



# 구해봐요 성대에 집

Software Requirements Specification

Introduction to Software Engineering

SWE 3002\_42

Team 2

김강산

정안승

한재웅

안진영

진송찬

강준모

문보현

조예은

# Table of Contents

## 1 Introduction

- 1.1 Purpose
- 1.2 Scope
- 1.3 Definitions, Acronyms and Abbreviation
- 1.4 References
- 1.5 Overview

## 2 Overall Description

- 2.1 Product perspective
  - 2.1.1 Market Status
- 2.2 Product Function
  - 2.2.1 로그인 및 회원가입
  - 2.2.2 프로필
  - 2.2.3 프로필 수정
  - 2.2.4 채팅
  - 2.2.5 자취방 정보 공유 기능
  - 2.2.6 커뮤니티 기능
- 2.3 User Classes and Characteristic
  - 2.3.1 User
  - 2.3.2 System Manager
- 2.4 Operating Environment
  - 2.4.1 Hardware
  - 2.4.2 Software
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

## 3 Specific Requirements

- 3.1 External Interface Requirements
  - 3.1.1. User Interfaces
  - 3.1.2. Hardware Interfaces
  - 3.1.3. Software Interfaces
  - 3.1.4. Communication Interfaces
- 3.2 Function Requirements
  - 3.2.1. Use case

- 3.2.1.1 회원가입 및 프로필 생성
- 3.2.1.2 로그인 및 로그아웃
- 3.2.1.3 유저 게시판
  - 3.2.1.3.1 자유게시판
  - 3.2.1.3.2 중고거래게시판
  - 3.2.1.3.3 공지게시판
- 3.2.1.4 부동산 정보 알림
  - 3.2.1.4.1 부동산 정보
  - 3.2.1.4.2 매물 세부 정보
- 3.2.1.5 마이페이지
  - 3.2.1.5.1 마이페이지 열람
  - 3.2.1.5.2 프로필 수정 및 삭제
- 3.3 Non-functional Requirements
- 3.4 Logical Database Requirements
- 3.5 Design Constraints
- 3.6 Standards Compliance
- 3.7 Standards Architecture

# 1 Introduction

## 1.1 Purpose

본 문서는 성균관대학교 자연과학캠퍼스 학생들에게 부동산 정보와 게시판 플랫폼 구해봐요 성대에 집의 system requirements 및 user requirements에 대하여 서술한다. 학생들을 위한 부동산 플랫폼 구해봐요 성대에 집은 성균관대학교 2023 학년도 2학기 소프트웨어공학개론 제 2조 (김강산, 정안승, 한재웅, 안진영, 진송찬, 강준모, 문보현, 조예은)에 의하여 작성된다. 해당 요구사항이 정리된 내용을 기반으로 하여 시스템 설계 및 개발이 진행될 예정이다

기존에 많은 부동산 서비스들이 제시되어 왔지만 학생들만을 위한, 학생들끼리 생활하면서 공유할 수 있는 공간이 제한되는 불편함이 있었다. 이러한 제약으로 인해 Team 2는 성균관대학교 자연과학캠퍼스 자취생, 자취 희망생들을 위한 구해봐요 성대에 집을 개발하여 불편함을 해결하려고 한다.

이 문서에는 제품, 기능, 유저, 디자인 등 요구사항 전반에 대하여 서술한다. 그리고 유저, 하드웨어 및 소프트웨어 인터페이스, 기능적 요구사항, 비기능적 요구사항 등 구체적인 요구사항에 대하여 설명한다.

## 1.2 Scope

구해봐요 성대에 집은 성균관대학교 자연과학 캠퍼스 학생들끼리 자취방에 대한 정보를 얻고 주거와 관련된 노하우들을 공유할 수 있는 커뮤니티를 제공하는 것을 주 목적으로 한다. 유저인 성균관대학교 학생들은 부동산 게시판에서 학교 주변에서 주거할 수 있는 공간들에 대한 정보를 얻는다. 학생들은 자신이 주거하고 있는 공간 또한 게시판 기능은 유저들의 자유로운 의견과 정보 공유를 위해 제공된다.

게시판은 자유 게시판과 중고 거래 게시판 두 가지 방법으로 유저들에게 서비스된다. 자유게시판은 유저들의 활발한 커뮤니케이션과 정보 공유를 위해 제공된다. 중고 거래 게시판은 유저들 사이에 활발한 커뮤니티 연결 및 비용절감을 위해 다양한 물건을 거래할 수 있도록 서비스를 제공한다. 이를 통해 수익 창출이 될 수 있고 소비자는 다양한 선택권을 유지할 수 있게 된다. 각 게시판들은 댓글 기능이 추가되어 주어진 정보에 대한 자유로운 의견을 표출 할 수 있도록 한다. 유저들은 프로필 페이지에서 자신의 비밀번호와 닉네임을 변경할 수 있고, 작성한 글은 개인 글 관리 페이지에서 확인 및 관리가 가능하다.

서비스 이용 규칙 및 업데이트 내역을 공지할 수 있는 페이지를 통해 유저들은 플랫폼 이용 규칙과 업데이트로 인해 새롭게 제시되는 서비스에 접근할 수 있게 된다.

## 1.3 Definitions, Acronyms and Abbreviation

아래의 표는 문서에서 사용되는 약어에 대한 설명이다.

약어	설명
UI	User Interface
UX	User Experience
API	Application Programming Interface
AWS	Amazon Web Service
DB	Database
OS	Operating System

아래의 표는 문서에서 정의한 기술적 개념이다.

단어	설명
유저	성균관대학교 학생 중 구해봐요 성대에 집 가입자
관리자	구해봐요 성대에 집 작동 및 서버를 관리하는 자
백엔드	웹 사이트 솔루션의 프로세스와 관련된 서버 측과 데이터 베이스부분
프론트엔드	유저에게 시각적으로 보여지는 부분

## 1.4 References

IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements

Specifications, In IEEEExplore Digital Library

[https://github.com/skkusal/SWE3002\\_42/blob/main/2023s/Team13\\_SRS.pdf](https://github.com/skkusal/SWE3002_42/blob/main/2023s/Team13_SRS.pdf)

## 1.5 Overview

Software Requirement Specification 문서는 총 3장으로 이루어져있다.

두 번째 부분은 소프트웨어 제품의 전반적인 부분을 설명한다. 우선 보다 넓은 맥락 안에서의 제품 관점을 설명하고 주요기능, 사용자 클래스 및 그들의 특성, 일반 사용자와 시스템 관리자를 구분한다. 하드웨어 및 소프트웨어 요구 사항을 지정하는 운영 환경을 개요화하며, 개발 프로세스에 영향을 미칠 수 있는 설계 및 구현 제약 사항을 강조한다. 사용자 설명서, 사용자 메뉴얼 및 가이드를 포함한 사용자 문서와 외부 종속성에 대해서도 기술한다.

세 번째 장에서는 구체적인 요구 사항에 중점을 둔다. 이 부분은 외부 인터페이스 요구 사항을 포함하며, 다른 시스템 또는 사용자와의 상호 작용과 관련이 있다. 또한 소프트웨어의

원하는 동작 및 기능을 설명하는 기능 요구 사항과, 성능, 보안 및 사용 편의성과 같은 품질 속성을 정의하는 비기능 요구 사항을 다룬다. 논리적 데이터베이스 요구 사항, 설계 제약 사항 및 업계 표준에 부합하는 것을 보장하기 위한 표준 준수 사항을 다룬다. 마지막으로 소프트웨어가 변화하는 요구 사항과 기술적 진보를 충족시키기 위해 어떻게 적응하고 진화할 수 있는지를 나타내며, 이러한 부분은 소프트웨어의 기능 및 성능에 대한 상세한 명세와 기대치를 개요화한다

## 2 Overall Description

### 2.1 Product perspective

이 플랫폼은 성균관대학교 학생들이 자취방을 구하는 데 있어서 편리하게 방을 구할 수 있도록 돕는 플랫폼이다. 현재 다방, 직방과 같은 어플이 존재하지만 학생들이 방을 구하는 데 특화된 어플은 아니기 때문에 위의 어플을 사용하는 데에는 어려움이 있다. 따라서 온전히 ‘성균관대 학생들을 위한 자취방 구하기’에 집중된 플랫폼을 구현하고자 하였다. 사용자는 해당 플랫폼에서 쉽게 방을 찾을 수 있고, 커뮤니티를 만들어 관련 정보를 공유할 수 있다. 이를 통해 경제적이고 효율적이게 자취방을 구할 수 있어 자취방이 필요한 많은 학생들의 수고로움을 덜 수 있을 것이다.

