

Kingo Learning

Software Requirement Specification



Team 12

팀장 조건희

조원 박성빈

조원 박민지

조원 송재현

조원 이태기

조원 김정안

목차

1. Introduction

1.1. Purpose	4
1.2. Scope	4
1.3. Definitions, Acronyms and Abbreviation	4
1.4. References	5
1.5. Overview	6

2. Overall Description

2.1. Product Perspective	6
2.1.1. System Interfaces	6
2.1.2. User Interfaces	6
2.1.3. Hardware Interfaces	6
2.1.4. Software Interfaces	6
2.1.5. Memory Constraints	7
2.1.6. Operations	7
2.2. Product functions	7
2.2.1. 로그인	7
2.2.2. 메인 페이지 및 강의	8
2.2.3. 실시간 강의	8
2.2.4. 할 일 및 캘린더	8
2.3. User Characteristics	8
2.4. Constraints	9
2.5. Assumptions and Dependencies	9

3. Specific requirement

3.1. External Interface Requirements	10
3.1.1. User Interfaces	10
3.1.2. Hardware Interfaces	13
3.1.3. Software Interfaces	13
3.1.4. Communication Interfaces	13
3.2. Functional Requirements	13
3.2.1. 로그인	13
3.2.2. 메인 페이지	14
3.2.3. 강의	15

3.2.4. 강의 콘텐츠	16
3.2.5. 익명게시판	17
3.2.6. 채팅방	18
3.2.7. 실시간 강의	19
3.2.8. 할 일	20
3.2.9. 캘린더	21
3.2.10. 푸시 알림	22
3.2.11. Use Case Diagram	23
3.3. Nonfunctional Requirements	23
3.3.1. Product Requirements	23
3.3.1.1. Usability Requirements	23
3.3.1.2. Dependability Requirements	24
3.3.1.3. Security Requirements	24
3.3.2. Organizational Requirements	24
3.3.2.1. Environmental Requirements	24
3.3.2.2. Development Requirements	24
3.3.3. External Requirements	25
3.3.3.1. Regulatory Requirements	25
3.3.3.2. Ethical Requirements	25
3.4. Logical Database Requirements	25
3.5. Design Constraints	25
3.6. Standards Compliance	25
3.7. Standards Architecture	26
3.8. System Evolution	26
3.8.1. Change Limitation and Assumption	26
3.8.2. Change of User Requirements	26
3.8.3. To Accommodate the Changes Easily	27

1. Introduction

1.1 Purpose

이 문서는 성균관대학교 온라인 강의 어플리케이션 'Learning X'의 개선을 위한 추가 및 보완사항에 대하여 서술한다. 이 서비스는 성균관대학교 2021학년도 1학기 소프트웨어공학개론 제 12조(조건희, 김정안, 박민지, 박성빈, 송재현, 이태기)에 의하여 작성된다. 요구사항이 요약, 정리되며 내용을 기반으로 하여 시스템 설계 및 개발이 진행될 것이다.

제 12조는 온라인 강의 어플리케이션에서 익명으로 운영되는 문의게시판 또는 채팅방, 할 일과 캘린더 목록에서의 개선 그리고 자체 실시간 강의 기능을 설계하고 기능 구현을 위한 요구사항을 서술한다. 주된 독자는 어플리케이션의 제 12조 구성원과 주된 이용자인 성균관대학교 교수, 학생 및 조교 등이다.

이 문서의 목적은 어플리케이션 기능 중 문의게시판의 이용율 저조 원인을 실명제로 판단하고 익명게시판을 통한 질문 및 의견교류의 활성화를 제안한다. 또한, 현재 Learning X는 지나치게 세분화 되어있는 탭 구성과 사용자가 이용하기 불편하게 되어있는 사용자 인터페이스를 개선하여 단순화작업을 진행한다. 마지막으로 써드파티 어플리케이션(Webex, Zoom 등)을 이용한 강의 참여와 이중 업로드 문제 등 개선을 위해 어플리케이션 내 자체적인 실시간 강의 기능 제공을 위한 개발을 진행한다.

1.2 Scope

기능 개선을 통해 온라인 강의 어플리케이션이 간편한 사용자 인터페이스 제공과 유용한 사용기능을 제공할 수 있다. 먼저, 학생들의 익명 게시판 사용을 위한 데이터베이스 익명화 기능이 추가되어야 한다. 성균관대학교 학생들 중 해당 수업에 참여하는 학생들을 위한 데이터베이스 구축 및 익명화작업이 이루어져야 한다. 다음으로 할 일과 캘린더 기능 중 불필요한 사용자 인터페이스를 선택 통합하여 단순화한다. 마지막으로, 서버 구축을 통해 자체적인 실시간 강의를 제공할 수 있도록 한다.

1.3 Definitions, Acronyms and Abbreviation

아래 표는 이 문서에서 사용된 약어에 대한 설명이다

약어	설명
UI	User Interface

PC	Personal Computer
GLS	Golden Lawn Square
DB	Database
UI	User Interface
HTTP	Hypertext Transfer Protocol

아래 표는 이 문서에서 사용된 용어에 대한 정의이다

용어	정의
사용자	성균관대학교 구성원으로 어플리케이션 시스템을 이용하는 자
강의 관리자	성균관대학교 구성원으로 강의 콘텐츠를 제작하고 운영하는 자
소프트웨어	컴퓨터를 이용하여 운영체제와 프로그램이 제공하는 서비스
GLS	Golden Lawn Square의 약자로 성균관대학교 행정 및 정보시스템

1.4 References

- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEEExplore Digital Library
(<https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Web/Requirements/IEEE-standard.html>)
- skkuse/2021spring_41class_team12 (github.com)

