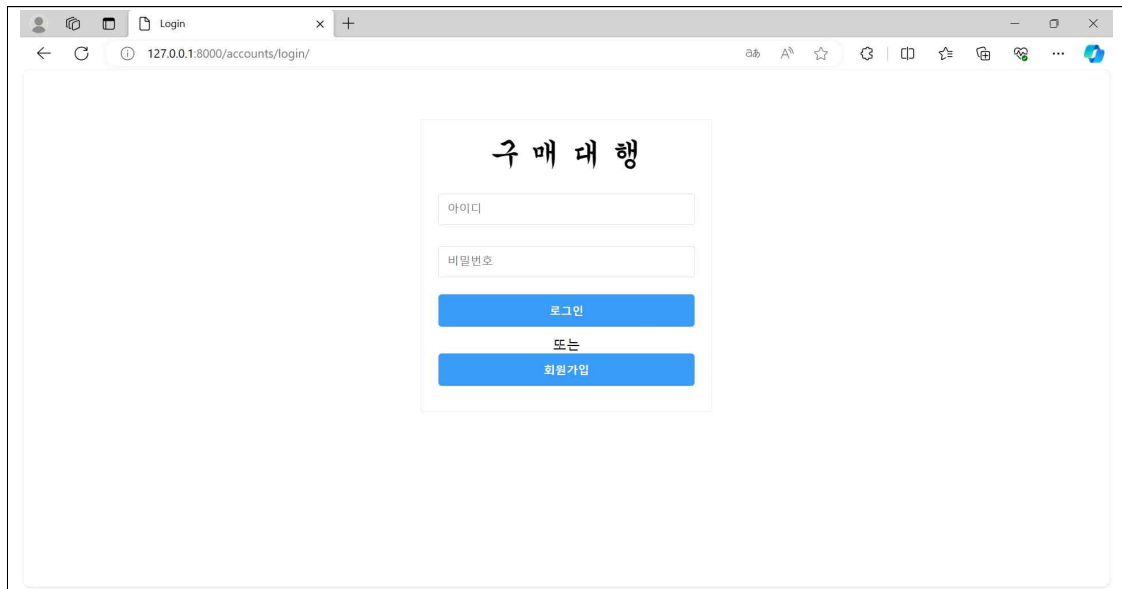
```
python CU_Images.py
```

이미지 다운로드가 완료되면 picture 폴더 안에 CU 폴더가
생성된다. 그 안에 이미지 파일들이 모두 저장된다.



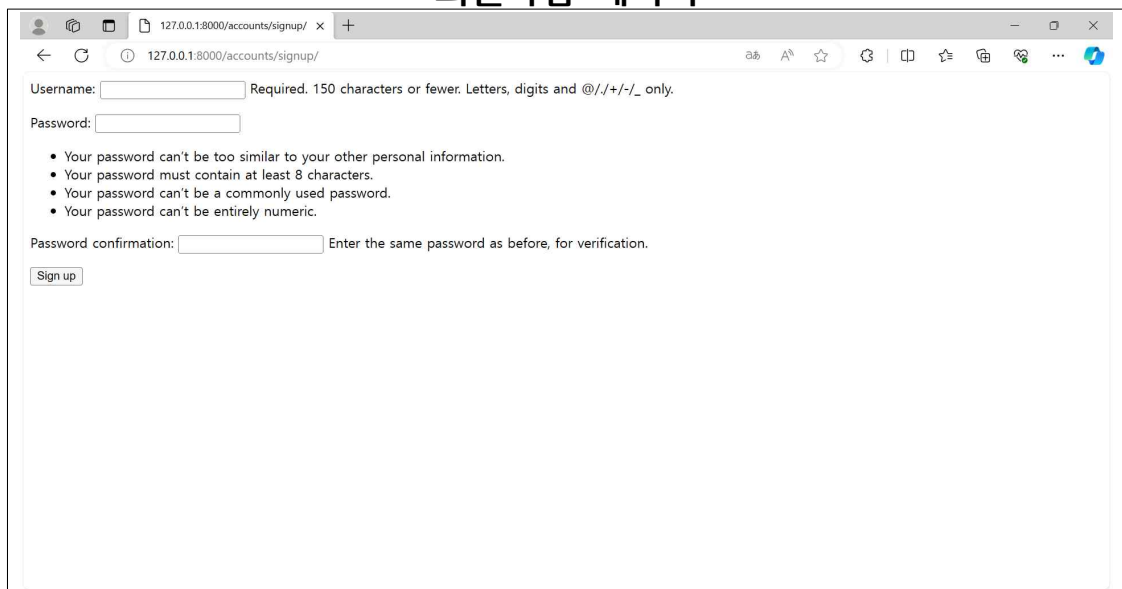
이미지는 모두 ALT 태그 값으로 저장된다.

[실행 시의 화면]



실행 시 로그인이 되어있지 않으면 보이는 첫 화면이다.
아이디와 비밀번호 입력 후 로그인 버튼으로 로그인이 가능하다.
회원가입 버튼을 눌러 회원가입 페이지로 이동할 수 있다.