#### 2.1.1 Market Status

현재 학생들이 학교 근처에 자취방을 구할 수 있는 대표적인 애플리케이션은 ‘직방’, ‘다방’, ‘에브리타임’ 등이 있다. 먼저 직방과 다방은 대표적인 온라인 부동산 플랫폼이다. 아무런 정보 없이 부동산을 방문해야 했던 과거와 달리, 소비자들은 자신의 예산과 조건에 맞는 집을 모바일로 미리 확인할 수 있다. 하지만 위 어플의 가장 큰 문제는 허위매물로 인한 이용자의 피해이다. 직방과 다방은 허위매물이 많아 이용자들은 편하게 집을 구하려 했다가 오히려 더 불편을 겪는 경우가 허다하다. 뿐만 아니라 사용자는 플랫폼에 등록된 매물만 확인할 수 있어 매물 비교가 제한적이다.

두 번째로 에브리타임은 학교 인증이 된 학생들 간의 정보 공유, 시간표 관리, 커뮤니케이션 기능 등 다양한 게시판을 통해 여러 종류의 서비스를 한꺼번에 제공하는 것이 특징인 애플리케이션이다. 하지만 게시판에 정해진 양식과 기능이 없고 단순히 사용자가 올리는 대로 올라가는 에브리타임 특성상 정확한 정보 전달에 한계가 있다. 집을 구하는 일은 단순한 거래가 아니기 때문에 정확한 정보 전달, 현실적인 집 내부의 모습, 외부 환경 조사 등이 필요하다. 하지만 에브리타임의 글로는 앞서 얘기한 내용을 제대로 확인할 수 없고 특히 원룸 게시판의 경우 활성화가 제대로 되어있지 않기 때문에 실질적인 도움이 되지 않는다.

### 2.2 Product Function

#### 2.2.1 로그인 및 회원가입

‘구해봐요 성대에 집’ 플랫폼을 이용하기 위한 첫 번째 단계이다. 회원이 아닌 경우에

회원가입 창을 통해 회원가입을 해야 하며, 회원인 경우에는 로그인을 하여 플랫폼에 접속한다. 회원가입 시에는 학교 인증을 통해 사용자를 인증하여 성균관대 학교 학생들만 플랫폼을 이용할 수 있도록 규제하고, 그 후에는 재학중인 캠퍼스를 선택하여 명륜동에 있는 인문사회과학 캠퍼스와 율전동에 위치한 자연과학 캠퍼스를 구분한다.

### 2.2.2 프로필

사용자의 계정과 관련된 정보를 작성 및 열람할 수 있는 기능이다. 사용자가 올린 글 목록을 열람할 수 있다.

### 2.2.3 프로필 수정

입력한 개인정보를 변경하는 기능이다. 닉네임, 프로필 사진, 그리고 선택한 캠퍼스를 변경 가능하다.

### 2.2.4 채팅

사용자 간에 소통을 위한 기능이다. 게시글의 글쓴이에게 바로 채팅을 보낼 수 있게 연결되어 있으며 채팅방 목록을 통해 채팅 내용을 관리할 수 있다. 채팅방을 클릭하면 상대방의 닉네임과 채팅내용을 확인할 수 있다.

### 2.2.5 자취방 정보 공유 기능

자취방의 가격, 평수, 특징 등을 공유하는 기능이다. 웹 상 존재하는 부동산 정보를 가져와 게시해 놓기도 하며 개인 대 개인간의 공유에는 가격, 평수, 자취방의 사진을 올리는 탭을 만들어 정확한 정보를 공유할 수 있도록 한다. 이 경우에는 개인간 채팅 기능을 통해 작성자에게 직접적으로 연락을 하여 더 필요한 정보에 대해 물을 수 있다. 뿐만 아니라 자취방 양도의 경우 계약 기간과 같은 추가적인 정보를 기입함으로써 이용자가 최대한 본인의 상황과 맞는 자취방을 구할 수 있도록 한다.

### 2.2.6 커뮤니티 기능

혼자 사는 학생들이 자유롭게 정보를 공유하는 공간이다. 예를 들면 집을 뺄 때 두고 갈 물품을 나눔하는데 사용할 수도 있고 집을 구하는 데 필요한 팁 등을 공유하는데 사용할 수도 있다. ‘좋아요’ 기능을 사용하여 좋아요 수 가 많은 글의 경우 게시판 상단에 위치하도록 한다. 반면 비매너 사용자들이 올리는 악성 글의 경우 신고 기능을 통해 신고가 일정 횟수 이상 쌓였을 경우 시스템에 통보하여 해당 사용자를 자동으로 블록 처리 할 것이다.

## 2.3 User Classes and Characteristic

유저는 일반 유저와 관리자로 나누어진다. 일반 유저는 게시판에 대한 열람 및 작성이

가능하다.

### 2.3.1 User

이 플랫폼의 사용자는 성균관대학교에 재학 중인 학생들로 제한한다. 사용자는 성균관대학교 근처에서 자취방을 구하고 있는 성균관대학교 학생이라고 가정한다. 사용자는 한국어를 사용할 수 있다고 가정하며, 자신의 기기에 애플리케이션을 설치하고 사용할 수 있는 기본지식이 있다고 가정한다.

### 2.3.2 System Manager

이 플랫폼의 시스템 관리자는 시스템 전문가로 제한된다. 이들은 ‘구해봐요 성대에 집’의 시스템에 대한 충분한 지식이 있어야 하고 전반적인 시스템 상황과 더불어 부동산 정보에 관한 전문적인 용어 및 지식을 이해할 수 있어야 한다. 또한 네트워크에 대한 충분한 지식이 있어야 하며, 일종의 커뮤니티를 운영하는 입장이기 때문에 도덕적, 윤리적 의식이 있어야 한다. 추가적으로 시스템 관리자는 개발 팀, 보안 인력, 고객 서비스 대표와 밀접하게 협력하여 시스템이 효과적으로 운영되고 사용자의 피드백이 애플리케이션 개발 주기에 통합되도록 해야 한다.

## 2.4 Operating Environment

### 2.4.1 Hardware

해당 웹 플랫폼은 windows, macOS, 혹은 Linux 운영 체제 기반의 데스크탑과 Android 및 iOS 운영체제 기반의 모바일 디바이스를 위해 설계되었다. ‘구해봐요 성대에 집’ 접속하는 사용자 기기는 최고 64bit CPU, 512MB of RAM, 200MB of drive space 이상의 빈 공간이 있음을 전제한다.

### 2.4.2 Software

해당 웹 플랫폼은 최소 112.0.5615.138 버전 이상의 chrome에서 사용되는 것을 전제하고 있다. 따라서 다른 웹 브라우저를 사용하는 경우 모든 서비스를 원활하게 이용할 수 없을 수도 있다.

## 2.5 Design and Implementation Constraints

해당 플랫폼은 다음 체크리스트를 고려하여 디자인하고 구현될 예정이다.

- 시스템은 사용자의 편의와 요구사항을 고려해야 하며, 해당 내용에 따라 수정이 가능하다.
- 사용자에게 친숙한 인터페이스를 제공할 수 있도록 개발하며, 자취방 정보에 쉽게 접근할수 있도록 한다.
- 시스템은 꼭 필요한 경우가 아니라면 가급적 무료로 제공되는 소프트웨어를 사용한다.