1.5 Overview

소프트웨어 요구사항 명세서는 총 3장과 부록으로 구성으로 되어 있다. 두 번째 장은 제품이 이해관계자의 니즈와 시스템의 기반을 어떻게 충족시키는지 제품 관점 설명을 위해 다수의 인터페이스, 시스템 기능과 시스템간의 상호 연결에 대하여 기술한다. 이 장은 또한 각 이해관계자 별로 시스템과의 관계성에 대하여 기술한다. 구체적으로는 시스템의 제약사항과 개선점의 상세한 내용에 대하여도 설명한다. 세 번째 장은 상세한 용어의 정립, 소프트웨어 시스템의 특징과 각 인터페이스에 대한 묘사를 기술한다. 이 장에서는 요구사항을 구체화하고 시각화하여 다양한 이해관계자가 각자 자신의 입장에서 시스템의 흐름과 요구사항 충족여부를 정확하게 판단할 수 있도록 도울 수 있다.

2. Overall Description

2.1 Product Perspective

이 소프트웨어는 성균관대학교 학생들을 위한 스마트 캠퍼스 어플리케이션입니다. 사용자는 스마트 캠퍼스 어플리케이션을 통해 온라인 강의(저장된 강의, 실시간 강의) 수강 기능, 개인 캘린더 기능, 익명 문의 게시판 기능 등을 사용할 수 있습니다. 이 어플리케이션은 강의를 수강하는 학생들뿐 아니라 각 강의의 관리자들도 또한 다양한 기능을 통해 좀 더 원활한 강의 진행이 가능하도록 설계되었습니다.

2.1.1. System Interfaces

사용자가 성균관대학교 계정으로 로그인하게 되므로 GLS학생인트라넷과 연동하여 사용자의 개인 정보와 수강 강의 목록 등을 받아옵니다. 받아온 사용자의 프로파일은 SQLite를통해 사용자의 로컬 영역에 저장되어 API호출을 통해 데이터베이스를 운영하므로 좀 더 빠른 동작을 기대할 수 있습니다.

2.1.2. User Interfaces

스마트폰의 화면을 통해 인터페이스가 제공되며, 사용자는 스마트폰을 통해 정보를 입력할 수 있습니다. 3장에 서술할 내용처럼 로그인, 캘린더 등 각각의 기능을 이용할 때 기능에 맞는 인터페이스를 제공합니다.

2.1.3. Hardware Interfaces

이 어플리케이션은 Android, iOS용입니다. 스마트폰에는 최소 1GB RAM과 1.0GHz 단일 프로세서가 있어야 합니다.

2.1.4. Software Interfaces

이 어플리케이션은 Android 6.0 (API 23) 이상의 Android OS 버전, Android 10 (API 29), iOS 13.0 이상의 iOS 버전을 대상으로 합니다.

2.1.5. Memory Constraints

어플리케이션은 최소 1GB RAM이 있는 모바일 기기에서 실행되어야 하며, 어플리케이션 설치 및 실행을 위해 최소 200MB의 저장 공간이 필요합니다.

2.1.6. Operations

■ 로그인

사용자는 성균관대학교 계정을 사용하여 로그인할 수 있습니다.

■ 강의 페이지

사용자가 수강 중인 강의 목록을 인터페이스를 통해 제공하며, 각각의 과목들의 정보(강의 수강 여부, 과제 제출 여부 등) 또한 사용자에게 제공합니다.

각각의 과목 버튼을 터치하면 상세한 정보를 볼 수 있으며, 과제 제출, 실시간 강의 또는 저장된 강의 동영상 수강 기능 또한 이 페이지에서 이용할 수 있습니다.

사용자는 다른 어플리케이션을 사용할 필요 없이 본 스마트 캠퍼스 어플리케이션에서 실시간 강의 기능 또한 사용할 수 있습니다.

■ 익명 문의 게시판 및 익명 채팅방

사용자는 익명 문의 게시판 또는 익명 채팅방을 이용함으로써 익명으로 더 원활하게 질의응답할 수 있습니다.

■ 할 일 및 캘린더

사용자는 할 일 및 캘린더 기능을 통해 더 효율적으로 개인의 스케줄을 관리할 수 있습니다.

■ 푸시 알림

사용자가 할 일 및 캘린더 기능에서 입력한 내용을 기반으로 작업 기한이 얼마 남지 않은 작업들의 정보를 푸시 알림을 통해 사용자에게 제공합니다.

2.2. Product Functions

2.2.1. 로그인

사용자는 성균관대학교 계정을 사용하여 스마트 캠퍼스 어플리케이션에 로그인할 수 있습니다. 로그인 시에 사용자는 자동 로그인 기능을 사용할지의 여부를 결정할 수 있으며,

자동 로그인 기능을 사용하기로 결정했다면 최초의 로그인 이후에는 다시 로그인하지 않아도 됩니다.

2.2.2. 메인 페이지 및 강의

사용자가 로그인을 했다면, GLS학생인트라넷과 연동하여 받아온 사용자의 정보를 바탕으로 사용자가 수강 중인 강의 목록을 인터페이스를 통해 나열합니다.

강의 목록은 각각의 강의 정보를 사용자에게 제공하며, 강의 버튼을 클릭하면 그 강의의 강의 콘텐츠를 이용할 수 있습니다. 수강생 사용자가 강의를 수강했다면 자동으로 출석 체크가 됩니다.

강의 관리자는 자신이 관리하는 강의들을 수강하는 수강생들의 정보를 제공받을 수 있고, 강의 콘텐츠를 업로드할 수 있습니다.

