



# 멘토 매칭 프로그램

## Software Requirement Specification

2021.04.25.

### Introduction to Software Engineering

#### TEAM 2

Team Leader 2015310783 서기용

Team Member 2015318579 김정훈

Team Member 2017310474 김시인

Team Member 2018312827 김민지

## CONTENTS

<b>1. Introduction</b>	<b>8</b>
1.1. Purpose	8
1.2. Scope	8
1.3. Definitions, Acronyms, and Abbreviation	8
1.4. References	10
1.5. Overview	10
<b>2. Overall Description</b>	<b>11</b>
2.1. Product Perspective	11
2.1.1. System Interfaces	11
2.1.2. User Interfaces	11
2.1.3. Hardware Interfaces	12
2.1.4. Software Interfaces	12
2.1.5. Communications Interfaces	12
2.1.6. Operations	12
2.1.6.1. System administrator	12
2.1.6.2. User	12
2.2. Product Functions	13
2.2.1. 회원가입	13
2.2.2. 사용자 정보 기입	14
2.2.3. 로그인/로그아웃	14
2.2.4. 멘토 멘티 매칭	14
2.2.5. 멘토 멘티 채팅	14
2.2.6. 소모임 참여	15
2.2.7. 소모임 채팅방	15

2.3.	User Characteristics	15
2.3.1.	System Administrator	15
2.3.2.	멘토	15
2.3.3.	멘티	15
2.4.	Constraints	16
2.5.	Assumptions and Dependencies	16
<b>3.</b>	<b>Specific Requirements</b>	<b>16</b>
3.1.	External Interface Requirements	16
3.1.1.	User Interfaces	16
3.1.2.	Hardware Interfaces	23
3.1.3.	Software Interfaces	23
3.1.4.	Communication Interfaces	24
3.2.	Functional Requirements	25
3.2.1.	Use Case	25
3.2.2.	Use Case Diagram	36
3.2.3.	Data Dictionary	37
3.2.4.	Data Flow Diagram	40
3.3.	Performance Requirements	40
3.3.1.	Static numerical requirement	40
3.3.2.	Dynamic numerical requirement	41
3.4.	Logical Database Requirements	41
3.5.	Design Constraints	41
3.6.	Standards compliance	42
3.7.	Software System Characteristics	42
3.7.1.	제품 요구사항 (Product Requirements)	42
3.7.1.1.	사용성 요구사항 (Usability Requirements)	42
3.7.1.2.	성능 요건 (Performance Requirements)	42

3.7.1.3. 사용성 요구사항 (Usability Requirements)	43
3.7.1.4. 보안 요구사항 (Security Requirements)	43
3.7.2. 조직 요구사항 (Organizational Requirements)	43
3.7.2.1. 운영 요구사항 (Operational Requirement)	43
3.7.3. 외부 요구사항 (External Requirements)	43
3.7.3.1. 안전/보안 요구사항 (Safety / Security Requirement)	43
3.8. Organizing the Specific Requirements	44
3.8.1. Context Model	44
3.8.2. Process Model	45
3.8.3. Interaction Model	45
3.8.3.1. Use Case Diagram	45
3.8.4. Behavior Model	46
3.8.4.1. Data Flow Diagram	46
3.8.4.2. Sequence Diagram	46
3.9. System Architecture	47
3.10. System Evolution	47
3.10.1. Limitation and Assumption	47
3.10.2. Evolutions of Hardware and Change of User Requirements	48
<b>4. Supporting Information</b>	<b>48</b>
4.1. Software Requirement Specification	48
4.2. Document History	49

## LIST OF FIGURES

[Figure 1] Use case diagram	36
[Figure 2] Entity Relationship Diagram	39
[Figure 3] Data flow diagram	40
[Figure 4] Context model	44
[Figure 5] Overall process model	45
[Figure 6] Sequence diagram	46
[Figure 7] System architecture of the system	47

## LIST OF TABLES

[Table 1] Table of acronyms and abbreviations	9
[Table 2] Table of terms and definitions	9
[Table 3] User interface of input processing using touchscreen	16
[Table 4] User interface of search	18
[Table 5] User interface of details	19
[Table 6] User interface of Register	20
[Table 7] User interface of Profile	21
[Table 8] User interface of Chatting	22
[Table 9] Hardware interface of applicable device for the system	23
[Table 10] Software interface of firebase real-time database	23
[Table 11] Communication interface of client and host	24
[Table 12] 회원가입 use case	25
[Table 13] 로그인/로그아웃 use case	26
[Table 14] 프로필 use case	27
[Table 15] 추천 멘토/멘티 use case	27
[Table 16] 멘토/멘티 검색 use case	28
[Table 17] 멘토/멘티 프로필 use case	28
[Table 18] 멘토/멘티 신청 use case	29
[Table 19] 멘토/멘티 매칭 use case	30
[Table 20] 멘토/멘티 활동 use case	30
[Table 21] 멘토/멘티 리뷰 use case	31
[Table 22] 멘토/멘티 추천 시스템 use case	32
[Table 23] 소모임 검색 use case	32
[Table 24] 소모임 프로필 use case	33
[Table 25] 소모임 신청 use case	33

[Table 26] 소모임 관리 use case	34
[Table 27] 소모임 활동 use case	35
[Table 28] 사용자	37
[Table 29] 검색 로그	37
[Table 30] 매칭 정보	37
[Table 31] 멘티 사용자	37
[Table 32] 멘토 사용자	38
[Table 33] 채팅 로그	38
[Table 34] 리뷰 데이터	38
[Table 35] 소모임 채팅 로그	38
[Table 36] 소모임 정보	39
[Table 37] Document History	49

# 1. Introduction

## 1.1. Purpose

이 문서는 '멘토 매칭 프로그램'을 설계하고 개발하기에 앞서 Requirement Specification 을 기술하기 위해 작성되었습니다. 앞으로 진행하게 될 design specification 에서도 여기에 기술된 requirements 를 토대로 작성될 것입니다.

이 문서는 시스템 설계를 위한 Specific Requirements 와 Constraints 를 포함합니다.

'멘토 매칭 프로그램'은 다른 커뮤니티 앱과 다르게 익명에서 벗어나 학생들 간에 1:1 소통을 할 수 있는 서비스를 제공합니다. 또, 일반 동아리 가입/개설이 부담스럽거나 원활한 학업을 위한 정보가 필요한 사용자에게 소모임 기능이 제공되기 때문에 사용자들은 본인의 상황에 맞게 타 사용자들과 교류할 수 있습니다.

## 1.2. Scope

'멘토 매칭 프로그램'은 코로나로 인해 학생간 비대면 친목, 교류가 힘든 학생들에게 1:1 멘토 매칭 서비스를 제공합니다. 사용자가 가입 시점에 기입 했던 정보를 이용하여 그에 적합한 멘토를 볼 수 있고, 멘토링을 신청할 수 있습니다. 사용자가 기입 한 정보는 데이터베이스에 저장되고 사용자는 실시간으로 멘토 성사 여부를 확인할 수 있습니다

동아리 가입이나 개설이 부담스러운 사용자에게 소모임 기능도 제공합니다. 검색어 입력을 통해 원하는 분야의 소모임 개설 여부를 확인하고 가입하거나 개설할 수 있습니다.

## 1.3. Definitions, Acronyms, and Abbreviation

아래 표는 이 문서에 사용된 약어를 설명합니다.



[Table 1] Table of acronyms and abbreviations

Acronyms	Explanation
HTTP	Hypertext Transfer Protocol
OS	Operating System
XML	Extensible Markup Language
취준생	취업준비생

아래 표는 이 문서에 사용된 특정 기술용어를 정의합니다.

[Table 2] Table of terms and definitions

Terms	Definitions
멘토링	경험과 지혜가 풍부한 사람이 1:1 로 지도와 조언을 하는 것
멘토	조언을 제공해줄 수 있는 상담 제공자
멘티	멘토에게 조언을 제공 받을 사람
소모임	관심사를 공유하거나 자기 계발을 위해 적은 인원으로 모임을 갖는 것
채팅	네트워크에서 두 명 이상의 사용자가 실시간으로 텍스트 정보를 주고 받는 것. 이 시스템에서는 기능에 따라 제공되는 채팅 서비스가 다름
System Administrator (시스템 관리자)	시스템을 개발하고 관리하는 사람
User	서비스를 제공받는 사람. 이 시스템에서는 멘토와 멘티가 있음
User Interface	사람과 시스템 사이의 소통을 위해 제공되는 가상 매개체
JSON	Key-value 의 쌍으로 이루어진 데이터 오브젝트를 전달하기 위한 포맷

## 1.4. References

- IEEE Recommended Practice for Software Requirements Specifications
- Ian Sommerville, 'Software Engineering 10<sup>th</sup> Edition'
- 2020 SKKUSE Team 1, '[https://github.com/skkuse/2020spring\\_41class\\_team1](https://github.com/skkuse/2020spring_41class_team1)'

## 1.5. Overview

이 문서는 총 4 개의 chapter 로 구성되어 있습니다. Chapter2 에서는 저희가 개발하는 제품의 전체적인 설명을 기술하고 있습니다. 시스템이 포함하고 있는 6 가지의 인터페이스와 시스템의 사용자인 System Administrator, User 가 취할 수 있는 Operation 을 설명합니다. Product Functions 에서는 시스템이 사용자에게 제공하는 기능과 그에 대한 설명들이 나열되어 있습니다. Constraints 와 Assumptions and Dependencies 에서는 개발을 진행함에 있어서 마주하게 될 제약들과 가정들을 포함하고 있습니다.

Chapter 3 에서는 Specification Requirements 를 설명합니다. External Interface Requirements 에서는 Chapter2 에서 나열한 여러 인터페이스를 table 을 이용하여 자세히 기술합니다. Functional Requirements 에서는 table 을 이용한 Use case, Use case diagram, Data Dictionary, Data Flow Diagram 을 확인할 수 있습니다. Performance Requirements 에서는 Static/Dynamic numerical requirements 를 설명합니다. 그 외에도 Logical Database requirements, Design Constraints, Standard Compliance 를 확인할 수 있습니다. Software system characteristics 에서는 시스템의 비기능적 요구사항을 제품요구사항, 조직요구사항, 외부요구사항으로 나누어 설명합니다. Organizing the Specific Requirements 에서는 Context Model, Process Model, Interaction Model, Use case Diagram 과 Behavioral Model 을 보여줍니다. Chapter3 의 마지막에서는 System Architecture 와 요구사항의 변경에 의해 예상되는 System Evolution 을 기술하고 있습니다.

Chapter 4 에서는 부가적인 정보를 제공합니다. 이 문서가 IEEE std 830 을 바탕으로 기술되었음을 설명하고 문서의 작성 기록이 표 형태로 첨부되어 있습니다.

