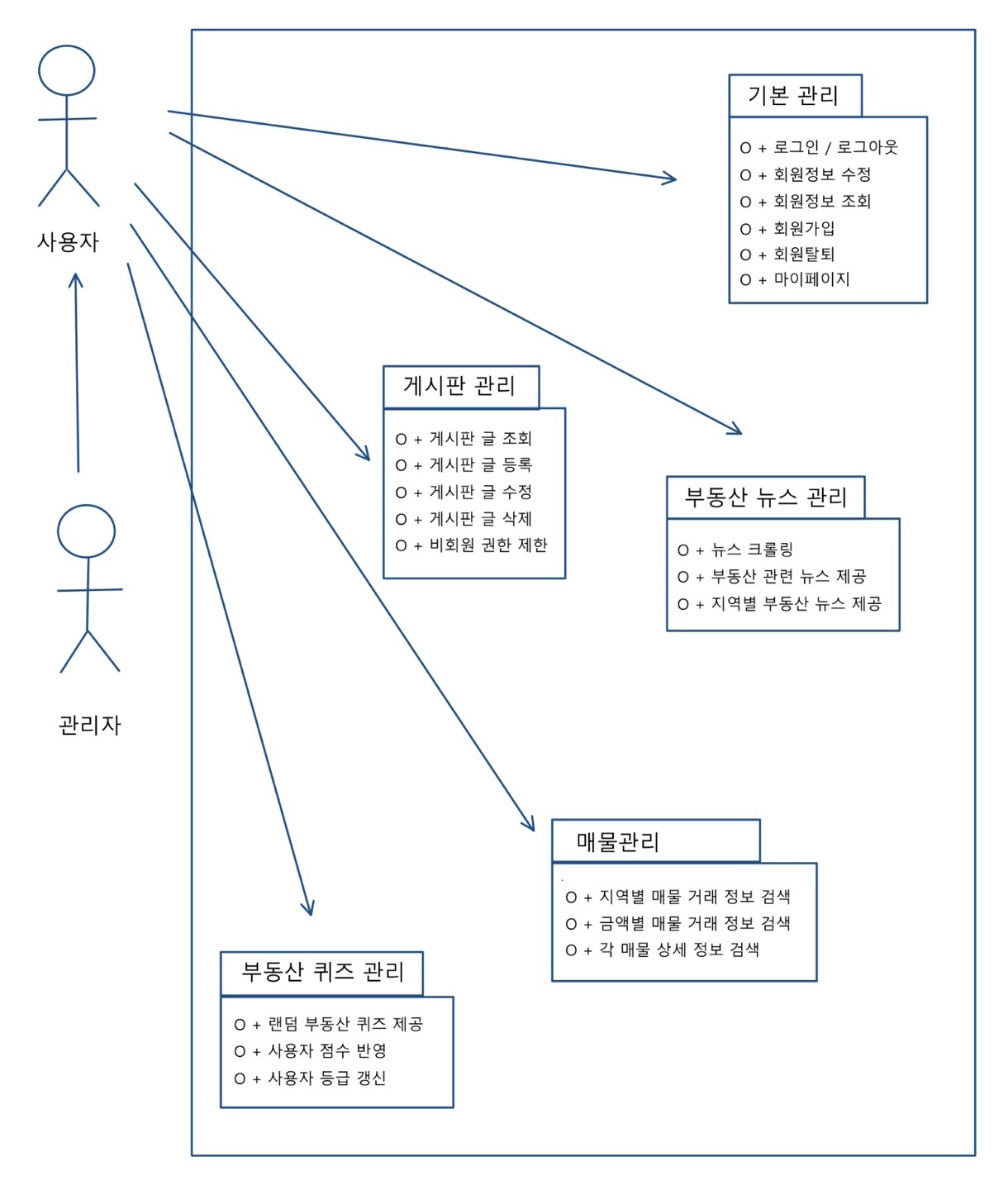
**<프로젝트 설계서>**

6기 광주 4반 최지수, 이명원, 이병석

**목차**

1. 요구사항 정의서(Usecase Diagram)
2. 테이블 구조도(ERD)
3. 클래스 다이어그램
4. 화면 설계서
5. **요구사항 정의서(Usecase Diagram)**

****

* 로그인을 포함한 기본 회원 CRUD 구현
* 회원 간의 소통을 위한 게시판 CRUD 구현
* 부동산 관련 뉴스를 볼 수 있는 페이지 구현
* 지역별, 금액별로 아파트 거래 정보를 검색하도록 구현
* 부동산 지식을 재밌게 익히기 위한 퀴즈, 점수 그에 따른 등급 구현

