## backend 관통 프로젝트

#### 목록

- 1. 기능 요구사항
- 2. Usecase Diagram
- 3. Class Diagram
- 4. 실행화면 캡쳐
  - a. 메인화면
  - b. 아파트 실거래가 검색
  - c. 회원 관리
  - d. 관심지역
  - e. 주변 업종/환경 정보조회
  - f. 공지사항

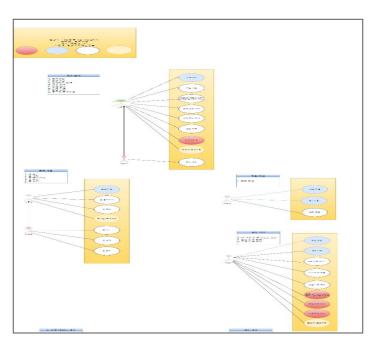
#### 기능 요구사항

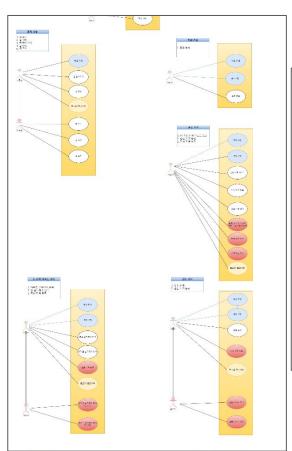
- 1. 메인화면
- 2. 아파트 실거래가 검색
  - a. 아파트별 실거래가 검색
  - b. 동별 실거래가 검색
- 3. 회원관리
  - a. 로그인/로그아웃
  - b. 회원 가입 / 회원 수정 / 회원 탈퇴
  - c. 비밀번호 찾기/변경
- 4. 관심 지역 설정
  - a. 관심지역 등록/삭제
  - b. 관심 지역 조회
- 5. 동네업종/오염정보
  - a. 지역별 상권 조회
  - b. 지역별 오염 정보 조회
- 6. 공지사항
  - a. 공지사항 등록, 수정,삭 제,목록조회,상세조회

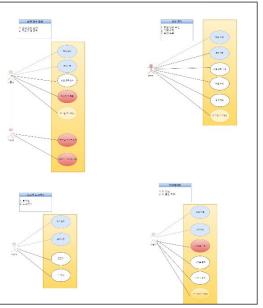
## 기능 요구사항

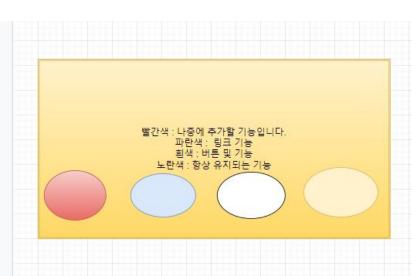
1. 메뉴 구성	메인화면	
	메뉴구성	
2. 아파트 실거래가 조회	동별 아파트 실거래가 조회	
	아파트별 실거래가 검색	
3. 회원 관리 / 로그인 관리	로그인 / 로그아웃 / 비밀번호 찾기	
	회원 가입 / 정보 수정 / 회원 탈퇴	
4. 관심지역 정보 관리	관심지역 등록 / 삭제	
	관심 지역 조회	
5. 주변 상권 정보, 오염 정보 제공	지역별 상권 조회	
	지역별 오염 정보 조회	
6. 공지사항 관리	공지사항 등록, 수정, 삭제, 검색	