- 가능한 범위 내에서 오픈소스 소프트웨어를 적극 활용한다.
- 어플리케이션 유지 보수를 고려하여 주석을 적극적으로 활용한다.
- 향후 어플리케이션의 확장성 및 가용성을 고려하여 소스 코드를 작성한다.
- 자원 낭비를 최소화할 수 있도록 최적화된 소스 코드를 작성한다.
- 사용자에게 친숙한 인터페이스를 제공할 수 있도록 개발한다.
- 한국어가 필요하다
- 최소 React 18.2.0 이상, Node 14.18.0 이상의 버전을 이용해 개발한다.
- UI/UX 디자인은 Figma를 사용하여 프로토타입으로 제작된다.
- 애플리케이션은 Flask 웹 프레임워크를 사용하여 구현되며, Python 3.8 이상의 버전에서 안정적으로 동작해야 한다.
- Windows 10 이상, chrome 버전 112.0.5615.138 이상으로 테스트한다.
- 시스템은 512MB 이상의 RAM을 가진 디바이스에서 실행되어야 한다.
- 시스템 설치와 구동을 위해 최소 200MB의 여유공간이 디바이스에 존재해야 한다.
- 유저 디바이스와 서버는 HTTP 프로토콜로 통신한다.
- 시스템은 Windows, Linux, MacOS 세 운영체제에서 실행 가능하다.
- 시스템은 Microsoft Edge, Firefox, Chrome, Opera, Safari 등의 웹 브라우저에서 실행이 가능하다.

## 2.6 User Documentation

사용자가 해당 플랫폼을 잘 활용할 수 있도록 돕기 위해 몇 가지 문서가 필요하다. 첫째, 하드웨어 및 소프트웨어 요구사항을 다룬 문서가 필요하며, 둘째 사용자 설명서가 필요하다. 사용자 설명서의 경우 각 탭은 어떤 목적을 위해 개설되었으며 각 탭에 글을 쓸 경우 어떤 양식을 지켜서 글을 써야 하는지, 신고와 차단 기능은 어떤 목적인지에 대한 내용을 포함해야 한다. 또한 각 게시글과 부동산 시세 및 매물 내역에 대한 예시가 담겨있는 스크린샷이 포함되어야 한다. 마지막으로 시스템 개발자의 연락처를 제공하여 기타 버그나 문제점을 보고할 수 있도록 한다.

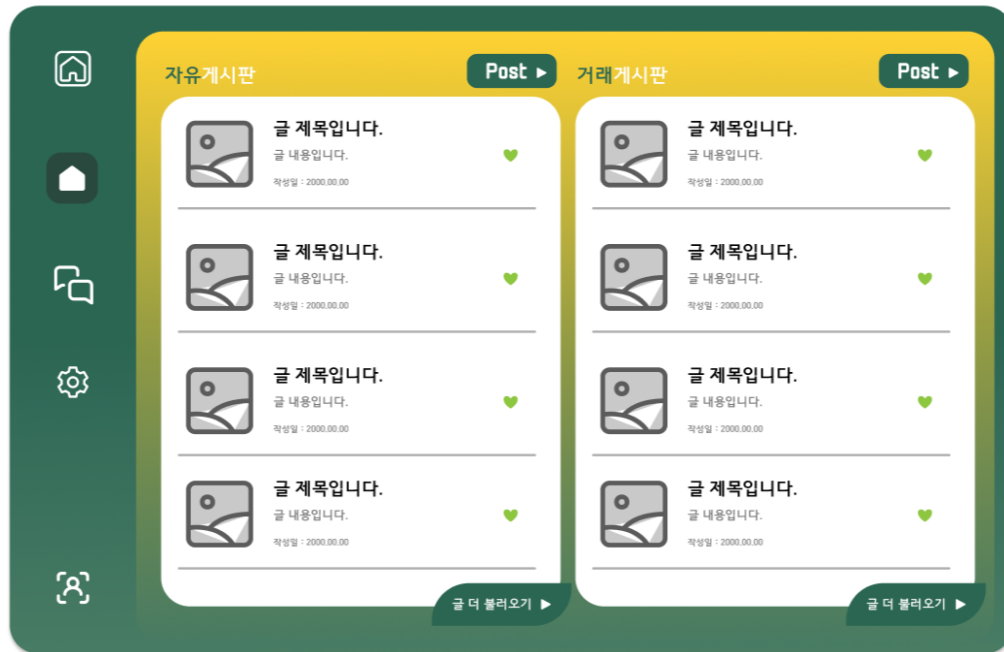
## 2.7 Assumptions and Dependencies

본 문서에서 구현할 웹 플랫폼은 최소 windows 10 이상, macOS Ventura, 우분투 22.04.2 LTS 버전 이상의 운영체제를 사용하고 112.0.5615.138 버전 이상의 chrome 웹 브라우저를 사용하는 기기에서 사용된다는 가정하에 작성되었다. 단, 웹개발언어를 이용한 웹 앱 어플리케이션이 제품의 전신이므로, 다른 운영체제의 모바일 디바이스 및 데스크탑에서도 접근하여 이용할 수 있다.