수강생 사용자는 각 과목별로 익명 게시판, 익명 채팅방 기능을 사용할 수 있어, 원활한 질의응답이 가능합니다.

2.2.3. 실시간 강의

사용자는 다른 어플리케이션을 사용할 필요없이 본 스마트 캠퍼스 어플리케이션을 통해 실시간 강의 기능을 사용할 수 있습니다.

강의의 관리자는 실시간 강의를 시작할 수 있고, 자신의 화면과 마이크, 카메라 정보를 수강생에게 공유할 수 있습니다. 강의 관리자가 실시간 강의를 켜놓은 상태라면, 수강생들은 강의 페이지에서 실시간 강의에 참여할 수 있습니다.

실시간 강의를 종료되면, 강의에 참여한 수강생 목록과 강의 녹화본이 데이터베이스에 저장됩니다. 이를 통해 자동 출석 체크와 실시간 강의 녹화본 공유가 가능합니다.

2.2.4. 할 일 및 캘린더

사용자는 할 일 및 캘린더 기능을 통해 개인의 작업 목록을 관리할 수 있습니다. 각각의 강의 목록들로부터 사용자가 수강해야 할 강의나 해야 할 과제의 리스트를 할 일 기능에서 제공하고, 이를 날짜별로 사용자가 캘린더 기능에서 관리할 수 있습니다. 사용자가 관리한 내용을 바탕으로 수행 기한이 얼마 남지 않은 작업들의 정보를 푸시 알림으로 사용자에게 보냅니다.

2.3 User Characteristics

이 문서에서 사용자는 수강생과 강의 관리자로 합니다. 수강생은 한국어 또는 영어를 원활하게 사용할 수 있는 능력이 있고, 현재 하나 이상의 강의를 수강하고 있다고 가정합니다. 강의 관리자는 관리 중인 강의를 수강하는 수강생들의 데이터를 처리할 수 있는 능력이 있고,

수강생들의 학습 내용(강의 수강 여부, 제출된 과제 내용)을 바탕으로 수강생들을 평가할 수 있는 능력이 있어야 합니다.

2.4 Constraints

이 어플리케이션은 본 문서에서 서술된 내용을 바탕으로 설계 및 구현됩니다. 또한 개발 시 다음 사항을 준수합니다.

- 널리 입증된 기술을 사용합니다.
- 어플리케이션에 꼭 필요한(유일한) 기술이 아니라면 가급적 무료로 제공되는 소프트웨어를 사용합니다.
- 가능한 범위 내에서 오픈 소스 소프트웨어를 적극적으로 사용합니다.
- 어플리케이션 유지 보수를 고려하여 주석을 적극적으로 활용합니다.
- 향후 어플리케이션의 확장성 및 가용성을 고려하여 소스 코드를 작성합니다.
- 자원 낭비를 최소화할 수 있도록 최적화된 소스 코드를 작성합니다.
- 사용자에게 친숙한 인터페이스를 제공할 수 있도록 개발합니다.
- 최소 Android 6.0 (API 23) 이상의 Android OS 버전, iOS 13.0 이상의 iOS 버전으로 개발합니다.
- Android 11, iOS 14 버전을 사용하여 어플리케이션을 테스트합니다.

2.5 Assumptions and Dependencies

본 문서는 구현할 스마트 캠퍼스 어플리케이션은 최소 Android 6.0 (API 23) 이상의 Android OS 버전, iOS 13.0 이상의 iOS 버전을 사용하는 모바일 기기에서 사용된다는 가정 하에 작성되었습니다. 따라서 다른 운영 체제 또는 버전에는 적용되지 않을 수 있습니다.

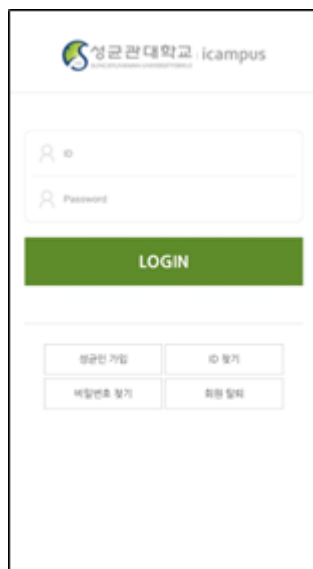
3. Specific Requirements

3.1 External Interface Requirements

이하 내용은 학습 시스템의 인터페이스 requirements을 설명한다. user가 어떻게 시스템과 상호작용하는지, 그리고 데이터를 저장하고 검색하기 위해 software가 필요로 하는 hardware와 communication interface를 정의한다.

3.1.1. User Interfaces

User interface는 직관적인 아이콘과 사용자 친화적인 디자인을 통해 쉽게 이용할 수 있도록 설계되어야 한다. 기능별로 구분하여 각 user interface에 대해 설명한다. 안드로이드와 OS를 지원하며 터치스크린을 통해 사용자 입력을 인식한다.



[Figure 1] 로그인 화면의 user interface

1. 웹앱을 통해 성균관대학교 icampus 로그인 페이지를 불러온다.
2. 이 앱의 사용자는 성균관대학교에 재학중인 학생으로 설정하며 이미 계정을 가지고 있을 것이라고 가정한다.
3. 사용자는 ID와 Password를 입력한다. 로그인에 성공하면 [Figure 2]의 화면으로 이동하고 로그인에 실패했을 시, 재입력 팝업창을 띄운다.
4. 최초 로그인에 성공하면 사용자가 로그아웃을 하기 전까지 로그인 세션을 유지하며 앱을 실행했을 때 [Figure 2]를 기본으로 보여준다.