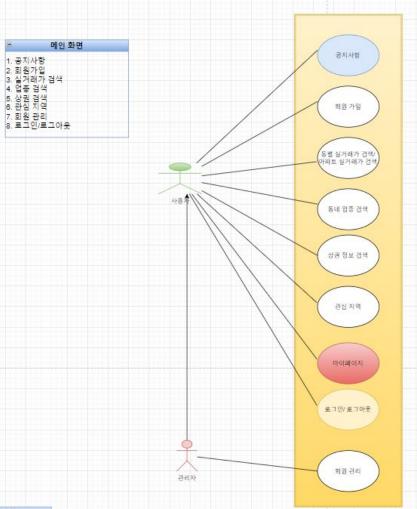
**Usecase Diagram** 

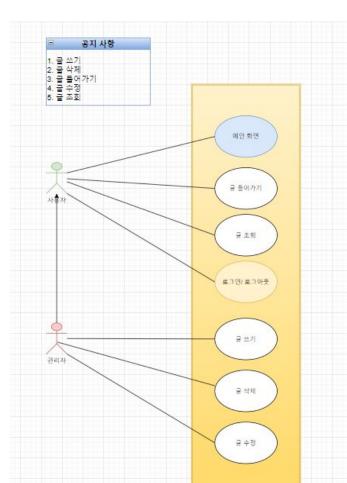


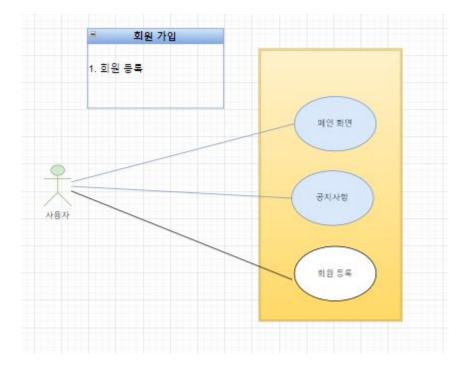


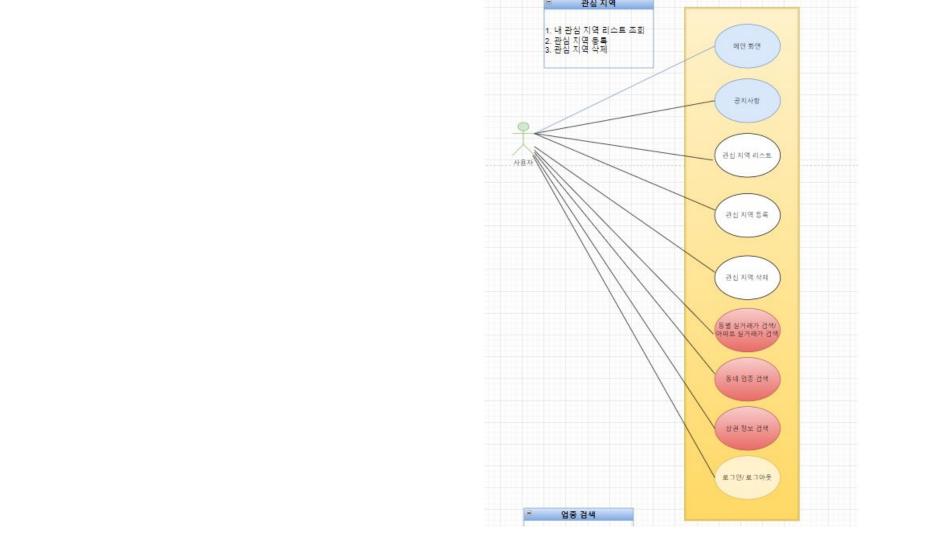


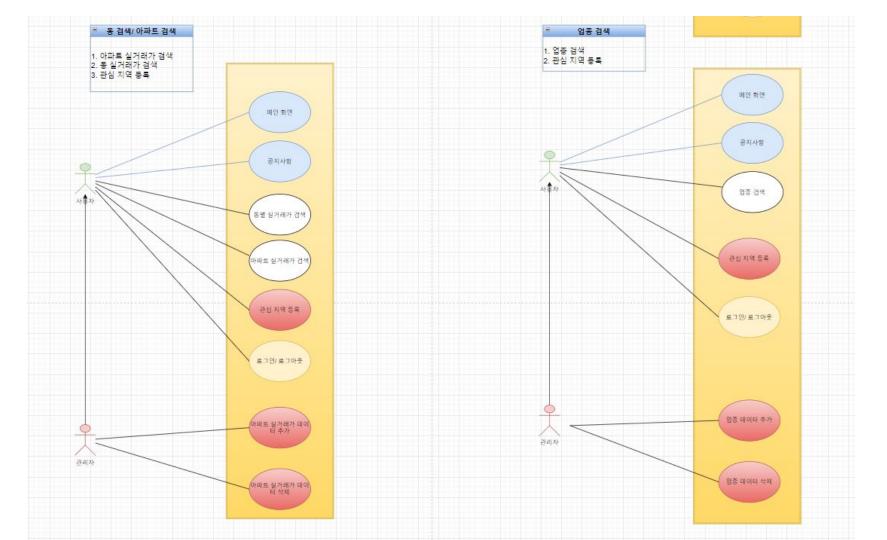


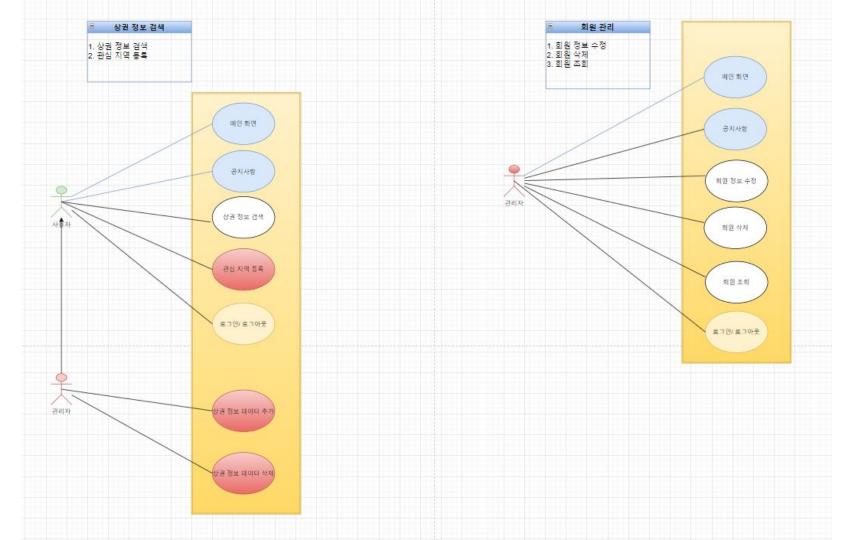


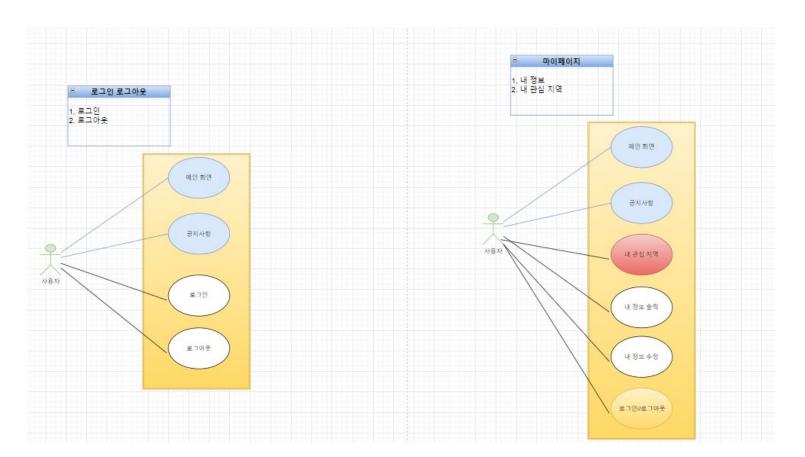








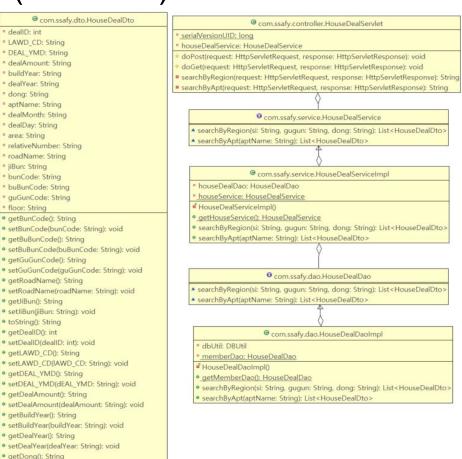




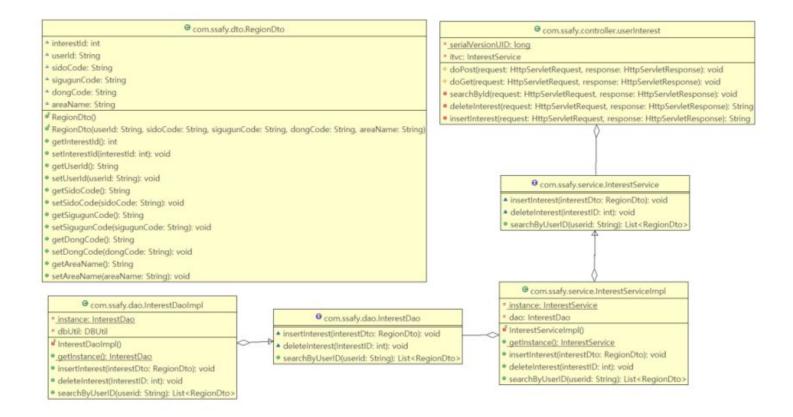
클래스 다이아그램

## 1. HouseDeal (실거래가)

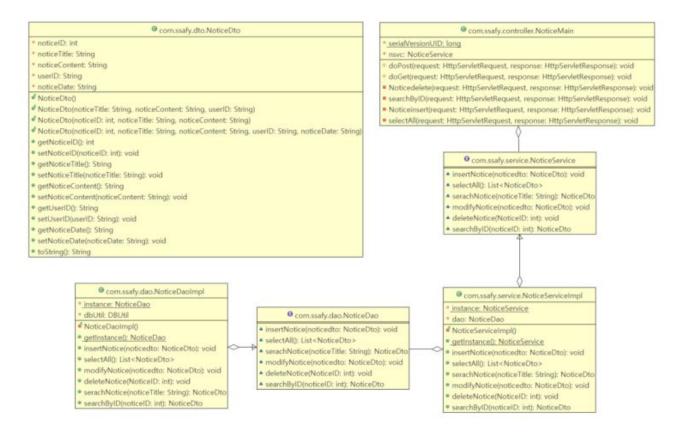
setDong(dong: String): voidgetAptName(): String



## 2. Interest (관심지역)

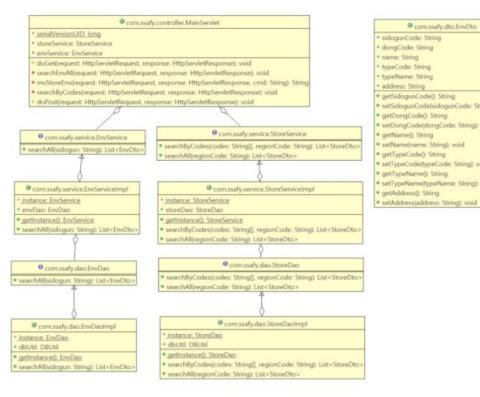


## 3. Notice (공지사항)



## 4. StoreEnv(상권&환경)





@com.ssafy.dto.EnvDto sidogunCode: String dongCode: String name: String typeCode: String typeName: String address: String agetSidogunCode@: String setSidogunCode(sidogunCode: String): void getDongCode(): String setDongCode(dongCode: String): void getName(): String setName(name: String): void agetTypeCode(): String setTypeCode(typeCode: String): void getTypeName(): String setTypeName(typeName: String): void aetAddressil: String