# 3 Specific Requirements

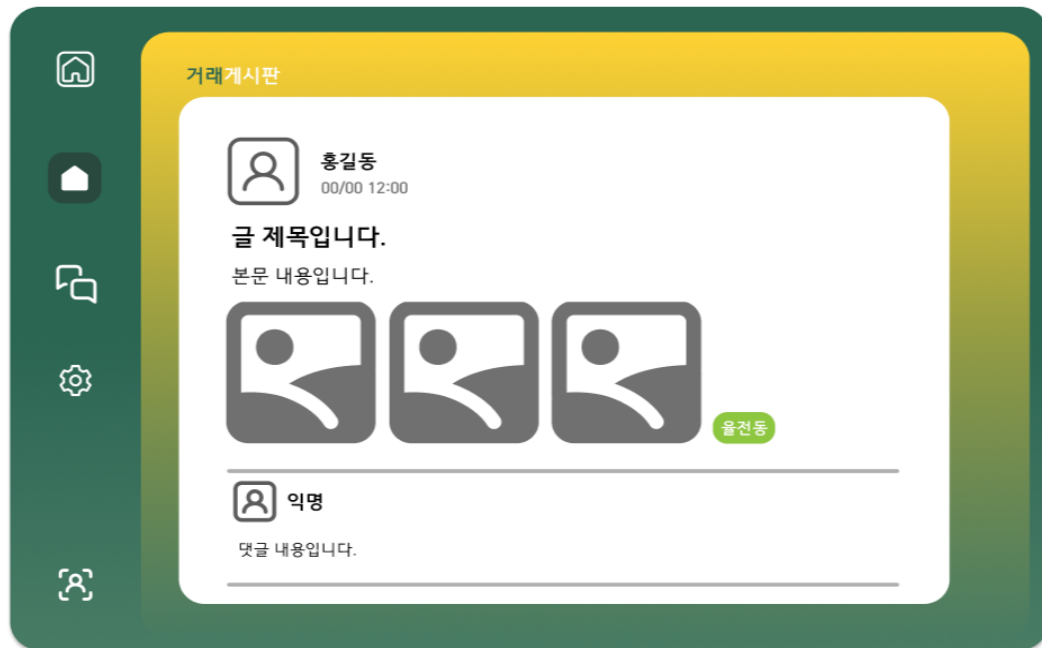
## 3.1 External Interface Requirements

### 3.1.1 User Interfaces



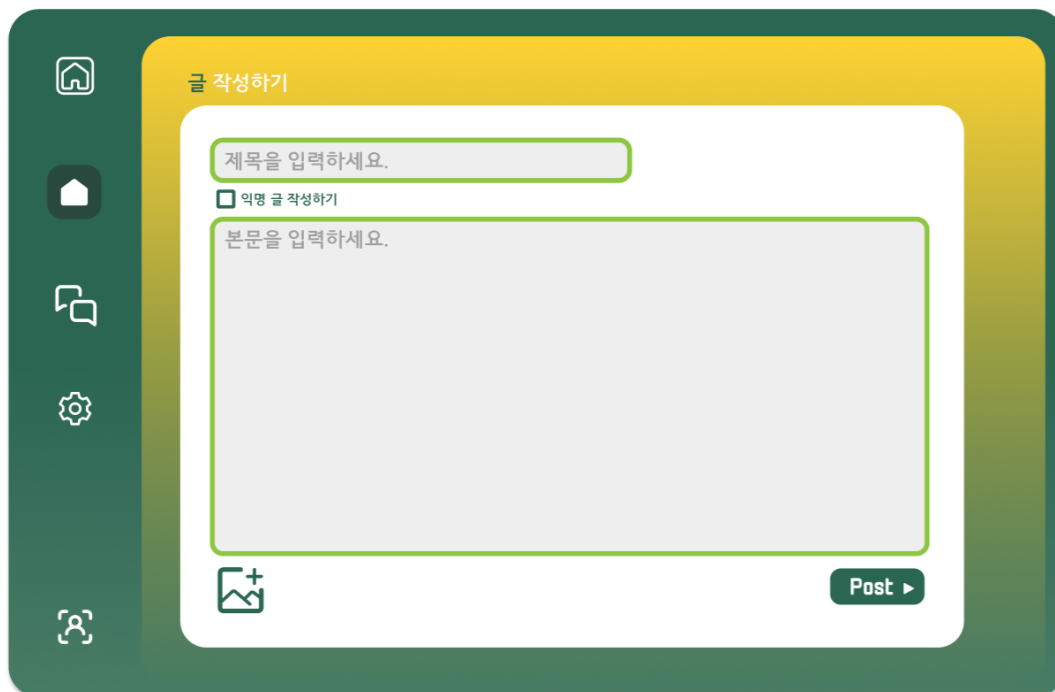
[Figure 1] Bulletin Board의 user interface

1. 좌측 메뉴의 버튼으로 홈페이지, 채팅, 설정, 내 정보 페이지로 이동한다.
2. 자유 게시판, 거래 게시판의 글 목록이 있다. 각 게시판의 글은 최신 작성된 글 순서대로 나열된다.
3. Post 버튼으로 각 게시판에 새 글을 작성한다. Figure 3 화면으로 넘어간다.
4. 글 더 불러오기 버튼으로 이 전에 작성된 글을 확인한다.
5. 게시글 박스를 클릭하여 각 글의 상세 내용으로 이동한다. Figure 2 화면으로 넘어간다.



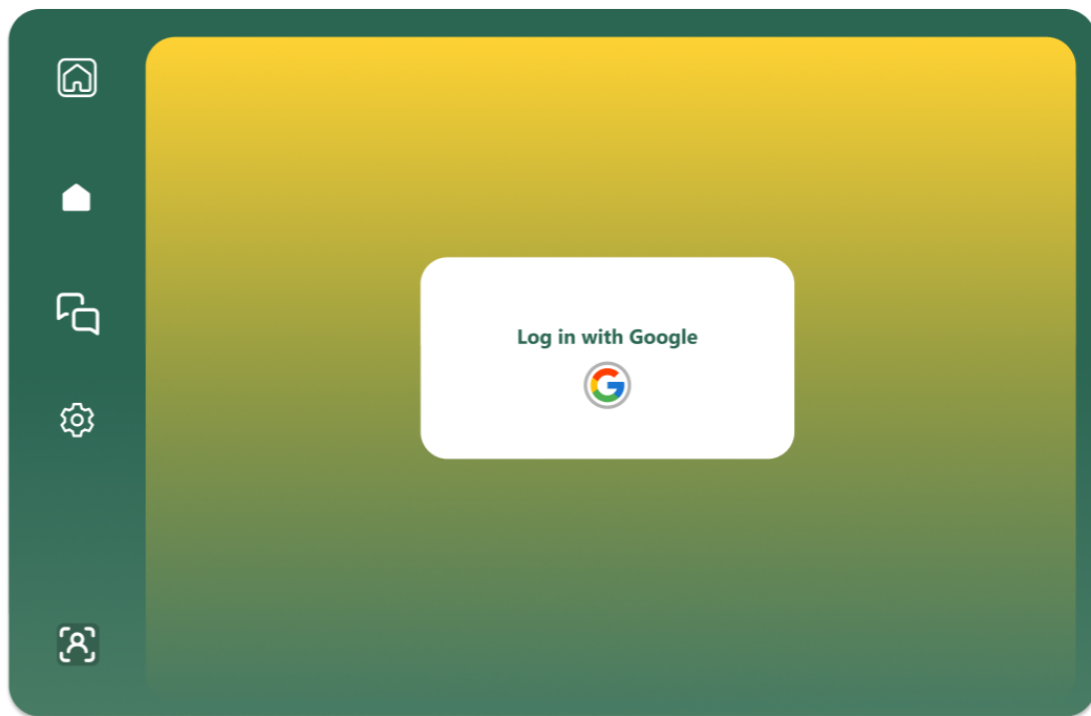
[Figure 2] Post Information의 user interface

1. 게시글 본문 내용을 확인할 수 있다.
2. 글 상단에 글 작성자, 글이 작성된 시간, 날짜가 표시된다.
3. 작성자가 업로드한 사진을 클릭해 사진을 확대하여 확인한다.
4. 거래게시판에서는 거래 가능 지역이 표시된다.
5. 하단에는 게시글에 달린 댓글이 표시된다.



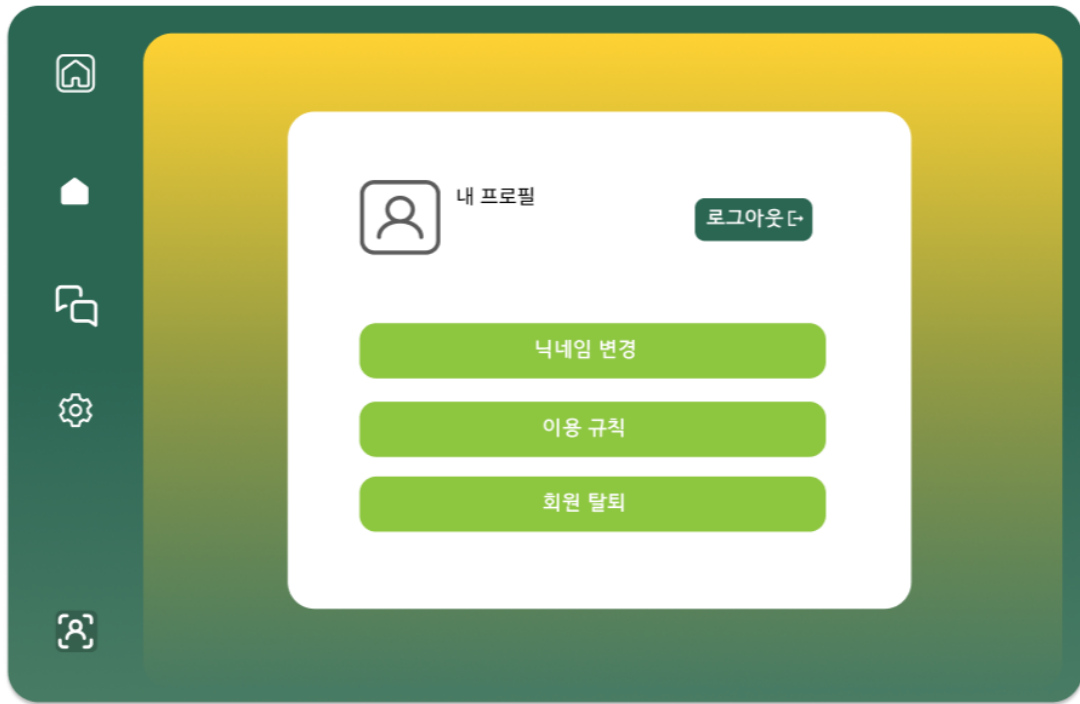
[figure 3] 게시글 작성의 user interface

- 1.글의 제목과 본문 내용을 입력한다.
- 2.체크 박스를 통해 글의 익명 여부를 결정할 수 있다.
- 3.게시글의 사진을 첨부할 수 있다.
- 4.Post 버튼을 통해 글을 게시한다.



[figure 4] 로그인 user interface

- 1.성균관대학교(@g.skku.edu) Google 계정을 통해 로그인이 가능하다.
- 2.G 아이콘을 클릭 시 구글로 링크된다.
- 3.로그인에 성공하면 figure 5 화면으로 넘어간다.