[Figure 2] 기본 홈 화면의 user interface

1. 상단 좌측에는 학기를 선택하여 각 학기에 수강한 강의를 볼 수 있는 목록 버튼이 있다. 상단 우측에는 시간표 아이콘이 있어 버튼을 클릭하면 시간표 화면으로 이동한다.
2. 메인 레이아웃에는 과목들을 클릭할 수 있는 네모 박스의 리스트가 나타난다. 사용자가 원하는 과목의 박스를 클릭하면 그 과목에 해당하는 [Figure 3]의 화면으로 이동한다.
3. 각 네모 박스는 과목명과 수강해야 하는 강의 수, 제출해야 하는 과제 수를 보여준다.



4. 레이아웃 하단에는 다섯 개의 탭 메뉴가 있다. 차례대로 클릭하면 홈(현재 화면)/캘린더/할 일/익명채팅방/메시지 화면으로 이동한다. 할 일, 익명채팅방, 메시지 아이콘의 경우 알림 배지가 있어 몇 개의 새로운 알림이 있는지 바로 확인할 수 있다. 이 탭 메뉴는 모든 화면 UI에서 보인다.



[Figure 3] 강의 페이지의 user interface

1. 상단에 과목명을 표시한다. 하단에 차례로 실시간 강의룸 페이지[Figure 8]로 이동하는 버튼과 여러 항목들이 리스트 형태로 보여진다.
2. 실시간 강의를 진행 중일 때에는 강의룸 버튼에 체크 표시가 나타난다.



[Figure 4] 익명 문의 게시판의 user interface

1. 제목으로 게시글을 검색할 수 있는 text 입력 창과 글쓰기 버튼, 게시판의 질문들을 리스트 형태로 보여준다.
2. 사용자는 글쓰기 버튼을 눌러 익명으로 질문을 남기거나 원하는 질문에 답변을 남길 수 있다.
3. 각 리스트는 게시글의 제목과 조회수, 답변 수를 보여준다.



[Figure 5-1, 5-2] 익명 채팅방의 user interface

1. 하단의 탭 메뉴 중 네번째 아이콘 버튼을 클릭했을 때 볼 수 있는 화면이다.
2. 참여할 수 있는 채팅방의 목록을 리스트 형태로 보여준다. 채팅방 이름은 과목명으로 설정되며 리스트 항목 우측에 메시지 알림 배지를 보여준다.
3. 리스트 항목을 클릭하면 해당 채팅방 화면으로 이동한다.

4. 사용자는 해당 과목의 수강생 전원과 익명으로 자유롭게 채팅을 주고 받을 수 있다.



[Figure 6-1, 6-2] 할 일의 user interface

1. 강의와 과제, 두 개의 탭과 할 일의 내용을 리스트 형태로 보여준다.
2. 완료된 항목은 회색 글씨로 나타나며 제출 기한이 지나면 리스트에서 삭제된다.
3. 강의자가 새로운 강의나 과제를 업로드 하면 수강생은 하단의 탭 메뉴에서 배지 알림으로 알 수 있다.



[Figure 7] 캘린더의 user interface

1. 달력 레이아웃 하단에 그 달에 해야 하는 일들을 리스트 형태로 나타낸다.
2. 사용자는 추가(+) 버튼을 눌러 원하는 날짜에 원하는 일정을 등록할 수 있다. 또한 할 일 리스트 항목을 클릭하여 내용을 수정하거나 삭제할 수 있다.



[Figure 8] 실시간 강의 화면의 user interface

1. 가로 화면을 고정으로 한다.
2. 좌측에 강의자 화면을 표시하며 우측 상단에 수강생 카메라 화면을 표시한다.
3. 수강생의 출석 현황과 참여하고 있는 수강생의 목록을 리스트 형태로 보여준다.

4. 사용자는 채팅창에 입력하거나 다른 수강생의 채팅을 확인할 수 있다.
5. 사용자는 마이크나 카메라를 끄거나 켤 수 있다.

3.1.2. Hardware Interfaces

6.0 이상 버전의 Android와 13.0 이상 버전의 iOS를 지원한다. 이 애플리케이션을 설치하기 위해서는 최소 200MB의 여유 저장공간이 있어야 한다. 또한 앱의 기능을 사용하기 위해서는 추가적으로 마이크, 주소록, 카메라 접근 권한을 허용해야 한다.

3.1.3. Software Interfaces

이 애플리케이션을 사용하기 위해서는 성균관대학교 성균인(학생/교수) 계정이 있어야 하며 학교 자체에 서버가 구축되어 있어야 한다.

3.1.4. Communication Interfaces

수강생 사용자는 서버에 로그인하여 학습을 하거나 자신의 학습 현황을 확인한다. 강의를 사용하는 사용자는 서버에 로그인하여 학습 자료를 업로드하거나 전체 수강생의 학습 현황을 확인한다. 서버에서 필요한 정보를 받아오기 위해서는 인터넷과 연결되어 있어야 한다. 로그인 세션을 계속 유지하기 위해서는 사용자의 로그인 세션을 유지하고 있어야 한다.

3.2. Functional Requirements

3.2.1. 로그인

Name	로그인
Description	ID와 비밀번호를 입력한다.
Actor	학교를 선택한 사용자, 로그인 관리 시스템
Action	<div>1. 사용자가 이전에 로그인 한 기록이 있다</div> <div>1 - A. 로그인 관리 시스템은 사용자에게 따로 ID 입력을 요구하지 않고, 이전의 기록을 바탕으로 자동으로 로그인이 되게 한다.</div> <div>2. 사용자는 이전에 로그인 한 기록이 없다.</div> <div>2 - A. 로그인 관리 시스템은 사용자에게 ID와 비밀번호를 요구한다.</div>

	2 - B. 사용자가 ID와 비밀번호를 입력하면, 시스템은 ID 데이터베이스에서 이 ID와 비밀번호를 검증하고, 데이터베이스의 정보와 일치하면 로그인을 승인한다.
Pre-Condition	사용자는 학교를 선택했지만 로그인은 하지 않은 상태이다.
Post-Condition	로그인에 성공하면 메인 페이지로 화면이 넘어간다.

