2022년 충북대학교 소프트웨어학과 졸업작품연구과제

**[ TensorFlow를 활용한 인공지능 클라이밍 피드백 서비스 ]**

**2차 빌드 개발 명세서**

2022년 12월 8일

| 문서번호 | : | 2022-클클-Doc-006 |
| --- | --- | --- |
| 소 속 | : | 충북대학교 소프트웨어학과 |
| 팀 명 | : | 클클(Climber Claim) |
| 팀 원 | : | 이승하, 이영은, 김예원 |
| 교 수 | : | 이건명 교수님 |

**제/개정 이력**

| **버전** | **날짜** | **작성자 성명** | **제/개정 사항** | **비고** |
| --- | --- | --- | --- | --- |
| v0.1 | 22. 11. 25. | 김예원 | 초안 작성 |  |
| v0.2 | 22. 12. 06. | 이승하 | 테스트 시나리오 및 결과 목록 작성 |  |
| v0.3 | 22. 12. 06. | 김예원 | 기타 구현 참고사항 작성 |  |
| v0.4 | 22. 12. 08. | 이영은 | Class Diagram 수정  Window Navigation Diagram 추가 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

목 차

[**1. 개발 개요**](#_heading=h.gjdgxs) **1**

[**2. 분석 명세**](#_heading=h.3znysh7) **1**

[**2.1 Use Case Diagram**](#_heading=h.dr5i7g5gevi1) **1**

[**2.2 Class Diagram**](#_heading=h.3cjnl47gyhzy) **2**

[**3. 설계 명세**](#_heading=h.2et92p0) **3**

[**3.1 Window Navigation Diagram**](#_heading=h.tyjcwt) **3**

[**3.2 User Interface Design**](#_heading=h.xszg9nl4v9pq) **4**

[**3.3 Database**](#_heading=h.32hioqz) **5**

[**4. 테스트 데이터 및 결과 목록**](#_heading=h.1hmsyys) **5**

[**5. 기타 구현 참고사항**](#_heading=h.vx1227) **6**

[**5.1 OSS / Reuse**](#_heading=h.3fwokq0) **6**

[**5.2 차기 구현 시 반드시 고려할 사항**](#_heading=h.1oim5m4nkznp) **6**

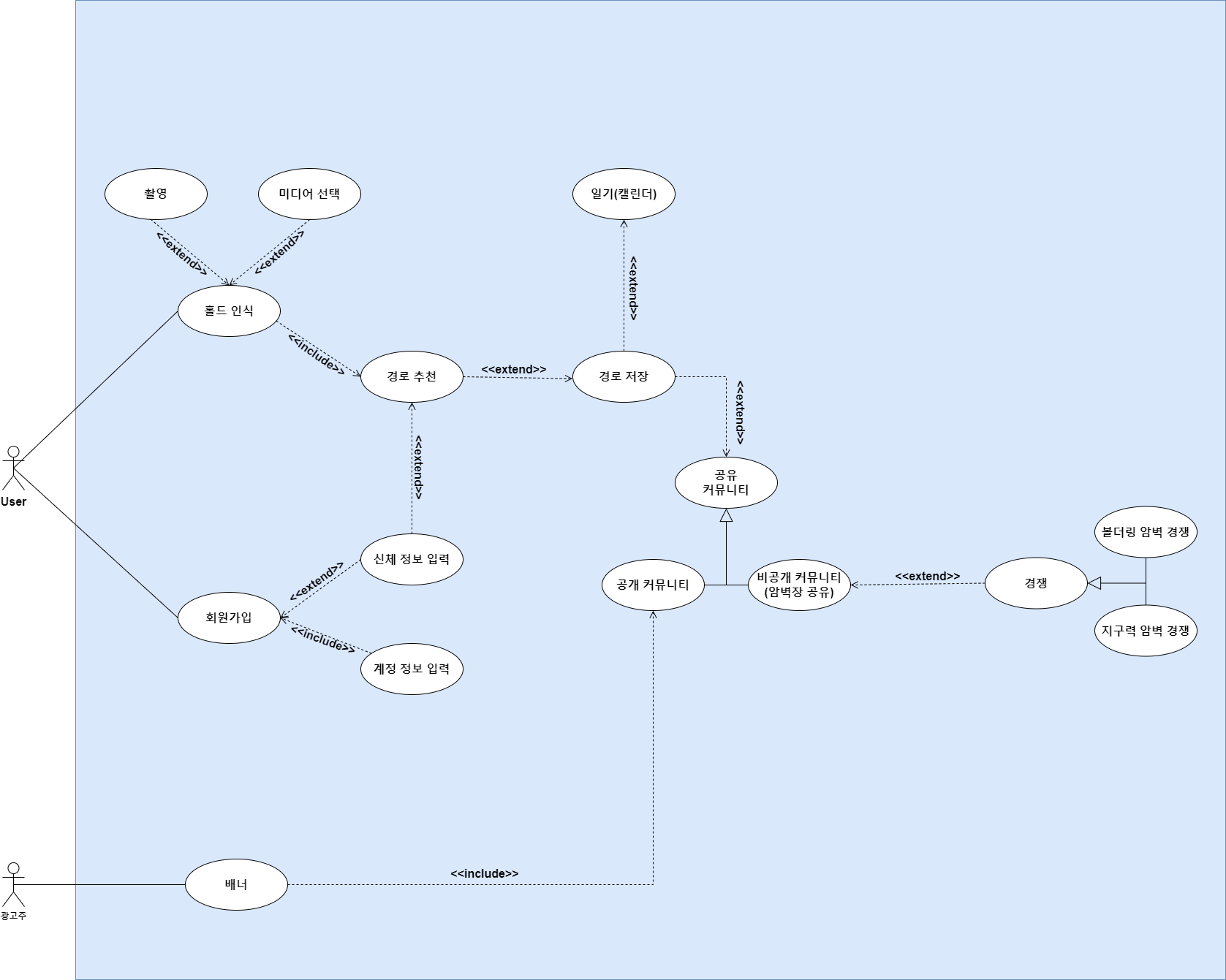
# **1. 개발 개요**

| 번호 | 태스크명 | 담당자 | 시작일 | 종료일 | 비고 |
| --- | --- | --- | --- | --- | --- |
| 1 | 모델 성능 향상을 위한 데이터 수집 및 전처리 | 이승하 | 22. 11. 18. | 22. 11. 22. |  |
| 2 | 모델 재학습 | 이승하 | 22. 11. 25. | 22. 12. 03. |  |
| 3 | UI 재구조화 | 이영은 | 22. 11. 18. | 당일 |  |
| 4 | UI 추가 개발 | 이영은 | 22. 11. 19. | 22. 11. 26. |  |
| 5 | 웹 API 개발 | 김예원 | 22. 12. 03. | 22. 12. 06. |  |
| 6 | 회원가입 완성 | 김예원 | 22. 11. 17. | 22. 11. 19. |  |