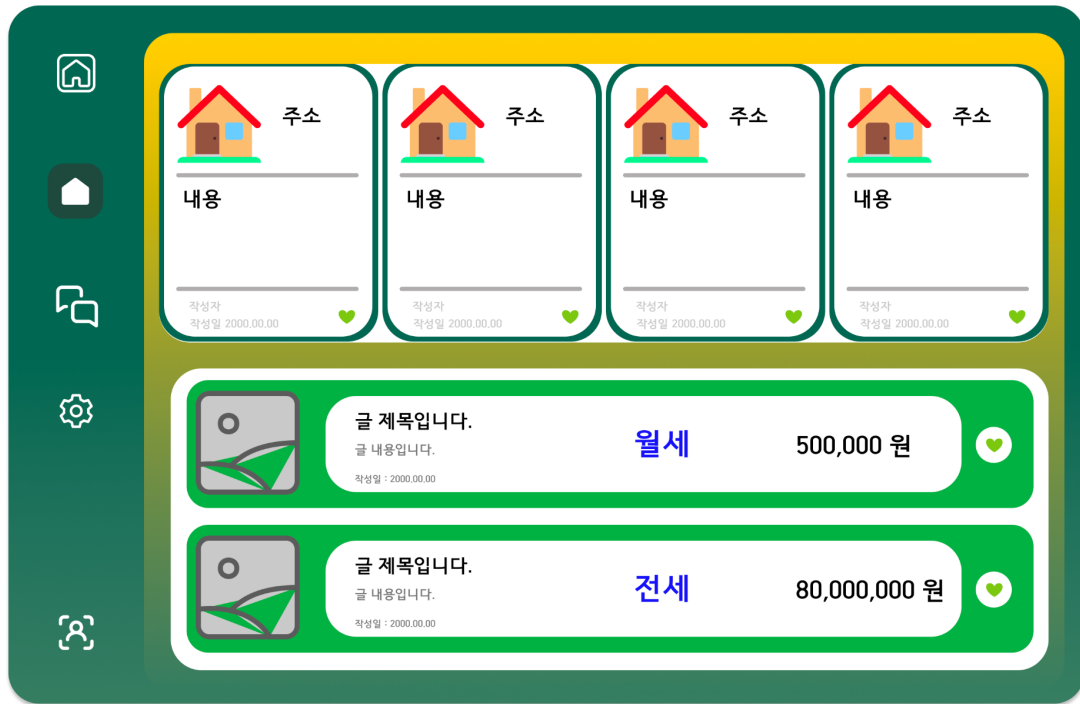
[figure 5] 내 프로필의 user interface

1. 닉네임 변경, 회원 탈퇴, 로그아웃이 가능하다.
2. 이용 규칙을 확인할 수 있다.



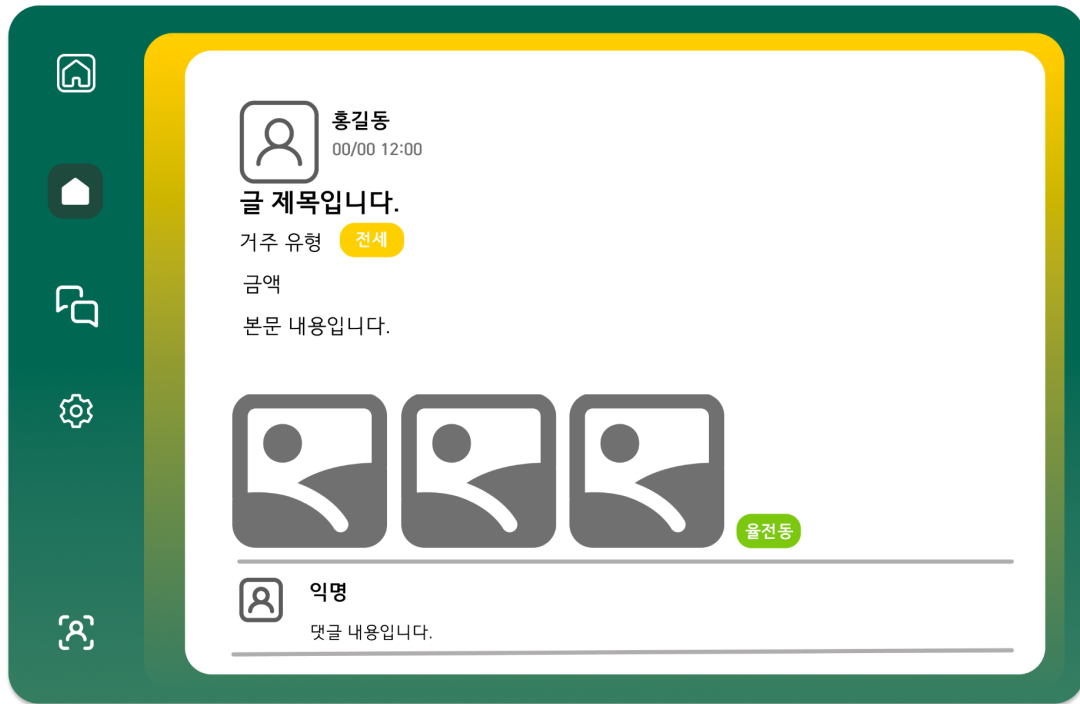
[figure 6] 홈 화면의 user interface

1. 부동산 정보를 제공해 준다.
2. 자유게시판 정보를 제공해 준다
3. 거래 게시판 정보를 제공해 준다
4. 각 게시물 마다 좋아요를 누를 수 있는 버튼이 있다
5. 글 더 불러오기 버튼을 누르면 figure 1 화면으로 넘어간다
6. 설정 버튼을 누르면 figure 5 화면으로 넘어간다
7. 프로필 버튼을 누르면 figure 5 화면으로 넘어간다



[figure 7] 부동산 정보 화면의 user interface

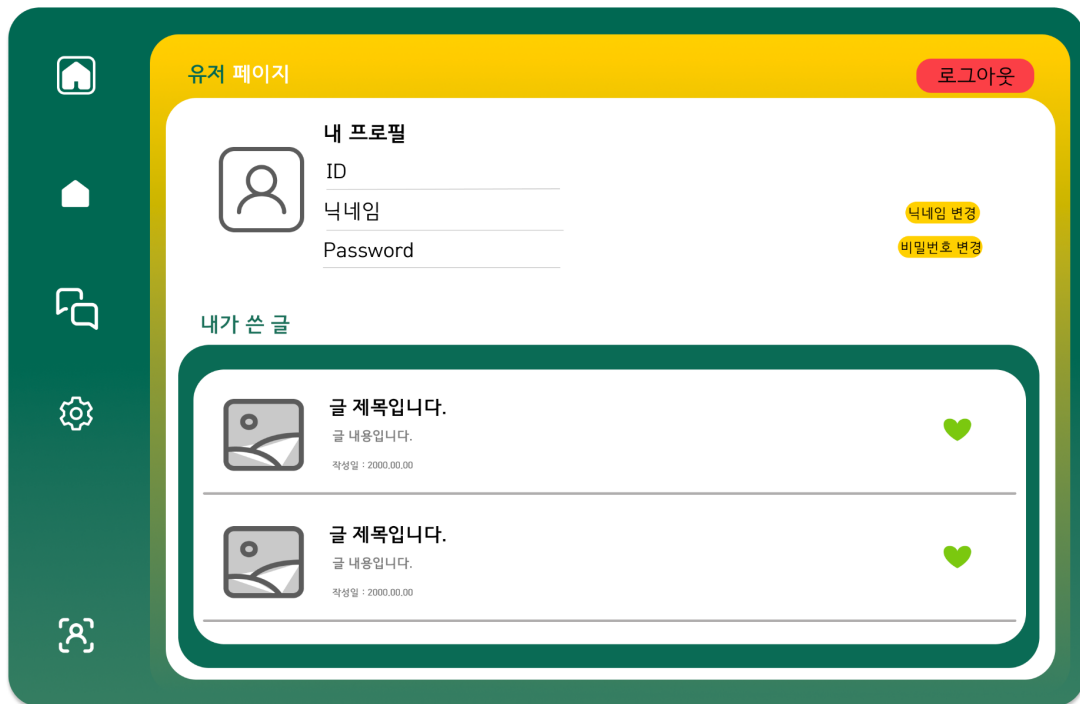
1. 정보 글을 누르면 figure 8 화면으로 넘어간다
2. 각 게시물 마다 좋아요를 누를 수 있는 버튼이 존재한다
3. 부동산 정보글의 형태는 두 가지 방법으로 나타난다.
4. 전세 월세를 확인할 수 있는 정보 글이 있다.



