



Software Requirements Specification

for

성균문고

Version 1.0 approved

소프트웨어공학개론 SWE3002_42
TEAM8

신승훈 2018310242
김보미 2015314519
신경준 2018310739
곽도원 2019312044
김준서 2019313442
김재현 2020315610

2023.11.05

Index of SRS

Index of SRS	ii
Revision History	ii
Table of Contents	ii
Table of Figures	ii
1. Introduction	1
1.1 Purpose	1
1.2 Scope	1
1.3 Definition, Acronyms and Abbreviation	2
1.4 References	3
1.5 Overview	4
2. Overall Description	5
2.1 Product Perspective	5
2.1.1 System Interfaces	5
2.1.2 User Interfaces	5
2.1.3 Hardware Interfaces	6
2.1.4 Software Interfaces	6
2.1.5 Communicate Interfaces	6
2.1.6 Memory Constraints	6
2.1.7 Operations	7
2.1.7.1 사용자 프로필 등록	7
2.1.7.2 사용자 프로필 평가	7
2.1.7.3 거래 글 작성	7
2.1.7.4 맞춤 검색	7
2.1.7.5 즐겨찾기 / 알림	7
2.2 Product Functions	8
2.2.1 사용자 프로필 및 유저 관리	8
2.2.2 거래 기능	8
2.3 User Classes and Characteristics	9
2.4 Constraints	10
2.5 Assumptions and Dependencies	10
3. External Interface Requirements	11
3.1 User Interfaces	11
3.1.1 회원가입 User Interface	11
3.1.2 회원가입 인증 User Interface	12
3.1.3 회원가입 (아이디 & 비밀번호) User Interface	13
3.1.4 회원가입 (닉네임 & 자기소개) User Interface	14
3.1.5 로그인 User Interface	15
3.1.6 메인페이지 User Interface	16
3.2 Hardware Interfaces	17
3.3 Software Interfaces	17
3.4 Communications Interfaces	18

4. Functional Requirements	18
4.1 Use Case	18
4.1.1 회원가입	18
4.1.2 로그인/로그아웃	19
4.1.3 프로필 수정	19
4.1.4 게시물 작성	20
4.1.5 거래 채팅	20
4.1.6 거래 완료	21
4.1.7 유저 평가	21
4.1.8 이상 유저 제한	22
4.1.9 게시물 검색	22
4.2 Use-case Diagram	23
4.3 Data Flow Diagram & Dictionary	24
5. Other Nonfunctional Requirements	26
5.1 Product Requirements	26
5.1.1 Usability Requirements (1)	26
5.1.2 Usability Requirements (2)	26
5.1.3 Dependability Requirements	27
5.1.4 Security Requirements	27
5.2 Organizational Requirements	29
5.2.1 Environments Requirements	29
5.2.2 Development Requirements	29
5.3 External Requirements	30
5.3.1 Regulatory Requirements	30
5.3.2 Ethical Requirements	31
5.3.3 Legislative Requirements	31
5.4 Organizing the Specific Requirements	32
5.4.1 Context Model	32
5.4.2 Activity Diagram	33
5.4.3 Sequence Diagram	34
5.5 Design Constraints	35
6. System Features	36
6.1 System Architecture	36
6.2 System Evolution	37
6.2.1 Limitation and Assumption	37
6.2.2 Anticipated Changes	37

Revision History

Name	Date	Reason For Changes	Version
SKMG	2023.11.05	Initialize SKMG project	v1.0.0

Table of Contents

Table 1 Acronyms and Abbreviation	2
Table 2 Definition	3
Table 3 Hardware Interfaces	17
Table 4 Software Interfaces	17
Table 5 Communication Interfaces	18
Table 6 회원가입 Use Case	18
Table 7 로그인/로그아웃 Use Case	19
Table 8 프로필 수정 Use Case	19
Table 9 게시글 작성 Use Case	20
Table 10 거래 채팅 Use Case	20
Table 11 거래 완료 Use Case	21
Table 12 유저 평가 Use Case	21
Table 13 이상 유저 제한 Use Case	22
Table 14 게시물 검색 Use Case	22
Table 15 Data dictionary(User)	24
Table 16 Data dictionary(Product)	25
Table 17 Data dictionary(Hashtag)	25