1. **테이블 구조도(ERD)**

테이블이(가) 표시된 사진

자동 생성된 설명

* 회원 정보 관리를 위한 ssafy\_member 테이블

1. userid : 회원 id 저장(Primary Key)
2. username : 회원 이름 저장
3. userpwd : 회원 비밀번호 저장
4. email : 회원 이메일 저장
5. joindate : 가입한 시간 저장
6. score : 회원별 퀴즈 점수 저장

* 게시판 정보 관리를 위한 board 테이블

1. articleno : 게시물의 순번 저장(Primary Key)
2. userid : 작성자 id 저장 (Foreign Key from ssafy\_member)
3. subject : 게시물 제목 저장
4. content : 게시물 내용 저장
5. hit : 조회수 저장
6. regtime : 게시물 작성 시간 저장

* 시 정보 관리를 위한 sidocode 테이블

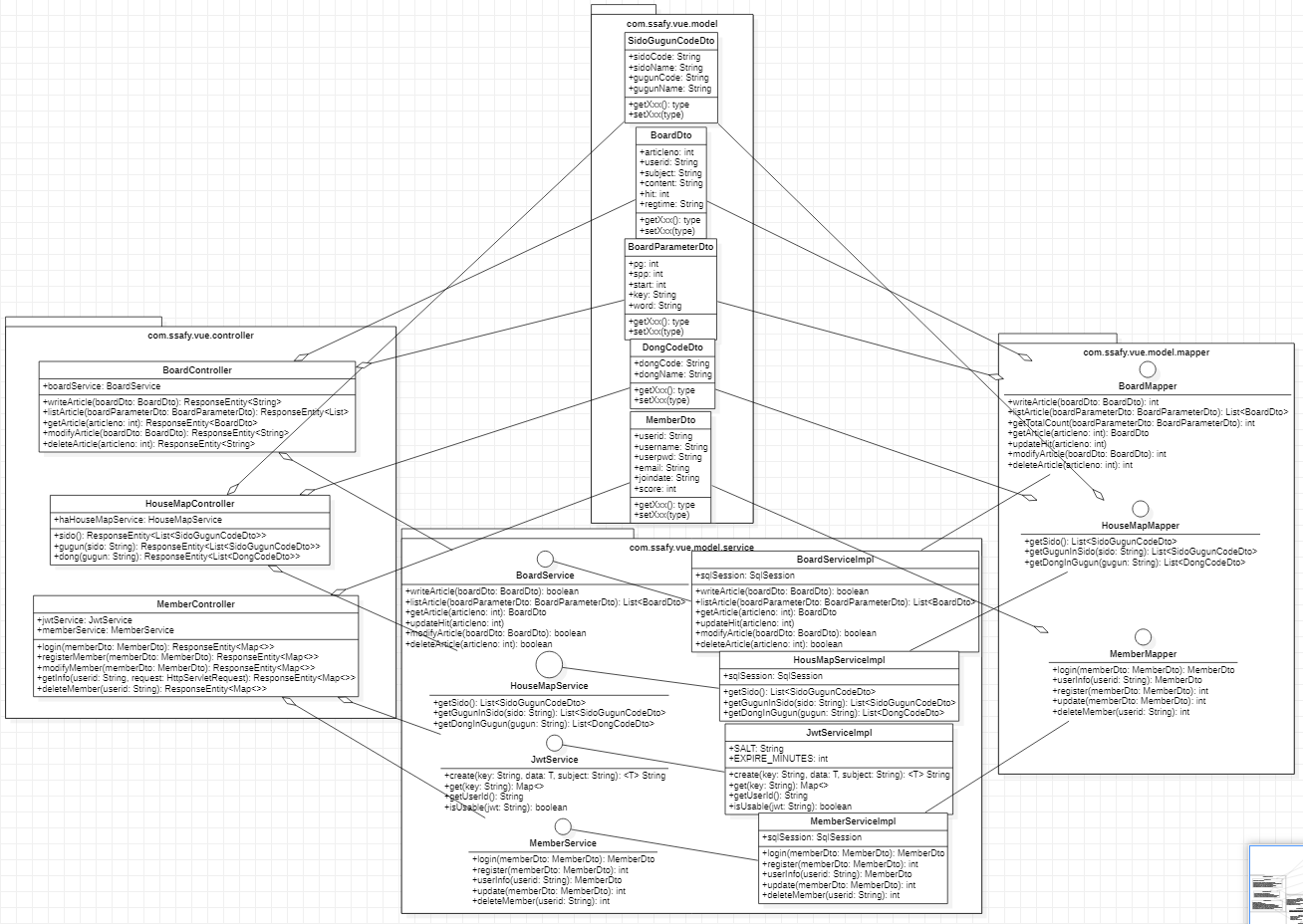
1. sidoCode : 시 별 코드 저장(Primary Key)
2. sidoName : 코드 별 이름 저장