3.2.2. 메인 페이지

Name	메인 페이지
Description	강의와 핵심 기능의 목록을 보여준다.
Actor	학교에 로그인을 완료한 사용자, 강의 시스템
Action	<p>1. 강의 시스템은, 강의들의 목록을 메인 페이지 상단에 나열해 놓는다.</p> <p>2. 강의의 이름은 강의 위에 있는 점 세 개 버튼을 통해 수정할 수 있는데, 사용자가 이름을 수정하면 강의 시스템은 커스텀 이름을 저장해, 실행 시 마다 이 커스텀 이름으로 강의의 이름을 대체한다.</p> <p>3. 메인 페이지 하단에는, 각각 할 일, 스케줄, 채팅창으로 이동할 수 이동할 수 있는 버튼들이 존재하는 네비게이션 바가 있다.</p>
Pre-Condition	사용자는 로그인을 한 상태이다.

Post-Condition	-
----------------	---

3.2.3. 강의

Name	강의
Description	해당 강의와 관련된 게시판의 리스트를 보여준다.
Actor	강의목록에서 강의를 터치한 사용자, 강의 관리자, 강의 시스템
Action	<p>1. 강의 시스템은 해당 강의와 관련된 게시판의 목록을 불러와 나열하고, 실시간 강의에 접속할 수 있는 버튼을 보여준다.</p> <p>2. 게시판의 목록은 강의의 관리자가 설정할 수 있으며, 주로 수업 계획서, 공지사항, 강의 콘텐츠, 익명 게시판이 존재한다.</p> <p>3. 익명 게시판과 강의 콘텐츠를 제외한 게시판에서는, 작성자의 실명과, 작성 날짜, 조회수가 존재하며, 실명으로 댓글을 남길 수 있다.</p>
Pre-Condition	-
Post-Condition	-

3.2.4. 강의 콘텐츠

Name	강의 콘텐츠
Description	강의 영상의 리스트를 보여준다.
Actor	강의 영상을 보고자 하는 사용자, 강의 관리자, 강의 시스템
Action	<ol style="list-style-type: none">1. 게시판 리스트에서, 강의 콘텐츠를 누르면, 강의 시스템이 강의 데이터베이스에서 영상 과제 자료 등을 읽어온다.2. 사용자에게 강의의 관리자가 올린 영상, 과제, 자료의 목록이 주간 별로 나열된다.3. 영상의 제목을 누르면, 강의 시스템은 운영체제에 내장된 브라우저를 불러와 영상을 재생한다.4. 영상을 끝까지 재생하면, 강의 시스템은 사용자가 강의에 출석한 것으로 처리한다.
Pre-Condition	-
Post-Condition	-

3.2.5. 익명게시판

Name	익명 게시판
------	--------

Description	자유롭게 글을 쓸 수 있는 게시판이나 작성자의 이름이 가려진다.
Actor	익명으로 의견을 나누고 싶은 사용자, 게시판 시스템
Action	<ol style="list-style-type: none"> 1. 게시판 시스템은 익명 게시판 데이터베이스에서 익명 게시판의 글 목록을 읽어온 후, 작성자 부분을 전부 '익명'이라 표시한다. 2. 사용자에게 익명 게시판의 글들이 익명으로 나열된다. 3. 글 작성 버튼을 누르면 사용자는 익명 게시판에 글을 업로드 할 수 있다. 4. 댓글 또한 게시판 시스템이 댓글 작성자를 전부 '익명'이라 표시해 누가 댓글을 작성했는지 알지 못하게 한다.
Pre-Condition	-
Post-Condition	-

3.2.6. 채팅방

Name	채팅방
Description	개설된 채팅방을 한 눈에 보이게 해서 관리를 돕는다.
Actor	다른 학우들과 채팅을 하고 싶은 사용자, 채팅방 시스템

Action	<p>1. 채팅방 시스템은, 채팅 목록에 사용자가 현재 수강하고 있는 강의의 이름을 나열해 놓는다.</p> <p>2. 사용자가 이 나열된 채팅방 중 하나를 터치하면, 이 수업을 수강하는 모든 인원이 참석하고 있는 익명 채팅방에 접속할 수 있다.</p> <p>3. 상단의 푸시 알림을 켜고 끄는 버튼을 통하여, 사용자는 이 채팅방에 새로운 채팅이 존재 할 시 알림을 받을지 받지 않을지 설정할 수 있다.</p>
Pre-Condition	-
Post-Condition	-

3.2.7. 실시간 강의

Name	실시간 강의
Description	다른 어플리케이션을 따로 구동하지 않고 실시간 강의를 수강할 수 있다.
Actor	실시간 강의를 들으려는 사용자, 실시간 강의 시스템, 채팅 시스템, 강의 시스템

