

**Smart Safety Home**

For One-Person Households

**Software Requirement Specification**

2022.05.01.

**Introduction to Software Engineering**

**TEAM 8**

| Team Leader | 김승현 |
| --- | --- |
| Team Member | 김영욱 |
| Team Member | 윤진희 |
| Team Member | 임성용 |
| Team Member  Team Member | 하솔비  이상협 |

CONTENTS

[**1. Introduction** -](#_heading=h.30j0zll) 8 -

[1.1. Purpose -](#_heading=h.1fob9te) 8 -

[1.2. Scope -](#_heading=h.2et92p0) 8 -

[1.3. Definitions, Acronyms, and Abbreviation -](#_heading=h.3dy6vkm) 9 -

[1.4. References -](#_heading=h.17dp8vu) 10 -

[1.5. Overview - 1](#_heading=h.26in1rg)1 -

[**2. Overall Description** - 1](#_heading=h.35nkun2)2 -

[2.1. Product perspective - 1](#_heading=h.44sinio)2 -

[2.1.1. Market status - 1](#_heading=h.z337ya)2 -

[2.1.2. Overall structure - 1](#_heading=h.1y810tw)3 -

[2.2. Product functions - 1](#_heading=h.1ci93xb)3 -

[2.2.1. 회원가입 및 로그인 - 1](#_heading=h.2bn6wsx)3 -

[2.2.2. 화재감지 - 1](#_heading=h.3as4poj)3 -

[2.2.3. 가스밸브 - 1](#_heading=h.49x2ik5)4 -

[2.2.4. 비상연락망 설정 및 신고 설정 - 1](#_heading=h.147n2zr)4 -

[2.2.5. 모션감지 카메라 - 1](#_heading=h.23ckvvd)4 -

[2.2.6. 설정 - 14 -](#_heading=h.32hioqz)

[2.2.6.1. 얼굴 등록 - 14 -](#_heading=h.32hioqz)

[2.2.6.2. 사용할 기능 선택 - 14 -](#_heading=h.32hioqz)

[2.2.6.3. 비상연락망 설정 - 14 -](#_heading=h.32hioqz)

[2.2.6.4. 카메라 연동 - 1](#_heading=h.32hioqz)4 -

[2.3. User Classes and Characteristics - 1](#_heading=h.41mghml)5 -

[2.3.1. User - 1](#_heading=h.vx1227)5 -

[2.3.2. System Manager - 1](#_heading=h.1v1yuxt)5 -

[2.4. Operating Environment - 1](#_heading=h.4f1mdlm)5 -

[2.4.1. Hardware - 1](#_heading=h.2u6wntf)5 -

[2.4.2. Software - 1](#_heading=h.19c6y18)5 -

[2.5. Design and Implementation Constraints - 1](#_heading=h.3tbugp1)5 -

[2.6. User Documentation - 1](#_heading=h.28h4qwu)6 -

[2.7. Assumptions and Dependencies - 1](#_heading=h.nmf14n)6 -

[**3. Specific Requirements** - 1](#_heading=h.37m2jsg)7 -

[3.1. External Interface Requirements - 1](#_heading=h.1mrcu09)7 -

[3.1.1. User Interfaces - 1](#_heading=h.46r0co2)7 -

[3.1.2. Hardware Interfaces -](#_heading=h.2r0uhxc) 29 -

[3.1.3. Software Interfaces -](#_heading=h.3q5sasy) 30 -

[3.1.4. Communication Interfaces - 3](#_heading=h.kgcv8k)1 -

[3.2. Functional Requirements - 32 -](#_heading=h.1jlao46)

[3.2.1. Use Case - 32 -](#_heading=h.43ky6rz)

[3.2.2. Use Case Diagram - 4](#_heading=h.haapch)0 -

[3.2.3. Data Dictionary - 4](#_heading=h.1gf8i83)1 -

[3.2.4. Data Flow Diagram -](#_heading=h.279ka65) 46 -

[3.3. Product Requirements -](#_heading=h.36ei31r) 47 -

[3.3.1. Performance Requirements -](#_heading=h.1ljsd9k) 47 -

[3.3.2. Space Requirements](#_heading=h.45jfvxd) - 47 -

3.3.3. Security Requirements [-](#_heading=h.45jfvxd) 47 -

3.3.4. Usability Requirements - 48 -

3.3.5. Dependability Requirements - 48 -

[3.4. Organizational Requirements -](#_heading=h.2koq656) 48 -

[3.4.1. Environmental Requirements - 48 -](#_heading=h.1ljsd9k)

3.4.2. Operational Requirements - 48 -

[3.4.3. Development Requirements - 49 -](#_heading=h.1ljsd9k)

[3.5. External Requirements -](#_heading=h.zu0gcz) 49 -

[3.5.1. Regulatory Requirements - 49 -](#_heading=h.1ljsd9k)

[3.5.2. Ethical Requirements - 49 -](#_heading=h.1ljsd9k)

[3.5.3. Accounting Requirements - 49 -](#_heading=h.1ljsd9k)

[3.5.4. Safety/Security Requirements - 50 -](#_heading=h.1ljsd9k)

[3.6. Organizing the Specific Requirements - 5](#_heading=h.3x8tuzt)1 -

[3.6.1. Context Model - 5](#_heading=h.2ce457m)1 -

[3.6.2. Process Model - 5](#_heading=h.3bj1y38)2 -

[3.6.3. Interaction Model - 5](#_heading=h.4anzqyu)7 -

[3.6.4. Behavior Model - 5](#_heading=h.2pta16n)8 -

[**4. Supporting Information** -](#_heading=h.42ddq1a) 60 -

[4.1. Software Requirement Specification -](#_heading=h.2hio093) 60 -

[4.2. Document History - 6](#_heading=h.wnyagw)1 -

**LIST OF FIGURE**

[[Figure 1] System Structure - 1](#_heading=h.2xcytpi)3 -

[[Figure 2] Use Case Diagram -](#_heading=h.319y80a) 40 -

[[Figure 3] Entity Relationship Diagram - 4](#_heading=h.3s49zyc)5 -

[[Figure 4] Data Flow Diagram -](#_heading=h.meukdy) 46 -

[[Figure 5] Context Model -](#_heading=h.rjefff) 51 -

[[Figure 6]](#_heading=h.1qoc8b1) Process model - 총괄 [-](#_heading=h.1qoc8b1) 52 -

[[Figure 7] Process model - 계정관리 -](#_heading=h.14ykbeg) 53 -

[[Figure 8] Process model - 비상연락망 - 54 -](#_heading=h.14ykbeg)

[[Figure 9]](#_heading=h.243i4a2) [Process model -](#_heading=h.14ykbeg) 화재, 가스, 카메라 [-](#_heading=h.243i4a2) 55 -

[[Figure 10]](#_heading=h.243i4a2) [Process model -](#_heading=h.14ykbeg) 설정 - 56 -

[Figure 11] Use Case Diagram(Figure 2 동일) - 57 -

[Figure 12] Data Flow Diagram(Figure 4 동일) - 58 -

[Figure 13] Sequence Diagram - 59 -

**LIST OF TABLES**

[[Table 1] Table of acronyms and abbreviations -](#_heading=h.4d34og8) 9 -

[[Table 2] Table of terms and definitions -](#_heading=h.2s8eyo1) 9 -

[[Table 3] Starting Interface - 1](#_heading=h.2lwamvv)7 -

[[Table 4] Main Interface - 1](#_heading=h.111kx3o)8 -

[[Table 5] Account Registration Interface 1 - 1](#_heading=h.206ipza)9 -

[[Table 6] Account Registration Interface 2 -](#_heading=h.4k668n3) 20 -

[[Table 7] Setting Interface -](#_heading=h.2zbgiuw) 21 -

[[Table 8] Face Registration Interface - 2](#_heading=h.1egqt2p)3 -

[[Table 9] Function Enabling Setting Interface - 2](#_heading=h.3ygebqi)4 -

[[Table 10] Reports Setting Interface - 2](#_heading=h.2dlolyb)5 -

[[Table 11] Emergency Contact Interface - 2](#_heading=h.sqyw64)6 -

[[Table 12] Camera Interface 1 - 2](#_heading=h.3cqmetx)7 -

[[Table 13] Camera Interface 2 - 2](#_heading=h.1rvwp1q)8 -

[[Table 14] Hardware Interface - 2](#_heading=h.4bvk7pj)9 -

[[Table 15] Hardware Interface IOT -](#_heading=h.1664s55) 29 -

[[Table 16] Software Interface -](#_heading=h.25b2l0r) 30 -

[[Table 17] Communication Interface -](#_heading=h.34g0dwd) 31 -

[[Table 18] 회원가입 Use Case -](#_heading=h.2iq8gzs) 32 -

[[Table 19] 로그인 Use Case - 3](#_heading=h.xvir7l)2 -

[[Table 20] 자동로그인 Use Case - 3](#_heading=h.3hv69ve)3 -

[[Table 21] 로그아웃 Use Case - 3](#_heading=h.1x0gk37)3 -

[[Table 22] 아이디/비밀번호 찾기 Use Case - 3](#_heading=h.4h042r0)4 -

[[Table 23] 화재감지기 Use Case - 3](#_heading=h.2w5ecyt)5 -

[[Table 24] 가스밸브 Use Case - 3](#_heading=h.1baon6m)5 -

[[Table 25] 비상연락망 관리 Use Case - 3](#_heading=h.3vac5uf)6 -

[[Table 26] 음성인식 단어 관리 Use Case - 3](#_heading=h.2afmg28)6 -

[[Table 27] 신고 Use Case - 3](#_heading=h.pkwqa1)7 -

[[Table 28] 카메라 Use Case - 3](#_heading=h.39kk8xu)7 -

[[Table 29] 얼굴 등록 관리 Use Case -](#_heading=h.1opuj5n) 38 -

[[Table 30]](#_heading=h.48pi1tg) [사용 기능 선택 Use Case](#_heading=h.1opuj5n) [-](#_heading=h.48pi1tg) 39 -

[[Table 31] User -](#_heading=h.2nusc19) 41 -

[[Table 32] Function -](#_heading=h.1302m92) 42 -

[[Table 33] Fire Detector -](#_heading=h.3mzq4wv) 42 -

[[Table 34] Gas -](#_heading=h.2250f4o) 42 -

[[Table 35] Camera -](#_heading=h.40ew0vw) 43 -

[[Table 36] Contact -](#_heading=h.2fk6b3p) 43 -

[[Table 37] Word - 4](#_heading=h.upglbi)4 -

[[Table 38] Image - 4](#_heading=h.3ep43zb)4 -

[[Table 39] Face - 4](#_heading=h.1tuee74)4 -

[[Table 40] Document History -](#_heading=h.3gnlt4p) 60 -

# **1. Introduction**

## **1.1. Purpose**

이 문서는 Smart Safety home 서비스를 제공하기 위한 소프트웨어 요구 사항 Software Requirements Specification, SRS) 이다. Smart Safety home 서비스는 증가하는 1인 가구 거주자들이 보다 효과적으로 집을 관리할 수 있는 애플리케이션으로, 안전하고 편안한 생활을 할 수 있도록 돕는 것을 목적으로 한다. 이 서비스는 성균관대학교 소프트웨어 공학개론 8팀이 이 목적에 맞는 요구사항을 요약하고 정리하였으며, 또한 정리한 정보를 분석하여 아래 기술한 내용을 바탕으로 시스템을 설계 및 구현한다. 따라서 주요 저자는 8 팀이며, 주요 독자 및 발표 대상은 소프트웨어 공학 개론 강의의 교수, TA 및 수강생이다. 앞서 말한 것처럼 이 문서의 궁극적인 목적은 1인 가구의 거주자들이 편안하고 안전한 주거 생활을 할 수 있도록 돕는 새로운 시스템 프로그램 요구사항의 사양을 개략적으로 설명하고 게시하는 것이며, 소프트웨어 공학개론 8팀은 이에 맞춰 발전하는 스마트홈 시장을 따라 다양한 목적의 센서와 기능을 사용해 주요 서비스 기능을 구현 및 개발한다.

## **1.2. Scope**

이 서비스는 사용자가 외·내부에서 보다 효율적으로 집을 관리할 수 있게 하고, 편안하고 안전한 생활을 하도록 돕는 것을 목적으로 하는 애플리케이션이다. 이 서비스를 지원하기 위하여 온도감지, 움직임 감지, 연기 감지 등 다양한 기능을 가진 센서들이 사용된다. 이 센서들을 사용해 서비스는 화재감지 및 가스누출 예방, 수상한 움직임 감지 및 인물 판별, 그리고 비상 신고 기능을 지원한다. 이용자는 이 서비스를 자신에게 필요한 것을 골라 취사 선택하여 사용할 수 있다. 우리는 이용자가 그러한 기능들을 잘 이용하여 편안한 주거 생활을 할 수 있기를 바란다.

## **1.3. Definitions, Acronyms, and Abbreviation**

The following table explains the acronyms and abbreviations used in this document.

[Table 1] Table of acronyms and abbreviations

| **Acronyms & Abbreviations** | **Explanation** |
| --- | --- |
| RAM | Random Access Memory |
| IoT | Internet of Things |
| UI | User Interface |
| CPU | Central Processing Unit |
| OS | Operating System |
| GUI | Graphical User Interface |
| API | Application Programming Interface |
| BIOS | Basic Input/Output System |

The following table defines certain technical terms used in this document.