## 2. Overall Description

### 2.1. Product Perspective

‘멘토 매칭 프로그램’은 새내기, 취준생, 복수전공생 등 누군가의 도움이 필요한 사람들에게 적절한 멘토를 찾아서 매칭을 시켜주는 프로그램입니다. 가입 할 때 멘토로서 혹은 멘티로서 가입 신청을 하고, 멘티인 경우에는 누군가의 도움이 필요한 분야, 관심사항 등을 입력하면 그 내용을 토대로 본인에게 맞는 멘토를 소개시켜 주는 프로그램입니다.

부가적으로 소모임 기능을 제공하여 동아리, 스터디그룹, 독강인 사람들이 한데 모여 정보를 공유하거나 관심사를 나눌 수 있습니다.

#### 2.1.1. System Interfaces

사용자의 정보, 사용자가 참여하고 있는 멘토링 정보와 소모임 채팅 정보는 Firebase 를 이용하여 저장됩니다. 사용자가 입력한 데이터는 사용자의 디바이스에서 서버로 전송 된 뒤 서버에서 입력 데이터를 프로세스 한 뒤 사용자가 원하는 데이터를 보여줍니다. 사용자가 원하는 내용을 검색하면 데이터 베이스에 있는 데이터를 알고리즘에 따라 검색합니다. 사용자들이 데이터를 사용하면서 수정/변경 되었던 사항들은 실시간으로 데이터 베이스에 반영됩니다.

#### 2.1.2. User Interfaces

사용자는 TextView 와 ImageView 로 구성된 XML 파일을 Activity Screen 을 통해 볼 수 있습니다. 사용자가 원하는 행동을 실행하기 위해서는 모바일 기기의 터치스크린을 클릭합니다. 멘티로 가입한 사용자는 가입 시에 기입한 사항을 토대로 본인에게 적합한 멘토를 추천받거나 키워드를 입력하여 멘토를 찾을 수 있습니다. 멘토는 자신에게 멘토링을 신청한 멘티를 확인할 수 있습니다.

사용자는 검색어 입력을 통해 원하는 개설된 소모임 리스트를 확인할 수 있습니다. 검색 결과가 만족스럽지 않으면 스크롤을 내려 다른 게시물을 확인할 수 있습니다.

### 2.1.3. Hardware Interfaces

1GB RAM, 1.0GHZ, Dual Processor 이상의 성능을 갖는 모바일 기기에서 사용 가능합니다.

### 2.1.4. Software Interfaces

Firebase 를 이용하여 데이터 사용자가 변경한 내용이 데이터 베이스에서 실시간으로 반영되며, 안드로이드 버전 6.0 이상의 환경에서 지원합니다.

### 2.1.5. Communications Interfaces

서비스를 제공받기 위해서는 데이터 베이스의 서버와 연결되어 있어야 합니다.

### 2.1.6. Operations

#### 2.1.6.1. System administrator

- 사용자에게 추천 멘토/멘티 출력
- 가입시 사용자가 기입한 정보로부터 키워드 추출
- 사용자의 데이터 처리
- 데이터베이스 관리

#### 2.1.6.2. User

- 회원가입
  - 학교 인증을 거친 후 계정을 만들 수 있습니다. 회원 가입에 필요한 정보는 고유 닉네임, 비밀번호, 성별, 멘토/멘티 선택 여부, 멘토/멘티 활동 분야입니다.
- 로그인/로그아웃
  - 회원 가입시 생성한 계정을 이용하여 로그인 할 수 있습니다. 로그아웃 버튼을 누르거나 사용자가 어플리케이션을 닫으면 세션은 임의로 닫힙니다.

- 개인 정보 수정
  - 가입 시 기입한 정보와 관심분야를 수정할 수 있습니다.
- 멘토/멘티 찾기
  - 회원 가입 시 입력한 정보를 바탕으로 멘토/멘티를 추천 받을 수 있습니다. 또는, 직접 검색을 통해 찾을 수 있습니다.
- 프로필 열람
  - 멘토/멘티는 마음에 든 타사용자의 프로필을 확인할 수 있습니다.
- 멘토링 신청/수락/거부
  - 멘티는 기입한 정보를 바탕으로 멘토를 추천 받고 멘토링 신청을 할 수 있습니다. 멘토는 이에 대해 수락하거나 거부할 수 있습니다.
- 채팅
  - 매칭이 성사된 멘토 멘티는 서로만 참여할 수 있는 채팅방을 통해 소통할 수 있습니다.
- 리뷰작성
  - 멘토와 멘티는 서로에게 평점을 주고 리뷰를 할 수 있습니다.
- 소모임 참가 요청/수락/거부
  - 개설된 소모임 관리자에게 참가 요청서를 보낼 수 있고, 소모임 관리자는 이를 확인하고 수락 여부를 결정합니다.
- 소모임 채팅
  - 개설된 소모임은 2 인 이상의 채팅방을 제공받습니다.

## 2.2. Product Functions

### 2.2.1. 회원가입

사용자는 서비스를 이용하기 위해서 가입을 필수적으로 해야 합니다. 로그인 페이지에서

‘회원가입’ 버튼을 통해 회원 가입 페이지로 이동해서 회원 가입을 진행합니다. 성균관대학교 재학생 또는 졸업생임을 증명할 수 있도록 성균관대학교 이메일을 필수적으로 기입해야 합니다. 성균관대학교 이메일로 발송된 인증번호를 기입하여 성균관대학교 재학생/졸업생임을 인증받습니다.

### 2.2.2. 사용자 정보 기입

‘멘토 멘티 매칭 프로그램’을 사용하기 위해서는 회원 가입과는 별개로 멘토 멘티 매칭을 하기 위한 사용자 정보를 필수적으로 기입해야 합니다. 사용자는 필수적으로 고유 닉네임, 비밀번호를 입력하고 본인이 멘토로서 활동할 것인지, 멘티로서 활동할 것인지를 선택해야 합니다. 다음으로 멘토가 필요한 분야 또는 멘토로서 도움을 줄 수 있는 분야를 선택합니다. 멘토는 추가적으로 해당 분야에서 왜 멘토가 필요한지, 어떤 어려움이 있는지를 기입해야 합니다. 멘티로서 가입하는 사람은 해당 분야에서 어떤 도움을 줄 수 있는지 등 본인을 소개하는 글을 간략하게 기입합니다. 선택적으로 입력할 수 있는 필드에서는 기타 특이사항을 입력합니다.

### 2.2.3. 로그인/로그아웃

이미 회원 가입을 마친 사용자에게 한하여 로그인 할 수 있습니다. 회원 가입 시 설정한 웹메일 주소와 비밀번호를 이용하여 로그인 합니다. 사용을 마치고 로그아웃 버튼을 눌러 로그아웃 하거나 애플리케이션을 종료하면 로그아웃 할 수 있습니다.

### 2.2.4. 멘토 멘티 매칭

가입 시 입력했던 사용자 정보를 바탕으로 사용자에게 적합한 멘토/멘티를 제시합니다. 멘티로 가입한 사용자는 본인의 활동분야와 일치하는 멘토들과 그들이 멘토링과 관련하여 기입한 정보를 볼 수 있습니다. 원하는 멘토를 찾으면 ‘멘토 신청’ 버튼을 눌러 해당 멘토에게 멘토링을 요청합니다. 멘토는 자신에게 멘토 요청을 보낸 사용자가 기입한 멘토링에 관한 정보를 읽을 수 있고 이를 바탕으로 멘토링 요청을 받을지, 거절할지 선택할 수 있습니다. 멘토가 멘토링 요청을 수락하면 멘토 멘티 매칭이 성사됩니다.

### 2.2.5. 멘토 멘티 채팅

멘토-멘티 페어에 대해서 서로만 참여하는 채팅 프로그램이 제공됩니다. 채팅 내용은 일정 기간 동안 데이터베이스에 저장됩니다. 채팅 중 부적절한 언행을 한 사용자를 신고할 수 있습니다.

### 2.2.6. 소모임 참여

홈에서 '소모임' 버튼을 눌러 소모임 활동을 할 수 있는 페이지로 이동합니다. '소모임' 기능에서 사용자는 자유롭게 소모임을 개설 하거나 참여할 수 있습니다. 또, 검색창에 키워드를 입력함으로써 관심 있는 소모임의 개설 여부를 확인하고 참가 신청을 할 수 있습니다.

### 2.2.7. 소모임 채팅방

검색을 통해서 참여하고자 하는 채팅방을 찾으면 관리자에게 쪽지를 통해 참여 의사를 보낼 수 있습니다. 관리자가 참여 코드를 제공하면 사용자는 소모임 채팅방에 참가하게 됩니다. 소모임 채팅방에서는 두 명 이상이 동시에 활동할 수 있습니다.

## 2.3. User Characteristics

### 2.3.1. System Administrator

System Administrator 는 System architecture 지식이 있고 시스템이 제공하는 서비스에 대해 전반적인 이해를 갖는 사람입니다. 데이터 베이스 시스템과 네트워크 시스템에 대한 전반적인 지식이 필요합니다. 시스템 관리 도구를 사용할 수 있고 브라우저를 통해 시스템에 접근할 수 있으며 부적절한 게시글을 삭제할 수 있는 권한이 있습니다.

### 2.3.2. 멘토

멘토 멘티 매칭 프로그램은 기본적으로 성균관대학교 재학생과 졸업생을 대상으로 서비스를 제공하기 때문에 성균관대학교 재학생이거나 졸업생이어야 합니다. User Interface 는 한국어로 제공되므로 프로그램을 사용할 수 있을 정도로 한국어를 구사할 수 있어야 합니다. 멘토가 되기 위한 특별한 자격요건은 없으나 해당 분야에서 누군가에게 조언을 줄 수 있어야 합니다.

### 2.3.3. 멘티

멘티는 멘토와 마찬가지로 성균관대학교 재학생이거나 졸업생이어야 합니다. 프로그램을

사용할 수준의 한국어를 구사할 수 있어야 합니다.

## 2.4. Constraints

앞으로 진행될 시스템 설계는 이 문서에 있는 내용에 맞춰 진행될 것입니다. 그 밖의 제약 사항은 다음과 같습니다.

- 가능한 한 범용되는 오픈 소스를 활용할 것입니다.
- 검색에 소요되는 시간은 5 초를 넘기지 않도록 합니다.
- 대안이 없는 경우를 제외하고 사용료를 지불해야 하는 제품은 사용하지 않습니다.
- 프론트엔드는 안드로이드 스튜디오와 Kotlin 을 사용하여 개발합니다.
- 백엔드는 Python 과 Firebase 를 이용하여 개발합니다.
- 사용자의 피드백을 기반으로 제품을 향상시킵니다.
- 향후에 발생하는 User requirement change 에 대비해 코드를 간결하게 작성하고 code 에 대해 가능한 한 comment 를 달도록 합니다.