회원가입 페이지



회원가입은 DJANGO의 UserCreationForm을 사용하였다.
규칙에 맞는 아이디와 비밀번호로 계정을 생성할 수 있다.

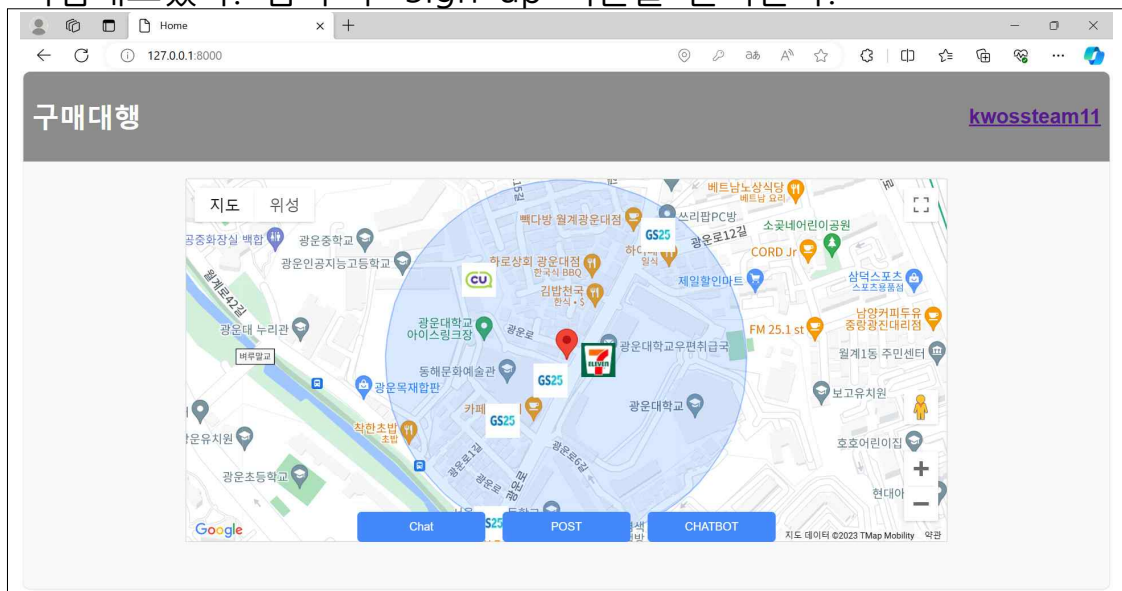
Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

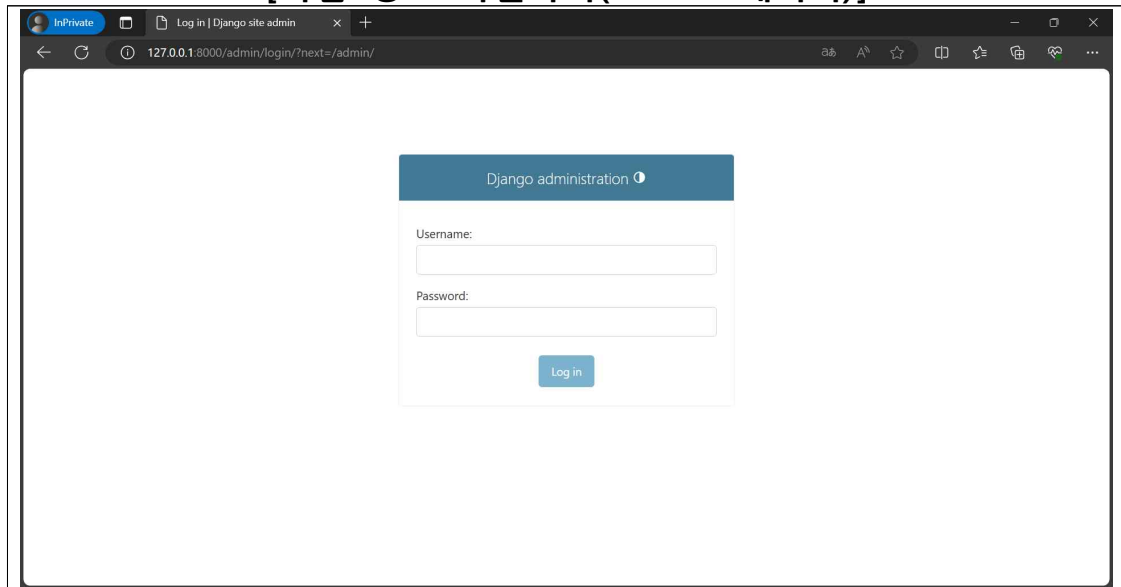
Password confirmation: Enter the same password as before, for verification.

예시로 'kwossteam11' 라는 아이디를 비밀번호 'rkskekfk1'로 가입해보겠다. 입력 후 Sign up 버튼을 클릭한다.