[Table 2] Table of terms and definitions

| **Terms** | **Definitions** |
| --- | --- |
| User | Someone who uses a system |
| System administrator | Someone who quantify the keywords included in the reviews for each laptop and manage the system |
| Query | request for data or information from a database table or combination of tables. |
| Algorithm | A set of rules or procedures followed by a computer in problem-solving operations |
| Client (user device) | A user device/user that connected to server |
| Host | any hardware device that has the capability of permitting access to a network via a user interface, specialized software, network address, protocol stack, or any other means |
| Server | A computer or computer program which manages access to a centralized resource or service in a network |
| Software | The programs and other operating information used by a computer |
| Network | For connect devices together so that they can share information. In this system, it usually means internet |
| Open source | refers to something people can modify and share because its design is publicly accessible |

## **1.4. References**

* IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEEXplore Digital Library  
  http://ieeexplore.ieee.org/Xplore/guesthome.jsp
* Team 7. “Software Requirement Specification”. SKKU, Last Modified: April. 25, 2021.  
  https://github.com/skkuse/2021spring\_41class\_team7/blob/main/IEEE\_SRS.doc

https://github.com/skkuse/2021spring\_41class\_team7/blob/main/doc/Team7\_SRS.docx

* Smart Home Project (SHP). “System Requirements Document for Safe Home“. Dr. S. Mitra, Dr. J. Wong, and Dr. C. Chang. (2004)

## **1.5. Overview**

이 소프트웨어 요구사항 사양 문서는 총 네 개의 장으로 구성되어 있다. 첫 번째 장에서는 문서의 개략적인 설명을 통해 목적을 나타내며, 두 번째 장에서는 이 서비스의 보다 자세한 목적과 서비스가 필요한 market status, 여러 인터페이스, 시스템 기능 및 다른 시스템과의 상호작용을 나타낸 diagram과 각 기능의 간략한 소개가 포함된다. 각 기능들에 대한 간단한 소개가 필요하다면 두 번째 장에서 설명하고 있으므로 살펴보길 바란다. 세 번째 장에서는 다양한 시스템 인터페이스 및 두 번째 장에서 설명했던 소프트웨어/하드웨어의 기능과 특성에 대해 보다 자세한 설명이 포함되어 있다. 사용자는 각 기능들이 어떤 기능의 센서를 필요로 하는지, 그 센서를 구동하기 위해서는 어떤 요구사항을 가지는지에 대해 궁금증을 가졌을 때, 세 번째 장에서 도움을 받을 수 있다. 이 섹션에서는 이해를 돕기 위해 각 기능들의 기초적인 구동 사진이 포함되어 있다. 네 번째 장에서는 SRS 문서의 설명이 포함되어 있다. 이 SRS가 어떤 문서 기준을 기반으로 했는지 알 수 있다. 또한 Smart Safety home Service의 SRS 작성 연력이 기록되어 있다. 팀 8의 구성원들은 이 프로젝트의 개발에 똑같이 기여했으며, 우리가 즐겁게 문서를 작성하고 체크한 만큼 많은 이용자들이 이 문서를 읽어주길 바란다.

# 

# 

# **2. Overall Description**

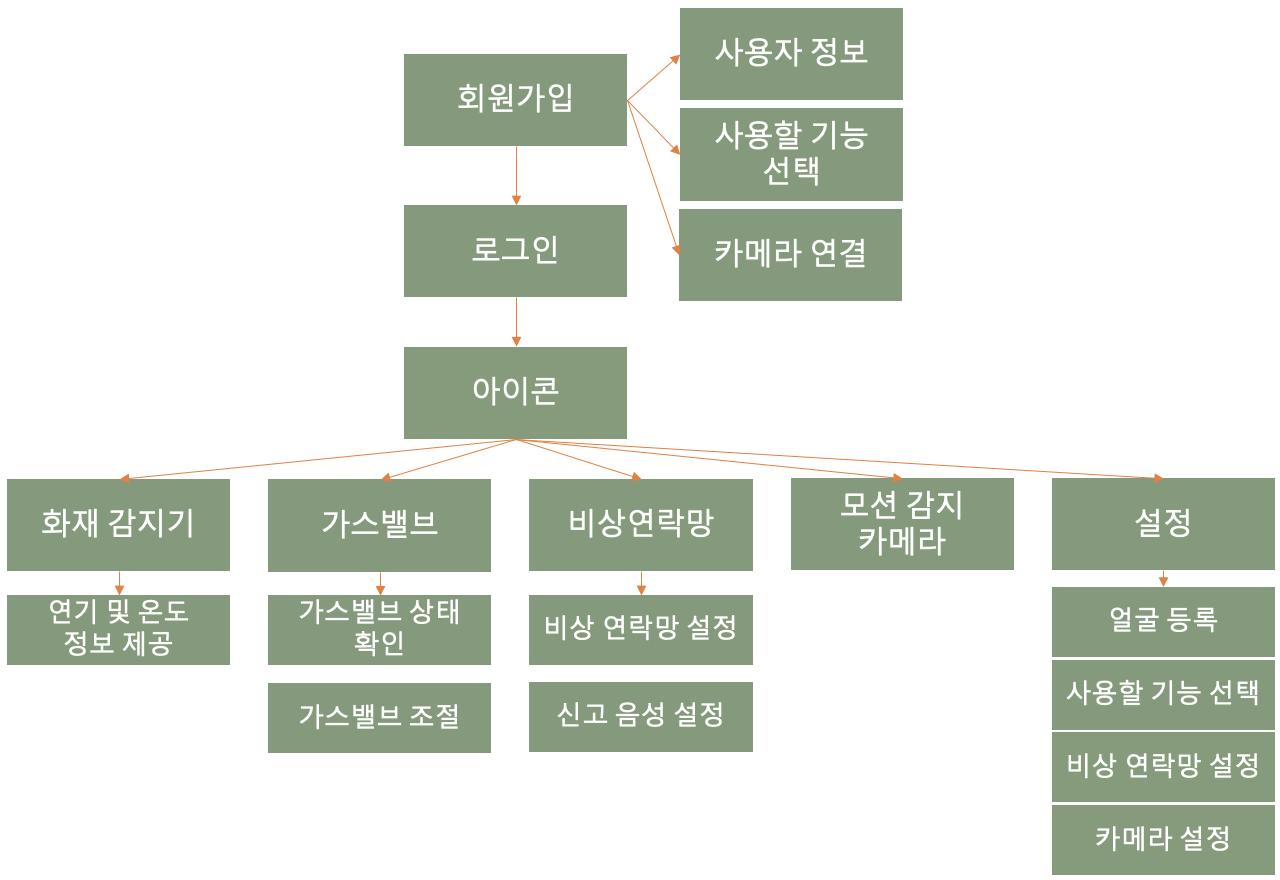
## **2.1. Product perspective**

“Smart Safety Home”은 1인 가구가 안전사고를 예방할 수 있고, 빠르게 대처할 수 있는 환경을 제공할 수 있도록 디자인하였다. “Smart Safety Home”은 사용자의 상황에 맞게 어플리케이션을 사용할 수 있도록, 본인이 사용할 기능을 선택할 수 있다. 해당 어플리케이션은 사용자의 가스 밸브 상태를 확인하고 조절할 수 있으며, 연기온도감지기를 이용하여 화재를 예방하고, 화재가 발생하였을 때 푸쉬 알람을 이용하여 사용자가 빠르게 대처할 수 있도록 한다. 그리고 등록되지 않은 얼굴이 카메라에 인식되었을 때 푸쉬 알람을 보내 사용자가 카메라를 확인하여, 위급 상황 시 빠르게 신고 및 연락할 수 있도록 한다. 신고에는 사용자가 등록한 연락처가 사용되며, 신고를 보다 편하게 지원하기 위해 어플리케이션의 위젯을 제공하고, 사용자가 등록한 신고 음성을 인식하여 자동으로 신고되도록 한다.

### **2.1.1. Market status**

현재 1인 가구의 비중은 청년들의 경제, 취업, 주거문제 등의 경제적인 요인들 및 결혼에 대한 가치관의 변화 등 개인적인 요인들로 인해 지속해서 높아지고 있는 추세이다. 1인 가구 밀집 지역에서 비밀집지역에 비해 2-3배의 범죄가 발생하며, 대부분의 1인 가구는 아침에 집을 나서면 저녁에 집을 들어오는 생활패턴을 보이고 있다. 따라서 범죄를 예방하고, 집에 없는 시간 동안 원격 조정이 가능한 smart safety에 대한 수요가 증가하고 있다.

### **2.1.2. Overall structure**



[Figure 1] System Structure

## **2.2. Product functions**

### **2.2.1. 회원가입 및 로그인**

어플리케이션을 처음 다운로드 받은 신규 사용자들은 본인의 계정 정보와 사용할 기능을 선택하고, 얼굴인식에 사용할 얼굴을 등록하여 새로운 계정을 생성할 수 있다. 추가로, 어플리케이션과 연동이 가능한 카메라가 있다면 카메라의 일련번호를 입력하는 방식으로 카메라를 연결할 수 있다.

사용자는 생성한 계정을 통해 로그인할 수 있으며, 로그인 후 본인이 선택한 기능들에 한해 어플리케이션을 사용할 수 있다.

### **2.2.2. 화재감지**

카메라에 포함되어 있는 온도연기감지 센서를 이용하여 집안의 온도 및 연기에 관한 정보를 어플리케이션에 표시한다. 특정 수치 이상의 연기나 온도가 감지되면 사용자에게 푸쉬 알림을 보내서 빠르게 대처할 수 있도록 한다.

### **2.2.3. 가스밸브**

가스 밸브의 상태 (열림 / 닫힘)를 확인할 수 있으며, 사용자가 버튼을 누를 시 가스 밸브의 상태를 변경할 수 있다.

### **2.2.4. 비상연락망 설정 및 신고 설정**

사용자가 신고를 원할 때 연락을 보낼 비상연락망을 설정할 수 있다. 신고는 홈화면에 사용자가 추가한 어플리케이션 위젯을 이용하거나, 사용자가 설정한 신고음성을 통해 할 수 있다. 사용자가 등록해놓은 단어로 사용자의 음성이 인식되면 자동으로 사용자가 설정한 비상연락망으로 사용자가 등록해놓은 연락이 가며, 추가로 긴급신고전화 (112, 119)로 신고가 자동으로 접수된다. 비상연락망을 설정해놓은 화면에서 특정 번호를 클릭하면, 해당 번호로만 연락을 보낼 수 있다.

### **2.2.5. 모션감지 카메라**

모션감지 카메라를 통해 집에 들어온 사람들의 얼굴을 인식할 수 있으며, 사용자가 등록하지 않은 얼굴이 인식되었을 때 사용자에게 푸쉬알람을 보내어 사용자가 바로 카메라 화면을 확인 및 필요 시 신고를 할 수 있도록 한다.

### **2.2.6. 설정**

**2.2.6.1. 얼굴 등록**

모션 감지 카메라를 통해 인식이 되어도 신고하지 않을 얼굴, 즉 집에 거주하는 사람들의 얼굴을 등록 및 관리할 수 있도록 한다.

**2.2.6.2. 사용할 기능 선택**

사용자가 이용할 기능을 선택 및 관리할 수 있는 기능으로, 사용자의 환경에 맞춰 기능을 사용할 수 있도록 한다.

**2.2.6.3. 비상연락망 설정**

2.2.4 비상연락망 설정 및 신고 설정과 동일한 기능이며, 사용자의 사용성을 고려하여 설정 메뉴에서 사용할 수 있도록 추가하였다.

**2.2.6.4. 카메라 연동**

사용자가 회원가입 시 연동한 카메라의 목록을 삭제 및 추가할 수 있다.

## **2.3. User Classes and Characteristics**

### **2.3.1. User**

### 특정 사용층으로 제한되지 않으며, 해당 서비스를 이용하고 싶은 사용자는 본인의 환경에 맞춰 어플리케이션을 사용할 수 있다. 하지만 어플리케이션과 연동이 가능한 카메라를 가지고 있지 않으면 움직임 감지 기능 등 카메라를 이용한 기능을 사용하는데에 제한이 있을 수 있다.

### **2.3.2. System Manager**

“Smart Safety Home”의 전체적인 관리 업무를 하는 사용자이다. system manager는 사용자의 계정을 만들 수 있으며, 어플리케이션의 전반적인 기능 및 설정에 관여할 수 있다. 하지만 해당 어플리케이션이 사용자의 개인정보와 밀접한 관련이 있으므로, 사용자의 설정 및 상태에는 관여할 수 없다.

## **2.4. Operating Environment**

### **2.4.1. Hardware**

사용자의 기기는 최소 2GB의 RAM용량과 1.6GHz프로세서가 기기에 내장되어 있어야 한다.

### **2.4.2. Software**

Android OS의 경우, 최소 Android 8.0 (API 26) 이어야 하며, Android 11.0(API 31)에 최적화 되어있다. IOS의 경우,iOS 14.5에 적합한 환경을 제공한다.

## **2.5. Design and Implementation Constraints**

“Smart Safety Home” 어플리케이션은 이 문서에 기술된 내용을 기반으로 구현되며, 아래의 제약사항들은 반드시 지켜져야 한다.

* 사용자가 기기에 대한 접근 권한을 승인해야 어플리케이션을 실행할 수 있으며, 이 권한을 변경할 수 있어야 한다.
* 신고 접수는 1초 이상 소요되어서는 안된다.
* 카메라화면 및 사용자의 데이터 다운로드는 어플리케이션이 정상적으로 작동할 때에만 진행되어야 한다.
* 기본 UI를 설계할 때 색각 이상자 사용가능여부 등 다양한 변수를 고려해야 한다.
* 카메라 화면, 사용자의 얼굴 설정 등 사용자의 사생활과 관련된 데이터를 사회적 윤리를 해치는 행위에 이용해서는 안된다.
* 소스코드를 작성할 때, 다른 개발자와의 협업 및 유지보수에 부정적인 영향을 미치지 않도록 주석을 작성해두어야 한다.
* 추가적인 비용을 사용하지 않기 위해, 최대한 오픈소스를 이용하여 구현한다.
* 불필요한 리소스를 소모하지 않기 위해 최적화에 힘써야 한다.