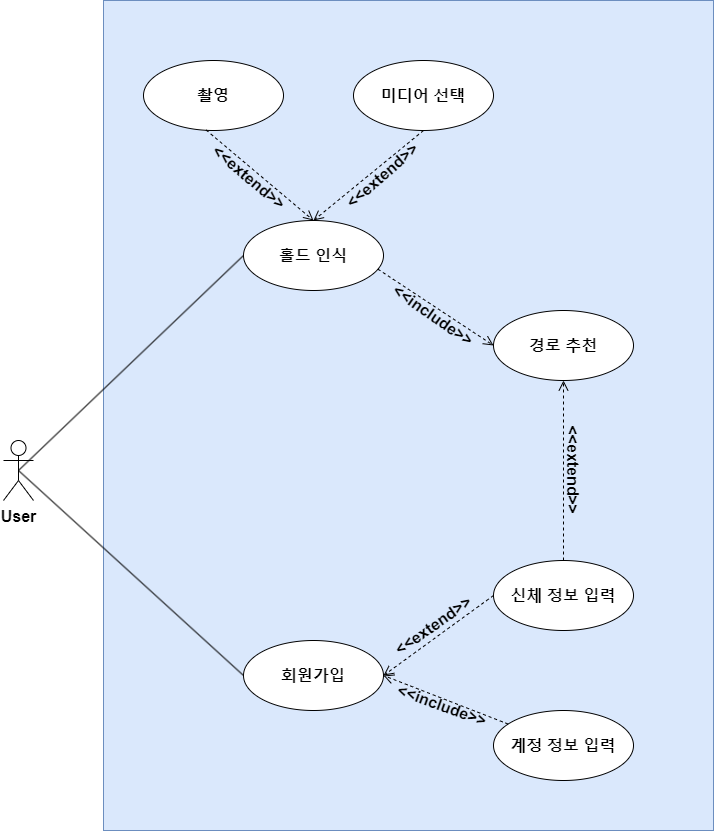
# **2. 분석 명세**

## **2.1 Use Case Diagram**

for whole system

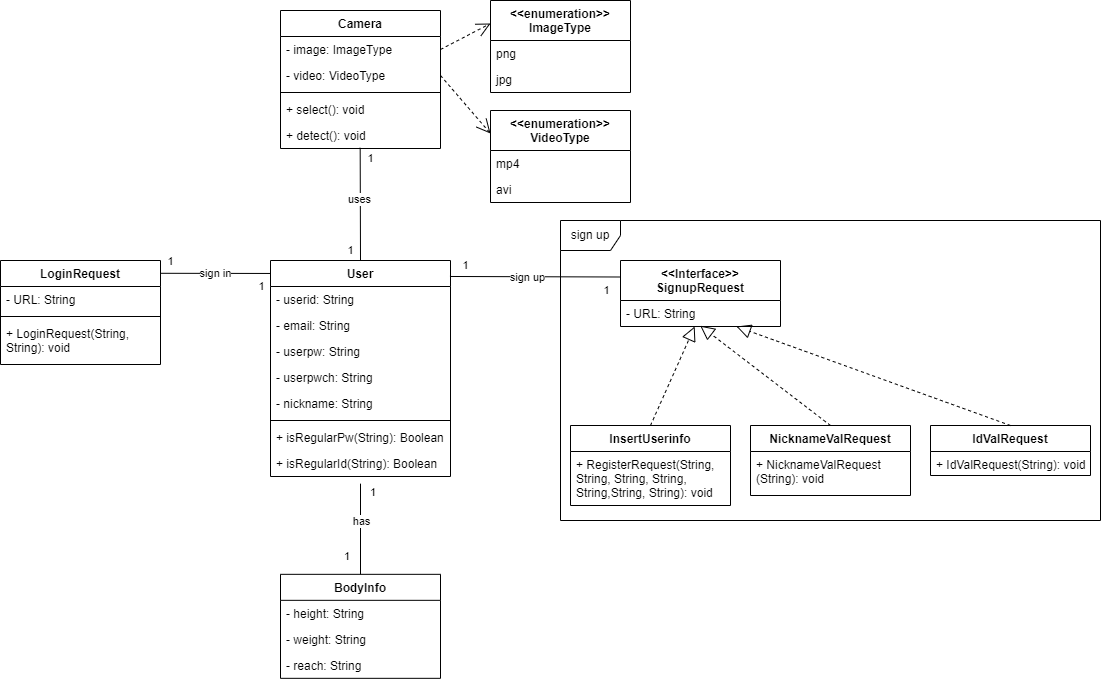


for the function



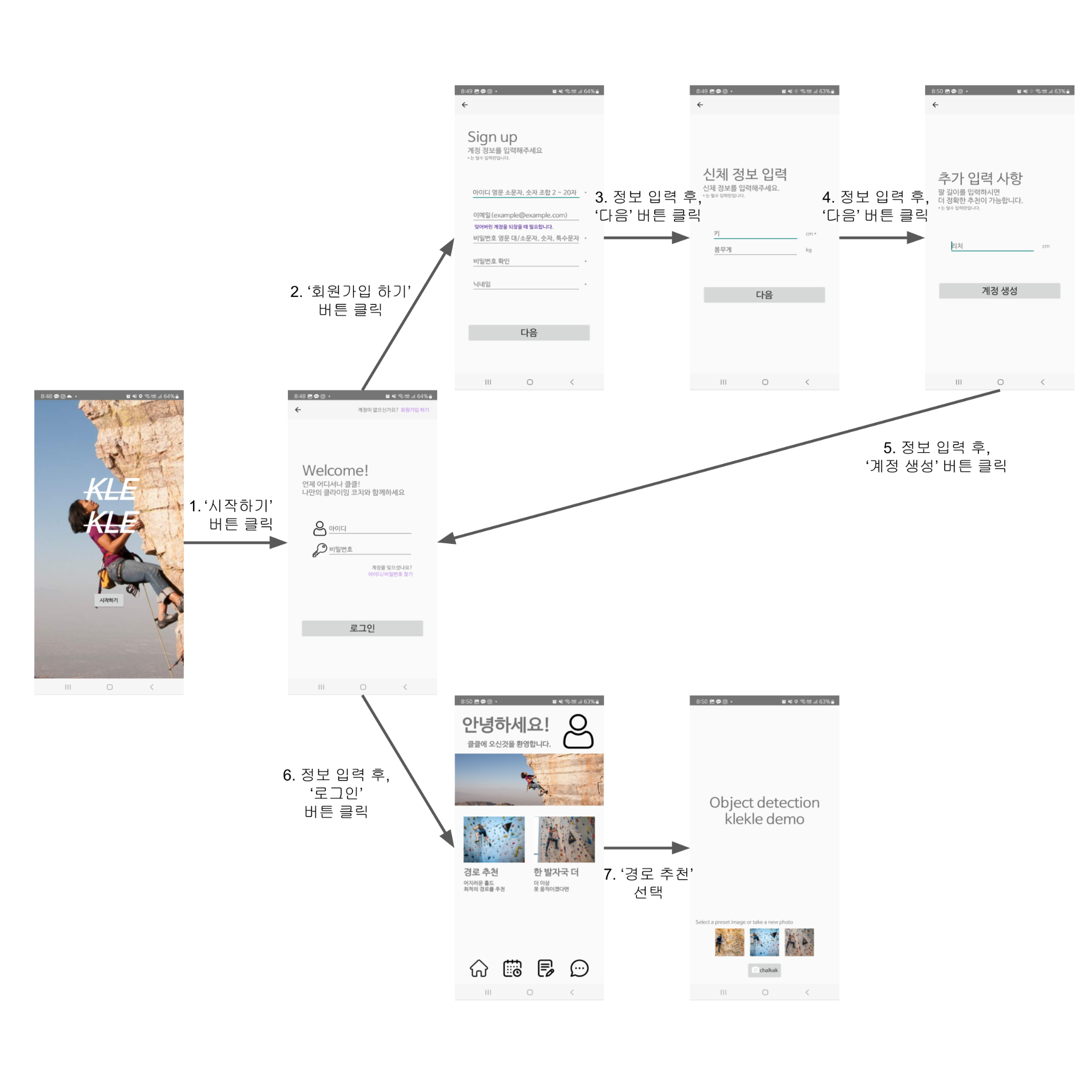
## **2.2 Class Diagram**

for the function



# **3. 설계 명세**

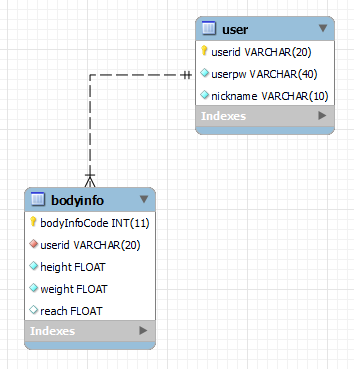
## **3.1 Window Navigation Diagram**



## **3.2 User Interface Design**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## **3.3 Database**



# **4. 테스트 데이터 및 결과 목록**

| 번호 | 테스트 시나리오 | 테스트 결과 |
| --- | --- | --- |