[figure 8] 부동산 정보 화면의 user interface

1. 게시글 본문 내용을 확인할 수 있다.
2. 글 상단에 글 작성자, 글이 작성된 시간, 날짜가 표시된다.
3. 작성자가 업로드한 사진을 클릭해 사진을 확대하여 확인한다.
4. 거래게시판에서는 거래 가능 지역이 표시된다.
5. 하단에는 게시글에 달린 댓글이 표시된다.
6. 거주 유형과 금액이 표시된다.





[figure 9] 유저 개인 화면의 user interface

1. 프로필을 관리하는 화면이다
2. 닉네임과 비밀번호를 변경할 수 있다.
3. 내가 쓴 글을 확인하고 관리할 수 있다.



[figure 10] 홈 화면의 user interface

1. 공지사항 및 이용규칙이 제공되는 화면이다.
2. 각 게시물을 누르면 정보 화면으로 넘어가 자세한 정보를 확인할 수 있다.
3. 관리자에 의해 작성되고 관리된다.

### 3.1.2 Hardware Interfaces

- 64bit CPU, 512MB of RAM, 200MB of drive space 이상의 하드웨어 사양을 가진 모바일 및 PC 로 접속해야 한다.
- AWS EC2 프리티어 서버를 활용해 개발을 진행한다.
- 사용되는 인스턴스인 t2.micro의 경우 아래 표와 같은 리소스를 지원한다.

인스턴스	vCPU	시간당 CPU 크레딧	메모리 (GiB)	스토리지	네트워크 성능
t2.micro	1	6	1	EBS 전용	낮음에서 중간

### 3.1.3 Software Interfaces

- 구해봐요 성대에 집을 사용하기 위해서는 성균관대학교 성균인 학생 계정이 있어야 한다.
- 사용자는 Chrome, Microsoft Edge, Safari, Firefox 등의 Web Browser로 접속해야한다

### 3.1.4 Communication Interfaces

사용자는 사이트에 로그인하여 자취방을 알아보거나 여러 게시판들을 통해 자유롭게 소통하고 중고물품을 거래한다. 서버에서 필요한 정보를 받아오기 위해서는 인터넷과 연결되어 있어야 한다. 로그인 세션을 계속 유지하기 위해서는 사용자의 로그인 세션이 계속 유지되고 있어야 한다. 관리자는 공지사항 게시판을 통해 사용자에게 공지를 전달할 수 있다. 자유 게시판, 중고거래 게시판, 세부 매물정보에 댓글을 달아 사용자끼리 혹은 사용자와 관리자가 서로 소통할 수 있다. 관리자는 게시판이나 매물정보에 달린 댓글을 관리할 수 있으면 필요시 수정 혹은 삭제할 수 있다

## 3.2 Function Requirements

### 3.2.1 Use case

#### 3.2.1.1 회원가입 및 프로필 생성

Use case name	회원가입 및 프로필 생성
Actor	회원가입 하지 않은 유저
Description	신규 유저가 회원가입을 하고 프로필을 생성한다.
Normal Course	<ol style="list-style-type: none"><li>1. 신규 유저는 회원가입 버튼을 눌러 회원가입 페이지로 이동한다.</li><li>2. 신규 유저는 자신의 학교 이메일과 사용할 비밀번호를 입력한다.</li><li>3. 이미 가입된 학교 이메일 계정이라면 이미 가입한 이메일이라고 알림을 보낸다.</li><li>4. 가입되지 않은 학교 이메일 계정이라면 해당 학교 이메일 계정으로 인증번호를 전송하고 인증을 진행한다.</li><li>5. 인증이 끝나면 프로필 사진을 업로드하고 중복검사를 진행하여 닉네임을 설정한다.</li><li>6. 프로필 사진과 닉네임을 설정하지 않았다면 기본 프로필 사진과 자동 생성된 닉네임으로 프로필을 생성한다.</li><li>7. 회원가입 완료 후 메인 페이지로 이동한다.</li></ol>
Pre-Condition	<ol style="list-style-type: none"><li>1. 학교 이메일 계정을 소유한 사람이어야 한다.</li><li>2. 홈페이지에 가입한 이력이 없는 사람이어야 한다.</li></ol>
Post-Condition	개인정보 보호에 유의하여야 한다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.1.2 로그인 및 로그아웃

Use case name	로그인 및 로그아웃
Actor	회원가입을 한 유저
Description	메인 페이지에서 로그인 / 로그아웃 기능을 사용해 로그인 / 로그아웃한다.
Normal Course	1. 학교이메일과 비밀번호를 입력하여 로그인을 시도한다. 2. 학교이메일과 비밀번호가 올바르지 않다면 재시도한다. 3. 로그아웃 버튼을 눌러 로그아웃한다.
Pre-Condition	회원가입이 완료된 유저여야만 한다.
Post-Condition	
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.1.3 유저 게시판

#### 3.2.1.3.1 자유게시판

Use case name	자유게시판
---------------	-------

Actor	관리자, 로그인한 유저
Description	유저가 자유롭게 논의하고 싶은 사항이 있을 때 게시물을 작성할 수 있다. 다른 사용자들이 남긴 게시물에 대해 댓글로 의견을 개진할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 유저가 자유게시판 페이지에 들어간다.</li> <li>2. 게시물 작성 버튼을 눌러 자유롭게 논의하고 싶거나 공유하고 싶은 주제에 대해 글을 작성한다.</li> <li>3. 작성자는 해당 게시물을 언제든지 수정, 삭제할 수 있다.</li> <li>4. 로그인 한 유저들은 해당 게시글을 열람할 수 있고 게시물에 댓글을 달 수 있다.</li> <li>5. 관리자는 게시글이나 댓글에 문제가 있을 경우 수정 혹은 삭제할 수 있다.</li> </ol>
Pre-Condition	<ol style="list-style-type: none"> <li>1. 사용자가 로그인을 한 상태다.</li> <li>2. 불건전한 이용으로 인해 관리자로부터 제재를 받지 않은 사용자다.</li> </ol>
Post-Condition	<ol style="list-style-type: none"> <li>1. 게시물 작성 이후 모든 유저가 해당 게시글 열람이 가능해야 한다.</li> <li>2. 게시물 작성 직후에 작성한 게시글로 바로 접속되어 게시글을 작성한 유저가 바로 확인할 수 있도록 한다.</li> <li>3. 유저가 게시글의 삭제나 수정을 원할 경우에 삭제와 수정이 이뤄질 수 있어야 한다.</li> <li>4. 사용자가 작성한 게시글 및 댓글이 마이페이지 목록에 등록된다.</li> </ol>

Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.
-------------	---------------------------

### 3.2.1.3.2 중고거래게시판

Use case name	중고거래게시판
Actor	관리자, 로그인한 유저
Description	자취생들의 중고거래가 이뤄질 수 있는 게시판이다.
Normal Course	<ol style="list-style-type: none"> <li>1. 유저가 중고거래 게시판에 들어간다.</li> <li>2. 게시물 작성 버튼을 눌러 팔고자 하는 물품에 대한 설명과 가격정보 등을 작성한다.</li> <li>3. 게시물 작성자는 자유롭게 자신의 게시물을 수정, 삭제할 수 있다.</li> <li>4. 유저들은 게시물에 댓글을 달면서 판매자와 소통할 수 있다.</li> <li>5. 판매가 완료되었을 경우 표시가 되어야 한다.</li> <li>6. 관리자는 게시글이나 댓글에 문제가 있을 경우 수정 혹은 삭제할 수 있다.</li> </ol>
Pre-Condition	<ol style="list-style-type: none"> <li>1. 사용자가 로그인한 상태다.</li> <li>2. 불건전한 이용으로 인해 관리자로부터 제재를 받지 않은 사용자다.</li> </ol>