* 구,군 정보 관리를 위한 guguncode 테이블

1. gugunCode : 구,군 별 코드 저장(Primary Key)
2. gugunName : 코드 별 이름 저장

* 동 정보 관리를 위한 dongcode 테이블

1. dongCode : 동 별 코드 저장(Primary Key)
2. sidoName : 동이 소속된 시의 이름(Foreign Key from sidocode)
3. gugunName : 동이 소속된 구,군의 이름
4. dongName : 코드 별 이름 저장
5. **클래스 다이어그램**



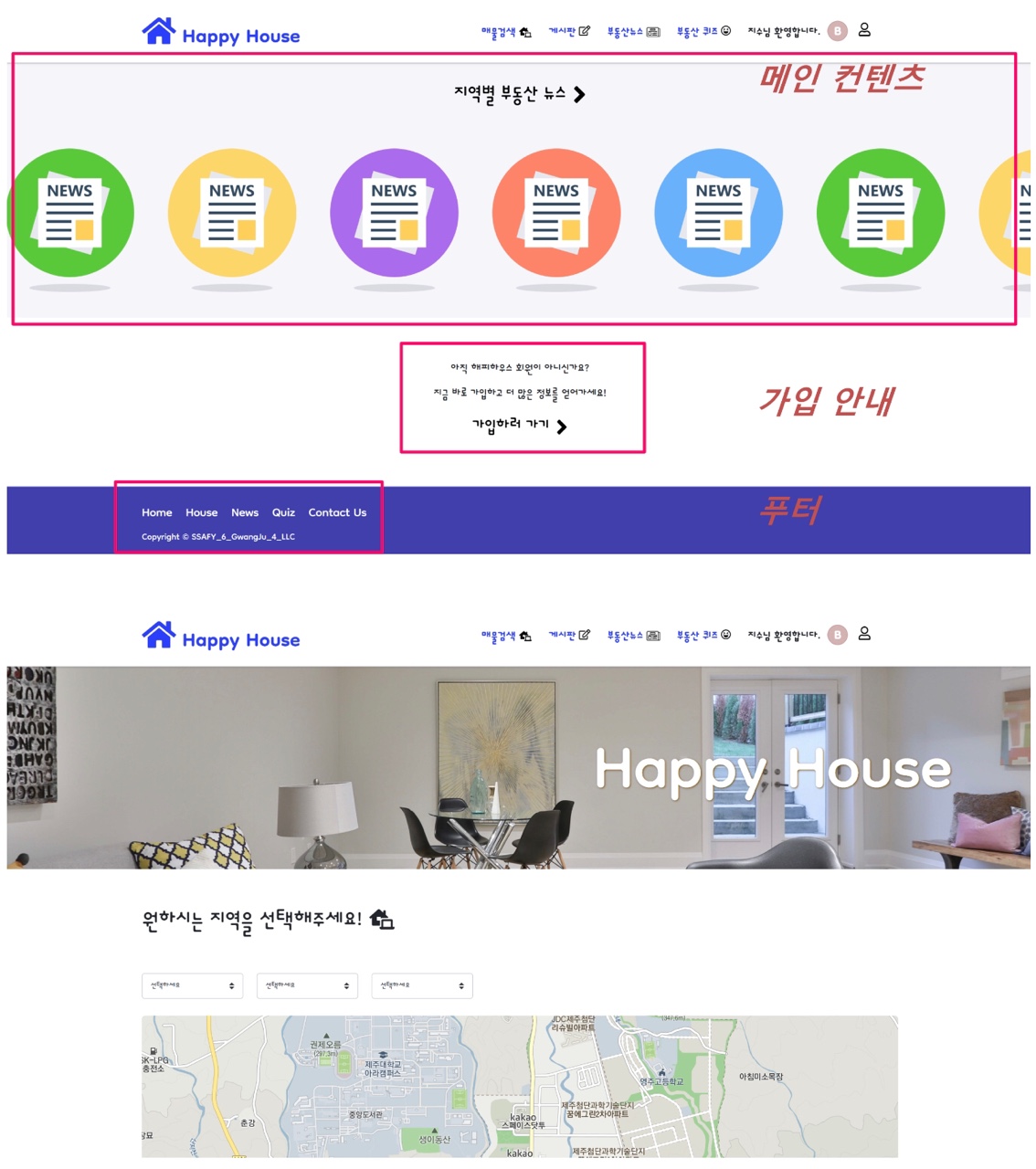
* controller는 Board, HouseMap, Member 세 종류 사용
* 각각의 controller는 Service 사용 (Member는 토큰 사용을 위해 jwt까지 2개의 서비스 연결)
* 각각의 서비스는 연결된 Mapper를 통해 DB에 접근하여 CRUD 등 처리
* 데이터들이 이동할 때 각각에 맞는 Dto에 담겨 이동