#### 5. User (회원정보)

#### @com.ssafv.dto.UserDto userld: String userPwd: String userEmail: String userName: String: userBirth: String userGender: String registDate: String manager: String userPhoneNum; String \* getUserGender(): String \* setUserGenderiuserGender: Stringt: void \* getUserId(): String \* setUserId(userId: String): void \* getUserPwd(): String \* setUserPwd(userPwd: String): void \* getUserEmail(): String \* setUserEmail(userEmail: String): void \* getLiserName(): String setUserName(userName: String); void · getUserBirth@: String setUserBirth(userBirth: String): void getRegistDate(): String \* setRegistDate(registDate: String): void getManager(): String \* setManager(manager; 5tring); void \* getUserPhoneNumű: String \* setUserPhoneNum(phoneNum: String): void \* toString(): String

# ■ serialVersion/LID: long \* serialVersion/LID: long \* userService: UserService ■ doPost/request: HttpServletRequest, response: HttpServletResponse): void ■ doGet/request: HttpServletRequest, response: HttpServletResponse): void ■ findPard(request: HttpServletRequest, response: HttpServletResponse): String ■ searchAllUser/request: HttpServletRequest, response: HttpServletResponse): String ■ updateUser/request: HttpServletRequest, response: HttpServletResponse): String ■ updatePard(request: HttpServletRequest, response: HttpServletResponse): String ■ deleteUser/request: HttpServletRequest, response: HttpServletResponse): String ■ idCheck/request: HttpServletRequest, response: HttpServletResponse): int ■ registerMember/request: HttpServletRequest, response: HttpServletResponse): String ■ logon/User/request: HttpServletRequest, response: HttpServletResponse): String ■ logon/User/request: HttpServletRequest, response: HttpServletResponse): String ■ ogen/User/request: HttpServletRequest, response: HttpServletResponse): String ■ ogen/User/request: HttpServletRequest, response: HttpServletResponse): String