## 2.5. Assumptions and Dependencies

이 문서는 안드로이드 개발 환경을 설정하고 작성되었습니다. 사용자는 안드로이드 5.1 버전 이상에서 서비스를 제공 받을 수 있습니다.

# 3. Specific Requirements



## 3.1. External Interface Requirements

### 3.1.1. User Interfaces

[Table 3] User interface of input processing using touchscreen

Name	Basic User Interaction
Purpose/Description	사용자는 기기의 터치 스크린을 통해 지침을 전송한다
Input source/ Output	Android 운영 체제가 탑재 되어 있는 스마트폰



destination	
Range / Accuracy / Margin of error	<ul style="list-style-type: none"> <li>- 핸드폰 Screen size 에 따라 7~10 개 사이</li> <li>- 사용자로부터 터치의 정확성에 따라</li> <li>- 핸드폰 스크린에 따른 민감도</li> </ul>
Unit	클릭
Time/ Velocity	비동기 사용자의 입력 / 사용자의 명령에 즉각적으로 실행
Relationship with other input/outputs	사용자가 입력한 모든 데이터가 사용자의 디바이스에서 버서로 전송 후에 서버에서 입력 데이터를 프로세스 한 뒤 사용자가 원하는 아웃풋 데이터를 개시한다
Format and configuration of screen	<ul style="list-style-type: none"> <li>- Activity 스크린이 XML 파일로 연결             <ul style="list-style-type: none"> <li>➢ XML 파일은 TextView 와 ImageViews 로 구성 된다</li> </ul> </li> <li>- TextViews 와 ImageView 는 앱에 검색, 입력, 그리고 버튼들에 기본 베이스로 작동한다.</li> <li>- 사용자들은 원하는 행동을 실행하기 위해 클릭하거나 입력한다.</li> </ul> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>&lt; 앱 초기 화면 &gt;</p>  </div> <div style="text-align: center;"> <p>&lt; 게시판 &gt;</p>  </div> </div>
Format and configuration of window	N/A

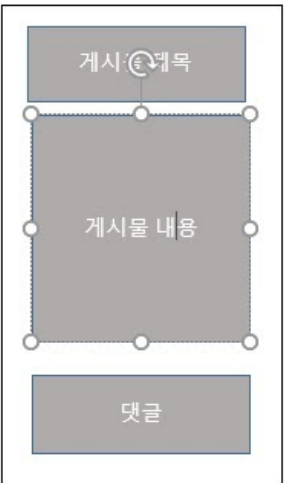
Data type	정수(int)
Instruction type	Instruction mapping 은 값에 맞춰 기능하도록 한다
Exit message	N/A

[Table 4] User interface of search

Name	Search Interface 1 – Search
Purpose/Description	<ul style="list-style-type: none"> <li>- 유저들이 검색하는 것을 키워드로 맞춰서 서버에 있는 데이터에 검색한다</li> <li>- 키워드와 적합한 게시물들을 게시한다</li> <li>- 게시물은 최근 게시된 것을 우선적으로 게시한다</li> </ul>
Input source/ Output destination	호스트 서버 / 클라이언트
Range / Accuracy / Margin of error	<ul style="list-style-type: none"> <li>- 검색한 모든 게시물들이 나온다</li> <li>- 키워드에 맞춰 검색 알고리즘에 따라 검색된다</li> <li>- Margin error set 에 알고리즘 문제가 발생할 수 있다</li> </ul>
Unit	스크린
Time/ Velocity	모든 입력 데이터가 디바이스에서 서버로 전송하는 시간이 최소한으로 소요되어야 유저가 디바이스 스크린에 빠르게 사용할 수 있다
Relationship with other input/outputs	유저들이 새로운 게시물을 추가 할 때마다 서버 데이터에 영향을 주기 때문에 결과가 나올 때 역학적으로 기능해야 한다.
Format and configuration of screen	<div> <div> <ul style="list-style-type: none"> <li>- Activity 스크린은 XML 로 연결된다. 코드는 Textview 로 구성이 되며 사용자가 원하는 검색 결과를 텍스트 파일로 게시한다.</li> <li>- 게시 된 검색 결과에 만족하지 않으면 스크롤을 내려 다른 게시글을 확인할 수 있다.</li> </ul> </div> <div> <p>&lt; 검색 &gt;</p> <div> <div>검색</div> <div>게시물</div> </div> </div> </div>

Format and configuration of window	XML documentation
Data type	Text, Widget, Image
Instruction type	N/A
Exit message	N/A

[Table 5] User interface of details

Name	Search Interface 2 – Details
Purpose/Description	- 게시물이 스크린에 게시가 되어 있을 때 각 게시글에 주제를 읽어서 클릭해서 게시글과 댓글을 읽을 수 있다
Input source/ Output destination	호스트 서버 / 클라이언트
Range / Accuracy / Margin of error	N/A
Unit	디바이스 화면
Time/ Velocity	유저에 비동기 입력 / 서버와 유저 디바이스에 통신 시간
Relationship with other input/outputs	N/A
Format and configuration of screen	<ol style="list-style-type: none"> <li>1. 게시글 제목과 게시글 내용을 읽을 수 있다. 또한 각 게시글에 댓글을 작성할 수 있다.</li> <li>2. 게시글은 사진과 텍스트로 표시된다.</li> </ol> 

Format and configuration of window	게시글에 댓글을 달 수 있는 칸이 존재한다.
Data type	Text, Widget, Image
Instruction type	N/A
Exit message	N/A

[Table 6] User interface of Register

Name	Customization Interface 1 – Register
Purpose/Description	- 유저들이 게시글을 작성하고 글을 게시할 때 제목과 내용을 익명 또는 고유 닉네임으로 글을 게시할 수 있다.
Input source/ Output destination	호스트 서버 / 클라이언트
Range / Accuracy / Margin of error	N/A
Unit	스크린
Time/ Velocity	유저가 작성한 글을 게시할 때 서버와 디바이스 통신 간의 소요 시간이 최소한으로 작동해야 한다.
Relationship with other input/outputs	N/A
Format and configuration of screen	<div> <p>1. 위칸에 게시글의 제목을 작성한다</p> <p>2. 하단 칸에 게시글의 본문 내용을 작성한다. 하단의 아이콘을 통해 사진을 추가하고 익명 여부를 설정할 수 있다.</p> </div> <div> <p>&lt; 게시물 작성 &gt;</p> </div>

Format and configuration of window	N/A
Data type	Text, Image, Query
Instruction type	사진과 글로 각각 설명되어 있다.
Exit message	"게시글이 게시되었습니다!"

[Table 7] User interface of Profile

Name	Customization Interface 2 – Profile
Purpose/Description	- 유저들은 로그인 한 후에 설정으로 들어가 계정 설정을 변경할 수 있다.
Input source/ Output destination	호스트 서버 / 유저
Range / Accuracy / Margin of error	N/A
Unit	페이지
Time/ Velocity	N/A
Relationship with other input/outputs	N/A
Format and configuration of screen	<div> <div> 1. 계정 설정에서 비밀번호와 이메일 등을 수정할 수 있다.  2. 커뮤니티에서는 닉네임과 프로필 사진, 문자 설정 등을 변경할 수 있다. </div> <div> <div>&lt;프로필&gt;</div> <div> <div>Profile</div> <div>계정</div> <div>비밀번호 변경</div> <div>이메일 변경</div> <div>커뮤니티</div> <div>닉네임 변경</div> <div>프로파일 수정</div> <div>설정 관리</div> <div>앱 규정</div> </div> </div> </div>

Format and configuration of window	N/A
Data type	Text, Query
Instruction type	N/A
Exit message	"수정 완료 되었습니다!"

[Table 8] User interface of Chatting

Name	Customization Interface 3 – Chatting	
Purpose/Description	유저들 간의 소통과 대화를 도와주면서 개인적인 정보 노출을 방지하기 위해서 필요한 기능	
Input source/ Output destination	호스트 서버 / 유저	
Range / Accuracy / Margin of error	N/A	
Unit	화면	
Time/ Velocity	유저들 간의 대화를 할 때 서버와 각 사용자 기기의 통신 시간을 줄여야 한다.	
Relationship with other input/outputs	N/A	
Format and configuration of screen	<div> <div>1. 일반적인 채팅 기능처럼 사용자 간의 대화 내용이 표시된다.</div> <div>2. 발화자 표시는 커뮤니티 닉네임으로 게시된다.</div> </div> <div> <div>&lt; Chatting &gt;</div> <div>커뮤니티 닉네임</div> <div>대화 내용 내용</div> <div>입력</div> </div>	

Format and configuration of window	N/A
Data type	Text, Query
Instruction type	N/A
Exit message	"수정 완료 되었습니다!"

### 3.1.2. Hardware Interfaces

[Table 9] Hardware interface of applicable device for the system

Name	Applicable device for the system
Purpose/Description	모든 안드로이드 디바이스는 버전 6.0 이상이어야 한다.

### 3.1.3. Software Interfaces

[Table 10] Software interface of firebase real-time database

Name	Firebase Real-time Database
Purpose/Description	쿼리가 모든 게시글에 있는 입력 값과 출력 값과 같은 멀티미디어와 메타 데이터를 관리한다
Input source / Output destination	Host server/ User, User/Host server
Range/ Accuracy/ Margin of error	보안 성능에 따라 다를 수 있다
Unit	Query
Time/ Velocity	즉각적으로 반응한다

Relationship with other input/outputs	서버에 있는 모든 입력값과 출력은 관계되어 있다
Format and configuration of screen	N/A
Format and configuration of window	N/A
Data type	Query
Instruction type	Query statement
Exit message	N/A

### 3.1.4. Communication Interfaces

[Table 11] Communication interface of client and Server

Name	Client and Server
Purpose/Description	<ul style="list-style-type: none"> <li>- 모든 유저들은 어플리케이션의 서버에 연결해야 한다.</li> <li>- 유저가 원하는 모든 게시글은 서버 데이터 베이스에서 제공한다.</li> </ul>
Input source/ Output destination	Client/Server
Unit	packet
Time/ Velocity	최소 10Mbps
Relationship with other input/outputs	서버의 모든 입력 값과 출력 값
Format and configuration of screen	N/A
Format and configuration of window	N/A
Data type	Query
Instruction type	Query statement
Exit message	N/A