## **2.6. User Documentation**

사용자가 어플리케이션 및 서비스를 원활하게 이용할 수 있도록 다양한 문서를 제공한다. 첫번째로 최소한의 하드웨어 및 소프트웨어의 성능을 명시한 문서를 제공한다. 두번째로, 어플리케이션 화면을 캡쳐하여 사용법을 설명한 사용자 메뉴얼 문서 및 영상을 제공한다. 그리고 사용자가 문제없이 해당 어플리케이션을 설치 할 수 있도록 설치 가이드 문서를 제공하며, 어플리케이션 이용 시 문제가 발생했을 때 문제를 보고할 수 있는 개발자의 정보를 명시한 문서를 제공한다.

## **2.7. Assumptions and Dependencies**

“Smart Safety Home”는 Android, IOS 기기를 기반으로 고안 및 구현되었으며, 이 문서에 명시되어 있는 버전에 최적화 되어있다. 따라서 어플리케이션의 기능 및 내용은 해당 버전이 아닌 기기에서는 작동하지 않을 수 있다.

# **3. Specific Requirements**

## **3.1. External Interface Requirements**

### **3.1.1. User Interfaces**

### [Table 3] Starting Interface

| **Name** | **Starting Interface** |
| --- | --- |
| Purpose/Description | 사용자는 로그인과 회원가입 중 선택한다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ user instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 사용자는 아이디와 패스워드를 입력한 후  로그인 버튼을 눌러 어플을 실행시킨다.  2. 아이디 등록을 하지 않은 경우 회원가입  버튼을 통해 새로운 아이디를 만든다.  3. 사용자의 요청에 따라 시스템은 대응되는  응답으로 처리한다. |
| Format and configuration of window | 해당사항 없음 |
| Data type | int type value of a button code, text, image |
| Instruction type | Instruction mapping according to the value of a button code |
| Exit message | “로그인에 성공하였습니다!” |

[Table 4] Main Interface

| **Name** | **Main Interface** |
| --- | --- |
| Purpose/Description | 사용자가 설정한 기능목록을 띄운다. 상단에는 화재감지기/가스벨브 상태를 볼 수 있고 하단에는 사용자가 설정한 |
| Input source/ Output destination | 사용자/ Android OS 또는 iOS가 장착된 사용자 장치 |
| Range/  Accuracy/  Margin of error | 화면의 버튼 수에 따른 범위/ 사용자의 터치 정확도에 따른 정확도/ 터치 감도 오차 한계 |
| Unit | 클릭 한번 |
| Time/ Velocity | 비동기적 user input/ user instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 사용자가 설정한 기능목록을 띄운다. 상단과 하단은 분리되어 있고 각 기능들을 card 형식으로 만들어 grid view로 띄운다.  2. 상단에는 화재감지기/가스벨브 상태를 띄운다.  3. 상단에 있는 가스벨브 상태표시는 클릭을 하여 상태를 즉각 바꿀 수 있다.  4. 하단에는 사용자가 설정한 기능들을 띄운다.  5. 하단에 기능들을 클릭하면 각 interface로 전환한다. |
| Format and configuration of window | 해당사항 없음 |
| Data type | int type value of a button code, text |
| Instruction type | Instruction mapping according to the value of a button code |
| Exit message | 해당사항 없음 |

[Table 5] Account Registration Interface 1

| **Name** | **Account Registration interface 1** |
| --- | --- |
| Purpose/Description | 사용자가 시스템을 사용하기 위해서 아이디와 패스워드가 필요하다. 이를 시스템에 등록하기 위해 회원가입을 진행해야 한다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ user instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 사용자는 자신이 사용할 번호와 아이디를 각 textfield에 채워넣는다.  2. next버튼을 눌러 얼굴등록 화면으로 이동한다.  3. 중복되지 않는 이메일과 적절한 password가 사용되었으면 성공적으로 얼굴등록 화면으로 넘어간다.  4. 적절하지 않은 값을 아이디/패스워드를 입력했을경우 경고문구가 뜨고 다시 입력하도록 요구한다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Starting interface로 돌아간다. |
| Data type | int type value of a button code, query |
| Instruction type | Instruction mapped to the button |
| Exit message | 해당사항 없음 |

[Table 6] Account Registration Interface 2

| **Name** | **Account Registration interface 2** |
| --- | --- |
| Purpose/Description | 모션감지 카메라를 위한 사용자의 얼굴등록을 진행한다. 카메라가 얼굴을 잘 감지할 수 있게 4각도에서 촬영한다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 해당사항 없음 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 사용자는 모션감지 기능을 위한 얼굴 촬영을 진행한다. 2. 총 4각도로 진행이 되며 얼굴이 인식이 되면 다음 각도의 얼굴 촬영을 요구한다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Account registration interface1-으로 돌아간다. |
| Data type | Image |
| Instruction type | 해당사항 없음 |
| Exit message | “회원가입이 완료되었습니다” |

[Table 7] Setting Interface

| **Name** | **Setting Interface** |
| --- | --- |
| Purpose/Description | 사용자에게 다양한 설정 기능을 모아둔 화면을 보여준다 |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 얼굴 등록, 사용 기능 선택, 신고 환경 설정 이렇게 총 3개의 설정 기능이 존재한다.  2. ‘얼굴 등록’ 버튼을 누를 경우 Face Registration 화면으로 이동한다  3. ‘사용기능선택’ 버튼을 누를 경우 Function Enabling Setting화면으로 이동한다  4. ‘신고 환경 설정’ 버튼을 누를 경우 Reports Setting 화면으로 이동한다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우Main interface로 돌아간다. |
| Data type | int type value of a button code, Text |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 8] Face Registration Interface

| **Name** | **Face Registration Interface** |
| --- | --- |
| Purpose/Description | 사용자에게 얼굴 등록 설정 화면을 제공한다. 해당 화면에서 신규 얼굴 추가, 등록된 얼굴 확인 그리고 얼굴 감지 기능 비 활성화가 가능하다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 추가된 얼굴들은 ‘얼굴1’, ‘얼굴2’ 와 같은 형태로 저장이 된다.    2. ‘얼굴 추가’ 옆의 ‘+’ 기호를 누를 경우 신규 얼굴을 추가할 수 있다.  3. ‘등록된 얼굴 확인하기’ 버튼을 누를 경우 등록된 모든 얼굴들 확인이 가능하다.  4. ‘얼굴 감지 해제’ 옆의 toggle버튼을 통해 얼굴 감지 기능을 활성화/비 활성화가 가능하다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 메인 Setting Interface로 돌아간다. |
| Data type | int type value of a button code, Text |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 9] Function Enabling Setting Interface

| **Name** | **Function Enabling Setting Interface** |
| --- | --- |
| Purpose/Description | 사용자에게 사용 기능 설정 화면을 제공한다. 해당 화면에서 여러 기능을 활성화/ 비 활성화가 가능하며 비 활성화할 경우 Main Interface에 해당 기능이 사라집니다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 각종 기능들이 나열 돼있으며 각 우측엔 toggle 버튼이 존재한다  2. ‘화재 감지기’ 옆의 toggle 버튼을 누르면 화재 감지 기능 활성화/ 비 활성화가 가능하다.  3. ‘가스 밸브’ 옆의 toggle 버튼을 누르면 가스 밸브 기능 활성화/ 비 활성화가 가능하다.  4. ‘비상 연락망’ 옆의 toggle 버튼을 누르면 비상 연락망 기능 활성화/ 비 활성화가 가능하다.  5. ‘카메라’ 옆의 toggle 버튼을 누르면 카메라 기능 활성화/ 비 활성화가 가능하다.  6. 비활성화된 기능들은 Main interface에서 사라진다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 메인 Setting Interface로 돌아간다. |
| Data type | Text |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 10] Reports Setting Interface

| **Name** | **Reports Setting Interface** |
| --- | --- |
| Purpose/Description | 사용자에게 신고 환경설정 화면을 제공한다. 해당 화면에서 비상연락망, 신고 음성, 그리고 긴급 문자 내용을 추가할 수 있다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 신고 환경설정에선 비상연락망 설정, 신고 음성 등록, 긴급 문자 내용 추가가 가능하다.  2. ‘비상연락망 설정’ 옆의 ‘+’ 버튼을 누르면 비상연락망을 등록할 수 있다.  3. ‘신고 음성 등록’ 옆의 ‘+’ 버튼을 누르면 신고 음성을 등록할 수 있다.  4. ‘긴급 문자 내용 추가’ 옆의 ‘+’ 버튼을 누르면 긴급 문자 내용을 등록할 수 있다.  5. 각 탭 아래에는 등록된 목록들이 나열 돼있다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Setting Interface로 돌아간다. |
| Data type | int type value of a button code, Text |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 11] Emergency Contact Interface

| **Name** | **Emergency Contact Interface** |
| --- | --- |
| Purpose/Description | 사용자가 설정한 비상연락처를 화면에 띄운다. 사용자는 비상시 리스트에 있는 연락처로 긴급 문자 및 전화를 할 수 있다. |
| Input source/ Output destination | 사용자/ Android OS 또는 iOS가 장착된 사용자 장치 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ user instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 기존에 설정해둔 비상연락처들이 list view 형태로 띄워진다.  2. List view내의 element에 왼쪽은 연락처의 정보인 이름과 번호가 표시되며 오른쪽에는 버튼이 있다.  3. 오른쪽에 있는 전화버튼과 문자버튼을 눌러 긴급 전화/문자를 실행할 수 있다. |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Main interface로 돌아간다. |
| Data type | int type value of a button code, text |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 12] Camera Interface 1

| **Name** | **Camera Interface 1** |
| --- | --- |
| Purpose/Description | 사용자에게 집안 곳곳에 설치돼 있는 카메라화면들을 한눈에 볼 수 있는 화면을 제공한다. |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 상시 작동중인 모션 감지 카메라의 화면들을 한눈에 볼 수 있다.  2. 사람의 움직임이 감지된 화면은 빨간색 테두리를 띄게 된다.  3. 각 화면을 두 번 연속으로 누를 경우 큰 화면으로 확대 된다. (Camera Interface 2 로 넘어간다) |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Main interface로 돌아간다. |
| Data type | Text, Video |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

[Table 13] Camera Interface 2

| **Name** | **Camera Interface 2** |
| --- | --- |
| Purpose/Description | 사용자에게 특정한 화면을 확대해서 보여준다 |
| Input source/ Output destination | 클라이언트/서버 |
| Range/  Accuracy/  Margin of error | 해당사항 없음 |
| Unit | Screen |
| Time/ Velocity | 비동기적 user input/ User instruction의 즉각적인 실행 |
| Relationship with other input/outputs | 해당사항 없음 |
| Format and configuration of screen | 1. 확대된 화면을 우측이나 좌측으로 slide할 경우 이전 또는 다음 카메라 화면으로 넘어간다.  ‘ |
| Format and configuration of window | 좌측상단탭의 ‘<’ 버튼을 누를 경우 Camera Interface 1로 돌아간다. |
| Data type | Text, Video |
| Instruction type | 해당사항 없음 |
| Exit message | 해당사항 없음 |

### 

### 

### 

### **3.1.2. Hardware Interfaces**

[Table 14] Hardware Interface

| **Name** | **Applicable device for the system** |
| --- | --- |
| Purpose/Description | Android OS smartphone (Android 6.0 이상), iOS기기 (iOS 13.0 이상) |

[Table 15] Hardware Interface IOT

| **Name** | **IOTs** |
| --- | --- |
| Purpose/Description | 화재 경보기, 가스, 카메라 데이터 전달 받을 IOT 장비 |
| Input source/ Output destination | IOT / 서버 |
| Range/  Accuracy/  Margin of error | IOT 성능에 따라 다르다 |
| Unit | 쿼리 |
| Time/ Velocity | 즉각적인 실행 |
| Relationship with other input/outputs | 서버의 모든 입출력과 연관된다 |
| Format and configuration of screen | 해당사항 없음 |
| Format and configuration of window | 해당사항 없음 |
| Data type | 쿼리 |
| Instruction type | 쿼리 statement |
| Exit message | 해당사항 없음 |

### **3.1.3. Software Interfaces**

[Table 16] Software Interface

| **Name** | **Real-time Database** |
| --- | --- |
| Purpose/Description | 멀티미디어 데이터 및 메타 데이터의 관리를 위한 쿼리 입출력 |
| Input source/ Output destination | 호스트서버/ 유저, 유저/ 호스트서버 |
| Range/  Accuracy/  Margin of error | Firebase 성능에 따라 다르다 |
| Unit | 쿼리 |
| Time/ Velocity | 즉각적인 실행 |
| Relationship with other input/outputs | 서버의 모든 입출력과 연관된다 |
| Format and configuration of screen | 해당사항 없음 |
| Format and configuration of window | 해당사항 없음 |
| Data type | 쿼리 |
| Instruction type | 쿼리 statement |
| Exit message | 해당사항 없음 |

### 

### **3.1.4. Communication Interfaces**

[Table 17] Communication Interface

| **Name** | **Client and Host** |
| --- | --- |
| Purpose/Description | 각 클라이언트에서 호스트 서버에 접속을 요청한다. 호스트서버에서 각 클라이언트에게 IOT로부터 얻은 화재경보기, 가스 밸브, 카메라 데이터 제공. |
| Input source/  Output destination | 클라이언트와 호스트서버 |
| Unit | 패킷 |
| Time/ Velocity | 최소 10Mbps 이상 |
| Relationship with other input/outputs | 서버와 관련된 모든 입출력 |
| Format and configuration of screen | N/A |
| Format and configuration of window | N/A |
| Data type | 쿼리 |
| Instruction type | 쿼리 statement |
| Exit message | N/A |