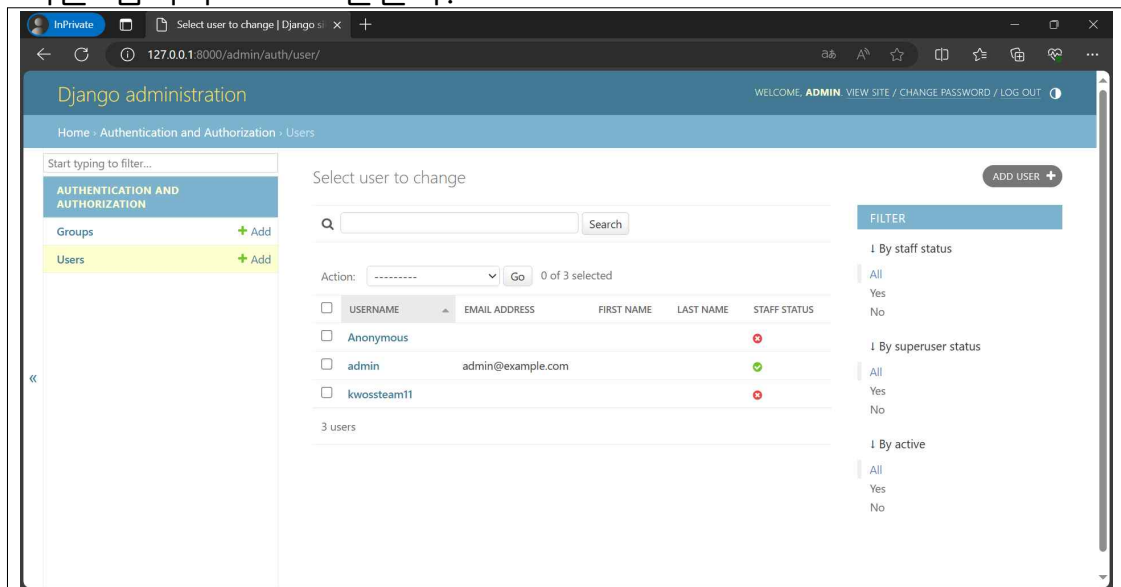


가입이 완료되면 자동적으로 로그인되며 홈 화면으로 이동한다. 가입 정보는 다음과 같이 확인할 수 있다.

[가입 정보 확인하기(admin 페이지)]

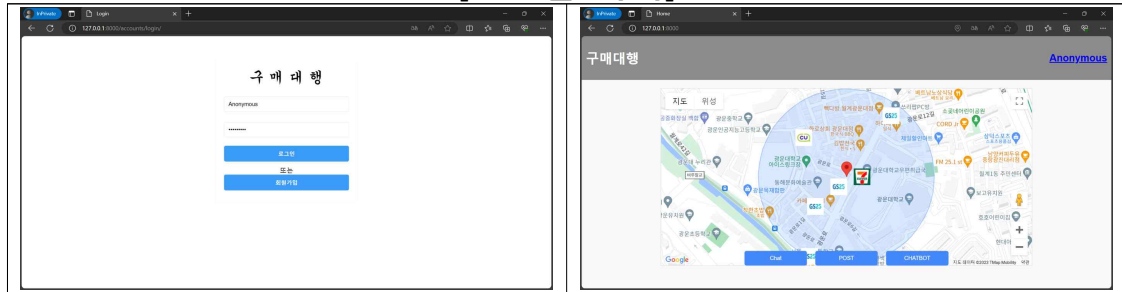


우선 127.0.0.1:8000/admin으로 이동한다. 현 프로젝트는 admin 계정이 admin 아이디에 admin 비밀번호로 되어 있다. 이를 입력하고 로그인한다.



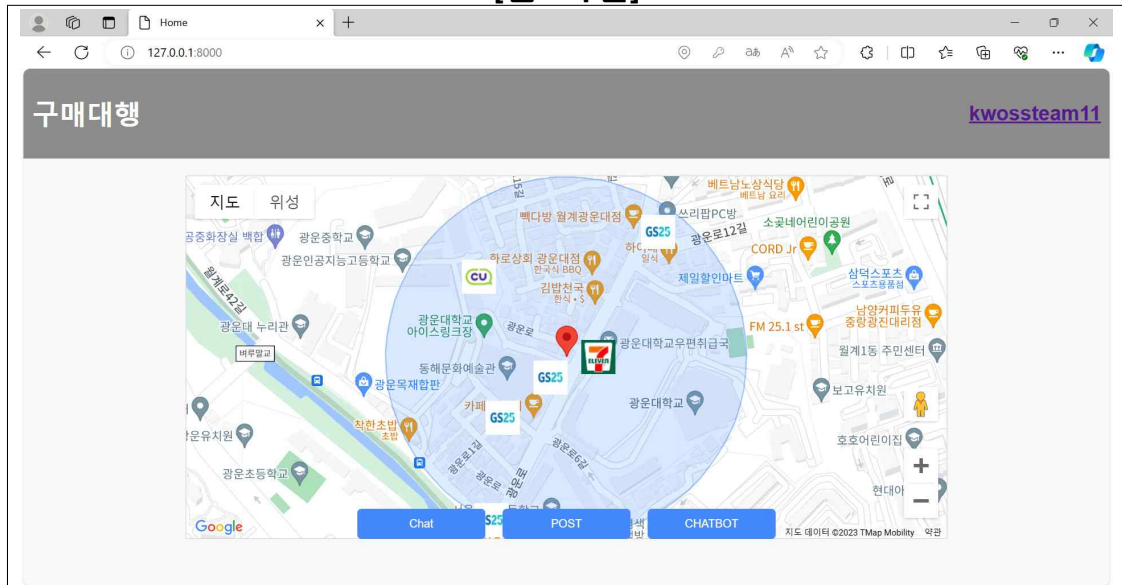
이동 후 Users를 누르면 kwossteam11의 존재를 볼 수 있다. admin 계정 외에 테스트로 쓴 Anonymous 계정 또한 보인다.