1. SidoGugunCodeDto -> HouseMapController, HouseMapMapper
2. BoardDto -> BoardController, BoardMapper
3. BoardParameterDto -> BoardController, BoardMapper
4. DongCodeDto -> HouseMapController, HouseMapMapper
5. MemberDto -> MemberController, MemberMapper
6. **화면 설계서**

* 메인화면 1



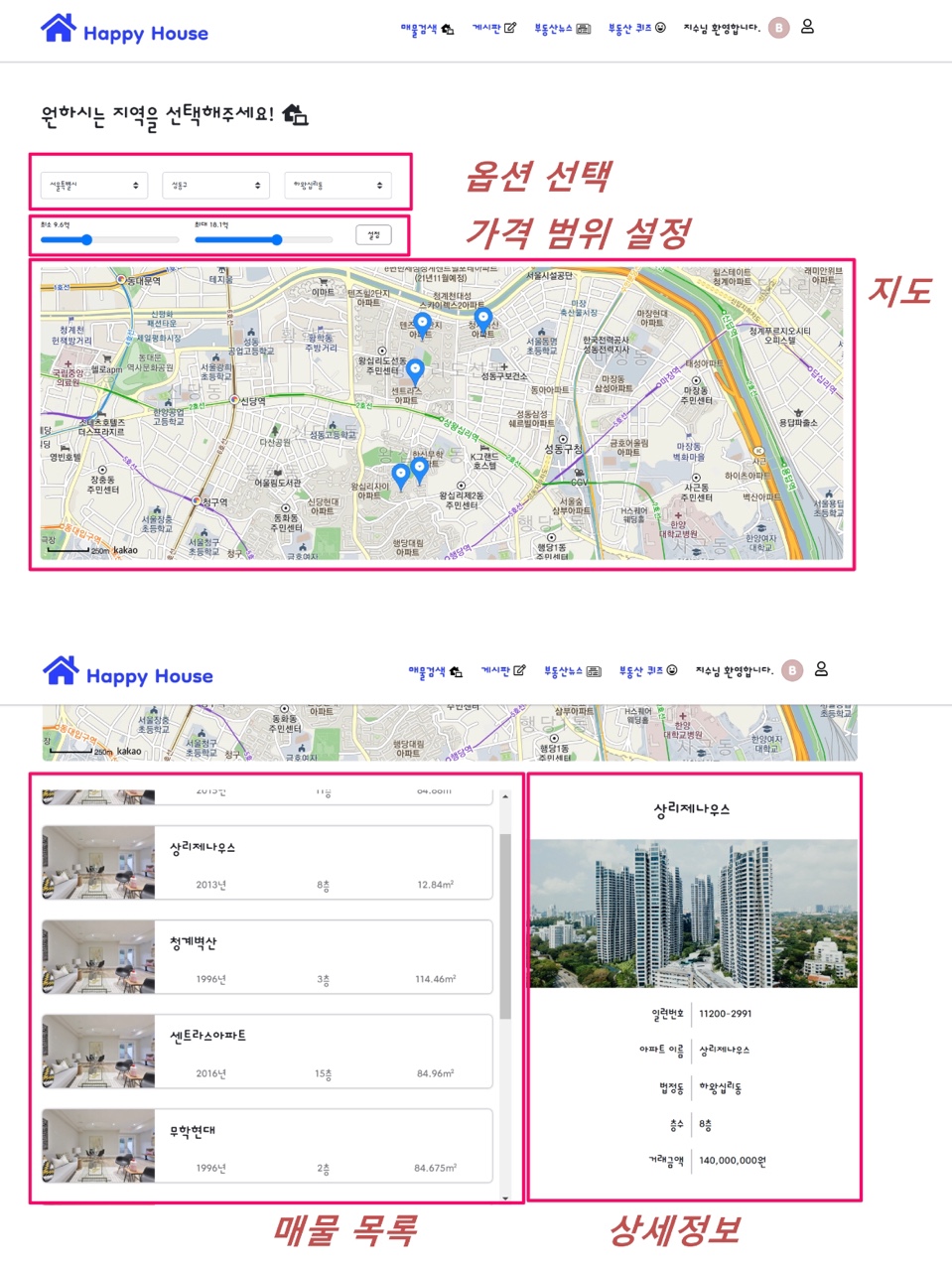
* 메인화면 2



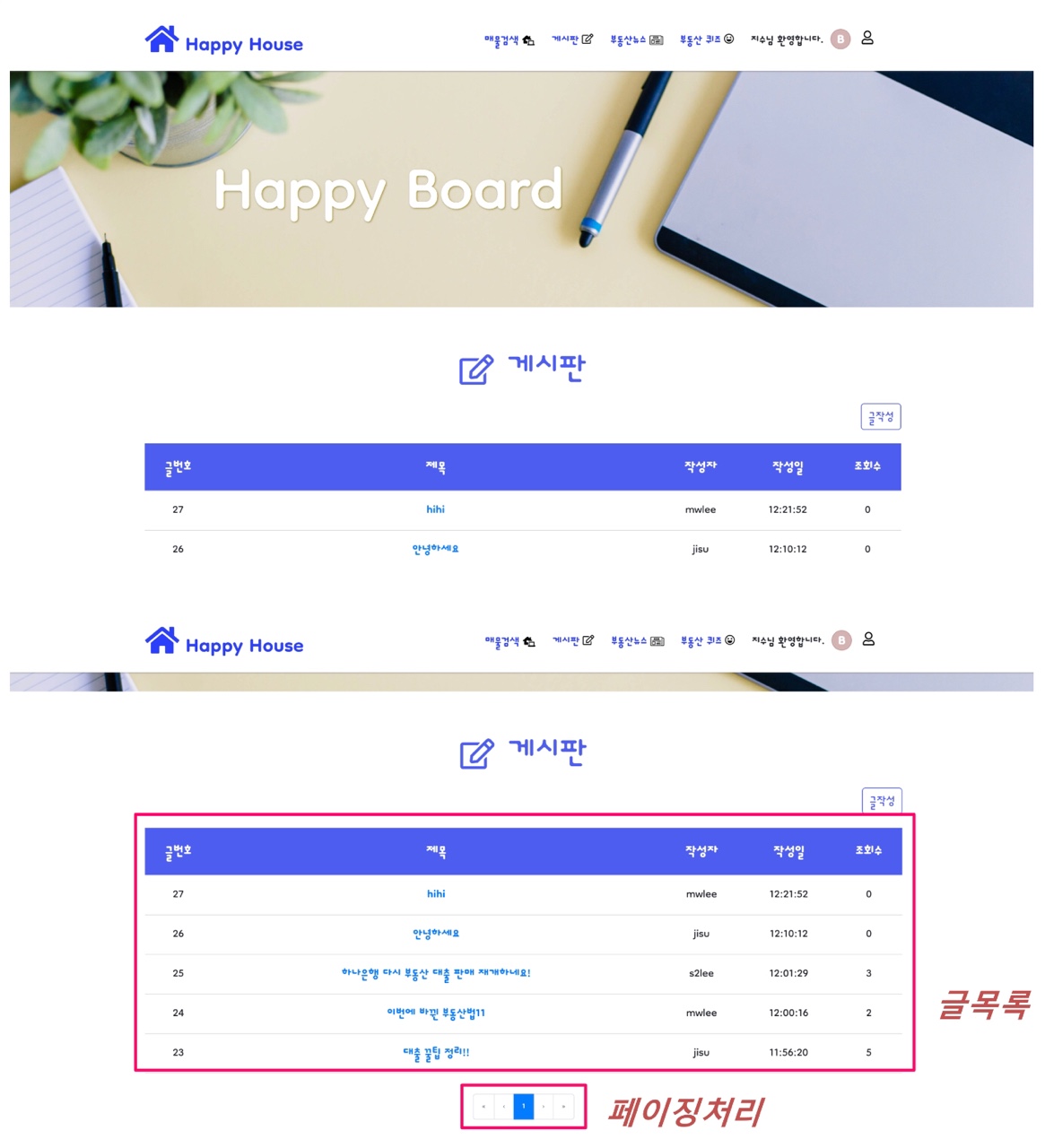
* 로그인, 회원가입 화면



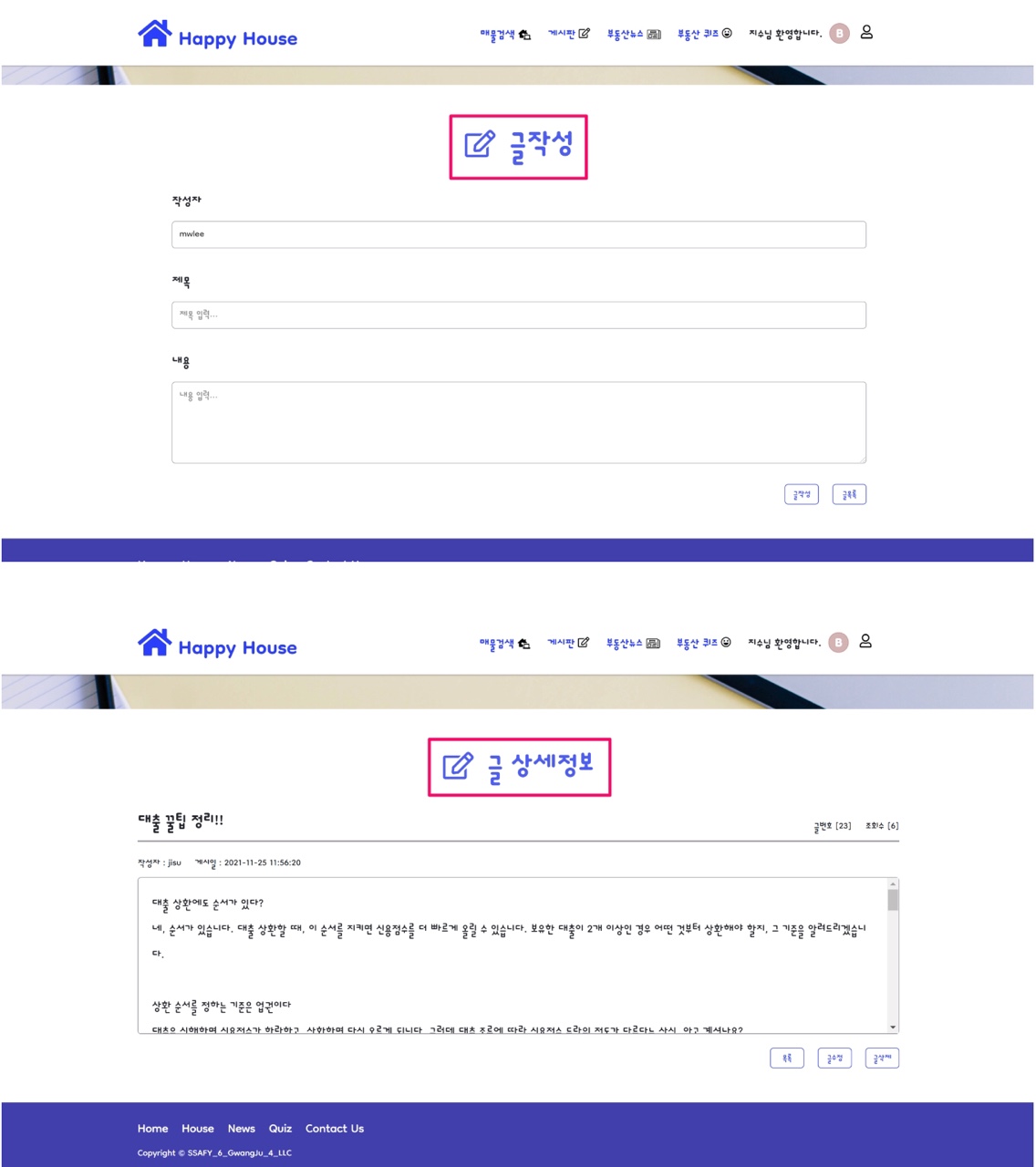