## 

## 3.2. Functional Requirements

### 3.2.1. Use Case<계정 관리>

[Table 18] 회원가입 Use Case

| **Use case name** | **회원가입** |
| --- | --- |
| Actor | 등록되지 않은 사용자 |
| Description | 애플리케이션을 처음 사용하거나 새로운 계정을 만들고자 하는 사용자는 회원가입을 통해 사용자 계정을 생성할 수 있습니다. |
| Normal Course | 1. 회원가입 화면에서 다음과 같은 정보들을 입력하여 회원가입을 진행합니다.    1. 아이디    2. 비밀번호    3. 비밀번호 확인    4. 이름    5. 생년월일    6. 닉네임    7. 이메일 주소    8. 핸드폰 번호 2. 아이디, 비밀번호를 암호화하여 서버에 저장합니다. 3. 거주자 얼굴 인식을 위한 사용자 얼굴을 등록합니다. 4. 사용할 기능 목록을 선택합니다. 5. 사용할 기능 목록 중 모션감지 카메라 기능을 선택했다면, 카메라 일련번호 입력을 통해 카메라와 애플리케이션을 연결합니다. |
| Pre-condition | 중복된 아이디, 닉네임을 허용하지 않습니다.  단어들을 조합하여 아직 등록되지 않은 닉네임을 제공합니다.  비밀번호는 특수문자, 영/한/숫자를 포함하여 10자 이상입니다. |
| Post-condition | 암호화된 아이디, 비밀번호가 서버에 저장됩니다. |
| Assumptions | 해당사항 없음 |

[Table 19] 로그인 Use Case

| **Use case name** | **로그인** |
| --- | --- |
| Actor | 등록된 사용자 |
| Description | 로그인을 통해 애플리케이션의 모든 기능을 사용할 수 있습니다. |
| Normal Course | 1. 아이디와 비밀번호를 입력합니다. 2. 아이디와 비밀번호가 정확하게 입력되었다면, 메인 화면으로 넘어갑니다. 3. 아이디와 비밀번호가 등록된 것이 아니라면 잘못된 정보임을 사용자에게 명시합니다. |
| Pre-condition | 등록된 아이디와 비밀번호를 입력해야 합니다. |
| Post-condition | 로그인 성공 시 랜덤으로 uid를 반환합니다.  로그인 실패 시 -1 값을 반환합니다. |
| Assumptions | 해당사항 없음 |

[Table 20] 자동로그인 Use Case

| **Use case name** | **자동로그인** |
| --- | --- |
| Actor | 등록된 사용자 |
| Description | 로그인된 상태에서 애플리케이션이 종료되고, 다시 실행되었을 때 불필요한 로그인 과정을 거치지 않습니다. |
| Normal Course | 1. 로그인에 성공했다면 반환된 uid를 서버에 저장합니다. 2. 애플리케이션을 종료하고 다시 실행시켜도 동일한 계정으로 로그인이 되어 메인 화면으로 넘어갑니다. 3. 로그아웃을 하게 되면 서버에 저장된 uid를 삭제합니다. |
| Pre-condition | 로그인 성공 시 반환된 uid를 사용합니다. |
| Post-condition | 자동로그인된 계정이 서버에 저장된 uid 정보와 동일합니다. |
| Assumptions | 로그인에 성공했다고 가정합니다. |

[Table 21] 로그아웃 Use Case

| **Use case name** | **로그아웃** |
| --- | --- |
| Actor | 등록된 사용자 |
| Description | 더 이상 로그인된 계정을 사용하지 않거나 다른 계정을 사용할 수 있습니다. |
| Normal Course | 1. 로그아웃 버튼을 클릭하면 로그인 화면으로 넘어갑니다. |
| Pre-condition | 해당사항 없음 |
| Post-condition | 서버에 저장된 uid가 없습니다. |
| Assumptions | 로그인이 되어 있다고 가정합니다. |

[Table 22] 아이디/비밀번호 찾기 Use Case

| **Use case name** | **아이디/비밀번호 찾기** |
| --- | --- |
| Actor | 등록된 사용자 |
| Description | 잊어버린 계정 아이디, 비밀번호를 찾을 수 있습니다. |
| Normal Course | <아이디 찾기>   1. “아이디 찾기" 버튼을 클릭하면 아이디 찾기 화면으로 넘어갑니다. 2. 계정 생성 시 등록했던 사용자 이름, 생년월일, 핸드폰 번호를 입력하고, 핸드폰 인증을 통해 아이디를 찾을 수 있습니다.   <비밀번호 찾기>   1. “비밀번호 찾기" 버튼을 클릭하면 비밀번호 찾기 화면으로 넘어갑니다. 2. 계정 생성 시 등록했던 사용자 이름, 생년월일, 핸드폰 번호를 입력합니다. 3. 찾고자 하는 비밀번호의 아이디를 입력하면 계정 생성 시 등록했던 이메일로 비밀번호를 재설정할 수 있습니다. |
| Pre-condition | 등록된 계정에 대한 사용자 이름, 생년월일, 핸드폰 번호를 입력합니다. |
| Post-condition | 아이디/비밀번호 찾기를 통해 아이디를 찾거나 비밀번호가 재설정 됩니다. |
| Assumptions | 해당사항 없음 |

**<화재감지기>**

[Table 23] 화재감지기 Use Case

| **Use case name** | **Friend request** |
| --- | --- |
| Actor | 로그인된 사용자 & 화재감지기 기능을 추가한 사용자 |
| Description | 집 내부 연기(%), 온도(C)에 대한 정보를 얻을 수 있습니다. |
| Normal Course | 1. 사용할 기능 목록에 화재감지기 기능이 설정되어 있는지 확인합니다. 2. 일정 시간마다 화재감지기 IOT로부터 현재 집 내부에 측정된 연기와 온도에 대한 정보를 받아옵니다. 3. 가져온 정보를 화면에 보여줍니다. |
| Pre-condition | 연기 값은 0이상 100이하의 값입니다.  온도 값이 200 이상이면 273 뺀 값을 섭씨 온도 값으로 사용합니다. |
| Post-condition | 해당사항 없음 |
| Assumptions | 화재감지기 IOT가 있다고 가정합니다. |

**<가스밸브>**

[Table 24] 가스밸브 Use Case

| **Use case name** | **Delete friend** |
| --- | --- |
| Actor | 로그인된 사용자 & 가스밸브 기능을 추가한 사용자 |
| Description | 가스 밸브 잠금 상태를 확인하고 밸브를 조절할 수 있습니다. |
| Normal Course | 1. 사용할 기능 목록에 가스밸브 기능이 설정되어 있는지 확인합니다. 2. 일정 시간마다 가스밸브 IOT로부터 가스 밸브 잠금 상태 정보를 가져옵니다. 3. 가져온 정보를 화면에 보여줍니다. 4. 가스밸브 기능 아이콘 토글 버튼을 통해 가스 밸브를 조절합니다. |
| Pre-condition | 가스밸브 잠금 상태 값은 0 또는 1 입니다. |
| Post-condition | 해당사항 없음 |
| Assumptions | 가스밸브 IOT가 있다고 가정합니다. |

**<비상연락망>**

[Table 25] 비상연락망 관리 Use Case

| **Use case name** | **Block the user** |
| --- | --- |
| Actor | 로그인된 사용자 & 비상연락망 기능을 추가한 사용자 |
| Description | 비상 시 연락할 연락처를 추가/삭제할 수 있습니다. |
| Normal Course | 1. “+” 버튼을 클릭하면 비상 시 연락할 연락처를 추가할 수 있습니다. 2. 이미 등록된 연락처를 등록되지 않도록 합니다. 3. 연락처를 왼쪽이나 오른쪽으로 밀어내면 연락처를 삭제할 수 있습니다. |
| Pre-condition | 연락처는 연락처 포맷(010-0000-0000)으로 구성되어 있습니다. |
| Post-condition | 연락처를 추가하면 연락처 리스트의 사이즈가 1 증가합니다.  연락처를 삭제하면 연락처 리스트의 사이즈가 1 감소합니다. |
| Assumptions | 해당사항 없음 |

[Table 26] 음성인식 단어 관리 Use Case

| **Use case name** | **음성인식 단어 관리** |
| --- | --- |
| Actor | 로그인된 사용자 & 비상연락망 기능을 추가한 사용자 |
| Description | 음성인식을 통해 신고할 단어를 추가/삭제할 수 있습니다. |
| Normal Course | 1. “+” 버튼을 클릭하면 음성으로 인식할 단어를 추가할 수 있습니다.    1. 이미 등록된 단어를 등록되지 않도록 합니다. 2. 단어를 왼쪽이나 오른쪽으로 밀어내면 단어를 삭제할 수 있습니다. |
| Pre-condition | 단어는 한/영으로만 이루어져 있습니다. |
| Post-condition | 단어를 추가하면 단어 리스트의 사이즈가 1 증가합니다.  단어를 삭제하면 단어 리스트의 사이즈가 1 감소합니다. |
| Assumptions | 해당사항 없음 |

[Table 27] 신고 Use Case

| **Use case name** | **신고** |
| --- | --- |
| Actor | 로그인된 사용자 & 비상연락망 기능을 추가한 사용자 |
| Description | 위젯, 연락처, 음성인식을 통해 사전에 등록한 연락처에 신고를 할 수 있습니다. |
| Normal Course | <위젯 - 연락처>   1. 위젯 편집기능을 통해 신고 위젯을 추가/삭제할 수 있습니다. 2. 신고 위젯을 클릭하면 연락처 목록이 나타납니다. 3. 연락처를 클릭하면 신고가 성공적으로 이뤄집니다.   <위젯 - 음성인식>   1. 위젯 편집기능을 통해 신고 위젯을 추가/삭제할 수 있습니다. 2. 신고 위젯을 클릭하면 연락처 목록이 나타나는 동시에 음성을 인식합니다. 3. 등록된 단어가 인식되면 신고가 성공적으로 이뤄집니다.   <연락처>   1. 비상연락망 화면에서 등록된 연락처를 클릭하면 신고가 성공적으로 이뤄집니다.   <음성인식>   1. 애플리케이션이 실행되었을 때에만 음성을 인식합니다. 2. 등록된 단어가 인식되면 신고가 성공적으로 이뤄집니다. |
| Pre-condition | 연락처가 최소 1개라도 등록되어 있어야 합니다.  음성인식 단어가 최소 1개라도 등록되어 있어야 합니다. |
| Post-condition | 신고한 연락처에 전화 또는 문자가 전송됩니다. |
| Assumptions | 해당사항 없음 |

**<카메라>**

[Table 28] 카메라 Use Case

| **Use case name** | **Place review** |
| --- | --- |
| Actor | 로그인된 사용자 & 카메라 기능을 추가한 사용자 |
| Description | 사전에 등록한 거주인 얼굴 인식과 움직임 인식을 통해 외부 침입자가 있는지 확인할 수 있고, 장시간 동안 움직임이 없을 때 자동으로 등록된 연락처에 신고할 수 있습니다. |
| Normal Course | <외부 침입자 감지>   1. 사용할 기능 목록에 카메라 기능이 설정되어 있는지 확인합니다. 2. 카메라로 촬영된 이미지를 통해 사람을 탐지합니다. 3. 사람이 감지되면 거주인 얼굴과 비교를 진행합니다. 4. 거주인 얼굴이 아니라면 거주인에게 해당 이미지를 전송합니다.   <장시간 움직임 없음>   1. 사용할 기능 목록에 카메라 기능이 설정되어 있는지 확인합니다. 2. 카메라로 촬영된 이미지를 통해 사람을 탐지합니다. 3. 사람이 감지되면 거주인 얼굴과 비교를 진행합니다. 4. 탐지된 사람이 일정 시간동안 동일한 좌표에 위치한다면 등록된 연락처에 모두 메세지를 전송합니다. |
| Pre-condition | 거주인 얼굴이 등록되어 있어야 합니다. |
| Post-condition | 외부 침입자가 감지되었을 때 특정 이미지를 반환합니다. |
| Assumptions | 카메라 IOT가 있다고 가정합니다. |