Post-Condition	<ol style="list-style-type: none"> <li>1. 게시물을 작성한 이후 모든 유저가 게시판 열람이 가능해야 한다.</li> <li>2. 판매를 완료한 유저는 판매완료버튼을 눌러 판매를 종료한다.</li> <li>3. 사기의 위험으로 유저의 게시글 삭제는 금지된다.</li> <li>4. 사용자가 작성한 게시글 및 댓글이 마이페이지 목록에 등록된다.</li> </ol>
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.1.3.3 공지게시판

Use case name	공지게시판
Actor	관리자, 로그인한 유저
Description	관리자가 해당 웹페이지 공지사항 게시글을 작성한다.
Normal Course	<ol style="list-style-type: none"> <li>1. 관리자가 해당 웹페이지 이용방법과 주의사항 등에 대해 게시글을 작성한다.</li> <li>2. 로그인한 유저들은 게시물을 열람할 수 있다.</li> <li>3. 관리자는 유저들이 작성한 팁이 유용하다고 생각할 경우 공지사항에 등록할 수 있다.</li> </ol>



Pre-Condition	사용자가 로그인을 한 상태다.
Post-Condition	사용자가 작성한 댓글이 마이페이지 목록에 등록된다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

#### 3.2.1.4 부동산 정보 알림

##### 3.2.1.4.1 부동산 정보

Use case name	부동산 정보
Actor	시스템, 로그인을 한 유저
Description	부동산 정보를 원하는 유저가 필터링을 통하여 원하는 매물을 확인할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 시스템은 초기에 읍전동에 존재하는 여러 부동산 매물을 불러와 목록으로 보여준다.</li> <li>2. 로그인을 한 유저는 지도의 일부분을 클릭하거나 원하는 매물 조건을 설정한다.</li> <li>3. 시스템은 유저가 설정한 조건에 알맞은 부동산 매물을 목록으로 보여준다.</li> </ol>

Pre-Condition	사용자가 로그인을 한 상태다.
Post-Condition	사용자가 부동산 매물을 클릭하면 매물 세부정보 페이지로 이동한다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

#### 3.2.1.4.2 매물 세부 정보

Use case name	매물 세부 정보
Actor	관리자, 시스템, 로그인을 한 유저
Description	해당 매물에 대한 부동산 정보를 원하는 유저가 더 자세한 정보를 얻을 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>부동산 정보 목록에서 매물 세부 정보를 누르면 데이터베이스에서 선택한 매물 세부 정보를 가져온다.</li> <li>로그인을 한 유저는 댓글 칸에 해당 매물에 관한 정보나 의견을 달 수 있다.</li> <li>관리자는 해당 정보가 부정확하거나 논란의 여지가 있을 경우 삭제할 수 있다.</li> </ol>

Pre-Condition	로그인을 한 사용자가 부동산 정보 use case에서 해당 매물에 대한 세부 정보를 원해 클릭을 한 상태다.
Post-Condition	사용자가 확인한 매물 정보가 마이페이지에서 최근에 확인한 매물정보 칸에 등록되어야 한다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.1.5 마이페이지

#### 3.2.1.5.1 마이페이지 열람

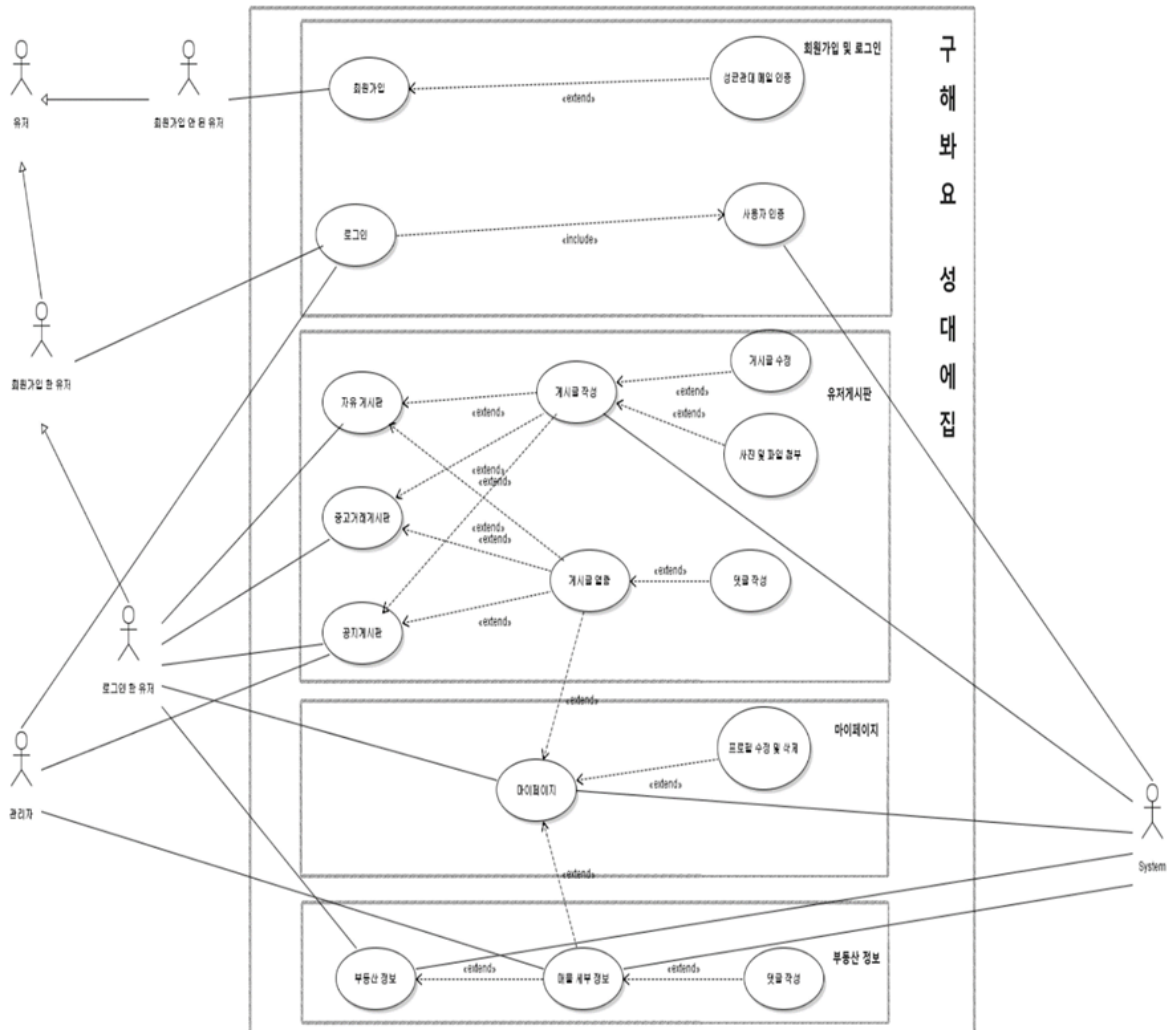
Use case name	마이페이지 열람
Actor	로그인을 한 유저
Description	웹사이트상에서의 자신의 기록을 확인할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 유저가 마이페이지에 들어간다.</li> <li>2. 마이페이지에서 유저는 본인의 프로필 및 작성한 게시글, 댓글, 최근에 본 매물 세부 정보목록을 확인할 수 있다.</li> <li>3. 사용자가 기록이 없을 경우 공란으로 표시된다.</li> </ol>

Pre-Condition	사용자가 로그인을 한 상태다.
Post-Condition	사용자가 해당 목록을 클릭하면 해당 페이지로 바로 이동한다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.1.5.2 프로필 수정 및 삭제

Use case name	프로필 수정 및 삭제
Actor	로그인을 한 유저
Description	마이페이지에서 자신의 프로필을 수정하거나 삭제한다.
Normal Course	<ol style="list-style-type: none"> <li>1. 프로필을 수정하거나 삭제하기 위해 다시 한 번 비밀번호를 검증한다.</li> <li>2. 비밀번호가 검증되었다면 기존 유저는 자신의 프로필을 자유롭게 수정하거나 삭제할 수 있다.</li> <li>3. 프로필에 대한 수정이나 삭제가 이루어졌다면 DB에 있는 기존 유저에 관한 모든 정보들도 업데이트를 한다.</li> </ol>
Pre-Condition	사용자가 로그인을 한 상태이고 마이페이지에 접속하였다.
Post-Condition	유저의 정보는 안전하게 보호되어야 한다.
Assumptions	사용자의 네트워크 사용이 가능하다고 가정한다.