[로그인 하기]



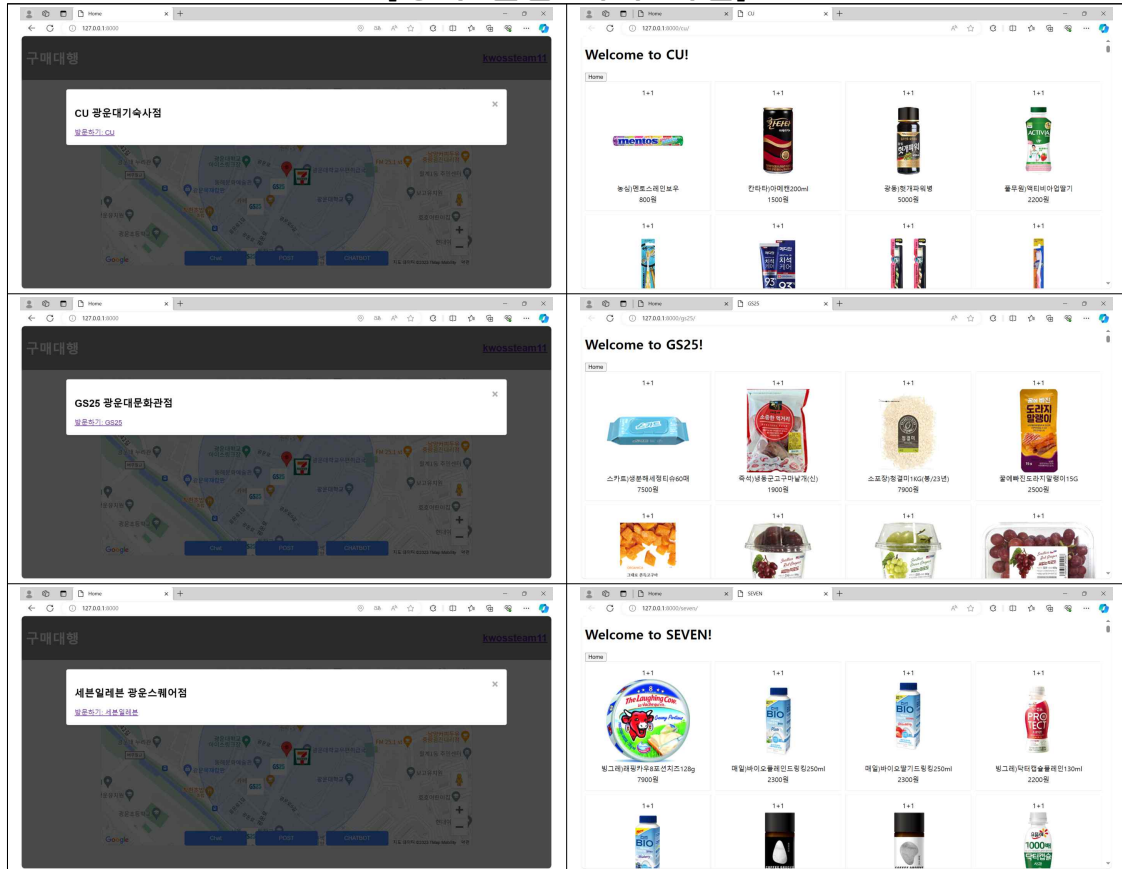
새 브라우저를 열어 다시 로그인을 진행해보겠다.
이번에 사용하는 계정은 테스트로 썼던 Anonymous 계정이다.
아이디와 비밀번호를 입력 후 로그인 버튼을 누르면
해당 계정이 존재하는 경우 홈 화면으로 이동된다.

[홈 화면]