| 1 | 아이디, 닉네임 중복 기능의 확인을 위해, 중복되는 아이디, 닉네임을 입력해본다. | 이미 데이터베이스에 입력되어 있는 아이디, 닉네임을 입력하고 ‘다음’ 버튼을 터치했을 시, “이미 존재하는 아이디/닉네임 입니다.”라는 에러 메시지를 출력하고, 다음 화면으로 넘어가지 않게 막혔다.(정상 작동) |
| 2 | 아이디, 닉네임 중복 기능의 확인을 위해, 중복되지 **않는** 아이디, 닉네임을 입력해본다. | 정상적으로 다음 화면으로 넘어가는 모습을 확인했다. |
| 3 | 회원가입 단계에서, 신체 정보 입력까지 완전히 입력에 완료했다. | ‘회원가입 완료’ 버튼 클릭 시 데이터베이스에 정상적으로 입력되었다.  비밀번호는 암호화 함수를 거쳐 정상적으로 암호화되어 입력되었다. |
| 4 | 학습이 완료된 인공지능 모델에 test data들을 넣어본다. | 정성적, 정량적으로 결과를 평가했을 때, 새로 학습한 인공지능 모델이 기존의 모델보다 성능이 좋아졌음을 확인할 수 있었다. |
| 5 | Window Terminal에서, Flask를 이용한 REST API의 호출이 정상적으로 이루어지는지 테스트한다.  curl -X POST -F image=@이미지경로 'http://localhost:5000/v1/object-detection/hold' | 터미널에 해당 이미지에서 hold를 detect한 결과가 JSON 형식으로 출력되었다. |

# **5. 기타 구현 참고사항**

## **5.1 OSS / Reuse**

MySQL - Database management

labelImg - 인공지능 학습 데이터 전처리에 이용

YOLOv5 - Climbing hold detection model 재학습에 이용

Flask - API 개발에 이용

## **5.2 차기 구현 시 반드시 고려할 사항**

경로 추천 기능 구현 시, 다음을 고려하도록 한다.

* Flask REST API에서 받아오는 detection 결과는, 바운딩 박스의 좌측상단, 좌측하단, 우측상단, 우측하단의 좌표를 JSON 형태로 되어있다.
  + 해당 데이터를 파싱하여 필요한 데이터를 적절히 사용하도록 한다.
* API가 올려져 있는 서버는 평상시에 다운된 상태이다.
  + 서버를 켜고 포트포워딩을 마친 후, 개발 및 테스트를 행하도록 한다.
* 경로 추천 결과를 데이터베이스에 저장할 수 있도록 한다.
  + 이에 필요한 database entity를 추가 설계하도록 한다.