#### O com ssafy day UserDay \* idChecklid: String): int. \* register(userDto: UserDto); void ▲ login(id: String, pwd: String): UserDto deleteUser(user(d: String): void \* updateInfoluserDto: UserDto': void updatePwd(userld: String, userPwd: String): void searchByld(userld: String): UserDto searchPwdBvid(userid: String): String searchAll(listParameterDto: ListParameterDto): List<UserDto> getTotalCount(listParameterDto: ListParameterDto); int com.ssafv.dao.UserDao(mpl. dbUtil: DBUtil userDao: UserDao ■ UserDao(mpl()) a getUserDao0: UserDao

searchAll(listParameterDto: ListParameterDto): List<UserDto>
 getTotalCount(listParameterDto: ListParameterDto): int

idClvcklid: Stringt: int
 register(userInfoDta: UserDta): vaid
 updatePwd(userId: String, userPwd: String): vaid

deleteUser(userid: String): void
 searchByld(userinfold: String): UserDto
 searchPwdByld(userid: String): String

login(id: String, pass: String): UserDto
 updateInfo(userDto: UserDto): void

#### com.ssafy.service.UserService \* idCheck(id: String): int \* registerMember(memberDto; UserDto); void login(id: String, pass: String): UserDto ▲ updateMember(memberDto: UserDto): void updatePwd(userld: String, userPwd: String); void \* deleteMember(id: String): void infoMember(id: String): UserDto userList(pg: String, key: String, word: String): List<UserDto> makePageNavigation(pg: String, key: String, word: String): PageNavigation searchPwdById(id: String): String @ comssafy.service.UserServiceImpl userDao: UserDao " userService: UserService d UserServiceImpl() getUserService(): UserService \* idChecklid: Stringl: int. # registerMember(userDta: UserDta): void \* loginfid: String, pass; String): UserDto

makePageNavigation(pg: String, key: String, word: String): PageNavigation

userList(pg: String, key: String, word: String): List < UserDto >

updateMember(userDto: UserDtol: void.