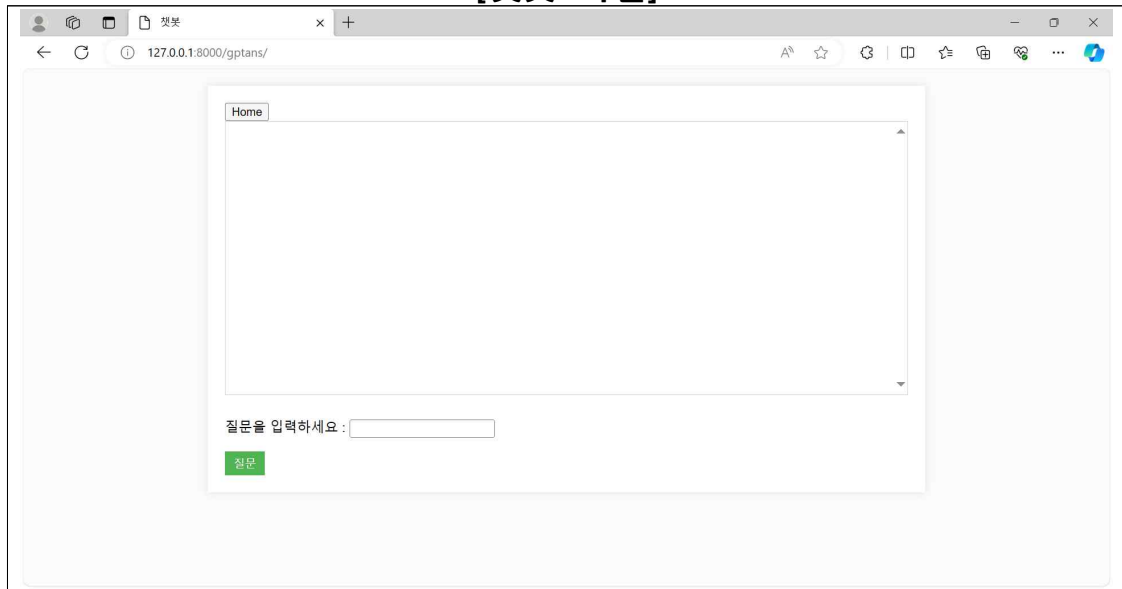
로그인 시 보이는 홈 화면은 사진과 같다. 로그인한 ID가 우측 상단에 표시되고, 현재 위치 기준으로 200m 반경의 편의점이 이미지로 뜬다. 지도 중앙 하단부에는 다른 기능으로 이동하는 버튼이 위치한다.

[행사 물품 목록 확인]

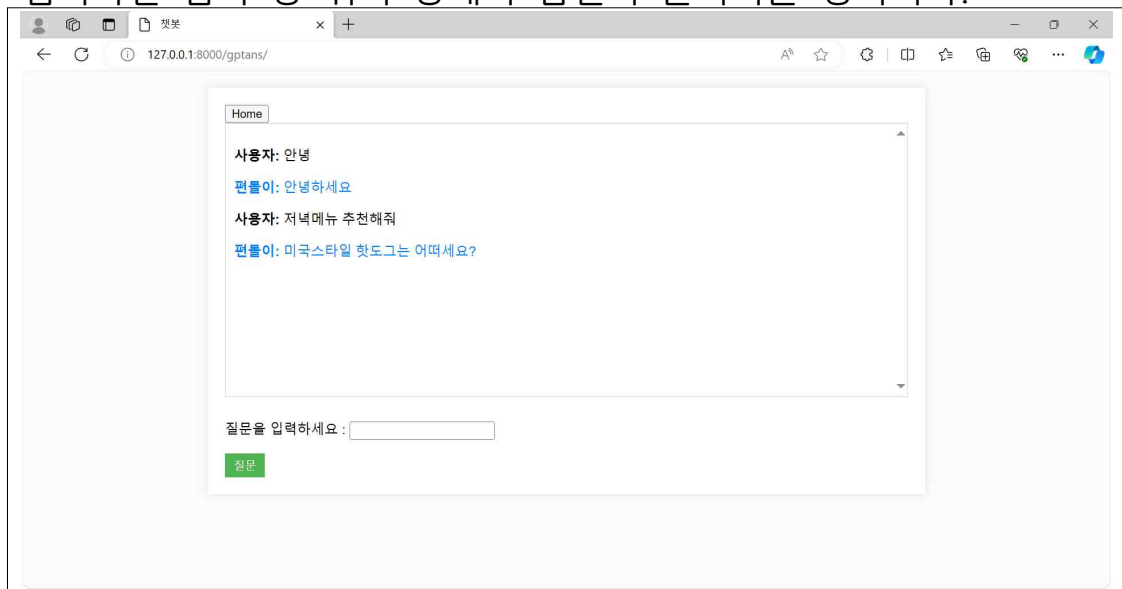


이미지를 누르면 해당 편의점의 이름과 링크가 나온다.
 링크를 클릭하면 우측의 이미지와 같이 해당 편의점의
 행사목록 페이지로 이동한다. 행사목록은 상단부터 행사분류,
 사진, 제품명, 가격으로 출력된다. 이는 CU, GS25, 7Eleven
 각각의 행사목록에 맞게 출력된다. 행사목록 페이지는
 새 창에서 출력되며, 좌측 상단의 Home 버튼으로 홈 화면으로
 이동할 수 있다.