* 매물 검색 화면



* 게시판 목록 화면

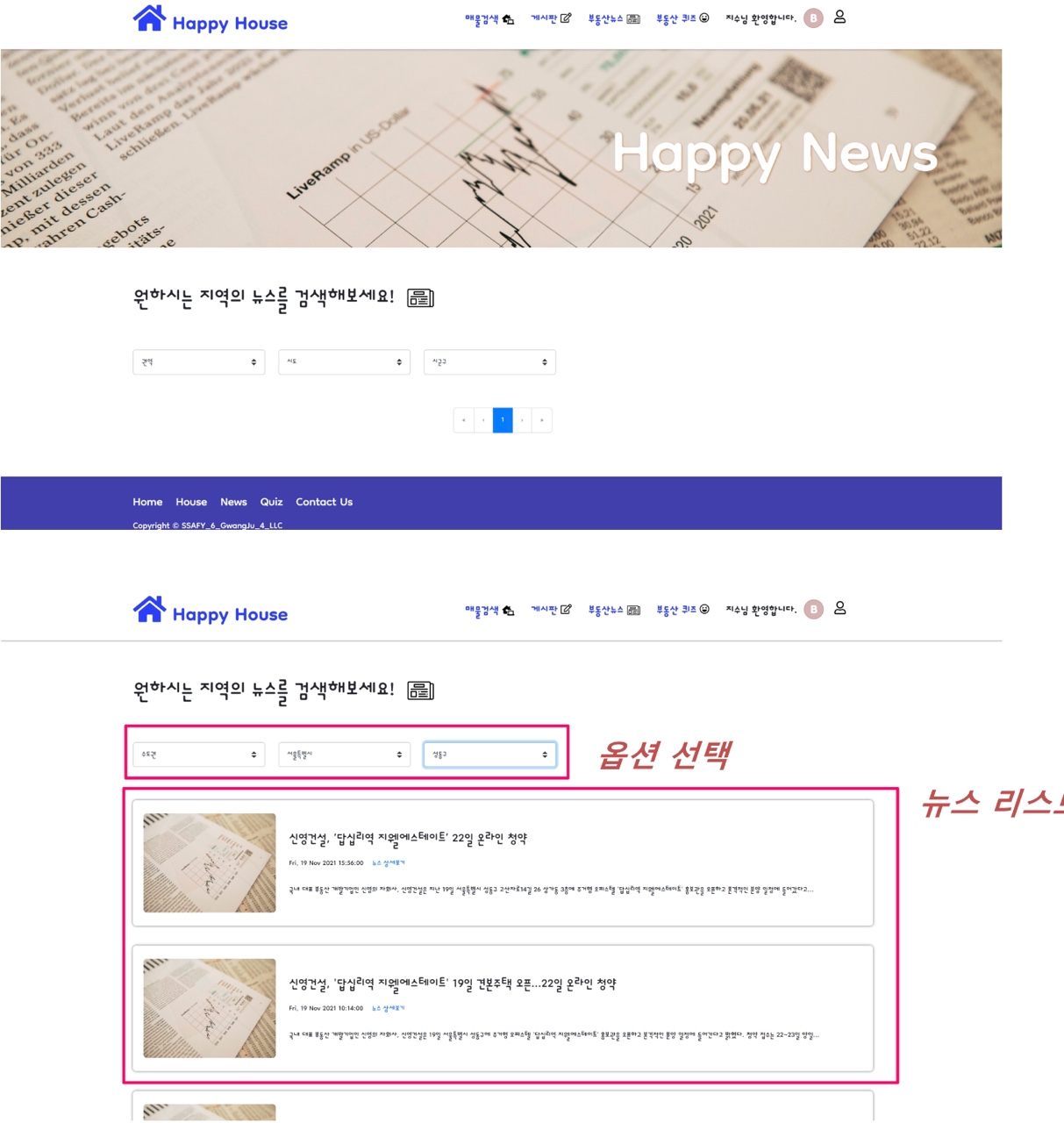


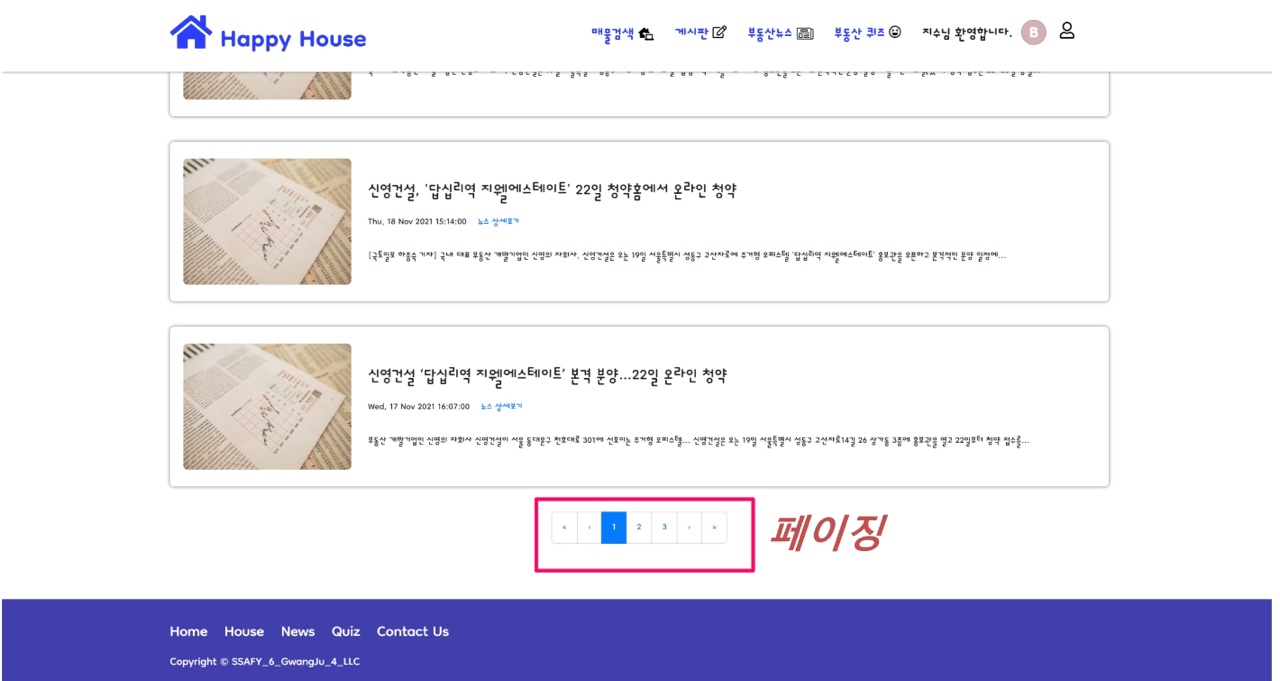
* 게시판 작성, 상세정보, 수정 화면