Table of Figures

Figure 1 Class Diagram	1
Figure 2 회원가입(캠퍼스 선택) User Interface	11
Figure 3 회원가입(이메일 인증) User Interface	12
Figure 4 회원가입(아이디, 비밀번호) User Interface	13
Figure 5 회원가입(닉네임, 자기소개) User Interface	14
Figure 6 로그인 User Interface	15
Figure 7 메인페이지 User Interface	16
Figure 8 Use Case diagram	23
Figure 9 Data diagram	24
Figure 10 Context model	32
Figure 11 Activity diagram	33
Figure 12 Sequence diagram	34
Figure 13 System Architecture	36

1. Introduction

1.1 Purpose

이 Software Requirements Specification 문서는 성균관대학교 재학생을 위한 웹 서비스인 ‘성균문고’의 System Requirements와 User Requirements를 기술한다. ‘성균문고’는 성균관대학교 2023년 2학기에 소프트웨어공학개론 수업 SWE3002_42 8조에 속하는 구성원 신승훈, 김보미, 신경준, 곽도원, 김준서, 김재현에 의하여 설계되고 개발될 예정이다.

‘성균문고’의 목적은 성균관대학교 양 캠퍼스간 재학생이 중고 전공서적을 합리적으로 거래할 수 있는 플랫폼을 제공함이다. 많은 학생들이 중고 전공서적을 저렴한 가격에 구매하거나 판매하고자하는 수요가 있지만 이를 위한 효과적인 거래 플랫폼이 부족했다. 또한, 일반 중고거래 사이트에서는 학교 내 특정 전공서적에 대한 정보가 제한적이고, 판매자/구매자 간 신뢰성 문제도 발생할 수 있다. ‘성균문고’ 플랫폼은 성균관대학교 학생들만을 위한 중고 전공서적 거래 플랫폼을 제공함으로써 교내에서 안전하고 신뢰성있는 거래를 가능하게 한다. 사용자는 ‘성균문고’를 이용해 캠퍼스, 전공, 가격 등 조건을 검색 혹은 등록해 적합한 서적을 구매 혹은 판매할 수 있다.

본 SRS에서는 ‘성균문고’에 대한 전반적인 요구사항에 대해 제품, 기능, 유저, 운영 환경, 디자인, 가정 등이 분석 및 기술되어있다.

1.2 Scope

‘성균문고’는 성균관대학교 재학생에게 전공 서적 거래에 특화된 웹 서비스를 제공한다. 판매를 원하는 사용자는 서비스에 로그인하여 서적의 정보, 사진, 상태, 가격, 캠퍼스, 전공 등을 등록하며, 구매를 원하는 사용자는 이러한 포스팅을 필터링 기능을 이용해 검색하고 판매자와 소통을 통해 거래방식과 일정을 조율하여 거래를 진행할 수 있다. 또한, 사용자는 개인 프로필을 통해 거래 신뢰도를 확인할 수 있으며, 맞춤 알림 서비스를 통해 원하는 서적이 등록되었을 때 알림을 받을 수 있다.

1.3 Definition, Acronyms and Abbreviation

아래 표는 이 문서에서 사용된 약어에 대한 설명이다.

Table 1 - Acronyms and Abbreviation

약어	설명
API	Application Programming Interface
App	Application
BE	BackEnd
CPU	Central Processing Unit
DB	Database
FE	FrontEnd
JWT	Json Web Token
LAN	Local Area Network
MVC	Model, View, and Controller
OS	Operating System
UI	User Interface
UX	User Experience

아래 표는 이 문서에서 정의한 특정 기술적 개념이다.

Table 2 - Definition

약어	설명
유저	성균문고 웹 서비스의 사용자
관리자	성균문고의 모든 권한을 갖는 모든 유저
디바이스	성균문고 서비스를 실행하는 하드웨어 장치
즐거찾기	중고 서적을 쉽게 검색할 수 있도록 키워드를 저장하는 기능
백엔드	웹서비스의 서버 측 로직 및 데이터베이스를 처리하는 부분
프론트엔드	사용자와 직접 상호작용하는 클라이언트 측 인터페이스
포스팅	성균문고 서비스에 게시되는 판매 글
캠퍼스	성균관대학교의 캠퍼스. 인문사회캠퍼스와 자연과학캠퍼스로 나뉨
데이터베이스	데이터를 저장, 검색, 검색 및 관리하는 시스템

1.4 References

IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEEExplore Digital Library

<http://ieeexplore.ieee.org/Xplore/guesthome.jspf>

Team 10, “RED:orm Dormitory life assistance platform”, SKKU

https://github.com/skkusal/SE_2022f

1.5 Overview

본 Software Requirements Specifications(SRS)는 성균문고 웹 서비스에 대한 상세한 시스템 및 사용자 요구사항을 체계적으로 기술한 것으로, 총 6장으로 구성되어있다.