[챗봇 화면]

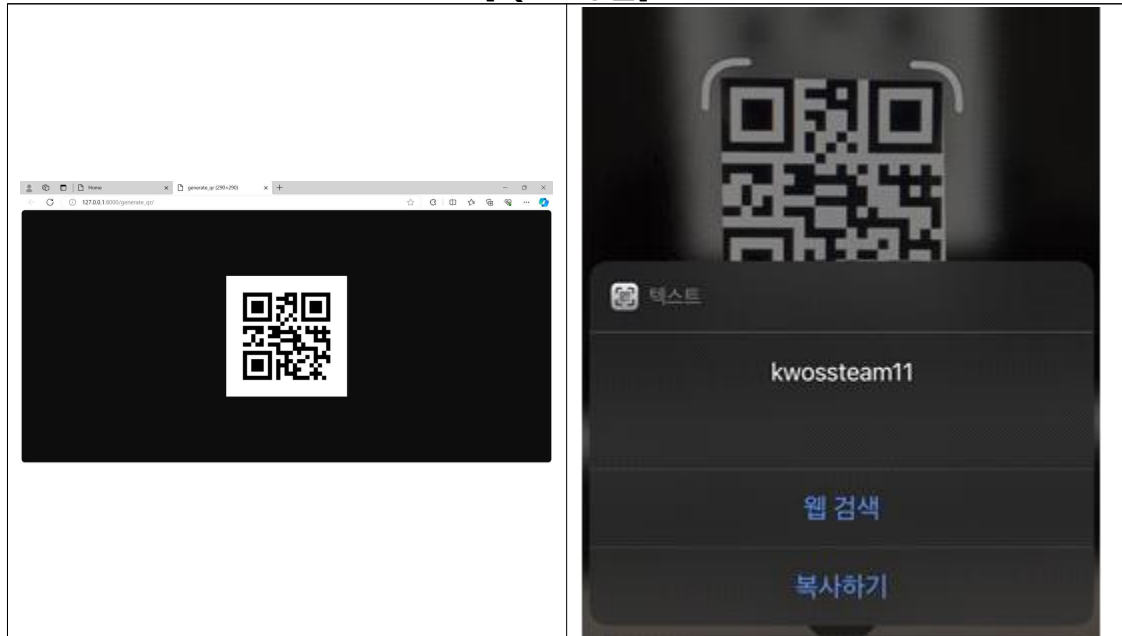


홈 화면에서 CHATBOT 버튼을 누르면 챗봇 페이지로 넘어간다. 챗봇 페이지는 사진과 같으며 하단의 질문 입력 창에 질문을 입력하면 입력 창 위의 창에서 답변이 출력되는 형식이다.



예시로 ‘안녕’이라고 질문을 했다. 챗봇에서 ‘안녕하세요’ 라고 답변이 온 점을 알 수 있다. 다른 질문으로 ‘저녁메뉴 추천해줘’ 라고 물어본 결과 ‘미국 스타일 핫도그’를 추천한 점을 알 수 있다. 챗봇의 데이터는 크롤링된 행사목록을 기반으로 하고 있다.

[QR 화면]

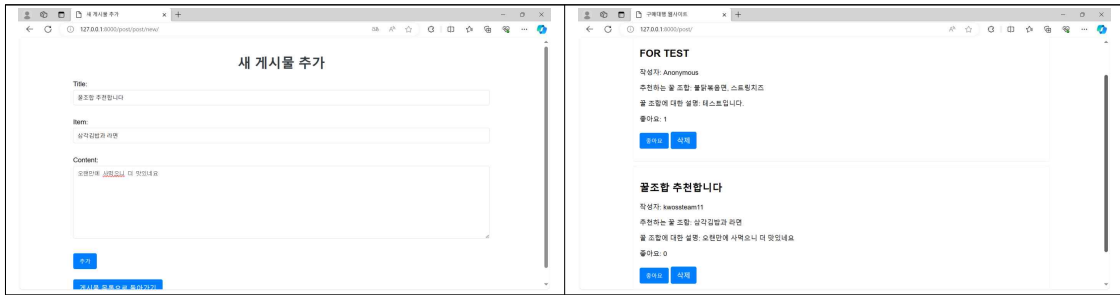


홈 화면에서 로그인한 사용자(현재 kwossteam11)를 클릭하면 좌측 사진과 같이 QR 코드가 뜬다. 이 QR 코드를 핸드폰으로 읽어보면 우측 사진처럼 'kwossteam11' 라고 뜬다. 이를 통해 회원 간 만남 시 서로 확인이 가능해진다.

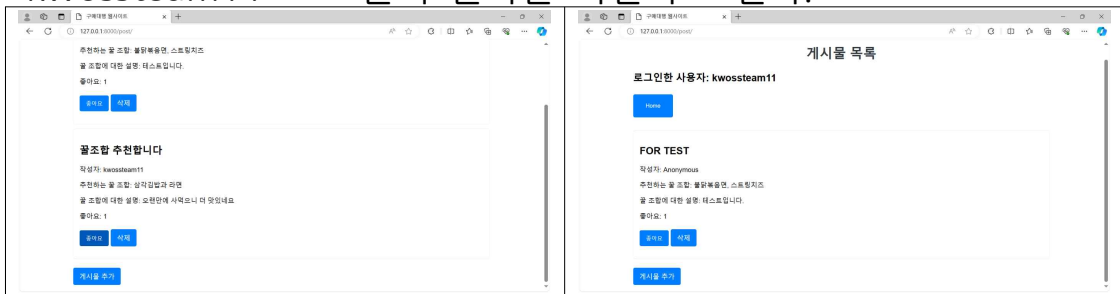
[포스트 기능]



홈 화면에서 POST 버튼을 누르면 포스트 페이지로 넘어간다. 포스트 페이지는 사진과 같으며 현재 테스트 계정으로 올린 포스트가 1개 올라가있는 상태이다.