## 3.2. Functional Requirements

### 3.2.1. Use Case

[Table 12] 회원가입 use case

Use case name	회원가입
Actor	비회원 사용자
Description	비회원 사용자가 어플리케이션에 접속하기 위해 회원으로 가입하려는 단계
Normal Course	<ol style="list-style-type: none"> <li>1. 모든 사용자는 어플리케이션 실행 후 로그인 페이지를 마주한다.</li> <li>2. 회원가입한 사용자에게만 서비스가 제공되는 것을 확인한 후, 비회원 사용자는 로그인 페이지에서 회원가입 버튼을 클릭한다.</li> <li>3. 비회원 사용자는 회원가입 페이지로 이동한다.</li> <li>4. 회원가입은 학교 웹메일을 통해 인증되었을 경우에만 가능하다.</li> <li>5. 비회원 사용자는 회원가입 양식에 따라 추가 정보를 입력해야 한다. 필요한 정보는 다음과 같다. <ul style="list-style-type: none"> <li>• 학교 웹메일(ID)</li> <li>• 고유 닉네임</li> <li>• 비밀번호</li> <li>• 성별</li> <li>• 멘토 / 멘티 선택 여부</li> <li>• 멘토 / 멘티 분야 선택</li> </ul> </li> <li>6. 시스템은 해당 학교 학생인지 확인하고 비밀번호 찾기 상황을 대비하기 위해 위에서 작성한 이메일 주소로 인증 코드를 전송한다.</li> <li>7. 인증코드 인증과 양식 작성이 완료되고 양식 마지막에 있는 회원가입 버튼을 클릭하면 로그인 페이지로 돌아간다</li> </ol>
Precondition	<p>사용자는 아직 프로그램에 회원가입 하지 않은 상태다.</p> <p>사용자가 올바른 정보를 입력한다.</p> <p>사용자의 웹메일 주소가 다른 사용자의 웹메일 주소와 일치해서는 안된다.</p> <p>잘못된 입력의 경우 프로그램은 웹메일 주소 및 비밀번호 형식을 확인한다.</p>

Post Condition	비밀번호는 보안을 위해 암호화되어 사용자 관리 데이터베이스에 저장한다.
Assumptions	N/A

[Table 13] 로그인 / 로그아웃 use case

Use case name	로그인 / 로그아웃
Actor	회원가입 된 사용자
Description	로그인은 프로그램에 등록된 사용자가 어플리케이션을 이용하기 위해 시스템에 접속을 시도하는 과정이다. 로그아웃은 로그인한 사용자가 어플리케이션에서 나가려고 하는 과정이다.
Normal Course	<p>&lt;로그인&gt;</p> <ol style="list-style-type: none"> <li>이미 회원으로 프로그램에 등록된 사용자가 시스템에서 어플리케이션을 이용하고자 하는 경우</li> <li>사용자가 회원가입을 위해 설정한 웹메일 주소와 비밀번호를 입력한 후 '로그인' 버튼을 클릭한다.</li> <li>웹메일과 비밀번호 정보가 데이터베이스에 저장된 정보와 일치하면 프로그램은 사용자가 프로그램에 접속할 수 있도록 허용하고 사용자는 이제 시스템에서 제공하는 모든 서비스를 이용할 수 있다.</li> </ol> <p>&lt;로그 아웃&gt;</p> <ol style="list-style-type: none"> <li>사용자가 시스템에서 나가기를 원할 경우 '로그 아웃' 버튼을 클릭한다.</li> <li>사용자가 로그아웃하지 않고 어플리케이션을 닫으면 프로그램은 해당 사용자의 세션을 임의로 닫는다.</li> </ol>
Precondition	<p>&lt;로그인&gt;</p> <p>사용자는 이미 프로그램에 회원가입 되어있어야 한다.</p> <p>&lt;로그아웃&gt;</p> <p>사용자는 이미 로그인한 상태여야 한다.</p>
Post Condition	사용자는 네트워크에 연결되어 있어야 한다.
Assumptions	N/A

[Table 14] 프로필 use case

Use case name	프로필
Actor	회원가입 된 사용자
Description	사용자가 회원가입 할 때 입력한 사용자 정보를 확인하고 수정할 수 있는 페이지다. 사용자는 멘토, 멘티 선택과 관심 분야를 수정할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 사용자가 '프로필' 버튼을 클릭하면 프로필 수정 페이지로 이동한다.</li> <li>2. 사용자는 회원가입시 선택했던 닉네임과 관심분야, 멘토 / 멘티 여부를 확인하고 수정할 수 있다.</li> <li>3. 정보를 수정한 사용자는 '수정' 버튼을 클릭해 이전 페이지로 돌아간다.</li> </ol>
Precondition	사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.
Post Condition	<p>프로필 수정 페이지가 종료되면 수정 정보가 사용자 데이터베이스로 전달되어 사용자 정보가 업그레이드 되어야 한다.</p> <p>업그레이드된 사용자 데이터베이스 바탕으로 추천 멘토 혹은 멘티를 추천한다.</p>
Assumptions	N/A

[Table 15] 추천 멘토 / 멘티 use case

Use case name	추천 멘토 / 멘티
Actor	회원가입 된 사용자
Description	사용자는 사용자에게 적합한 추천 멘토 혹은 멘티 리스트를 보고 원하는 사용자를 선택할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 로그인 후 사용자는 회원가입시 입력한 선호 키워드를 바탕으로 적합한 멘토 혹은 멘티 리스트를 추천 받는다.</li> <li>2. 추천 리스트에는 사용자와 가장 적합한 순으로 정렬된 최대 5 명의 멘토 혹은 멘티가 표시된다.</li> <li>3. 사용자는 더 많은 추천 멘토 혹은 멘티 리스트를 확인하기 위해 '더보기' 버튼을 클릭할 수 있다.</li> <li>4. 추천 멘토 혹은 멘티 리스트로 출력된 사용자를 선택하면 선택된 사용자의 프로필을 확인할 수 있다.</li> </ol>

Precondition	사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.
Post Condition	N/A
Assumptions	사용자는 마음에 드는 멘토 혹은 멘티를 선택해야 한다.

[Table 16] 멘토 / 멘티 검색 use case

Use case name	멘토 / 멘티 검색
Actor	회원가입 된 사용자
Description	사용자가 찾고자 하는 멘토 혹은 멘티를 검색한 후 선택할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>로그인 후 사용자는 '검색' 버튼을 클릭한다.</li> <li>찾고자 하는 멘토 혹은 멘티의 닉네임이나 웹메일 주소를 검색하면 검색 키워드와 관련이 높은 순으로 최대 10 명의 멘토 혹은 멘티가 표시된다.</li> <li>사용자는 더 많은 검색 결과를 확인하기 위해 '더보기' 버튼을 클릭할 수 있다.</li> <li>검색결과로 출력된 사용자를 선택하면 선택된 사용자의 프로필을 확인할 수 있다.</li> </ol>
Precondition	사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.
Post Condition	검색 기록은 로컬 데이터베이스에 저장되어야 한다.
Assumptions	N/A

[Table 17] 멘토 / 멘티 프로필 use case

Use case name	멘토 / 멘티 프로필
Actor	회원가입 된 사용자
Description	사용자는 특정 사용자를 선택해 해당 사용자의 프로필을 확인할 수 있다.

Normal Course	<ol style="list-style-type: none"> <li>1. 로그인한 한 사용자가 추천 멘토 혹은 멘티나 검색을 통해 표시된 다른 사용자의 '프로필' 버튼을 클릭한다.</li> <li>2. 사용자는 선택한 사용자의 세부 프로필 페이지로 이동한다.</li> <li>3. 사용자는 해당 사용자에 대한 닉네임과 자기소개, 관심 분야, 이전에 매칭된 멘토 혹은 멘티의 리뷰를 포함한 세부 정보를 확인할 수 있다.</li> <li>4. '멘티 신청' 혹은 '멘토 신청' 버튼을 통해 멘토 혹은 멘티를 신청할 수 있다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>검색을 통해 다른 사용자의 프로필을 확인한 경우 검색 과정을 완료해야 한다.</p> <p>사용자의 마음에 든 멘토 혹은 멘티가 한 명 이상 있어야 한다.</p>
Post Condition	N/A
Assumptions	N/A

[Table 18] 멘토 / 멘티 신청 use case

Use case name	멘토 / 멘티 신청
Actor	회원가입 된 사용자
Description	사용자는 자신의 마음에 든 특정 사용자를 선택해서 멘토 혹은 멘티로 신청할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 로그인한 사용자는 특정 사용자의 프로필에 위치한 '멘토 신청' 혹은 '멘티 신청' 버튼을 클릭한다.</li> <li>2. 멘토 혹은 멘티가 신청되었다는 팝업 메시지가 표시된다.</li> <li>3. 신청을 받은 사용자에게 멘토 혹은 멘티 신청이 도착했다는 푸시 알림이 전송된다.</li> <li>4. 신청을 받은 사용자는 '수락' 버튼이나 '거절' 버튼을 통해 멘토 혹은 멘티를 수락하거나 거절 할 수 있다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 특정 사용자의 프로필에 접속해 있어야 한다.</p>

Post Condition	N/A
Assumptions	N/A

[Table 19] 멘토 / 멘티 매칭 use case

Use case name	멘토 / 멘티 매칭
Actor	멘토 혹은 멘티 신청을 받은 회원가입 된 사용자
Description	멘토 혹은 멘티 신청을 받은 사용자는 신청자의 프로필을 확인하고 매칭을 수락할지 거절할지 결정한다.
Normal Course	<ol style="list-style-type: none"> <li>1. 멘토 혹은 멘티 신청을 받은 사용자는 '받은 신청' 페이지로 이동한다.</li> <li>2. '받은 신청' 페이지에 멘토 혹은 멘티를 신청한 사용자의 리스트가 표시된다.</li> <li>3. 리스트의 사용자 중 한 사용자를 선택하면 해당 사용자의 프로필을 확인할 수 있다.</li> <li>4. 해당 사용자가 마음에 든다면 멘토 혹은 멘티 신청을 받은 사용자는 '수락' 버튼을 클릭한다. 반면 마음에 들지 않는다면 '거절' 버튼을 클릭한다. <ol style="list-style-type: none"> <li>4-1. '수락'을 클릭한 경우 매칭이 성사된다.</li> <li>4-2. '거절'을 클릭한 경우 매칭이 결렬된다.</li> </ol> </li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 특정 사용자에게 멘토 혹은 멘티 신청을 받은 상태여야 한다.</p> <p>사용자는 '받은 신청' 페이지에 접속해 있어야 한다.</p>
Post Condition	매칭이 성사된 경우 성사된 매칭 정보가 데이터 베이스에 저장된다.
Assumptions	멘토 혹은 멘티 신청을 받은 사용자는 멘토 / 멘티 활동을 진행하려는 계획이 있다.

[Table 20] 멘토 / 멘티 활동 use case

Use case name	멘토 / 멘티 활동
Actor	멘토 / 멘티 매칭이 성사된 회원가입 된 사용자
Description	멘토 / 멘티 매칭이 성사된 사용자는 채팅을 통해 멘토 / 멘티 활동을 진행할 수 있다.