### 3.2.2 Use Case Diagram



### 3.3 Non-functional Requirements

### 3.3.1. Product Requirements

### 3.3.1.1 Usability Requirements

- 사용자가 웹사이트 이용방법을 별도의 도움 없이 이해할 수 있도록 사이트의 UI와 아이콘을 직관적으로 개발해야 한다.
- 페이지에 부여된 텍스트와 아이콘만으로 사용자가 기능을 쉽게 이해할 수 있어야 한다.

- 전체적인 사이트 디자인은 성균관대학교를 상징하는 진한 녹색 바탕에 세부적인 요소들에 흰색과 노란색을 사용하였다.
- 언어는 한국어를 기본으로 한다.

#### 3.3.1.2 Efficiency Requirements

- 게시글 등록, 각 페이지 이동은 2sec를 넘지 않도록 한다.
- 매물 세부 정보, 부동산 정보를 가져오는 속도는 5sec를 넘지 않도록 한다.

#### 3.3.1.3 Dependability Requirements

- 사용자가 ID, Password를 정확하게 입력했을 경우 항상 로그인 가능해야 한다.
- 유저가 사진을 업로드 할 때 끊김이 없어야 한다.
- 부동산 정보 및 매물 정보에 대한 업데이트가 3시간마다 자동으로 이루어져야 한다.
- 부동산 정보를 반영한 지도에 대한 줌아웃기능이 서버의 불안정으로 끊기는 빈도를 줄여야 한다.

#### 3.3.1.4 Security Requirements

- 사용자는 회원 가입시 성균관대학교 메일을 통해 성균관대학교 학생임을 인증해야만 회원가입을 할 수 있다.
- 회원가입을 한 사용자의 개인정보는 암호화되어 안전하게 보호되어야 한다.
- 사이트를 이용하는 도중 사용자의 부적절한 행위를 적발하면 관리자는 이를 확인하고 적절한 조치를 취할 수 있어야 한다.

### 3.3.2 Organization Requirements

#### 3.3.2.1 Environment Requirements

- 본 웹사이트는 React와 Figma을 기반으로 개발되며 Web browser(

Chrome v45, Edge v12, Firefox v34 등) 이상의 버전에서 사용할 수 있도록 개발되어야 한다

### 3.3.2.2 Development Requirements

- 웹 UI는 Figma를 통해 진행하며 아이콘, 텍스트를 직관적으로 표기한다.
- React를 사용해 해당 사이트의 Front End 개발을 진행하도록 한다.
- Flask와 MongoDB를 사용해 해당 사이트의 Back End 개발을 진행하도록 한다.

### 3.3.3 External Requirements

#### 3.3.3.1 Regulatory Requirements

- 본 명세서에서 사용하는 오픈소스 프로그램들의 사용 규정에 맞게 시스템이 개발되어야 한다.
- 개발에 있어 다루는 시스템들은 저작권 보호법을 위반해서는 안 되며 원작자의 요청 시 이를 삭제해야 한다.

#### 3.3.3.2 Ethical Requirements

- 해당 사이트를 운영함에 있어서 부적절한 방법을 통한 금전적 이득을 취하지 않아야 한다.
- 중고거래 과정에서 부적절한 금전 이득이나 사기 행위가 발생하지 않도록 주의하여야 한다.

### 3.3.4 Organizing the Specific Requirements

#### 3.3.4.1 Context Model

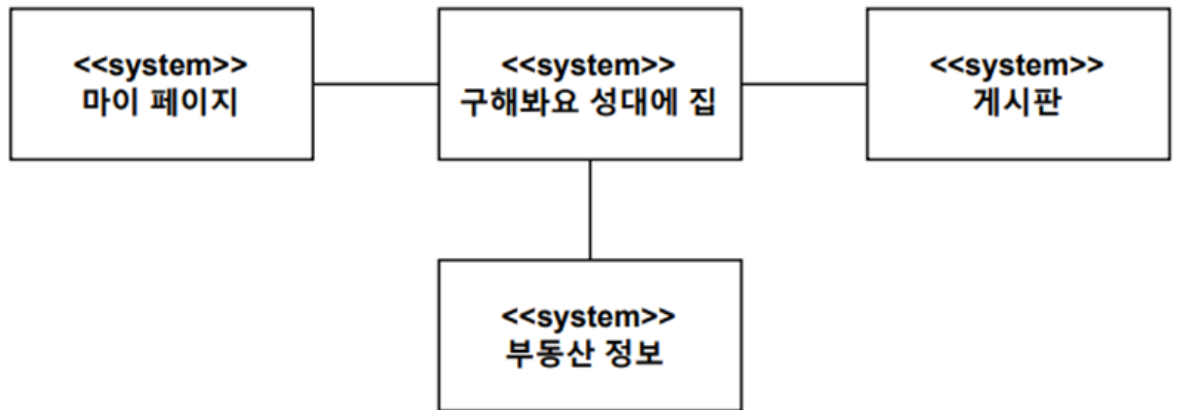


그림 1 - Context model

### 3.4 Logical Database Requirements

Mongo DB을 통해 유저 정보가 담긴 데이터베이스를 관리한다. 시스템은 이메일, password, 프로필 사진 등의 사용자 정보와 사용자가 작성한 게시글 및 댓글 정보, 부동산 정보, 매물 세부 정보를 데이터베이스에 저장한다. 게시글을 생성할 때나 부동산 정보를 업데이트할 때는 사용자와 관리자 계정을 구분하여 권한이 부여되어야 한다. 데이터베이스는 정보를 처리하기 위한 기본 성능을 가질 수 있도록 관리되어야 한다.

### 3.5 Design Constraints

본 시스템은 현재 에브리타임 앱과 직방 앱을 참고하여 대략적인 인터페이스 구조를 따른다. 모바일 및 OS 운영 체제를 사용하는 Web Browser로 시스템에 액세스할 수 있어야 하며 HTTP 통신이 가능하도록 설계되어야 한다.

### 3.6 Standards Compliance



- 웹 기반 시스템 관리 도구는 HTML 표준을 따른다.
- 프로그램 버전 관리 시스템으로는 github을 사용한다.
- 시스템 개발을 위해 Windows, macOS, Linux 모두 가능하며, 64-bit CPU 최소 512MB 이상 RAM에서 최소 500MB 의 여유공간이 디바이스 존재해야 한다.

## 3.7 Standards Architecture

### 3.7.1 Limitation and Assumption

본 시스템은 성균관대학교 학생들만 인증을 받을 수 있도록 설계되었다. 따라서 학교 근처에 사는 지역 주민들은 사용이 불가능하다고 가정하였다. 또한 부동산 정보와 매물 세부 정보에 대한 검증은 자체적으로 거치지 않고 정보를 얻어온 프로그램에 의존하기 때문에 정보에 대한 신뢰 정도에 한계를 가진다.

### 3.7.2 Anticipated Changes

본 시스템은 학교 근처에 사는 지역 주민들에게도 유용할 것이다. 그렇기에 위에 서술한 한계점을 극복하기 위하여 위치기반 회원가입 인증 시스템을 도입하여 한계점을 극복할 수 있을 것이다. 더불어, 부동산 관련 전문가 인력이 총원된다면 부동산 정보에 대한 자체 검증을 통해 정보에 대한 신뢰도 또한 높일 수 있을 것이다.