포스트 화면 하단의 게시물 추가 버튼을 누르면 좌측 사진같이 제목, 추천 조합, 설명을 입력하는 창이 뜬다. 예시로 제목을 ‘꿀조합 추천합니다’, 추천 조합을 ‘삼각김밥과 라면’, 그리고 설명을 ‘오랜만에 사먹으니 맛있네요’ 라고 입력하고 추가 버튼을 누른다. 우측 사진을 보면 해당 내용으로, 그리고 작성자 이름 ‘kwossteam11’으로 글이 올라간 사실이 보인다.

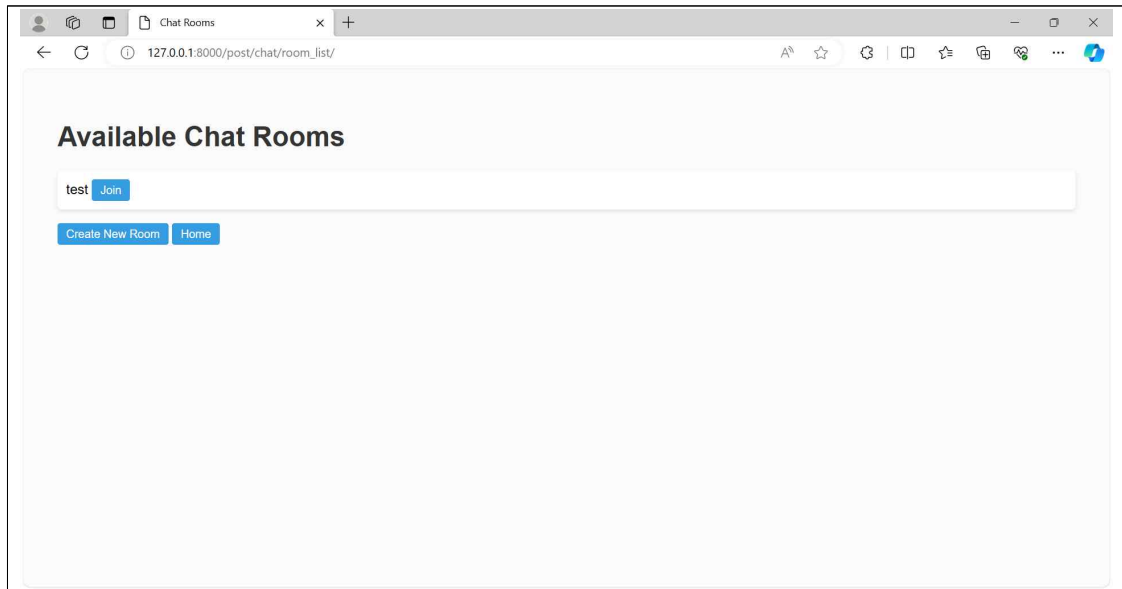


포스트마다 좋아요와 삭제 버튼이 존재한다. 위에서 예시로 만든 포스트의 좋아요를 누르면 해당 포스트의 좋아요 수가 올라간다. 삭제 버튼을 누르면 누른 사용자가 글을 쓴 사용자와 일치하는지 여부를 확인한 후 일치하면 글을 삭제한다. 좌측 사진을 통해 좋아요의 증가를, 우측 사진을 통해 포스트 삭제를 알 수 있다.

[사용자 간 채팅 기능]

<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	zealous. 6a82242a	redis:5	Running	N/A	6379:6379	0 seconds a	■ ⋮ 🗑

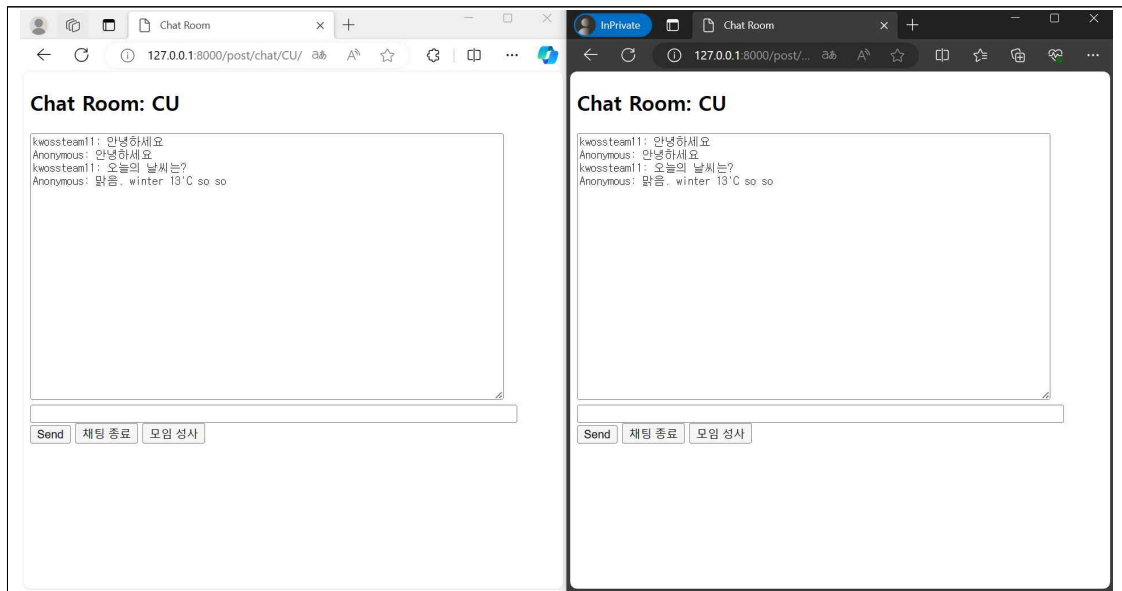
채팅 기능에 앞서 docker를 통해 redis 서버가 running 상태임을 알 수 있어야 한다. 사진은 docker desktop에서 redis 서버가 running 상태임을 보여주고 있다.