= deleteMember(id: String); void

\* infoMember(id: String); UserDto

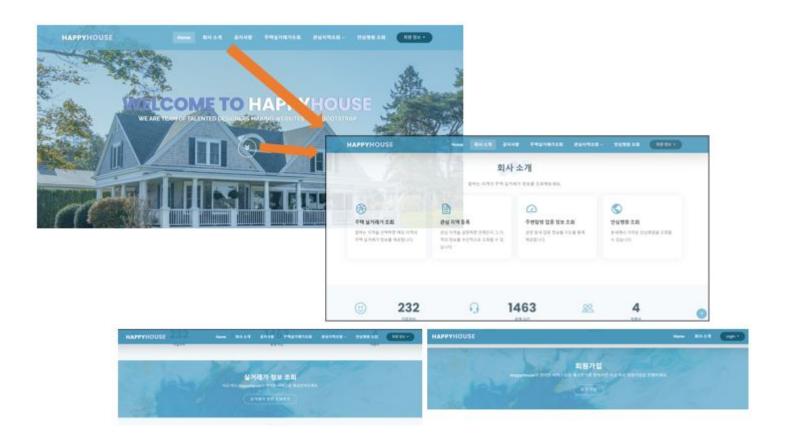
searchPwdByld(id: String): String

updatePwd(userid: String, userPwd: String): void

실행화면 캡처

메인화면

#### 1. 메인화면

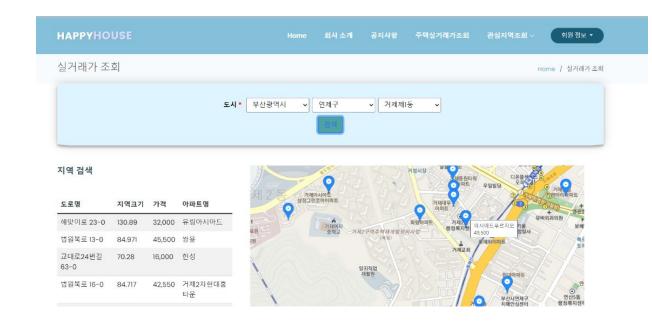


## 1. 메인화면



아파트 실거래가 검색

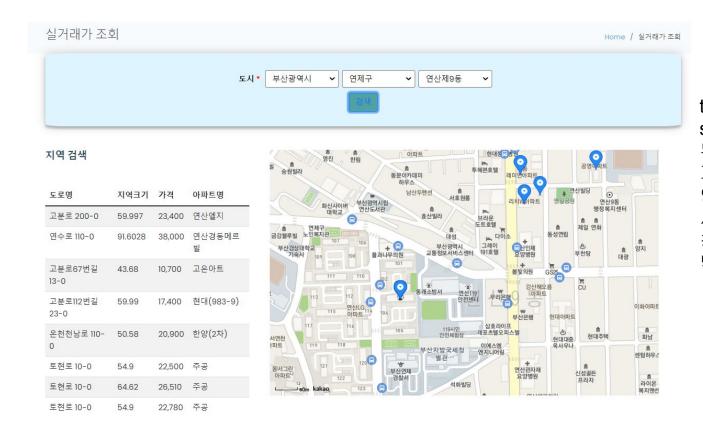
#### 2. 아파트실거래가 동으로 검색



아파트 실거래가 에 시, 구군, 동 정보를 선택하면 대분류된 동 단위로 아파트 정보 리스트와 지도에 해당 아파트의 위치 정보가 나옵니다.

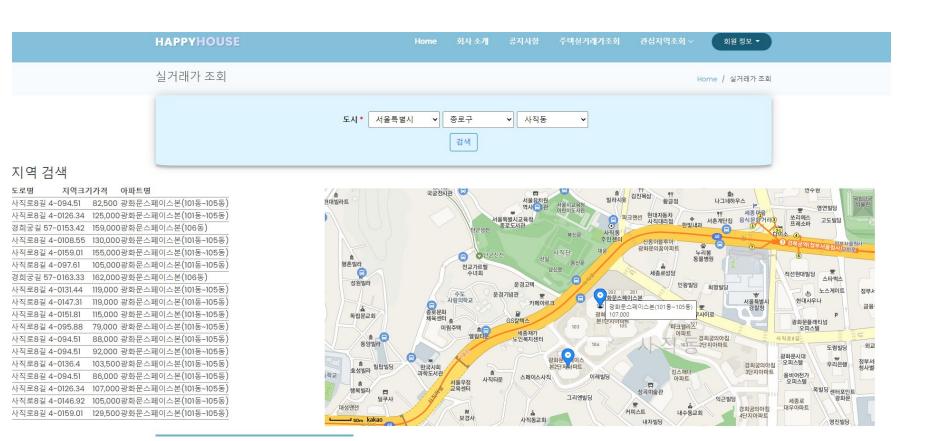
커서를 해당 marker에 올리면 해당 아파트 이름과 실거래가가 나타납니다.

#### 2. 아파트실거래가 동으로 검색

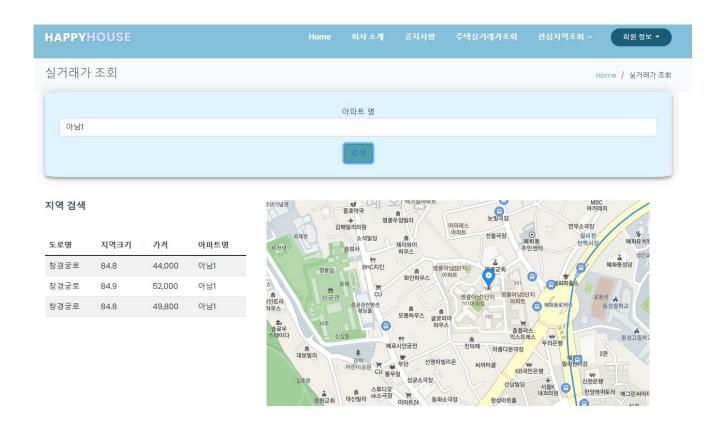


test시 참고사항. sql export 파일에는 부산광역시 연제구 거제동, 부산광역시 연제구 연산동 데이터, 서울시 종로구 사직동, 창신동, 숭인동 데이터를 넣어 놓았습니다.

#### 2. 아파트 실거래가 동으로 조회



#### 2. 아파트실거래가 아파트 주소로 조회



아파트 이름을 입력하면 db에서 해당 아파트 정보를 조회해서 지도와 실거래가 리스트를 보여줍니다.

커서를 해당 marker에 올리면 해당 아파트 이름과 실 거래가가 나타납니다.