Action	<p>1. 실시간 강의 버튼을 누르면 실시간 강의 시스템은 해당 강의의 관리자가 켜놓은 영상 채널과 사용자를 연결시켜준다.</p> <p>2. 카메라 버튼을 눌러 사용자가 웹캠을 가동하면, 실시간 강의 시스템은 사용자의 웹캠의 영상을 다른 사람들에게 실시간으로 송출한다.</p> <p>3. 사용자가 발언을 하면, 실시간 강의 시스템은 사용자 컴퓨터의 마이크를 통해 사용자의 음성을 녹음하고, 이 음성을 다른 사람에게 실시간으로 재생한다.</p> <p>4. 채팅 시스템은 채팅 서버에서 글들의 목록을 읽어와 실시간 채팅을 구현하고, 사용자가 채팅을 입력하면 서버에 글을 추가해 다른 사람에도 보이게 한다.</p> <p>5. 강의의 관리자가 실시간 강의를 종료하면, 실시간 강의 시스템은, 이 강의를 듣고 있었던 사용자의 목록과, 강의의 녹화본을 강의 시스템에게 넘겨준다.</p> <p>6. 강의 시스템은 사용자 목록을 바탕으로, 실시간 강의를 수강하였던 학생들을 자동으로 출석으로 처리를 하고, 녹화본을 강의 데이터베이스에 올려 사용자가 강의 목록을 통해 녹화본을 볼 수 있도록 한다.</p>
Pre-Condition	<p>강의의 관리자가 강의를 켜놓고 있는 상태이어야함.</p> <p>자신을 찍기 위해선 웹캠이 존재 해야하고, 발언을 하기 위해선 마이크가 존재 해야함.</p>
Post-Condition	-

3.2.8. 할 일

Name	할 일
Description	사용자가 아직 수강하지 않은 과제 등을 표시해준다.
Actor	다음에 뭘 해야할지 궁금한 사용자, 작업 목록 관리 시스템
Action	<p>1. 어플리케이션을 구동할 때, 작업 목록 관리 시스템은 강의의 목록들을 읽어서 해야하는 과제와 봐야하는 영상에 대한 리스트를 작성함.</p> <p>2. 이 리스트 중 완료된 작업은 회색으로 표시를 하고, 완료되지 않은 작업은 색을 입혀, 영상은 강의 리스트에, 과제는 과제 리스트에 올려 사용자들에게 보여줌.</p> <p>3. 리스트의 항목을 클릭하면, 시스템은 해당 항목과 가장 연관이 있는 강의의 게시판을 띄워 줌.</p> <p>4. 사용자가 과제를 제출하거나 영상에 출석을 하는 등 작업의 상태에 변화를 주는 일을 하면, 작업 목록 관리 시스템은 자체적으로 (1)의 강의의 목록을 재구성해 바로 상태의 변화를 반영해 회색으로 바꿔 사용자에게 보여줌.</p>
Pre-Condition	-
Post-Condition	-

3.2.9. 캘린더

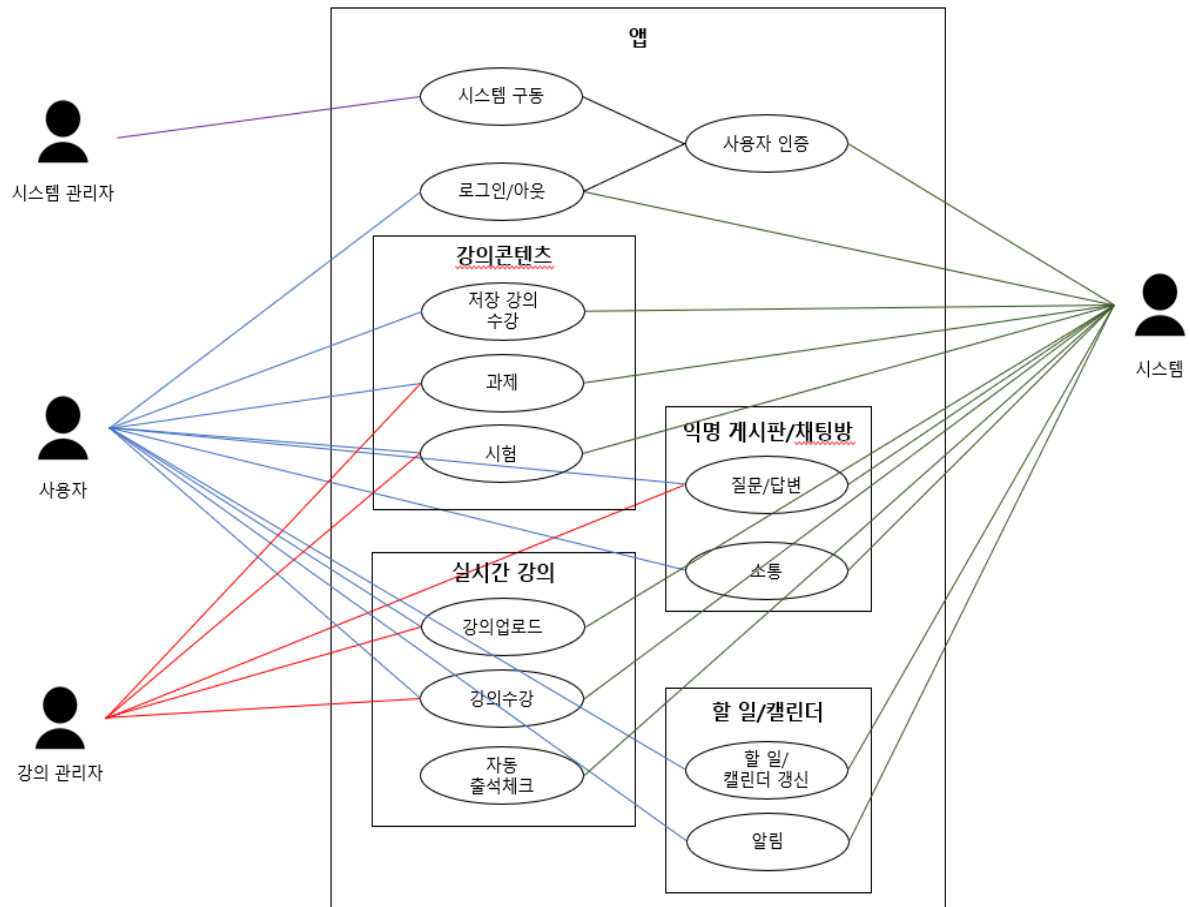
Name	캘린더
Description	날짜 별로 해야하는 강의와 과제에 대해 표시해준다.
Actor	달력을 보며 계획을 짜고 싶은 사용자, 캘린더 시스템, 작업 목록 관리 시스템
Action	<p>1. 사용자가 캘린더 버튼을 누르면, 캘린더 시스템은 작업 관리 시스템에게 8-1에서 구성한 강의의 목록을 받아와, 이 목록과 캘린더의 자체적인 일정 목록을 날짜 별로 분류해 달력에 표시해 사용자에게 보여준다.</p> <p>2. 사용자가 달력의 특정 날을 터치하면, 해당 날까지 해야 하는 작업을 확인할 수 있다.</p> <p>3. 사용자가 일정 추가 버튼을 누르면, 사용자는 자신만의 일정을 추가 할 수 있는데, 캘린더 시스템은 이 일정을 (1)의 캘린더의 자체적인 일정 목록에 저장해, 매 실행 때 마다 보이게 한다.</p>
Pre-Condition	-
Post-Condition	-