**<설정>**

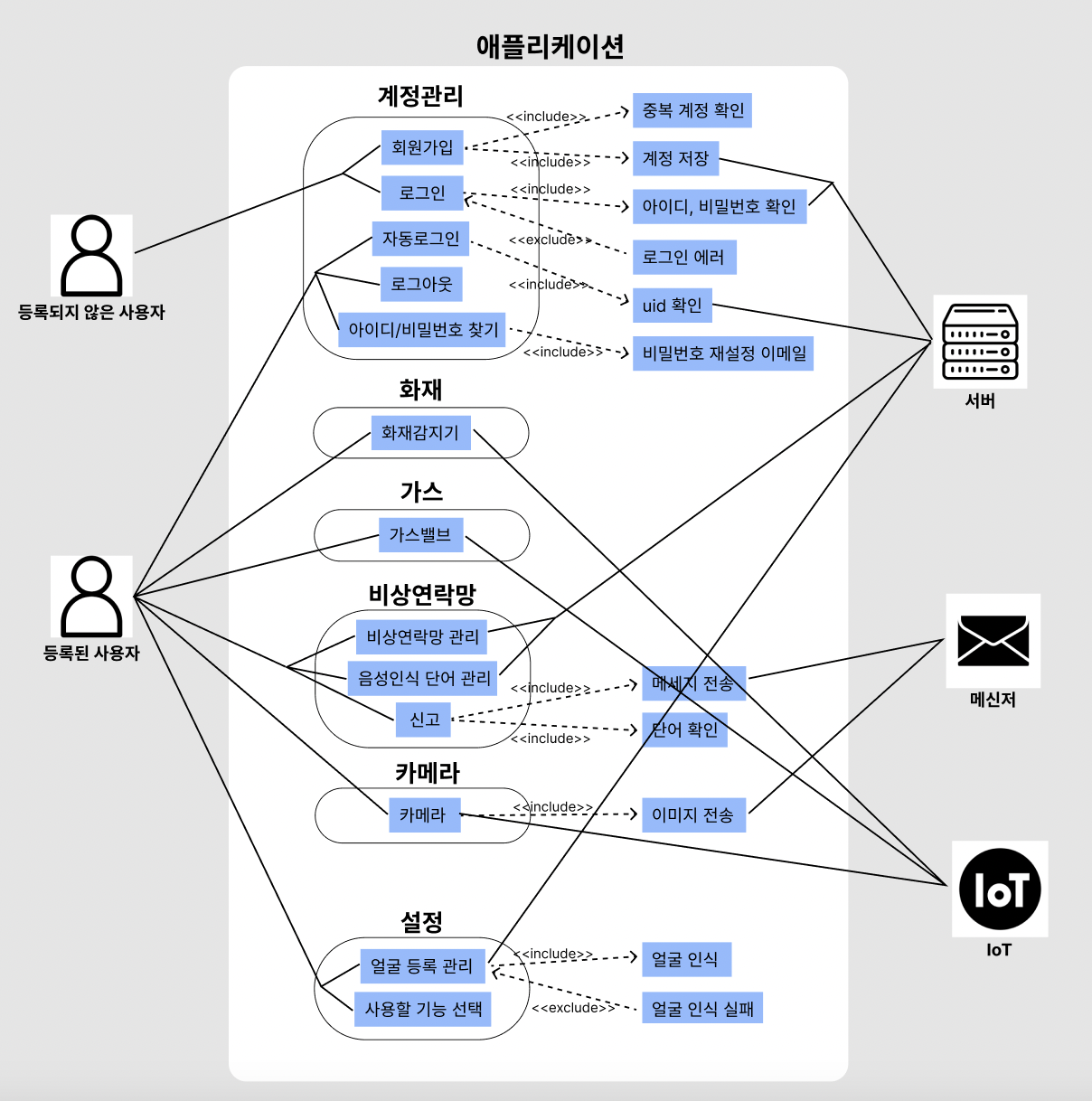
[Table 29] 얼굴 등록 관리 Use Case

| **Use case name** | **Open/Close a chat room** |
| --- | --- |
| Actor | 로그인된 사용자 |
| Description | 거주인의 얼굴을 추가/삭제할 수 있습니다. |
| Normal Course | <추가>   1. “얼굴 등록" 버튼을 클릭하면 얼굴을 인식할 카메라가 실행됩니다. 2. 얼굴 인식 알고리즘에 기반하여 얼굴을 등록합니다. 3. 등록이 완료되면 얼굴에 대한 이름을 지정합니다.   <삭제>   1. 등록된 얼굴 이름 중 하나를 길게 클릭하면 삭제 여부에 대해 묻습니다. 2. “확인” 버튼을 클릭하여 등록된 얼굴을 삭제합니다. |
| Pre-condition | 사람 얼굴만 등록 가능합니다. |
| Post-condition | 얼굴을 추가하면 얼굴 리스트의 사이즈가 1 증가합니다.  얼굴을 삭제하면 얼굴 리스트의 사이즈가 1 감소합니다. |
| Assumptions | 해당사항 없음 |

[Table 30] 사용 기능 선택 Use Case

| **Use case name** | **Enter/Leave the chat room** |
| --- | --- |
| Actor | 로그인된 사용자 |
| Description | 화재감지기, 가스밸브, 비상연락망, 카메라 기능 중 사용할 기능들만 선택적으로 사용할 수 있습니다. |
| Normal Course | 1. 스위치 버튼을 통해 사용할 기능을 on/off 할 수 있습니다. 2. 사용할 기능에 대해서만 메인 화면에 보여집니다. |
| Pre-condition | 해당사항 없음 |
| Post-condition | 해당사항 없음 |
| Assumptions | 해당사항 없음 |

### 3.2.2. Use Case Diagram



[Figure 2] Use Case Diagram

### 3.2.3. Data Dictionary

[Table 31] User

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | String | 사용자 아이디 |
| password |  | Not Null  특수문자 + 한/영/숫자 포함 10자 이상 | String | 사용자 비밀번호 |
| name |  | Not Null | String | 사용자 이름 |
| birth |  | Not Null  xxxx-xx-xx | String | 사용자 생년월일 |
| nickname |  | Not Null | String | 사용자 별명 |
| email |  | Not Null  xxxx@xxxxx.xxx | String | 사용자 이메일 |
| phone |  | Not Null  xxx-xxxx-xxxx | String | 사용자 핸드폰 번호 |
| functions |  | Nullable | List<Function> | 사용할 기능 리스트 |
| contacts |  | Nullable | List<Contact> | 비상연락망 리스트 |
| words |  | Nullable | List<Word> | 음성인식 단어 리스트 |
| faces |  | Nullable | List<Face> | 거주인 얼굴 리스트 |

[Table 32] Function

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 기능 id |
| name |  | Not Null | String | 기능 이름 |

[Table 33] FireDetector

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 화재감지기 IOT 일련번호 |
| smoke |  | Not Null  0이상 100이하 | Integer | 연기 percentage |
| temperature |  | Not Null | Integer | 섭씨 온도 |
| user\_id | FK | Not Null | String | 사용자 id |

[Table 34] Gas

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 가스밸브 IOT 일련번호 |
| isLock |  | Not Null  0 또는 1 | Integer | 가스밸브 잠금 여부 |
| user\_id | FK | Not Null | String | 사용자 id |

[Table 35] Camera

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 카메라 IOT 일련번호 |
| images |  | Nullable | List<Image> | 이미지 리스트 |
| user\_id | FK | Not Null | String | 사용자 id |

[Table 36] Contact

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 비상연락망 id |
| name |  | Not Null | String | 연락처 이름 |
| phone |  | Not Null  xxx-xxxx-xxxx | String | 연락처 |

[Table 37] Word

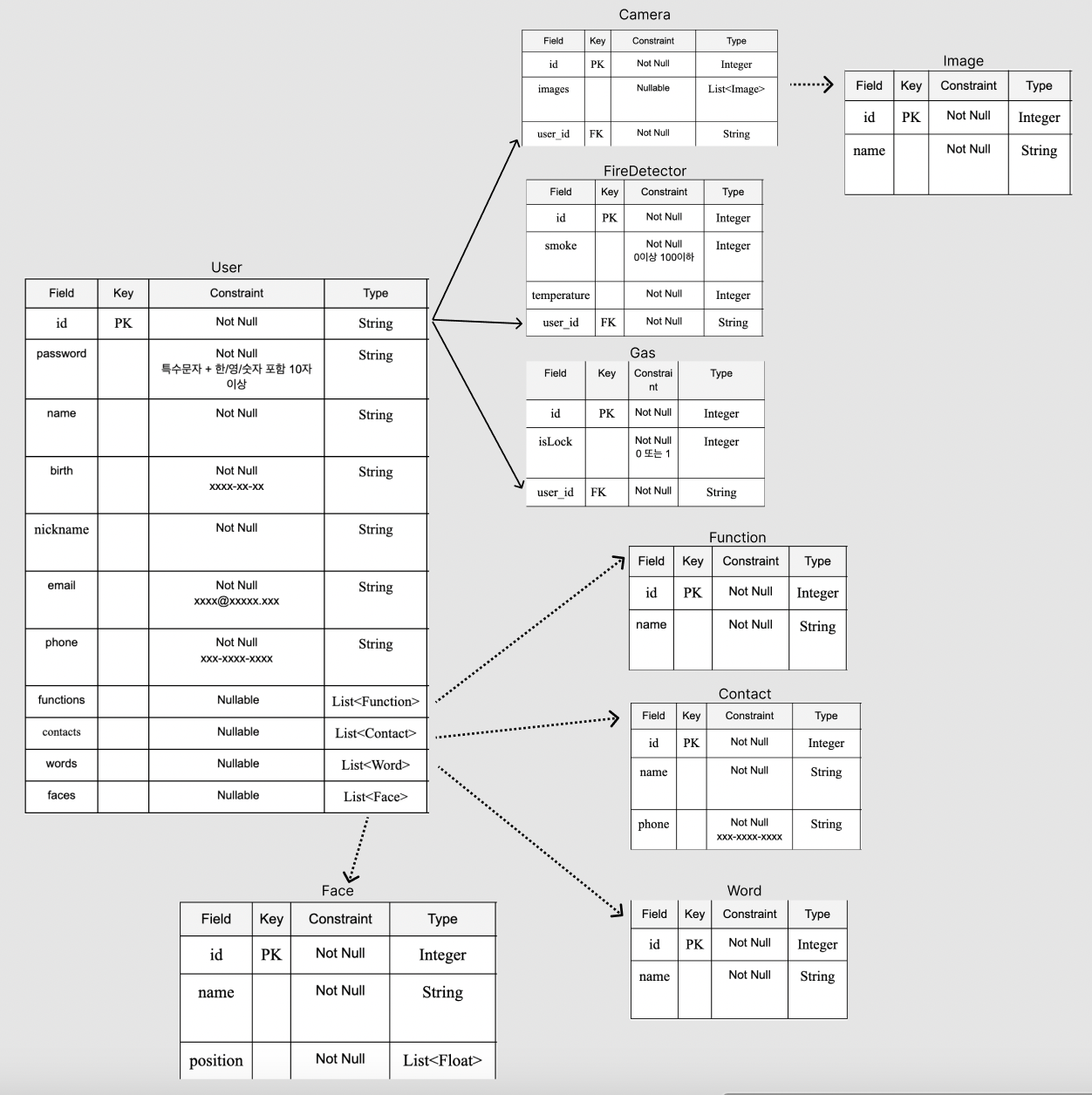
| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 단어 id |
| name |  | Not Null | String | 단어명 |

[Table 38] Image

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 이미지 id |
| url |  | Not Null | String | 이미지 url |

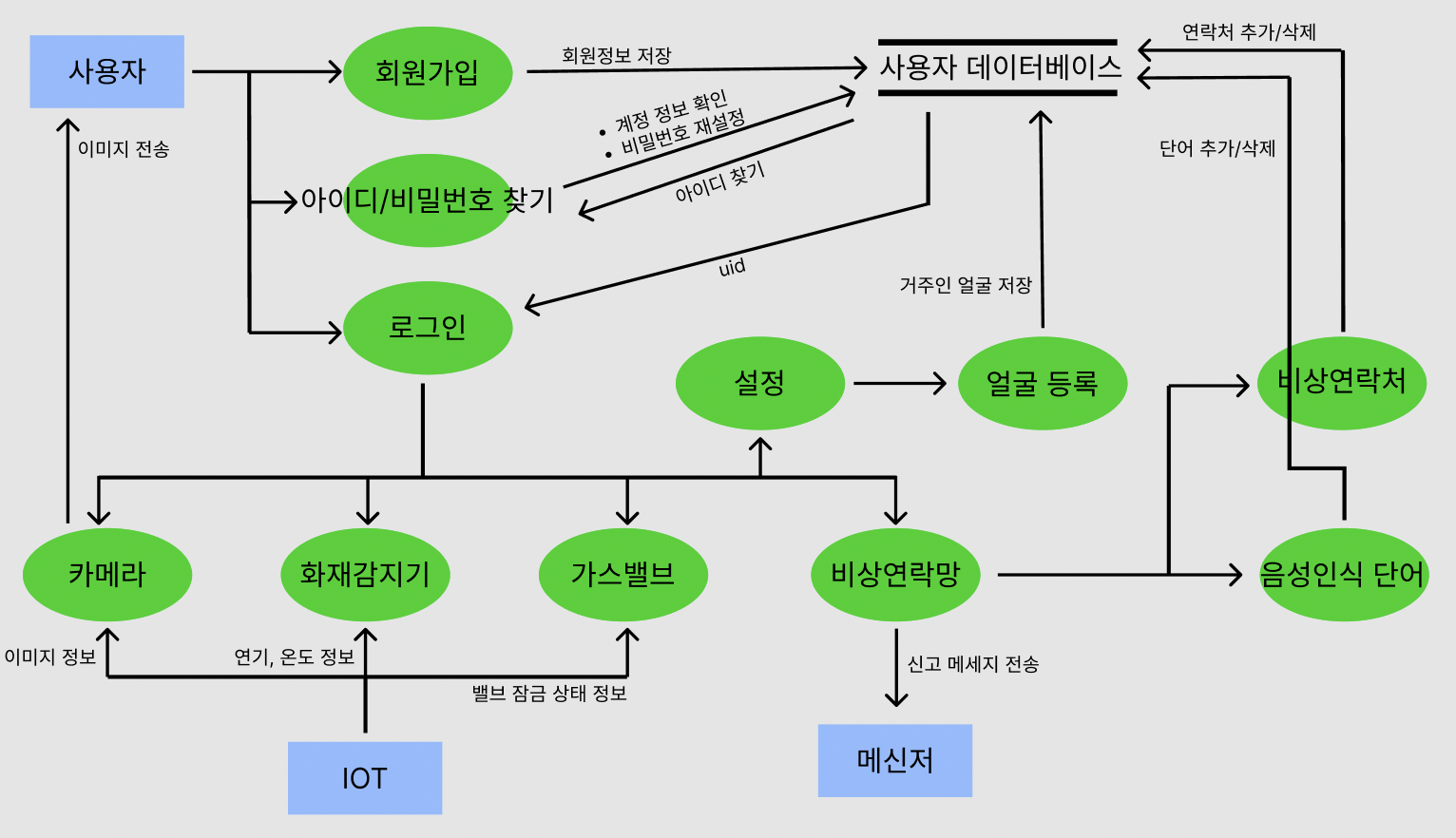
[Table 39] Face

| Field | Key | Constraint | Type | Description |
| --- | --- | --- | --- | --- |
| id | PK | Not Null | Integer | 얼굴 id |
| name |  | Not Null | String | 거주인 이름 |
| position |  | Not Null | List<Float> | 얼굴 좌표 |



[Figure 3] Entity Relationship Diagram

### 3.2.4. Data Flow Diagram



[Figure 4] Data Flow Diagram

## 

## 3.3. Product Requirements

해당 장에서는 비기능적 요구사항의 일부로서 본 명세서에서 다루고 있는 제품이 어떻게 작동해야 하는지에 대한 요구사항들을 규정한다. 요구사항은 다음과 같은 성능 요구사항, 공간 요구사항, 보안 요구사항, 사용성 요구사항, 신뢰성 요구사항들을 다룬다.