#### 2. 아파트실거래가 아파트 주소로 조회



#### 지역검색

	지역크	기가격	아파트명
로214번길	84.62	30,500	경남
	83.08	22,100	경남
	73.82	20,900	경남
	83.08	19,500	경남
	로214번길	로214번길84.62 83.08 73.82	73.82 20,900



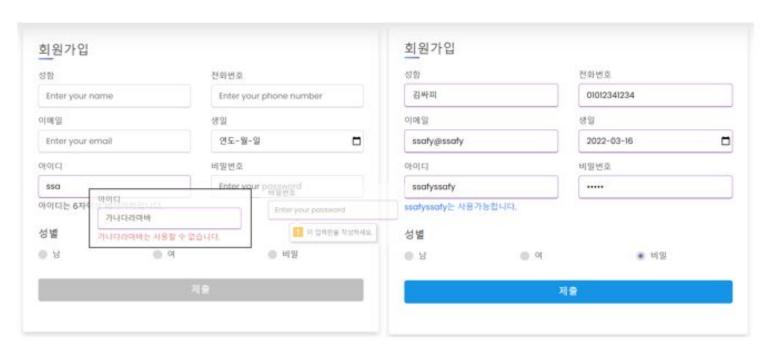
회원 관리 (/user)

1) 회원 등록



1) 회원 등록

기존에 있는 아이디와 중복되거나 글자 수에 맞지 않으면 제출 버튼이 비활성화 됩니다.

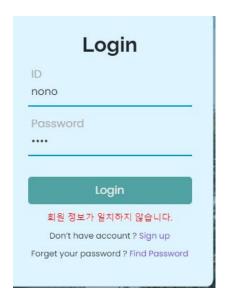


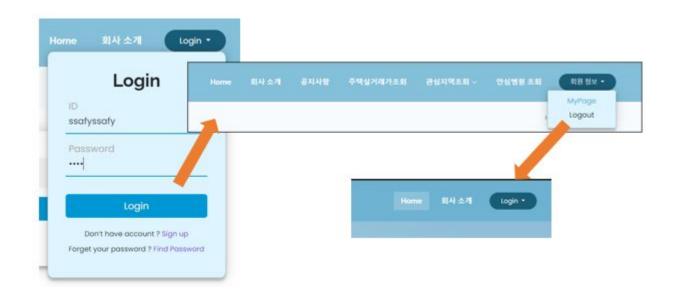
2) 비밀번호 찾기





3) 로그인/로그아웃 DB에 해당 회원이 존재하지 않으면, 메시지가 뜹니다. 로그인에 성공한다면, 서비스들 버튼이 활성화 됩니다.

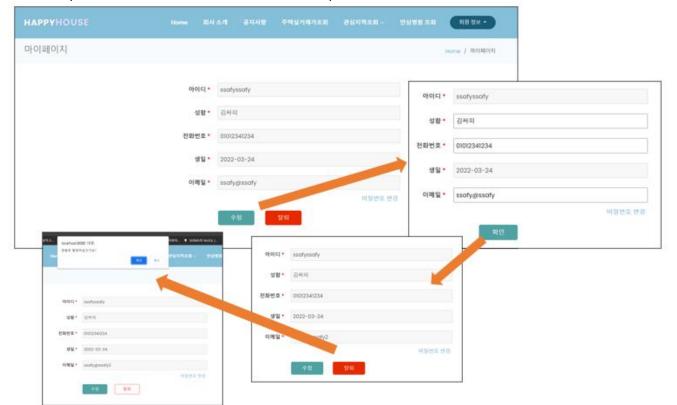




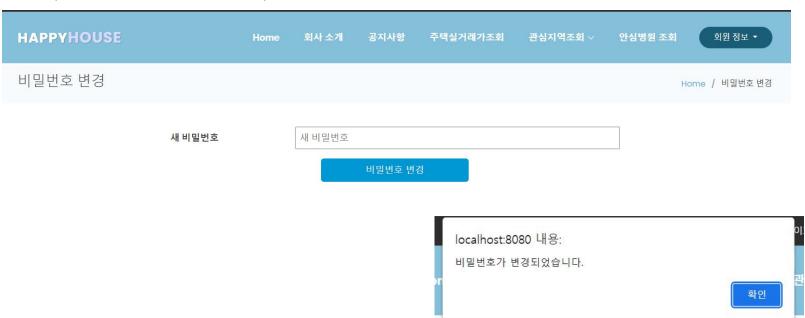
4. 마이페이지 해당 ID의 회원 정보를 조회합니다.



4) 마이페이지 회원정보(이름, 전화번호, 이메일)를 수정할 수 있습니다. 탈퇴 버튼을 클릭하면, 사용자에게 한번 더 확인한 후, 회원 정보를 삭제합니다.



5. 비밀번호 찾기 해당 ID (세션에 저장되어 있는 ID)의 비밀번호를 변경합니다.



## 관심지역 (/main)

## 4. 관심지역 설정

1) [관심 지역 조회] 메뉴를 통해 사용자의 관심 지역 리스트를 확인할 수 있습니다.