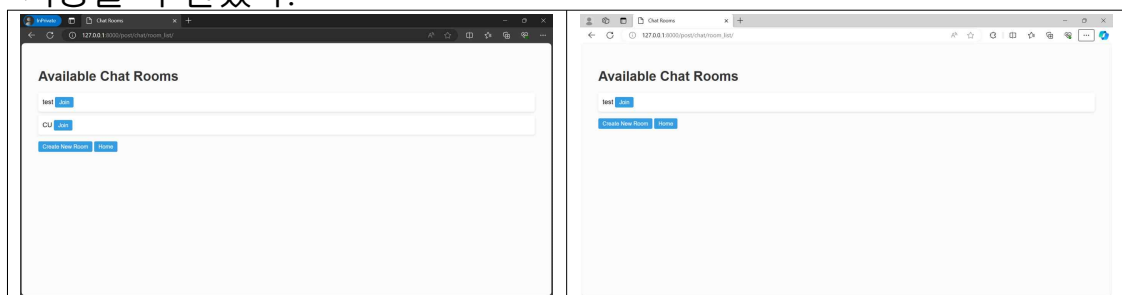
홈 화면의 Chat을 클릭하면 chat의 room_list 페이지로 넘어간다. 사진과 같이 현재 활성화된 채팅방들을 확인할 수 있다. 현재는 테스트 용도로 만든 test 채팅방만 남아있는 상태이다.



화면의 Create New Room을 누르면 좌측 사진같이 채팅방을 만들 수 있는 페이지로 이동한다. 예시로 이름을 CU라고 한다. 입력 후 create room 버튼을 클릭하면 room_list 페이지로 이동하며 가운데 사진같이 CU 채팅방의 생성을 알 수 있다. 채팅방 이름 옆의 Join 버튼을 누르면 우측 사진같이 해당 채팅방으로 이동하게 된다.



사진은 다른 브라우저(사진은 Edge의 시크릿모드)로 접속한 'Anonymous'와 'kwossteam11'이 CU 채팅방에서 채팅을 나누는 모습을 보여준다. 메시지를 입력 후 send 버튼을 누르면 어느 사용자가 어떤 메시지를 보냈는지 화면에 출력된다. 채팅방 하단에는 채팅 종료 버튼, 모임 성사 버튼이 존재한다. 채팅 종료 버튼을 누르면 채팅방은 유지되며 사용자만 나가진다. 반면 모임 성사 버튼을 누르면 해당 채팅방이 사라지게 된다. 위는 공동 구매를 위해 사용자가 모이게 되면 해당 방을 삭제하게 기능을 구현했다.



좌측 사진은 CU 채팅방에서 채팅 종료를 누른 모습을, 우측 사진은 CU 채팅방에서 모임 성사를 누른 모습을 보인다.

▶ 다음에 시도해볼 것

프로젝트 전체

디자인 개선, 배포

로그인 및 회원가입

비밀번호 찾기 기능

주변 편의점 확인

반경 넓히기, 더 많은 편의점 추가

행사 물품 목록 확인

상품 클릭 시 정보 출력, 크롤링 주기적 실행 등의 자동화

챗봇 시스템

자연어 처리를 활용한 챗봇 생성

포스트 기능

좋아요 순 정렬 기능, 댓글 기능

사용자 간 채팅

방 생성 시 물품 이름 등의 정보 도입

▶ 참조 문서

[로그인 및 회원가입, 포스트 기능]

<https://docs.djangoproject.com/ko/4.2/>

[행사 물품 목록 확인]

<https://dudumandu0321.tistory.com/46>

[사용자 간 채팅]

<https://channels.readthedocs.io/en/latest/index.html>

[챗봇 시스템]

<https://needjarvis.tistory.com/639>

<https://hashdork.com/ko/create-a-deep-learning-chatbot-with-python/>

▶ 프로젝트 저장소 주소

<https://github.com/kw-oss-team11/project>

▶ 개인별 작업 내용

이동호	이준석	장기현	최진우	황정묵
<ul style="list-style-type: none">· 행사 물품 목록 확인· 회원 이름의 QR 생성· 회원가입 및 로그인 기능	<ul style="list-style-type: none">· 사용자 간 채팅	<ul style="list-style-type: none">· 주변 편의점 확인	<ul style="list-style-type: none">· 포스트 기능	<ul style="list-style-type: none">· 챗봇 시스템