**3.3.1. Performance Requirements**

다음 성능 요구사항들은 제품이 얼마나 빠르게 작동해야 하는지에 대한 요구사항들을 다루고 있다. 요구사항들은 구동 기기의 최소 요구 사양에 기반하여 작성되었으나 기기의 사양에 따라 지켜지지 않을 수 있다. 구동 기기의 최소 요구 사양은 다음과 같다.

⬧ 2GB RAM, 1.6GHz CPU, 32GB 저장공간, Android 6.0 이상 또는 iOS 13.0 이상

⬧ 앱이 실행되면 3 초 안에 메인 화면에 진입해야 한다.

⬧ 로그인 프로세스는 2 초 내에 완료되어야 한다.

⬧ 한 화면에서 다른 화면으로 진입할 때는 1 초 내에 완료되어야 한다.

⬧ IOT로부터 받은 데이터(화재감지기, 가스밸브, 카메라 화면)를 실시간으로 반영해야한다.

⬧ 비상연락망 화면에서 등록된 연락처를 클릭하면 1초내에 신고가 이뤄져야한다.

⬧ 모든 데이터 업로드는 최소 10Mbps 최대 100Mbps 의 속도를 지원해야 한다.

⬧ 모든 데이터 다운로드는 최소 10Mbps 최대 200Mbps 의 속도를 지원해야 한다.

**3.3.2. Space Requirements**

다음 저장 요구사항들은 제품이 얼마만큼의 저장공간과 메모리를 사용해야 하는지에 대한 요구사항들을 다루고 있다. 요구사항들은 성능 요구사항과 마찬가지로 최소 요구 사양에 기반하여 작성되었으며 기기의 사양이나 상태에 따라 지켜지지 않을 수 있다.

⬧ 앱이 차지하는 저장 공간은 400MB 를 넘지 않아야 한다.

⬧ 모든 기능을 수행함에 있어서 프로그램은 200MB 이상의 메모리를 사용하지 않아야 한다.

**3.3.3. Security Requirements**

다음 보안 요구사항들은 제품이 작동하는데 있어서 보안을 유지하기위한 요구사항들을 다루고 있다.

⬧ 한 계정 당 동시에 하나의 접속 만을 허용해야 한다.

⬧ 한 기기 당 동시에 하나의 계정 접속 만을 허용해야 한다.

**3.3.4. Usability Requirements**

다음 사용성 요구사항들은 일반 사용자들이 제품을 사용하는데 있어서 오류를 발생시키지 않을 수 있도록 직관적인 사용성들에 대한 요구사항들을 다루고 있다.

⬧ 모든 사용자들은 별도의 설명서 없이 쉽게 사용할 수 있어야한다.

⬧ 모든 화면에서 이전 화면으로 넘어가는 버튼이 있어야 한다.

⬧ 모든 기능은 일관된 사용자 인터페이스를 가져야 한다.

⬧ 사용자 인터페이스는 쉽고 직관적이며 간결해야 한다.

**3.3.5. Dependability Requirements**

다음 신뢰성 요구사항들은 제공하는 기능들이 의도한 대로 작동할 수 있도록 규정하는 요구사항들을 다루고 있다.

⬧ IOT로부터 끊김없이 데이터를 전송 받아야한다.

⬧ 사용자가 비활성화한 기능에 대해선 데이터를 전송 받지않는다.

## 3.4. Organizational Requirements

해당 장에서는 비기능적 요구사항의 일부로서 시스템을 사용하는 고객이나 시스템을 개발하는 개발자들이 속한 조직의 규정이나 절차 또는 상황에 따른 제약조건들을 규정한다. 요구사항은다음과 같은 환경적 요구사항, 운영 요구사항, 그리고 개발 요구사항을 다룬다.

**3.4.1. Environmental Requirements**

별도의 환경적 요구사항은 없다.

**3.4.2. Operational Requirements**

다음 운영 요구사항들은 시스템이 운영되는 조직의 상황이나 규정 등을 반영하여 시스템 운영을 원활하게 하는데 목적이 있다.

⬧ 회원가입은 아이디, 비밀번호 그리고 얼굴 등록을 통해 이루어진다.

⬧ 비상연락망의 주소 접근 권한은 회원에게 밖에 존재하지 않는다.

⬧ IOT를 통해 전달받은 정보는 데이터베이스에서 관리되며 실시간으로 회원에게 제공된다.

**3.4.3. Development Requirements**

다음 개발 요구사항들은 시스템을 개발하는데 있어서 개발자들이 속한 조직의 규정과 절차에 따른 제약조건들을 다룬다.

⬧ 시스템 개발은 폭포수 모델에 기반하여 이루어져야 한다.

⬧ Android 와 iOS 앱 개발은 Flutter 2.0.3 을 통해 이루어져야 한다.

⬧ Android 와 iOS 앱 개발은 Dart 2.5 를 통해 이루어져야 한다.

⬧ Android 앱 개발은 Android 6.0 이상에 대하여 이루어져야 한다.

⬧ iOS 앱 개발은 iOS 13.0 이상에 대하여 이루어져야 한다.

## 3.5. External Requirements

해당 장에서는 비기능적 요구사항의 일부로서 외부에서 작용하는 시스템 또는 개발 과정에 대한 요구사항들을 규정한다. 요구사항은 다음과 같은 규제 요구사항, 윤리적 요구사항, 회계 요구사항, 안전/보안 요구사항을 다룬다.

**3.5.1. Regulatory Requirements**

다음 규제 요구사항은 시스템이 운영되는 성균관대학교 외부에서 작용하는 요구사항들을 다룬다.

⬧ 본 명세서에서 다루는 시스템은 저작권 보호법에 따라 원작자가 요청 시 컨텐츠를 차단할 수 있어야 한다.

⬧ 본 명세서에서 다루는 시스템이 사용하는 오픈소스 소프트웨어는 개발 규정에 맞게 개발되어야 한다.

**3.5.2. Ethical Requirements**

다음 윤리적 요구사항은 시스템이 개발되고 사용되는데 있어서 비윤리적인 행위가 개입할 수 없도록 규정하는 요구사항들을 다룬다.

⬧ 시스템이 개발되고 운영되는데 있어서 악의적인 의도를 통해 금전적 이득을 취하는 집단이 없어야 한다.

⬧ 시스템 사용자의 의도에 의해서 성적에 영향을 줄 수 있는 행위를 차단해야 한다.

**3.5.3. Accounting Requirements**

다음 회계 요구사항은 시스템이 운영되고 개발되는데 있어서 발생할 수 있는 금전적인 문제에 대한 규정을 다룬다.

⬧ 시스템이 개발되고 운영되는데 있어서 사용자의 추가적인 결제를 요구하지 않아야 한다.

⬧ 시스템 개발과 운영에 사용되는 비용은 계약서에 명시된 규정을 따라야 한다.

**3.5.4. Safety/Security Requirements**

다음 안전/보안 요구사항은 시스템이 운영되고 사용되는데 있어서 안전과 정보의 보안이 지켜질 수 있도록 하는 요구사항들을 다룬다.

⬧ 본 명세서에서 다루는 시스템은 개인정보보호법에 따라 모든 개인정보는 암호화되어 저장되고 사용자에 동의한 범위 내에서만 사용되어야 한다.

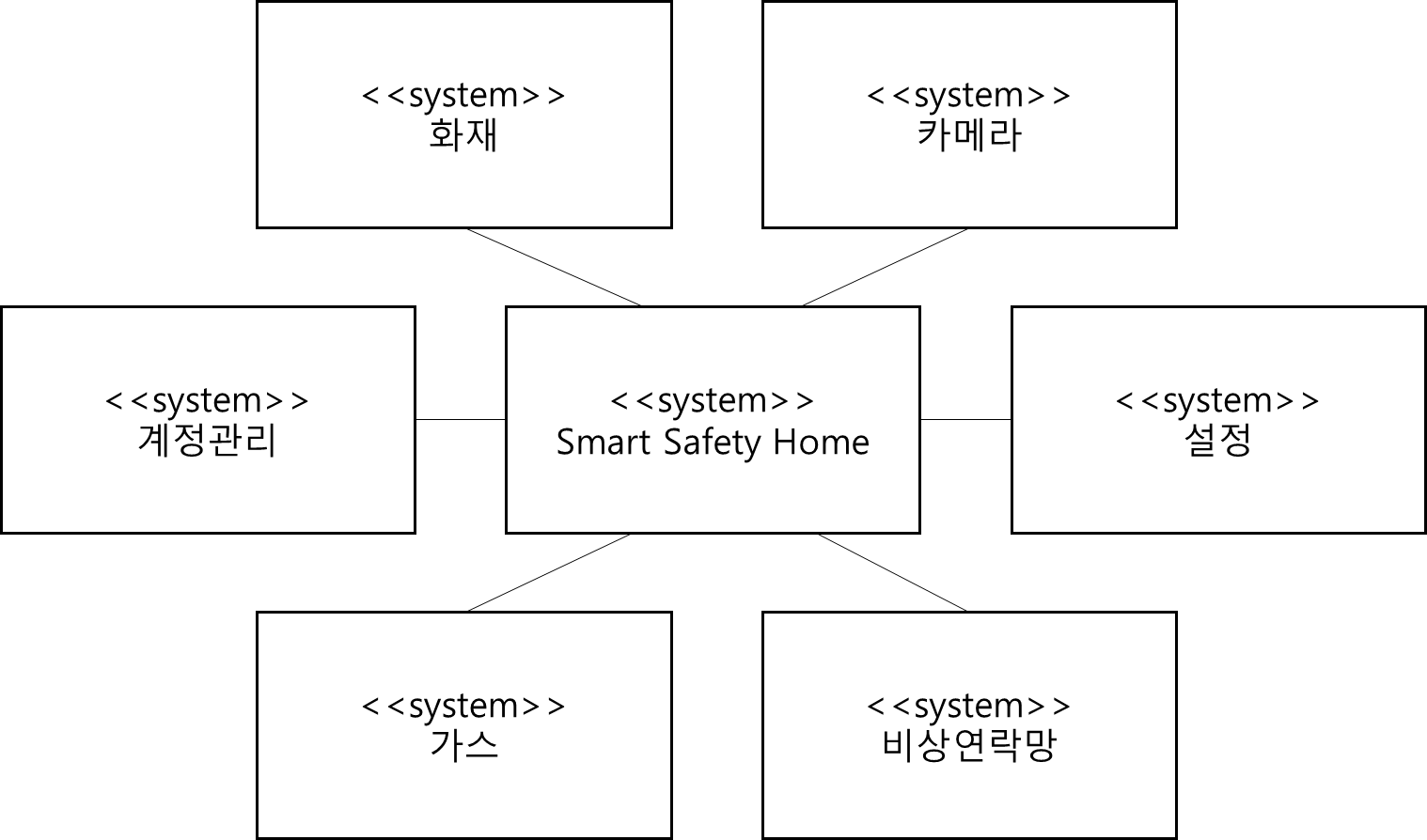
## 

## **3.6. Organizing the Specific Requirements**

해당 장에서는 통합 모델링 언어인 UML(Unified Modeling Language)을 통해 시스템 모델을 설명한다. 시스템 모델 다이어그램을 통해 구체적인 요구사항들을 확인할 수도 있다.

### 3.6.1. Context Model

컨텍스트 모델 다이어그램을 통해 시스템 내 서브 시스템 간의 관계를 보여줄 수 있다.