## 4. 관심지역 설정

2) [관심 지역 등록] 메뉴를 통해 관심 지역 등록합니다.



# 업종/환경 정보조회(/main)

업종/환경 정보조회(/main)

[사용 데이터]

- 1. 상권(업종) 정보 <u>소상공인시장진흥공단 상가(상권)정보</u> (강원도 지역에 한정하여 구현하였습니다.)
- 2. 환경 정보 서울시 종로구 환경 지도 점검 내역 (서울시 종로구 지역에 한정하여 구현하였습니다.)

- 백엔드는 구축한 상태이기 때문에, 다른 지역으로 확장시 데이터만 추가하면 됩니다.

## 4. 동네 업종 / 오염 조회 - 페이지 이동을 위한 인터페이스

1) -1 [관심지역조회] 하위의 [주변탐방 업종 정보 조회] 와 [주변환경정보 조회] 를 통해 지역의 상권과 환경 정보를 확인할 수 있습니다.

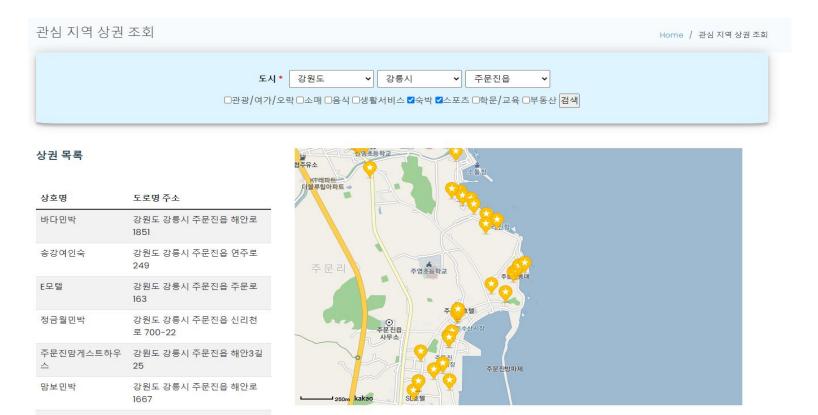


1) -2 관심지역 리스트에서 상권정보 / 환경 정보 메뉴를 통해 해당 지역의 상권과 환경 정보를 확인할 수 있습니다.



## 4. 동네 업종 / 오염 조회 - 상권 조회 페이지

2) 관심 지역 상권 조회 페이지에서 지역과 업종을 선택하면 해당 지역의 선택 업종 리스트가 출력되고 지도에 표시됩니다.



## 4. 동네 업종 / 오염 조회 - 환경 정보 조회 페이지

3) 환경 정보 조회 페이지에서 지역을 선택하면 해당 지역의 환경시설 정보가 출력되고 지도에 표시됩니다.



#### 환경 목록

업체 <b>/</b> 기관명	종류
혜화카프라자	폐수배출업소관리
혜화카프라자	폐수배출업소관리
삼성세차장	폐수배출업소관리
세란병원 종합검진센터	폐수배출업소관리
세란병원	폐수배출업소관리
(주)지에스이앤알 평창주유소	폐수배출업소관리
상명대학교	폐수배출업소관리
국군서울지구병원	폐수배출업소관리
서울적십자 병원	폐수배출업소관리
서울대학교 병원	폐수배출업소관리



공지사항

## 6. 공지사항 / 게시글 목록

1. 공지사항 메뉴를 클릭하면 게시글 목록이 뜹니다

공지	사항			Home / 공지사항
번호	제목	내용	작성자	작성일
1	안뇽	첫게시글!!!!!	3	2022-03-29 22:33:07
30	사이트 이용 안내드립니다.	실거래가 조회가능	3	2022-03-30 23:53:30
				등록

## 6. 공지사항 / 게시글 등록

2. 게시글 등록 화면과 등록 후 공지 목록입니다.



## 6. 공지사항 / 게시글 확인

3. 제목을 누르면 게시글을 확인할 수 있습니다.

제목		
사이트 이용 안내드립니다.		
내용		
실거래가 조회가능		fi.
	수정 수저	확인

## 6. 공지사항 / 게시글 수정

5. 수정 버튼을 누르면 제목과 내용을 수정할 수 있습니다. // 수정 후 바뀐 목록 확인(good -> bad)



## 6. 공지사항 / 게시글 삭제

6. 삭제버튼을 누르면 글을 삭제할 수 있습니다. // 삭제 후 사라진 cheeese cake 글