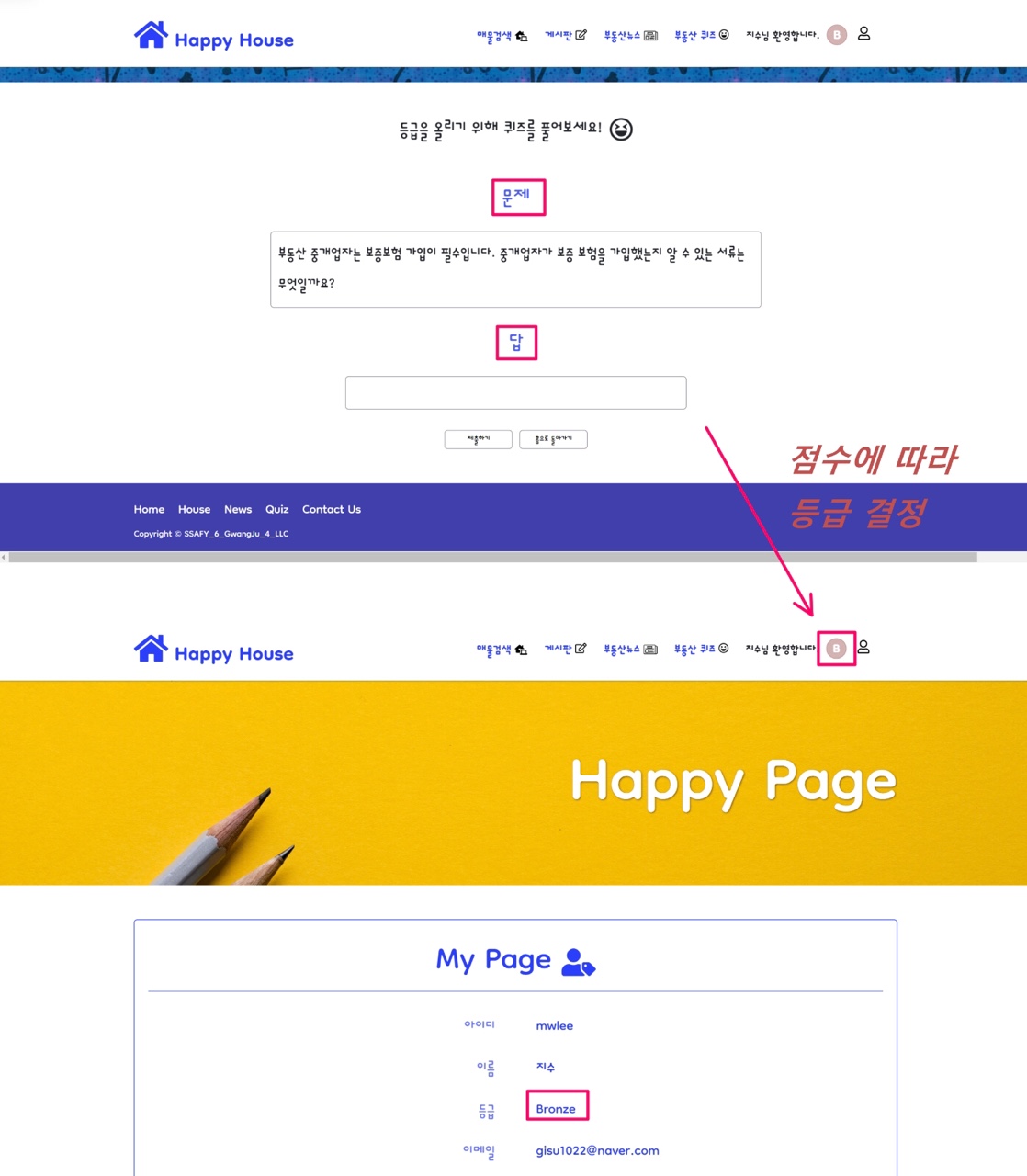
* 부동산 뉴스 화면





* 부동산 퀴즈 화면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 마이페이지 화면



* Contact us 화면