Normal Course	<ol style="list-style-type: none"> <li>1. 멘토 / 멘티 매칭이 성사된 사용자는 멘토 / 멘티 활동 탭으로 이동한다.</li> <li>2. 멘토 / 멘티 활동 탭에서 사용자들은 채팅을 나눌 수 있다.</li> <li>3. 하단의 입력 창을 누르면 키보드 자판이 띄워지고, 보내고자 하는 메시지를 입력하고 우측의 '전송' 버튼을 누르면 메시지가 전송된다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 한 명 이상의 특정 사용자와 멘토 / 멘티 매칭이 성사된 상태여야 한다.</p> <p>사용자는 멘토 / 멘티 활동 페이지에 접속해 있어야 한다.</p>
Post Condition	채팅 로그는 한달 주기로 데이터 베이스에 임시저장 된다.
Assumptions	N/A

[Table 21] 멘토 / 멘티 리뷰 use case

Use case name	멘토 / 멘티 리뷰
Actor	멘토 / 멘티 활동을 한달 이상 진행한 회원가입 된 사용자
Description	멘토 / 멘티 매칭이 성사된 후 활동을 한달 이상 진행한 사용자는 각자의 멘토와 멘티에게 리뷰 할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 멘토 / 멘티 매칭이 성사된 후 활동을 한달 이상 진행한 사용자는 멘토 / 멘티 활동 탭에 들어가 각자 멘토 혹은 멘티의 프로필에 접속한다.</li> <li>2. 각 멘토 혹은 멘티의 프로필에 접속한 사용자는 '리뷰' 탭을 클릭해 각 사용자에게 대해 별점과 리뷰를 남긴다. <ol style="list-style-type: none"> <li>2.1 별점은 최소 1 개에서 최대 5 개를 남길 수 있다.</li> <li>2.2 리뷰는 최소 50 자 이상 작성해야 한다.</li> </ol> </li> <li>3. '리뷰 저장' 버튼을 클릭하면 리뷰가 저장되며 각 사용자의 프로필 리뷰 탭에 표시된다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 한 명 이상의 특정 사용자와 멘토 / 멘티 매칭 활동을 한 달 이상 진행한 상태여야 한다.</p> <p>사용자는 매칭된 멘토 혹은 멘티의 프로필에 접속해 있어야 한다.</p>
Post Condition	<p>리뷰 내용과 별점은 데이터베이스에 저장된다.</p> <p>각 사용자의 프로필 리뷰 탭에 리뷰 내용과 별점이 표시된다.</p>
Assumptions	N/A

[Table 22] 멘토 / 멘티 추천 시스템 use case

Use case name	멘토 / 멘티 추천 시스템
Actor	시스템 관리자
Description	멘토 혹은 멘티 리뷰와 사용자의 관심분야를 분석해 매칭을 시도하는 사용자에게 추천 멘토 혹은 멘티를 출력한다.
Normal Course	<ol style="list-style-type: none"> <li>1. 사용자 관리 시스템은 사용자에 대한 리뷰나 관심분야와 같은 방대한 데이터를 수집한다.</li> <li>2. 데이터는 전처리한 후 데이터 베이스에 저장된다.</li> <li>3. 사용자의 관심분야 데이터를 처리하는 동안 리뷰 데이터 또한 처리되어 데이터 베이스에 저장된다.</li> <li>4. 저장된 데이터를 바탕으로 사용자와 관련성이 높은 사용자를 추천한다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 프로필에 관심 분야를 설정한 상태여야 한다.</p>
Post Condition	N/A
Assumptions	각 사용자의 리뷰 수는 분석에 필요한 최소 개수 이상이어야 한다.

[Table 23] 소모임 검색 use case

Use case name	소모임 검색
Actor	회원가입 된 사용자
Description	사용자가 찾고자 하는 소모임을 선택할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 로그인 후 사용자는 '검색' 버튼을 클릭한다.</li> <li>2. 찾고자 하는 소모임의 이름이나 키워드를 검색하면 검색 키워드와 관련이 높은 순으로 최대 10 명의 멘토 혹은 멘티가 표시된다.</li> <li>3. 사용자는 더 많은 검색 결과를 확인하기 위해 '더보기' 버튼을 클릭할 수 있다.</li> <li>4. 검색 결과로 출력된 사용자를 선택하면 선택된 사용자의 프로필을 확인할</li> </ol>



	수 있다.
Precondition	사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.
Post Condition	검색 기록은 로컬 데이터베이스에 저장되어야 한다.
Assumptions	N/A

[Table 24] 소모임 프로필 use case

Use case name	소모임 프로필
Actor	회원가입 된 사용자
Description	사용자는 특정 소모임을 선택해 해당 소모임의 프로필을 확인할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 로그인한 한 사용자가 검색을 통해 표시된 소모임 아이콘을 클릭한다.</li> <li>2. 사용자는 선택한 소모임의 세부 프로필 페이지로 이동한다.</li> <li>3. 사용자는 해당 소모임에 대한 소모임 명과 소모임 소개, 관심 분야 등의 세부 정보를 확인할 수 있다.</li> <li>4. '소모임 가입' 버튼을 통해 소모임에 가입 신청을 할 수 있다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>소모임 검색 과정을 완료해야 한다.</p> <p>사용자의 마음에 든 소모임이 하나 이상 있어야 한다.</p>
Post Condition	N/A
Assumptions	N/A

[Table 25] 소모임 신청 use case

Use case name	소모임 신청
Actor	회원가입 된 사용자
Description	사용자는 자신의 마음에 든 특정 소모임을 선택해서 가입 신청을 할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 로그인한 사용자는 특정 소모임의 프로필에 위치한 '소모임 가입' 버튼을 클릭한다.</li> <li>2. 소모임 가입이 신청되었다는 팝업 메시지가 표시된다.</li> <li>3. 신청을 받는 소모임 관리자에게 소모임 가입 신청이 도착했다는 푸시</li> </ol>

	<p>알림이 전송된다.</p> <p>4. 신청을 받은 소모임 관리자는 신청자의 프로필을 확인하고 '수락' 버튼이나 '거절' 버튼을 통해 신청자를 가입시키거나 거절 할 수 있다.</p>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 특정 소모임의 프로필에 접속해 있어야 한다.</p>
Post Condition	N/A
Assumptions	N/A

[Table 26] 소모임 관리 use case

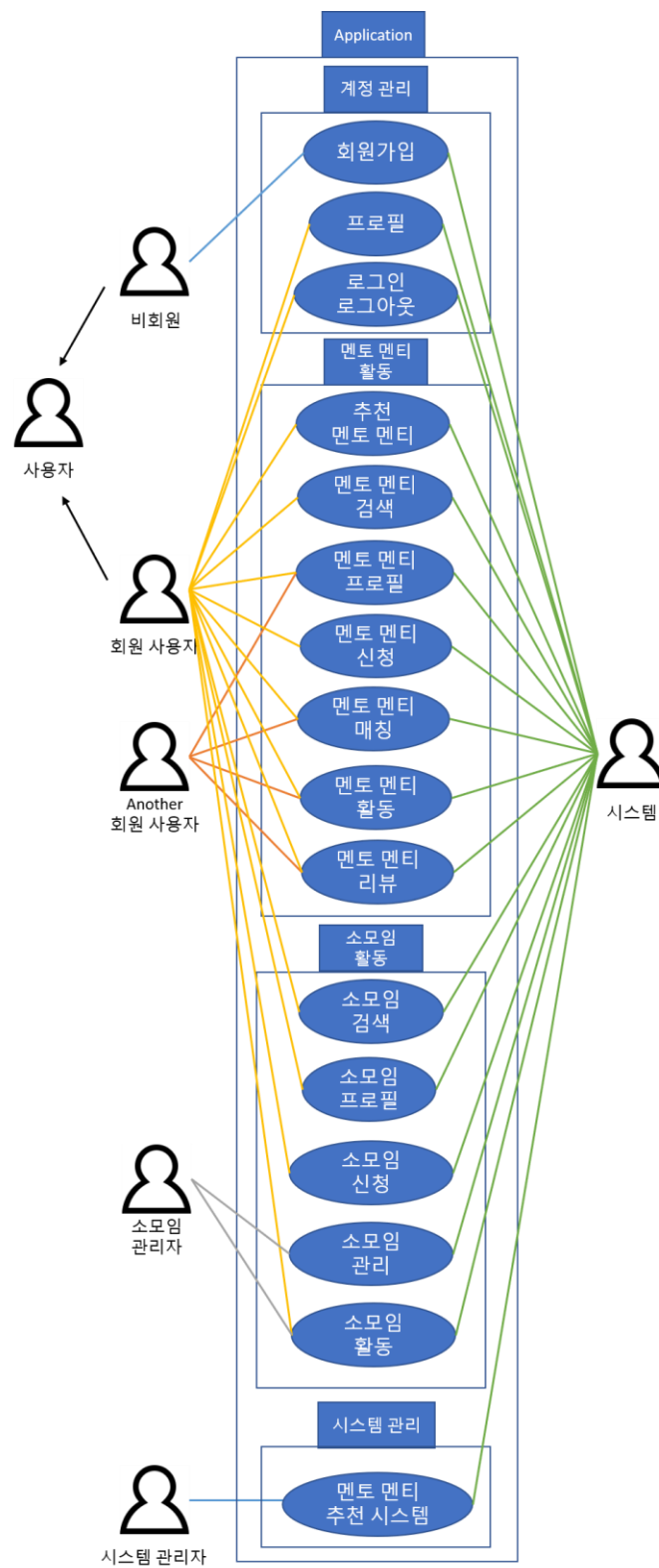
Use case name	소모임 관리
Actor	소모임 관리자
Description	소모임 가입 신청한 사용자의 신청을 수락할지 거절할지 결정할 수 있으며, 소모임 회원 관리를 비롯한 다양한 세부사항을 관리할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 소모임 관리자는 '소모임 관리' 페이지로 이동한다.</li> <li>2. '받은 가입 신청' 페이지에 소모임 가입 신청을 한 사용자의 리스트가 표시된다.</li> <li>3. 리스트의 사용자 중 한 사용자를 선택하면 해당 사용자의 프로필을 확인할 수 있다.</li> <li>4. 해당 사용자가 소모임에 적합하다 여겨진다면 소모임 관리자는 '수락' 버튼을 클릭한다. 반면 마음에 들지 않는다면 '거절' 버튼을 클릭한다. <ol style="list-style-type: none"> <li>4-1. '수락'을 클릭한 경우 매칭이 성사된다.</li> <li>4-2. '거절'을 클릭한 경우 매칭이 결렬된다.</li> </ol> </li> <li>5. '소모임 관리' 페이지에서 '세부사항 수정' 버튼을 클릭하면 게시된 소모임 홍보 글의 세부사항을 수정할 수 있다.</li> <li>6. '소모임 관리' 페이지에서 '회원 관리' 버튼을 클릭하면 현재 소모임 멤버들의 프로필을 확인하고 강퇴, 소모임 관리자 계승 등을 설정할 수 있다.</li> </ol>
Precondition	<p>소모임 관리자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>소모임 관리자는 '소모임 관리' 페이지에 접속해 있어야 한다.</p>

Post Condition	소모임 멤버나 세부사항이 변경되는 즉시 모든 데이터는 데이터 베이스에 저장되어야 한다.
Assumptions	소모임 관리자는 소모임 활동을 진행하려는 계획이 있다.