1장에서는 서비스의 목적과 범위, 정의, 약어, 참고문헌 및 전반적인 문서 구조에 대해 소개한다. 성균문고는 성균관대학교 재학생을 위한 중고 전공서적 거래 웹 서비스로 이 서비스의 목적, 필요성 및 타겟 사용자를 명확히 정의하여 프로젝트의 배경과 틀을 제공한다.

2장에서는 개발 중인 서비스의 전체적인 구조와 시스템의 관점, 제공될 기능, 사용자 클래스 및 특성, 제약 조건, 가정 및 의존성 등을 다룬다. 이 장에서는 성균문고가 어떻게 운영될 것인지, 시스템이 작동하는 동안 마주칠 수 있는 제약사항 및 해당 시스템이 어떻게 상호작용하는지에 대해 설명한다.

3장에서는 사용자 인터페이스, 하드웨어 및 소프트웨어 인터페이스, 통신 인터페이스에 대해 상세한 요구사항을 기술한다. 이 장은 성균문고의 사용자가 시스템과 어떻게 상호작용할지에 대한 구체적인 정보를 제공한다.

4장에서는 시스템의 핵심 기능과 사용 사례, 사용 사례 다이어그램, 데이터 흐름도를 포함하여 성균문고가 사용자의 요구를 어떻게 충족시킬 것인지 자세히 설명한다. 시스템이 사용자의 입력 및 시나리오에 어떻게 반응할지에 대한 기능적 측면을 중점적으로 다룬다.

5장에서는 제품 요구사항, 조직적 요구사항, 외부 요구사항 등 성균문고가 충족해야 할 비기능적 요구사항을 다룬다. 이 장은 시스템의 성능, 품질, 기준 준수 등을 보장하기 위해 필요한 사항들에 서술한다.

6장에서는 성균문고의 시스템 아키텍처, 시스템 진화 및 기타 중요 기능들에 대해 설명한다. 이 장은 전체적인 시스템 설계와 향후 발전 가능성에 대한 비전을 제공함으로써, 장기적인 관점에서는 시스템 발전 가능성을 이해하는데 도움을 준다.

2. Overall Description

2.1 Product Perspective

성균문고는 성균관대 양 캠퍼스 간 학생들이 전공 중고 서적을 합리적인 가격으로 빠르고 안전하게 거래할 수 있도록 도와주는 웹 서비스 플랫폼이다. 초기 회원가입을 진행할 때 성균관대 재학생임을 확인할 수 있는 이메일 인증 과정을 거치게 되고, 개인 유저에 대한 프로필 평가 시스템을 통해 신뢰성있고 안전한 거래가 성사될 수 있도록 지원한다. 전공 서적에 대한 사진 업로드, 상태, 캠퍼스, 전공과 같은 정보를 기입하여 거래글을 등록할 수 있으며 특정 키워드를 등록 한 경우, 조건에 부합하는 물품이 등장했을 때 맞춤 알림을 수신할 수 있다. 또한, 피쳐 간 값 필터를 세팅하여 유저 친화적인 검색 엔진을 제공하며 이를 통해 사용하는 유저가 빠르게 물품을 검색할 수 있으며 실시간 소켓 통신 기반 채팅 시스템을 통해 거래 관련 대화, 거래 약속 확정, 거래 성사 등의 외적인 프로세스를 수행할 수 있다.

2.1.1 System Interface

유저, 포스팅, 캠퍼스 메타데이터, 전공 메타데이터 테이블에 대한 데이터베이스는 NoSQL인 MongoDB로 관리하게 된다. 데이터베이스와 상호작용하기 위한 백엔드 인터페이스로 Node Express 프레임워크를 사용하고 API 또한 개발되었다.