3.2.10. 푸시 알림

Name	푸시 알림
------	-------

Description	푸시 알림 기능을 통해 사용자가 작업을 잊지 않도록 돕는다
Actor	푸시 알림의 도움을 받고 싶은 사용자, 푸시 알림 시스템
Action	<p>1. 사용자가 푸시 알림의 기능을 켜 상태라면, 푸시 알림 시스템은 9-1에서 구현한 날짜 별 일정을 기반으로, 매 자정, 24시간 남은 작업들의 목록을 사용자에게 푸시 알림으로 보낸다.</p> <p>2. 채팅방에서, 푸시 알림 시스템은, 채팅에 참여하고 있는 다른 사용자가 새로운 채팅을 보내면, 새 채팅이 존재한다고 사용자에게 푸시 알림을 보낸다.</p>
Pre-Condition	채팅방의 푸시 알림을 받으려면 채팅방의 푸시 알림이 켜져 있는 상태여야 한다.
Post-Condition	-

3.2.11. Use Case Diagram



3.3. Nonfunctional Requirement

3.3.1. Product Requirements

3.3.1.1. Usability Requirements

사용자가 앱을 사용함에 있어서 직관적인 UI를 갖추어야 한다. 전체 페이지에서 텍스트 혹은 아이콘만 봐도 어떤 기능인지 알 수 있어야 하며, 과목 별 페이지에서도 어떤 강의를 들었는지 여부를 파악하기 쉬워야 한다. UI는 기본적으로 흰 배경에, 메뉴 창은 진한 녹색을 기본으로 하며, 각 과목마다 주차별로 몇 개의 강의를 들었고, 얼마나 남았는지 표시하게 한다. 또한 언어는 한국어와 영어를 지원한다.

3.3.1.2. Efficiency Requirements

과제 제출, 게시글 등록, 페이지 이동 등의 모든 페이지 내에서의 행동 시 응답속도는 4sec를 넘겨서는 안된다.

3.3.1.3. Dependability Requirements

GLS학생인트라넷의 성균관대 홈페이지 가입 시 입력했던 정보를 제대로 반영해서, 본인의 정보를 입력했을 시 이에 맞는 계정으로 로그인 성공해야 한다. 서버가 불안정하여 접속이 잘 되지 않는 경우와 사용자가 이용 중에 오류로 인해 페이지가 멈추는 현상 등의 빈도를 줄여야 한다. 강의를 끝까지 들었을 때 강의의 상태를 시청 완료로 반영하며, 누락되지 않도록 해야 한다. 또한 과제에 대하여 제출한 파일이나 직접 입력한 답안이 손상 없이 제출되어야 한다.

3.3.1.4. Security Requirements

사용자의 개인정보는 암호화되어 보관하여 중간에 외부의 해킹으로 가로챌 수 없도록 한다. 비밀번호는 운영자 측에서도 알 수 없고 로그인 시 식별할 때만 사용된다. 그리고 사용자의 저장 공간, 카메라, 마이크 등의 권한 접근에 관해서는 앱 사용시만 허용할 것인지, 앱 미사용 시에도 허용할 것인지, 아니면 아예 허용하지 않을 것인지를 사용자가 직접 결정할 수 있게 해야 한다.

3.3.2. Organizational Requirements

3.3.2.1. Environmental Requirements

본 앱은 기본적으로 PC와 스마트폰 동시에 사용 가능한 프로그램 기반이므로, 항상 PC와 스마트폰에서 보여지는 정보가 같도록 동기화 되어야한다. 또한 본 시스템은 안드로이드 6.0 버전 이상에서 지원된다.

3.3.2.2. Development Requirements

사용자의 개인정보와 강의 시청 및 과제 제출 여부 등의 정보는 데이터베이스에 저장된다. 이 정보들은 단순한 페이지 이동 등의 정보에 영향을 주지 않는 행동 중에 변하지 않아야 한다.

3.3.3. External Requirements

3.3.3.1. Regulatory Requirements

실시간 강의 연동 시, 사용자의 초상권 및 카메라, 마이크 등의 접근 권한을 해당 프로그램에 제공하는 것에 대한 동의를 미리 얻어야 한다. 사용자는 이를 거부할 수 있으나, 그럴 경우 실시간 강의 프로그램 사용에 제약이 있을 수 있다. 또한 실시간 강의 영상의 외부 유출을 금지해야 한다.