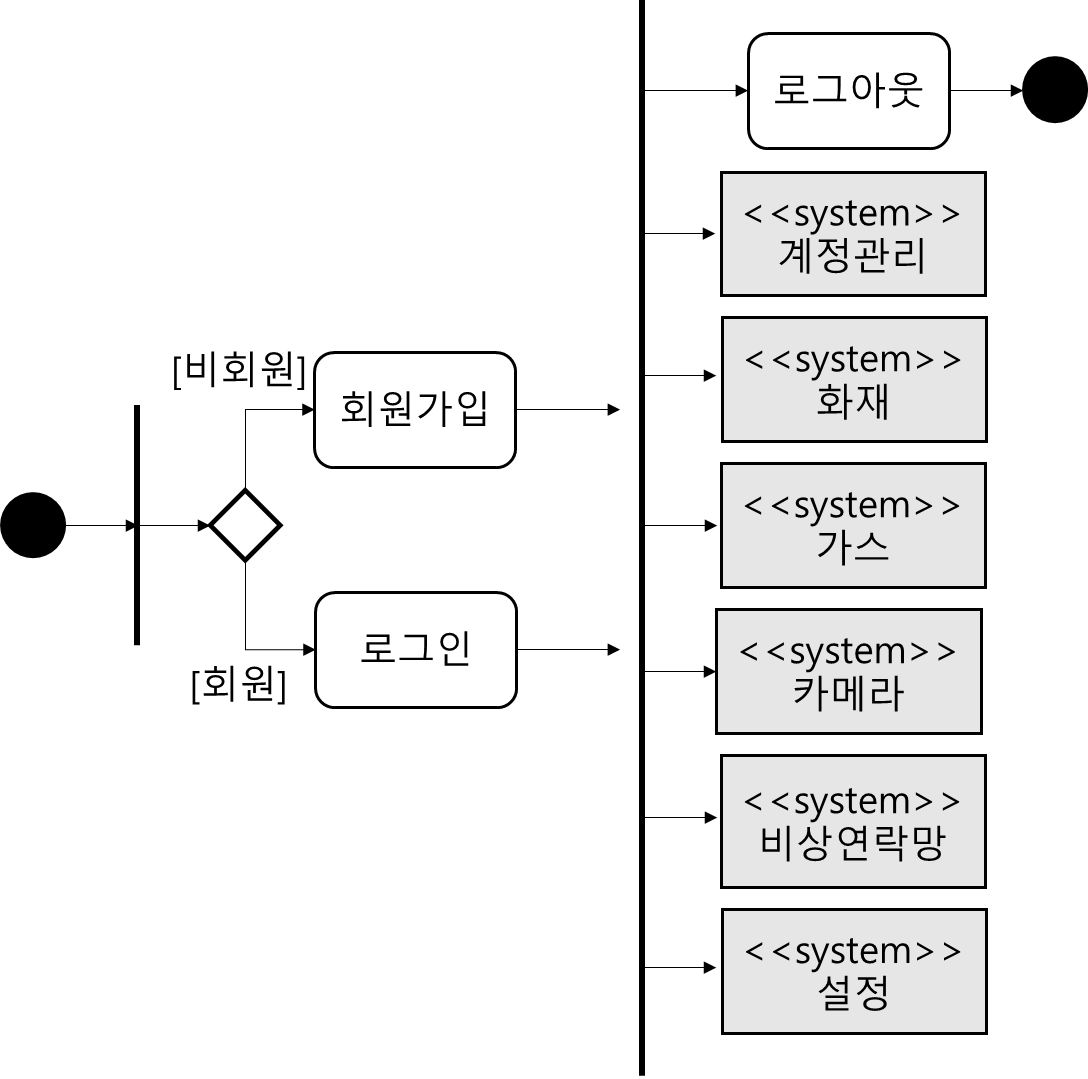


### [Figure 5] Context Model

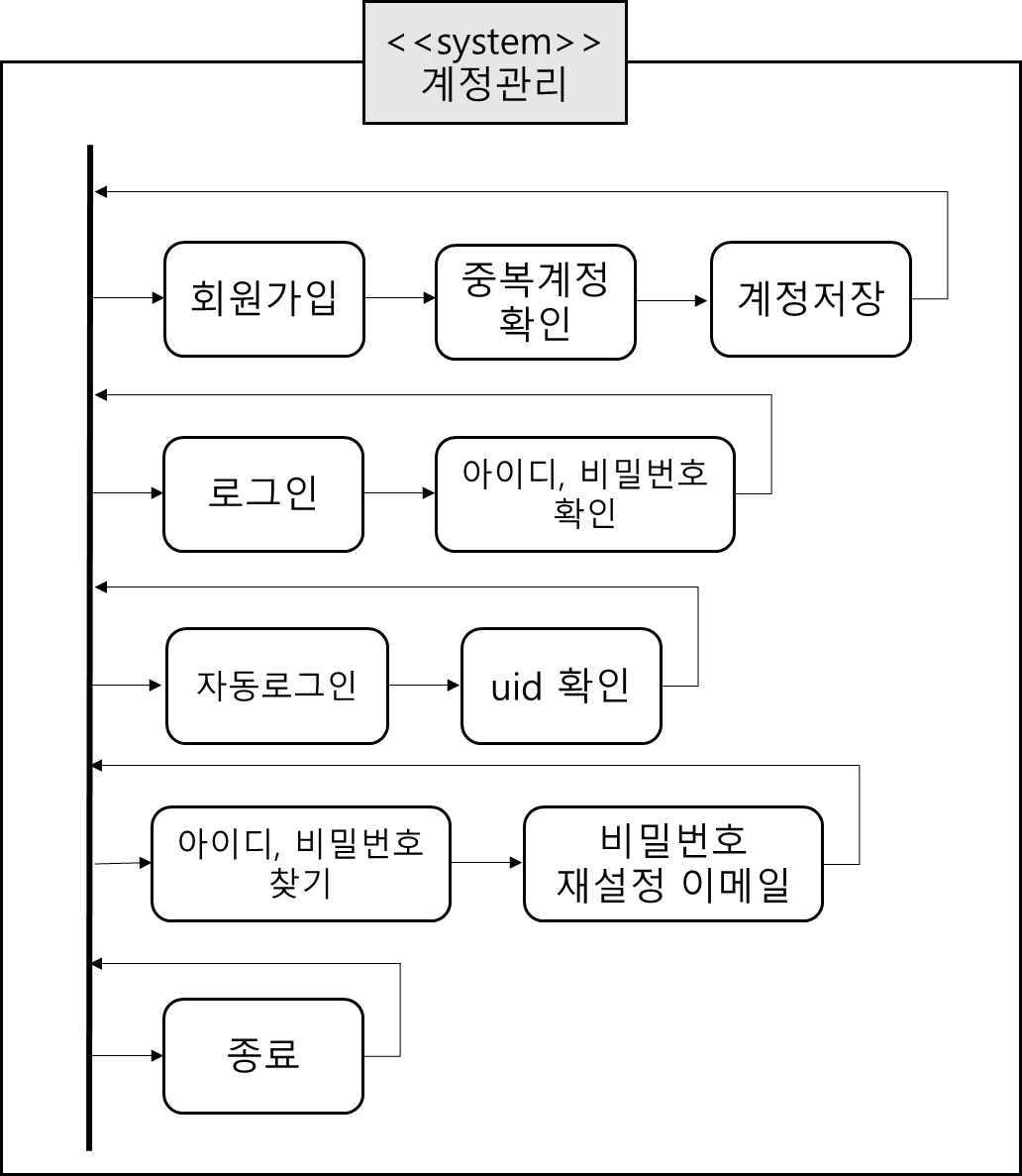
### 

### 3.6.2. Process Model

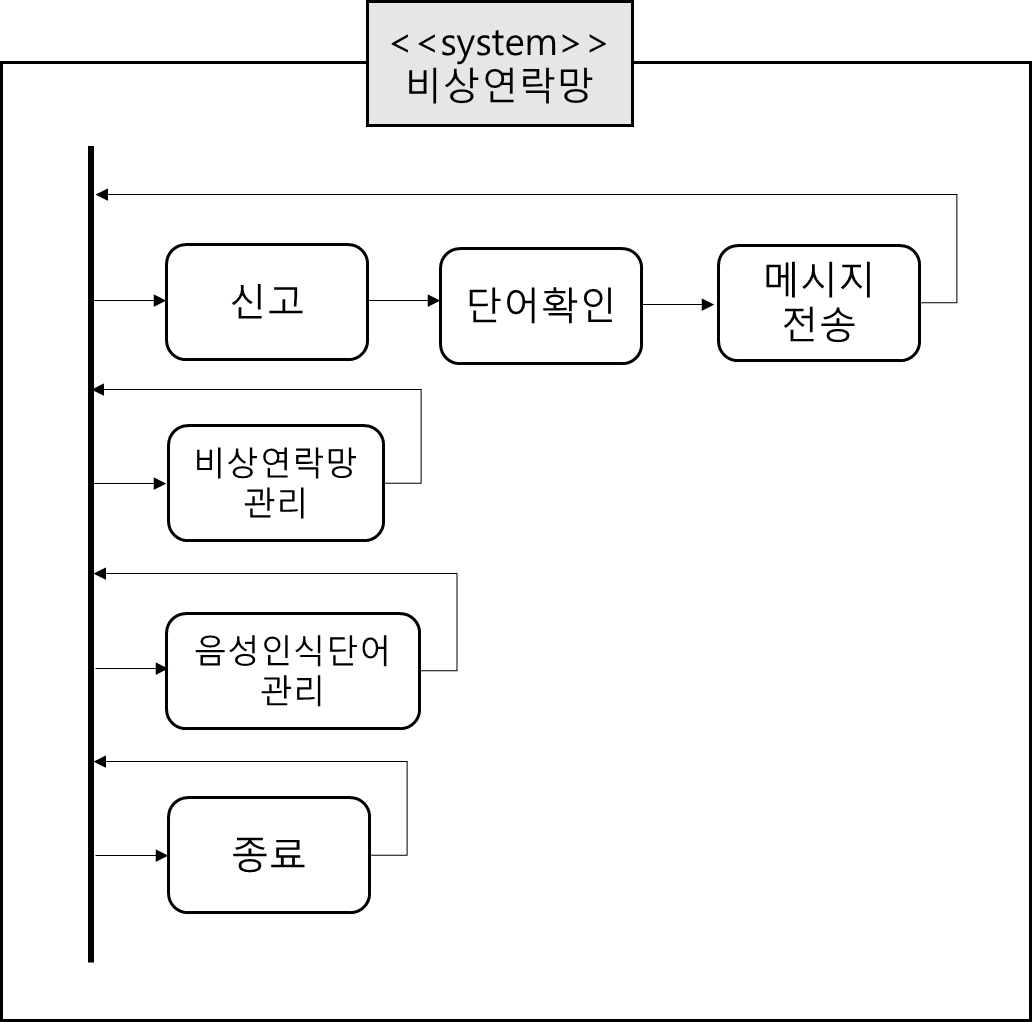
프로세스 모델을 통해 시스템 내 각 프로세스의 순서를 보여준다.



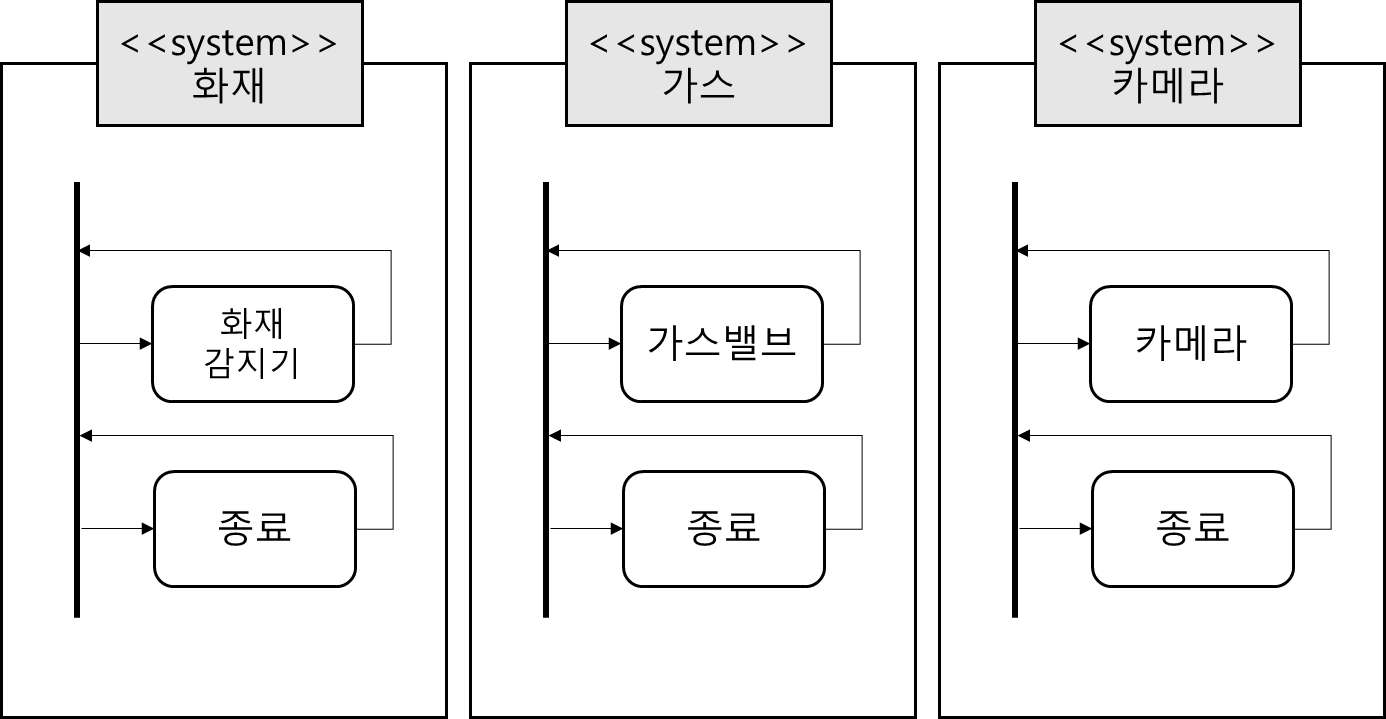
[Figure 6] Process model - 총괄



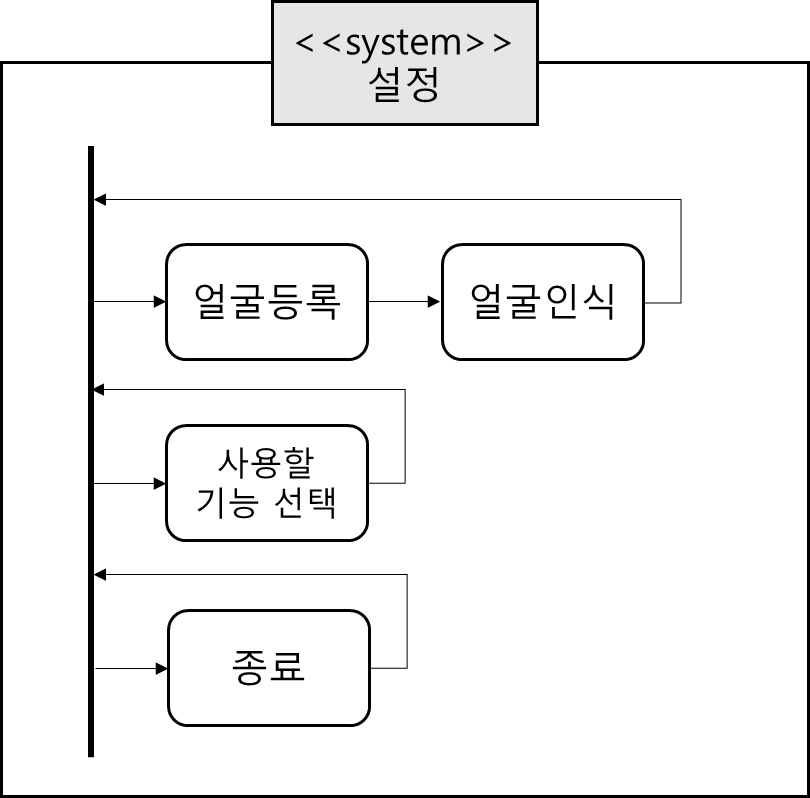
[Figure 7] Process model - 계정관리



[[Figure 8] Process model - 비상연락망](#_heading=h.14ykbeg)