[Table 27] 소모임 활동 use case

Use case name	소모임 활동
Actor	소모임에 가입된 사용자와 소모임 관리자
Description	소모임에 가입된 사용자와 소모임 관리자는 채팅을 통해 소모임 활동을 진행 할 수 있다.
Normal Course	<ol style="list-style-type: none"> <li>1. 소모임에 가입된 사용자는 소모임 활동 탭으로 이동한다. 소모임 관리자 또한 소모임 탭으로 이동한다.</li> <li>2. 소모임 활동 탭에서 사용자들은 2 인 이상 채팅을 나눌 수 있다.</li> <li>3. 하단의 입력 창을 누르면 키보드 자판이 띄워지고, 보내고자 하는 메시지를 입력한 후 우측의 '전송' 버튼을 누르면 메시지가 전송된다.</li> </ol>
Precondition	<p>사용자는 로그인 중이어야 하고 네트워크에 연결되어 있어야 한다.</p> <p>사용자는 소모임에 가입된 상태여야 한다.</p> <p>사용자는 소모임 활동 페이지에 접속해 있어야 한다.</p>
Post Condition	채팅 로그는 한달 주기로 데이터 베이스에 임시저장 된다.
Assumptions	N/A

### 3.2.2. Use Case Diagram



[Figure 1] Use case diagram

### 3.2.3. Data Dictionary

[Table 28] 사용자

Field	Key	Constraint	Description
유저 ID	PK	Not Null	사용자 고유 식별 id
웹메일		Not Null	사용자 학교 email
비밀번호		Not Null	사용자 비밀번호
닉네임		Not Null	사용자 고유 닉네임
멘토 멘티 여부			멘토 혹은 멘티 선택 여부
성별			

[Table 29] 검색 로그

Field	Key	Constraint	Description
검색 쿼리 ID	PK	Not Null	검색 쿼리 고유 식별 id
유저 ID		Not Null	사용자 고유 id
날짜		Not Null	검색 실행 날짜

[Table 30] 매칭 정보

Field	Key	Constraint	Description
팀 번호	PK	Not Null	팀별 고유 식별 번호
유저 ID		Not Null	사용자 고유 id
멘토 멘티 여부			멘토 혹은 멘티 선택 여부
관심 분야			팀이 가지는 관심 분야

[Table 31] 멘티 사용자

Field	Key	Constraint	Description
유저 ID	PK	Not Null	사용자 고유 식별 id
관심 분야			사용자가 보이는 관심 분야

[Table 32] 멘토 사용자

Field	Key	Constraint	Description
유저 ID	PK	Not Null	사용자 고유 식별 id
관심 분야			사용자가 보이는 관심 분야

[Table 33] 채팅 로그

Field	Key	Constraint	Description
팀 번호	PK	Not Null	팀별 고유 식별 번호
발화자 유저 ID	PK	Not Null	발화 사용자의 고유 식별 ID
발화 내용		Not Null	메시지 내용
날짜		Not Null	메시지 전송 날짜

[Table 34] 리뷰 데이터

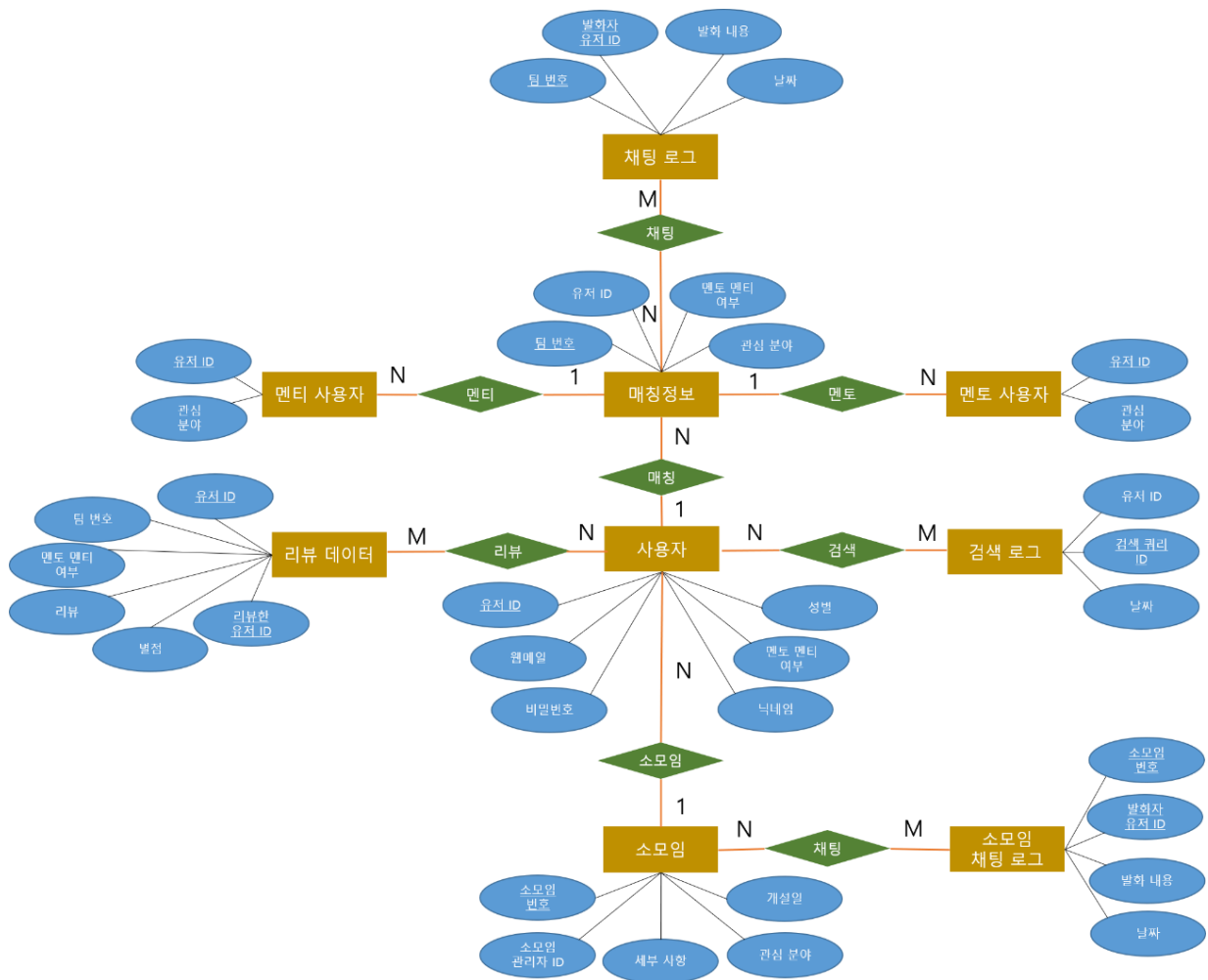
Field	Key	Constraint	Description
유저 ID	PK	Not Null	사용자의 고유 식별 ID
팀 번호		Not Null	팀별 고유 식별 번호
멘토 멘티 여부			멘토 혹은 멘티 선택 여부
리뷰		Not Null	리뷰 내용
별점		Not Null	별점 (1~5)
리뷰한 유저 ID	PK	Not Null	리뷰한 사용자의 고유 식별 ID

[Table 35] 소모임 채팅 로그

Field	Key	Constraint	Description
소모임 번호	PK	Not Null	소모임별 고유 식별 번호
발화자 유저 ID	PK	Not Null	발화 사용자의 고유 식별 ID
발화 내용		Not Null	메시지 내용
날짜		Not Null	메시지 전송 날짜

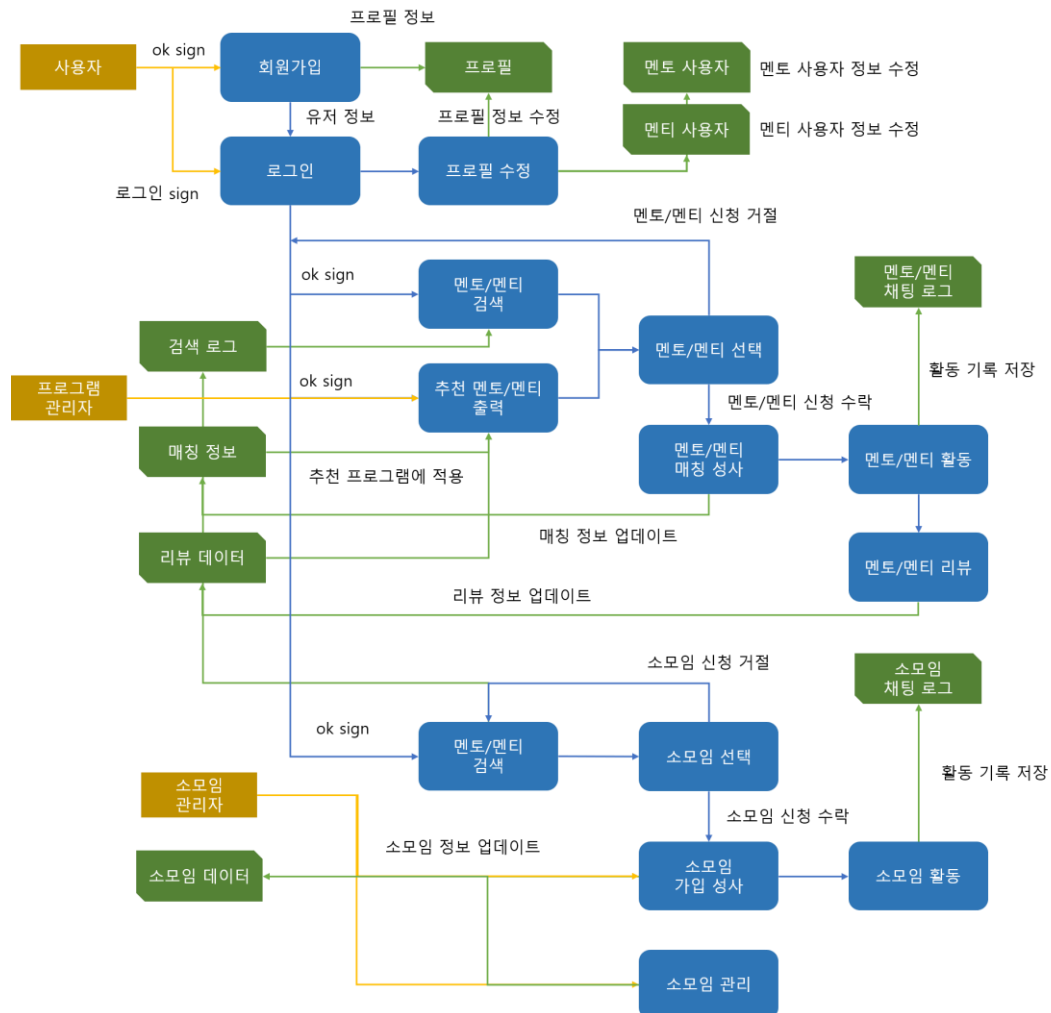
[Table 36] 소모임 정보

Field	Key	Constraint	Description
소모임 번호	PK	Not Null	팀별 고유 식별 번호
소모임 관리자 ID		Not Null	사용자 고유 id
개설일		Not Null	소모임 개설일
관심 분야			팀이 가지는 관심 분야
세부 사항			소모임 세부사항



[Figure 2] Entity Relationship Diagram

### 3.2.4. Data Flow Diagram



[Figure 3] Data flow diagram

## 3.3. Performance Requirements

이 모든 사항은 추정이므로 추후에 변경될 수 있습니다.