2.1.2 User Interface

성균문고 플랫폼은 웹앱 서비스이며 유저의 디바이스 크기에 따라 동적인 UI를 웹 브라우저를 통해 출력한다. 웹 브라우저 내 유저의 클릭, Drag & Drop 등의 웹 기반 입력 기능을 통해 반응하게 된다. 반응형 UI를 통해 여러가지 상황에 대해서도 서비스를 제공할 수 있도록 구성 하였으며 서비스 사용을 위해선 회원가입 및 로그인 프로세스가 필수적이다. 메인 홈 화면 내에서는 현재 거래가 완료되지 않고 진행되고 있는 중고 서적 물품들의 포스팅들이 Grid 형식으로 나열되어있고 클릭 시 상세 정보를 확인할 수 있다.

판매를 진행하고 싶을 경우 메인 홈 화면 내 포스팅 등록 버튼을 통해 포스팅 등록 페이지로 이동하게 되고 사진, 물품 정보를 기입하여 포스팅을 완료할 수 있다.

구매를 희망할 경우 포스팅 클릭 후 정보를 확인하고, 판매자와 채팅을 거래를 진행할 수 있으며 신뢰 여부를 판단하기 위해 판매자 프로필을 확인할 수도 있다. 거래가 정상적으로 완료 되었을 경우 판매자는 해당 포스팅에 판매 완료 토글을 활성화하여 구매자에게 알림을 전송하고 구매자는 판매자에게 평점을 부여할 수 있다.

유저 프로필 페이지 내에는 키워드 알림 기능을 위한 컴포넌트들이 배치되어있고 복수개의 키워드를 등록하여 새로운 포스팅에 해당 키워드가 포함되어 있을 경우 이메일로 맞춤 알림을 수신하게 된다.

2.1.3 Hardware Interface

성균문고는 웹 브라우저를 사용할 수 있는 모든 디바이스를 대상으로 구현되었으며, 구글사의 Chrome 브라우저에 최적화 되어있다. 추가적으로 해당 플랫폼을 사용하는 유저의 디바이스는 최소 2GB 메모리 공간과 1.0Hz 프로세스를 필요로 한다.

2.1.4 Software Interfaces

해당 SRS 문서가 작성되는 최신 시점 (2023.11.05)으로 구글 Chrome 웹 브라우저, Firefox 웹 브라우저, 애플 Safari 웹 브라우저의 최신 버전과 그 이전의 최신 버전을 지원한다. 페이지 간 라우팅은 next.js 기반으로 처리하게 되며 jwt 토큰 방식으로 유저 세션을 관리 및 인증을 담당한다.

2.1.5 Communicate Interfaces

프론트엔드에서 발생하는 상호작용을 백엔드로 전달하여 MongoDB 데이터베이스 내 트랜잭션을 수행하게 될 때 Mongoose 클라이언트 및 인터페이스를 통해 직접적인 통신을 하게 되며 유저 인터페이스에 접속한 유저는 Wi-Fi 또는 LAN 또는 셀룰러 데이터에 연결되어있어야 한다.

2.1.6 Memory Constraints

React 기반 앱은 웹 브라우저에서 실행되므로 최소 1GB 이상의 메모리를 요구한다. 더 유저 친화적인 UI의 렌더링 및 시스템 레이어와 빠른 통신을 위해 2GB~4GB 이상의 메모리를 추가적으로 할당하여 사용할 수 있다면 보다 더 긍정적인 유저 경험을 만들 수 있을 것이다.

2.1.7 Operations

2.1.7.1 사용자 프로필 등록

서비스 이용을 위해 회원가입 프로세스를 거쳐야하고 성균관대 재학생임을 인증한 뒤 로그인할 수 있다. 로그인 직 후, 헤더의 프로필 버튼을 클릭하여 유저의 전공, 캠퍼스, 키워드 등을 등록 및 수정하여 프로필을 등록할 수 있고 다른 유저에게 평가받은 점수를 확인할 수 있다.

2.1.7.2 사용자 프로필 평가

거래가 성사된 후 구매자는 판매자 프로필을 평가할 수 있으며 이는 더 신뢰성 있고 안전한 거래를 도와주는 척도로 작용한다. 판매가 완료 되었을 시점에 구매자에게 알림이 전송되고, 구매자는 판매자 프로필 페이지로 이동할 수 있다. 해당 페이지에서 판매자에게 별점을 부여할 수 있고 이는 모든 유저에게 공개되는 정보이다.