3.3.3.2. Ethical Requirements

본 앱은 성균관대 계정을 소유하고 있는 사용자만 이용 가능해야 한다.

3.4. Logical Database Requirements

성균관대학교 학생 데이터베이스를 이용해 사용자 정보를 관리한다. 재학생을 대상으로 하기 때문에 입학과 동시에 학생 데이터베이스에 사용자 정보가 만들어진다. 성균관대학교 GLS학생인트라넷과 연동되어 해당 과목의 수강생 목록을 받아오고 이 애플리케이션에서 학생이 학습한 결과를 토대로 학점을 책정한다.

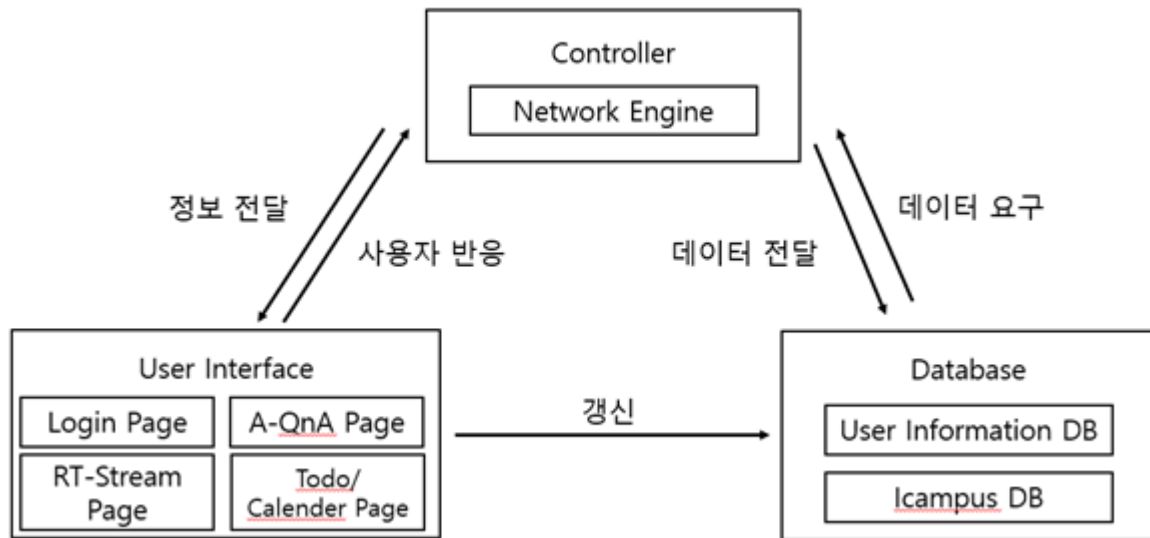
3.5. Design Constraints

본 시스템은 현재 아이캠퍼스 앱의 대략적인 인터페이스 구조를 따른다. 안드로이드 및 애플 등의 다양한 모바일 기기에서 접속할 수 있어야 하며, 시스템 관리자는 웹 브라우저를 통해 접속하고 관리할 수 있어야 한다.

3.6. Standards Compliance

Android Studio와 **Swift**로 각각 안드로이드, IOS를 지원하는 앱을 개발한다. 그리고 프로젝트 관리 시스템으로 **Github**을 사용하고 원격 서버 데이터베이스 프로그램으로 **GLS학생인트라넷**을 사용한다. 코드 내의 함수들과 변수들의 이름은 **camel case**를 사용하고, 데이터베이스의 함수와 변수들은 **underscore notation**을 사용한다.

3.7. Standards Architecture



* RT-Stream: Real-Time Stream(실시간 강의)

* A-QnA: Anonymous-QnA(익명게시판)

3.8. System Evolution

여기서는 재사용된 시스템의 변화, 사용자의 요구 변화 등에 따른 예상되는 변화에 대해서 이야기한다. 또한, 변화가 일어남에 따라 변화되지 않을 우리 시스템의 특성들도 정의한다. 이런 것들을 정의하는 이유는 후에 시스템 디자이너가 예상되는 변화를 제한할 디자인을 제안하는 것을 피하기 위함이다.

3.8.1. Change Limitation and Assumption

우리가 개발할 아이캠퍼스의 목적은 익명성이 보장되고 보기 좋은 할 일/캘린더 UI, 그리고 실시간 강의 지원이다. 따라서 게시판과 대화방의 익명성, 아이캠퍼스에 실시간 강의 시스템 포함, 할 일/캘린더 지원은 변하지 않을 특성이다. 또한 Android Studio와 Swift로 개발한다는 것, 그리고 그 외에 사용하는 시스템에도 거의 변하지 않을 것이다.

3.8.2. Change of User Requirements

UI 관련에서는 언제든지 변화가 가능하다. 사용자가 사용하면서 이러한 UI가 불편하다는 의견이 많으면 바로 변경할 수 있도록 해야 한다. 또한, 사용자가 원하는 기능이 피드백으로

들어오면 수용할 수 있도록 해야 한다. 그 예시로 익명 대화방에 이모티콘 기능 추가, 실시간 강의 수업 참가 방법 추가 등이다.

3.8.3. To Accommodate the Changes Easily

되도록 프로그래밍할 때 UI는 세분화해서 component들로 정리해두고 기능들을 추가하거나 수정할 수 있도록 가독성을 높이는 refactoring 과정을 자주 반복해줘야 할 것이다.