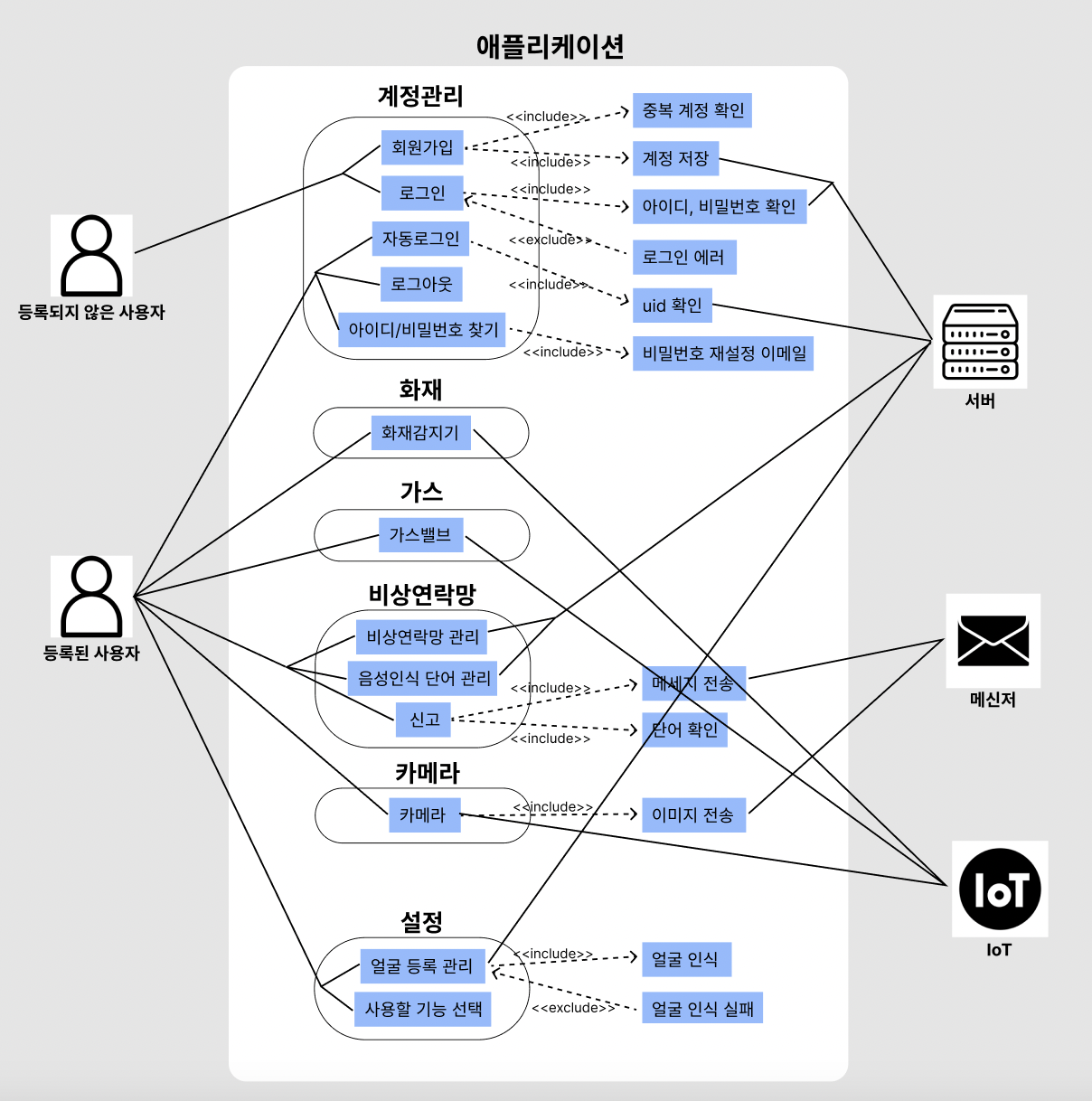
[Figure 9] Process model - 화재, 가스, 카메라



[[Figure 10]](#_heading=h.243i4a2) [Process model -](#_heading=h.14ykbeg) 설정

### 3.6.3. Interaction Model

Use case diagram을 통해 시스템에 참여하는 사람과 시스템 간의 관계를 볼 수 있다.

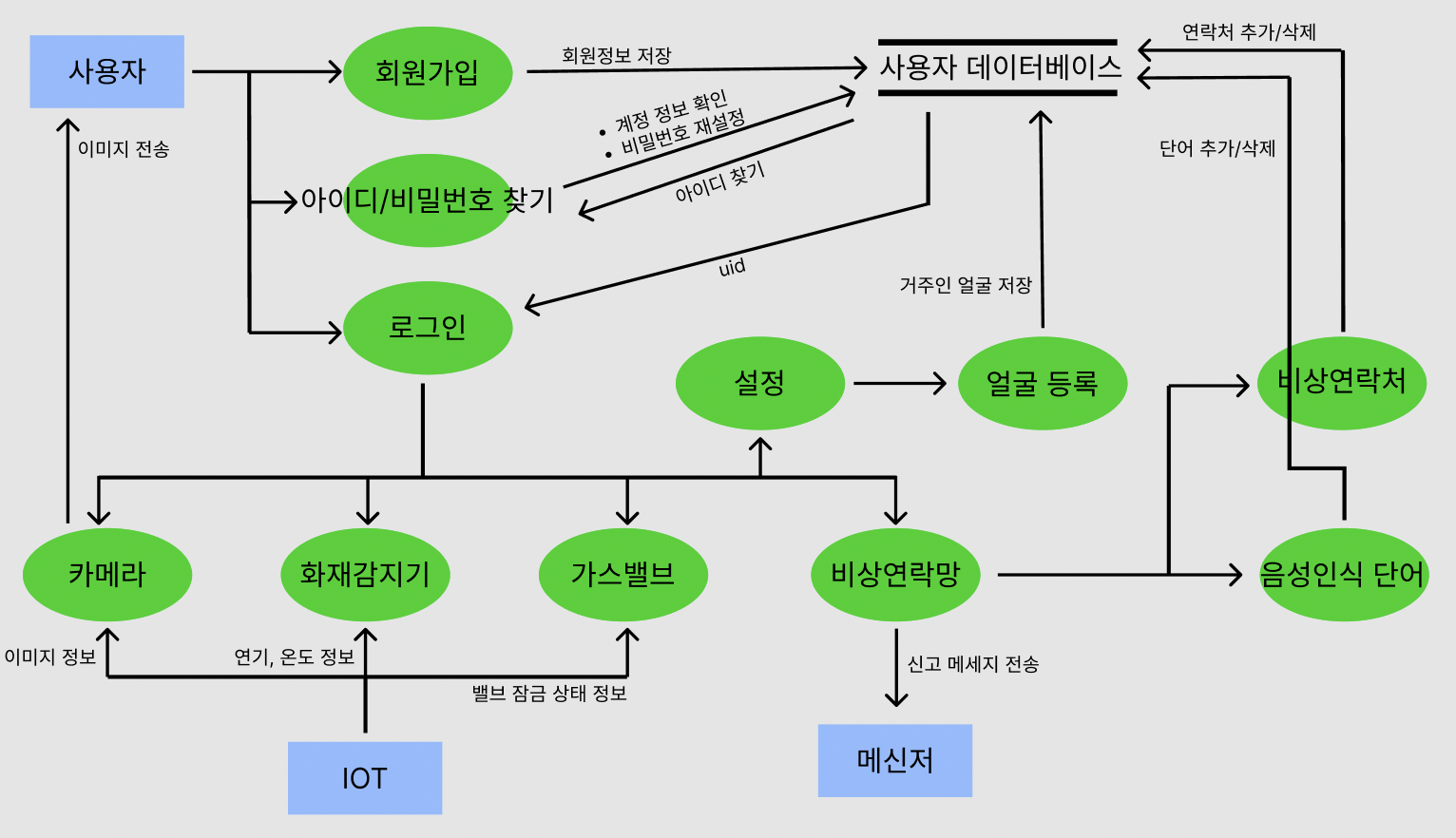


[Figure 11] Use Case Diagram(Figure 2 동일)

### 

### 3.6.4. Behavior Model

#### 3.6.4.1. Data Flow Diagram



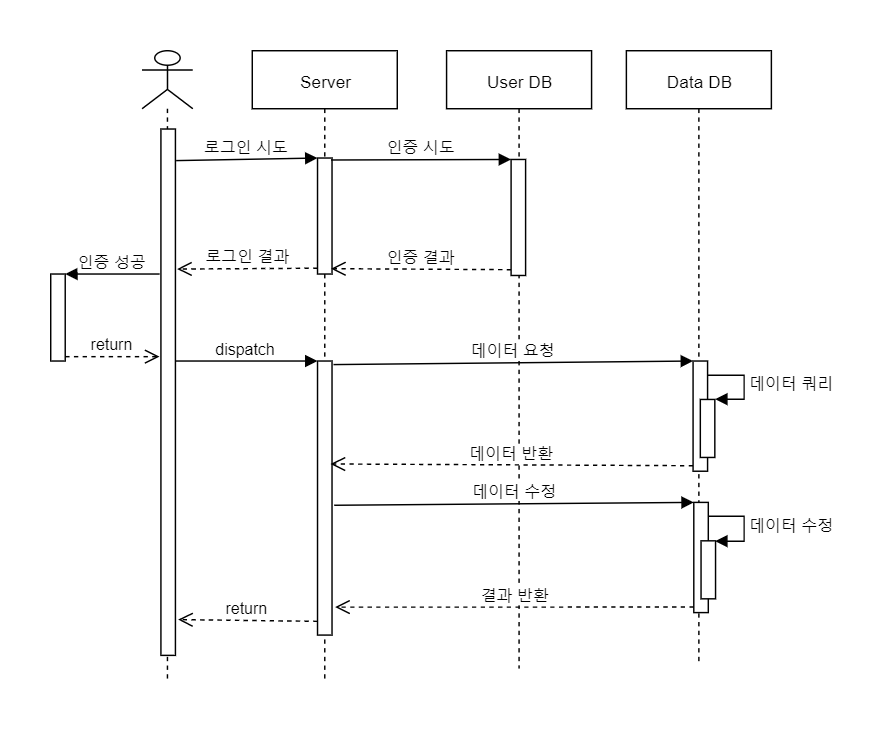
[Figure 12] Data Flow Diagram(Figure 4 동일)

데이터 흐름 다이어그램을 통해 시스템 내에서 데이터가 처리되는 과정을 확인할 수 있다.

#### 

#### 3.6.4.2. Sequence Diagram

Sequence Diagram을 통해서 프로세스의 진행 순서에 대해서 확인할 수 있다.



[Figure 13] Sequence Diagram

# 

# **4. Supporting Information**

## **4.1. Software Requirement Specification**

이 SRS 문서는 IEEE Recommendation (IEEE Recommended Practice for Software Requirements Specifications, IEEE-Std-830).에 따라 작성되었다.

https://ieeexplore.ieee.org/document/720574

## **4.2. Document History**

[Table 40] Document History

| **Date** | **Version** | **Description** | **Writer** |
| --- | --- | --- | --- |
| 2022/04/23 | 0.1 | Style and overview | 김승현 |
| 2022/04/23 | 1.0 | Addition of 2.1.1 | 하솔비 |
| 2022/04/23 | 1.1 | Addition of 3.1.1 | 이상협 |
| 2022/04/24 | 1.2 | Addition of 2.1.2, 2.2 | 하솔비 |
| 2022/04/25 | 1.3 | Addition of 3.1.2, 3.1.3 | 김영욱 |
| 2022/04/26 | 1.4 | Addition of 1, 1.1 | 윤진희 |
| 2022/04/26 | 1.5 | Addition of 2.2 | 하솔비 |
| 2022/04/27 | 1.6 | Addition of 3.1.4, 3.2 | 김승현/임성용 |
| 2022/04/27 | 1.7 | Addition of 2.3, 2.4, 2.5 | 하솔비 |
| 2022/04/28 | 1.8 | Addition of 1.2, 1.3 | 윤진희 |
| 2022/04/29 | 1.9 | Addition of 3.3, 3.4. 3.5 | 김승현/임성용 |
| 2022/04/29 | 1.10 | Addition of 3.6 | 김영욱 |
| 2022/04/30 | 1.11 | Addition of 2.6, 2.7 | 하솔비 |
| 2022/04/30 | 1.12 | Addition of 1.4, 1.5 | 윤진희 |
| 2022/04/30 | 1.13 | Revision of 2 | 하솔비 |
| 2022/04/30 | 1.14 | Revision of 1 | 윤진희 |
| 2022/05/01 | 1.15 | Revision of 3.6 | 이상협 |
| 2022/05/01 | 1.16 | Revision of 3.3, 3.4, 3.5 | 임성용 |
| 2022/05/01 | 1.17 | Revision of 3.1 | 김영욱 |
| 2022/05/01 | 1.18 | Revision of 3.2 | 김승현 |
| 2022/05/01 | 1.19 | Addition of 4 | 윤진희 |
| 2022/05/01 | 1.20 | revision of style and typing errors | 이상협 |
| 2022/05/01 | 1.21 | Final Revision | 김승현 |