### 3.3.1. Static numerical requirement

- 모든 시스템 관리 도구는 관리자들만 사용할 수 있습니다.



- 이 시스템은 한 기기 당 한 사용자만 이용 가능합니다. 하지만 로그아웃한 경우에는 다른 사용자도 이용할 수 있습니다.
- 이 시스템은 디바이스 하드웨어가 최소 1GB RAM, 1.0GHZ, Dual Processor 여야 하며, 소프트웨어는 Android 6.0 이상이어야 합니다.

### 3.3.2. Dynamic numerical requirement

- 이 시스템에 접속할 수 있는 최대 동시 접속자수는 500 명이며, 활동 사용자수는 15,000 명입니다.
- 이 어플리케이션은 5 초 이내로 실행 되어야 합니다.
- 로그인 시 소요되는 시간은 최대 5 초여야 합니다.
- 검색한 게시글은 최대 5 초 안에 게시되어야 합니다.
- 모든 기능들은 5 초~10 초 이내에 실행되어야 합니다.

## 3.4. Logical Database Requirements

이 시스템은 Firebase system 를 이용하므로 모든 사용자들의 프로필 정보를 비롯한 데이터들은 Firestore 에 저장됩니다. 또한 모든 게시글 정보 또한 데이터베이스로 저장되어야 하므로 데이터 베이스 시스템의 성능이 문제없이 작동해야만 오류를 최소화 할 수 있을 것입니다.

## 3.5. Design Constraints

이 시스템은 모든 학생들이 이용할 수 있습니다. 이 시스템은 모든 안드로이드 버전 6.0 이상의 운영체제에서 이용 가능합니다. 관리자들은 브라우저로 접근할 수 있고 부적절한 게시글을 삭제할 수 있습니다. 이 프로그램은 Firebase 로 설계되어 있으며 Firestore 데이터베이스를 이용하고 있습니다.

### 3.6. Standards compliance

시스템 내의 모든 프로그램은 Python 표준을 따라 작성하며, 그 외의 사항은 관용적인 프로그래밍 기법을 따릅니다. 프로그램 내의 함수 명과 변수 명은 camel 표기법을 따르며, 그 외의 사항은 ISO, IEEE 의 표준안을 따릅니다.

### 3.7. Software System Characteristics

소프트웨어 시스템 특성은 비기능적 요구사항(Non-functional Requirements)를 통해 드러납니다. 따라서 이 섹션에서는 시스템의 비기능적 요구사항에 대해 설명합니다. 비 기능적 요구사항은 다음과 같이 제품 요구사항(Product Requirements), 조직 요구사항(Organization Requirements), 외부 요구사항(External Requirements)으로 분류됩니다.

#### 3.7.1. 제품 요구사항 (Product Requirements)

제품 요구사항은 소프트웨어의 런타임 동작을 지정하거나 제한합니다. 본 프로그램은 다음 요구사항을 충족해야 합니다.

##### 3.7.1.1. 사용성 요구사항 (Usability Requirements)

본 프로그램의 가장 중요한 비기능적 요구사항 중 하나입니다. 프로그램은 비전문가인 사용자들도 쉽게 사용할 수 있어야 하고, 사용자가 발생할 오류를 최소화 할 수 있도록 구성되어야 합니다. 전문 용어의 사용을 최소화하고 필요한 경우 쉽게 설명해야 합니다. 각 사용자는 매뉴얼을 학습하지 않아도 모든 시스템 기능을 수월하게 사용할 수 있어야 합니다. 즉, 사용자 인터페이스는 단순하고 직관적으로 다루기 쉬워야 합니다.

##### 3.7.1.2. 성능 요건 (Performance Requirements)

멘토 / 멘티 추천 알고리즘은 시스템에서 가장 시간이 많이 걸리는 작업입니다. 시간이 오래 걸리는 만큼 사용성이 저하되므로 5 초 이내에 결과를 사용자에게 제공해야 합니다. 또한 채팅 프로그램은 원활한 소통을 위해 즉각적으로 전송된 메시지를 사용자에게 표시해야 합니다.

### 3.7.1.3. 사용성 요구사항 (Usability Requirements)

시스템은 주어진 시간에 목적에 맞는 서비스를 제공해야 합니다. 멘토 / 멘티 추천 시스템은 오류를 최소화하면서 서비스를 제공해야 하며, 추천 결과는 사용자의 예상과 부합해야 하며 신뢰감을 제공할 수 있어야 합니다. 시스템의 평균 오류 비율은 5%를 초과하지 않아야 합니다.

### 3.7.1.4. 보안 요구사항 (Security Requirements)

사용자는 프로그램을 사용하기 전에 적절한 인증을 받아야 합니다. 회원으로 가입되지 않은 사용자는 프로그램에 접근 할 수 없으며 사용 또한 할 수 없도록 해야 합니다. 또한 회원으로 가입되지 않은 사용자는 다른 프로그램 사용자의 개인정보와 ID, 비밀번호와 같은 기밀 정보에 접근할 수 없습니다.

## 3.7.2. 조직 요구사항 (Organizational Requirements)

후술한 요구사항은 고객 및 개발자 기구의 정책 및 절차에서 파생된 광범위한 시스템 요구사항입니다.

### 3.7.2.1. 운영 요구사항 (Operational Requirement)

시스템 사용자는 학교 웹 메일을 사용해 개인을 식별해야 합니다. 이 프로그램은 개인에게 적합한 멘토 혹은 멘티를 추천하고 검색할 수 있어야 합니다. 메신저의 역할로서 기능하며 멘토와 멘티가 소통할 수 있는 플랫폼을 제공합니다. 이 시스템은 스마트 폰에서 어플리케이션으로 작동하며 언제든지 작동 가능합니다.

## 3.7.3. 외부 요구사항 (External Requirements)

시스템 및 개발 과정 중 외부 요인에서 파생된 모든 요구사항을 다룹니다.

### 3.7.3.1. 안전/보안 요구사항 (Safety / Security Requirement)

프로그램은 외부 시스템이 개인 정보에 접근할 수 없도록 보장해야 합니다. 프로그램은 외부 자연 재해로 인해 사용자의 데이터베이스가 손상되지 않도록 충분히 안전해야 합니다.

## 3.8. Organizing the Specific Requirements

### 3.8.1. Context Model

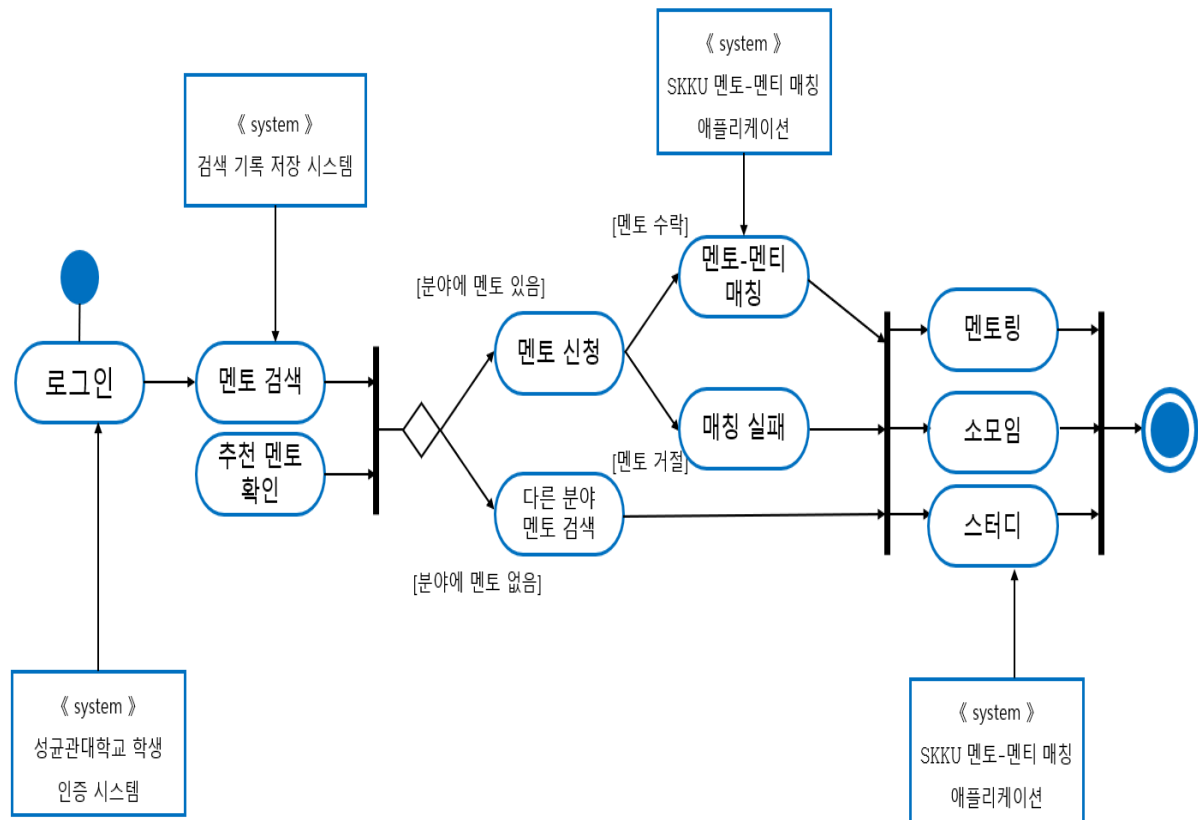
SKKU 멘토-멘티 매칭 어플리케이션의 Context Model 은 다음과 같습니다.



[Figure 4] Context model

### 3.8.2. Process Model

어플리케이션의 비즈니스 프로세스를 표현한 Activity Diagram 은 다음과 같습니다.



[Figure 5] Overall process model

### 3.8.3. Interaction Model

#### 3.8.3.1. Use Case Diagram

3.2.2 의 Use Case Diagram 을 참조하십시오.

다이어그램의 사용자 (멘티 학생과 멘토 학생)은 성균관대학교 신입생, 재학생, 졸업생, 원전공생, 복수전공생 등 모두 해당합니다.

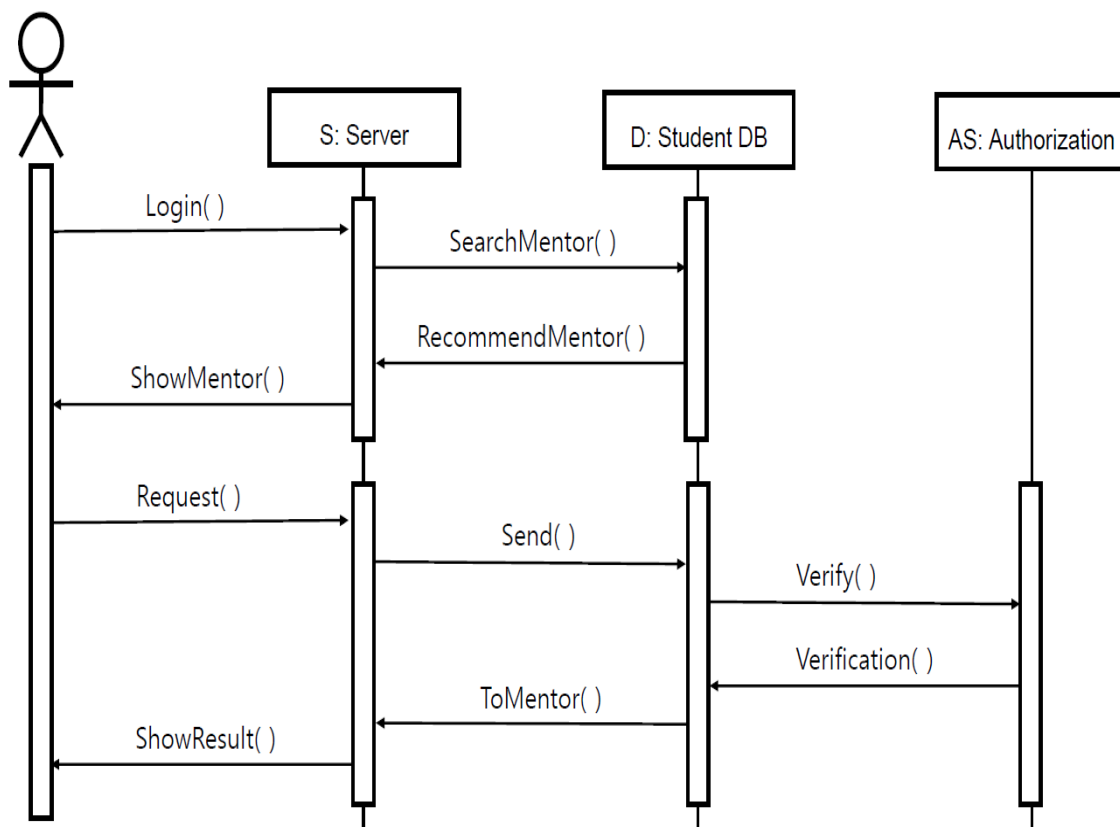
### 3.8.4. Behavior Model

#### 3.8.4.1. Data Flow Diagram

3.2.4 의 Data Flow Diagram 를 참조하십시오.

#### 3.8.4.2. Sequence Diagram

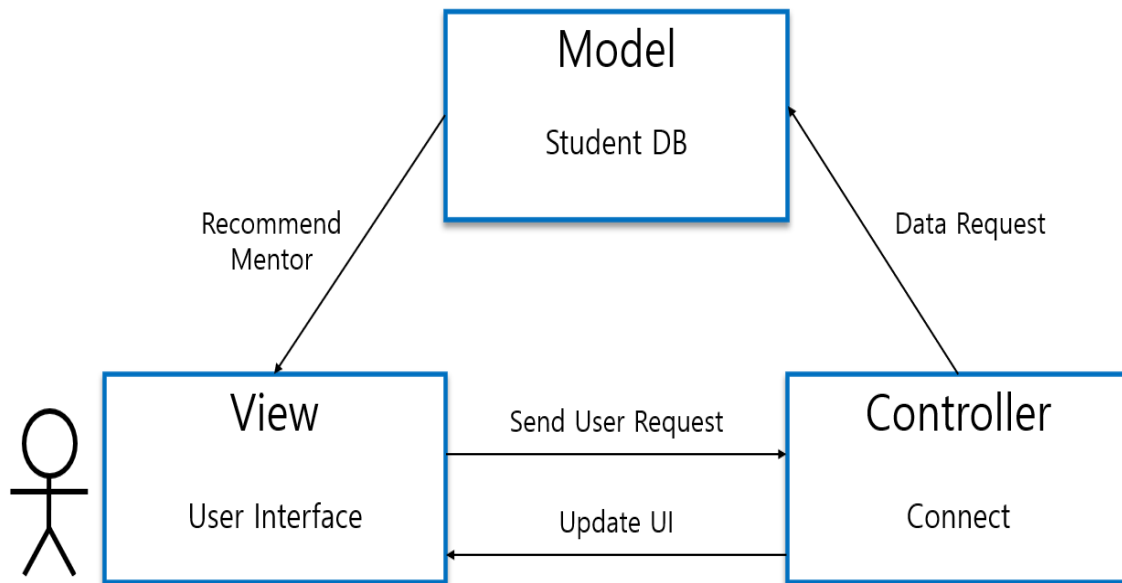
SKKU 멘토-멘티 매칭 애플리케이션의 대표 기능인 멘토-멘티 매칭 상호작용 순서를 Sequence Diagram 으로 나타냈습니다.



[Figure 6] Sequence diagram

### 3.9. System Architecture

전체적인 System Architecture 구조를 표현하면 다음과 같습니다. 소프트웨어 디자인 패턴 기법 중하나인 MVC Pattern 을 활용하여 표현하였습니다.



[Figure 7] System architecture of the system

### 3.10. System Evolution

#### 3.10.1. Limitation and Assumption

첫째로 기술 측면에서는 애플리케이션의 개발 환경은 안드로이드 개발 환경으로 가정합니다. 그렇기 때문에 애플리케이션 사용자는 안드로이드 운영 체제가 탑재된 스마트폰 (안드로이드 6.0 버전 이상)에서만 서비스를 제공 받을 수 있습니다.

둘째로 운영 측면에서는 멘티 대비 멘토가 부족한 경우 멘토-멘티 매칭 자체가 어려울 수 있습니다. 애플리케이션 내 우수 멘토 학생에게 소정의 상품을 지급하는 등 멘토 학생 수를 높이기 위한 활성화 방안이 함께 필요합니다.

셋째로 사용자 측면에서는 애플리케이션의 사용자는 성균관대학교 학생 인증이 성공적으로 완료된 학생으로 가정합니다. 성균관대학교 졸업생 역시 멘토로서 재학생들에게 취업준비, 진로, 전공에 대한 조언과 도움을 줄 수 있고, 또는 멘티로서 취업, 대학원 등 정보를 듣고 도움을 받을

수 있습니다. 따라서 애플리케이션 사용자는 재학 여부에 관계없이 모든 성균관대학교 학생으로 가정합니다.

### 3.10.2. Evolutions of Hardware and Change of User Requirements

애플리케이션의 개발 환경을 안드로이드 개발 환경으로 설정하였기 때문에 시스템 사용을 위해 사용기기는 안드로이드 6.0 버전 이상이어야 합니다. 이후 하드웨어의 기술적 성능 향상, 변화에 따라 애플리케이션 역시 추가적인 유지보수 등이 필요할 것으로 예상합니다.

또한 애플리케이션은 코로나와 언택트 시대 흐름을 배경으로 성균관대학교 학생 간 언택트 교류 활성화를 위해 제안 되었습니다. 이후 코로나가 종식되면, 학생들이 다시 대면수업을 하고 오프라인 모임과 행사가 늘어나는 등 애플리케이션의 외부 환경이 변화할 것입니다.

외부 환경이 변화하면 학생들의 사용자 요구사항도 달라지거나 새로운 사용자 요구사항이 나타날 수 있습니다. 코로나가 종식된 이후에는 사용자 요구사항이 온라인 비대면 교류 대신 오프라인 대면 교류로 바뀔 가능성이 높습니다.

그러나 대면, 비대면과 관계없이 교류가 활발한 학교 커뮤니티 애플리케이션은 코로나 이후에도 여전히 필요할 것으로 예상됩니다. 예를 들어 오프라인 대면 학교 행사, 모임 홍보 및 모집을 위한 장으로 애플리케이션을 활용할 수 있습니다. 또는 애플리케이션을 통해 멘토-멘티 매칭을 받고 멘토링 활동은 대면과 비대면을 혼용하여 진행 가능합니다. 이외에도 외부 환경 변화를 예상하고 이에 맞추어 보완하는 작업을 중장기적으로 수행해야 합니다.

## 4. Supporting Information

### 4.1. Software Requirement Specification

이 요구사항 명세서는 IEEE 표준안 (IEEE Std. 1012-2012, IEEE Standard for Software Verification & Validation) 에 따라 작성되었습니다.



## 4.2. Document History

[Table 37] Document History

Date	Version	Description	Writer
2020/04/17	0.1	양식 작성	김민지
2020/04/19	1.0	1.1, 1.2, .13 작성	김정훈
2020/04/20	1.1	3.1 작성	서기용
2020/04/20	1.2	3.2 작성	김민지
2020/04/22	1.3	3.8, 3.9 작성	김시인
2020/04/23	1.4	2.1, 2.2, 2.3, 2.4, 2.5 작성	김정훈
2020/04/23	1.5	3.3 작성, 3.1 수정	서기용
2020/04/24	1.6	3.2 수정, 3.6, 3.7 작성	김민지
2020/04/24	1.7	3.9 수정, 3.10 작성	김시인
2020/04/25	1.8	4.1, 4.2 작성, 최종본 양식 수합	김민지