2.1.7.3 거래 글 작성

메인 페이지 내 포스팅 등록 버튼을 누르게 되면 포스팅 작성 페이지로 이동하게 되고 물품 사진, 제목, 상태, 전공, 설명 등의 정보를 폼 형태로 기입할 수 있다. 사진을 필수적으로 최소 한 장 이상 업로드 하계끔 제약을 설정할 것이고 서적 상태는 enum으로 세팅하여 최하~최상까지 유저가 직접 선택할 수 있게끔 한다. 설명란에는 키워드를 해시태그로 작성할 수 있으며 이는 키워드 알림 전송에 필요한 트리거로 적용될 수 있다.

2.1.7.4 맞춤 검색

원하는 서적을 더 빠르고 쉽게 찾기 위해 맞춤 검색 기능을 메인 페이지에서 제공한다. 텍스트 기반의 폭 넓은 검색 엔진이 아닌 전공, 캠퍼스, 상태를 직접 유저가 카테고리 별로 설정할 수 있으며 양 캠퍼스 간 거래 기능을 지원해 복수전공생의 교재 구매 편의성을 향상시킨다.

2.1.7.5 즐겨찾기 / 알림

키워드 알림을 활성화/비활성화 할 수 있으며 해시태그를 기반으로 키워드를 등록할 수 있다. 또한 전공을 복수개로 등록할 수 있고 전공과 키워드가 일치하는 포스팅이

등록되었을 경우 키워드 알림을 활성화 해놓은 유저에 한해서 메일 알림을 전송한다.

2.2 Product Functions

2.2.1 사용자 프로필 및 유저 관리

해당 서비스는 성균관대 재학생 신분이 인증된 유저만 접근이 가능하며 이에 따라 회원가입 시 성균관대 메일을 통해 인증 과정을 거치게 된다. 정상적으로 회원가입 완료 후 프로필을 인스타그램 및 당근마켓 서비스와 유사하게 커스터마이징 할 수 있으며 유저의 전공, 캠퍼스, 관심 키워드를 등록할 수 있다.

관심 키워드를 복수 개로 설정 해놓고 키워드 알림을 활성화 시키면 해당 조건에 부합하는 거래글이 등록되었을 경우 유저에게 알림이 전송되며 사용자의 요구 사항을 빠르게 충족 시킬 수 있다. 또한 기존 시스템의 양 캠퍼스간 교류가 쉽지 않았던 단점을 극복할 수 있으므로 복수전공생의 편의성을 도모할 수 있다.

대면 거래를 직접 하는데 있어 신뢰성과 안전성이 중요한 만큼 각 사용자 프로필에 평가 시스템을 도입할 예정이며, 거래가 성사된 후 구매자는 판매자 프로필에 점수를 부여할 수 있으며 몇 명의 인원이 점수를 평가했으며 평점은 어떻게 되는지 등을 공개적으로 확인할 수 있다.

2.2.2 거래 기능

로그인 후 성균문고 메인 페이지에서 거래글 등록 프로세스를 수행할 수 있다. 핵심적인 기능으로 거래글을 등록할 때 매물에 대한 사진은 필수적으로 등록해야하며 뿐만 아니라 전공, 책 제목, 상태, 해시태그 기반의 키워드를 남길 수 있다.

등록된 거래글은 성균문고 서비스를 사용하는 모든 유저에게 공개적으로 보여지며 카드 뷰의 형태로 메인 페이지에 보여지게 된다. 구매를 희망하는 유저는 맞춤 검색, 키워드 알림 등의 소스를 통해 접근할 수 있으며 자신의 조건에 부합하는 매물일 경우 판매자와 채팅을 통해 거래 약속을 만들 수 있다.

거래를 정상적으로 마무리 했다면 해당 거래글의 상태는 거래 완료로 변경되고, 상태가 변경되는 시점에 구매자에게 판매자 평가 페이지 알림 메시지를 발신하게 된다. 따라서 거래 기능의 End-to-End 파이프라인이 위와 같이 구성된다고 볼 수 있다.

2.3 User Classes and Characteristics

성균문고 웹 서비스 유저 클래스는 최상위 클래스로 Product와 Composition 관계이다.

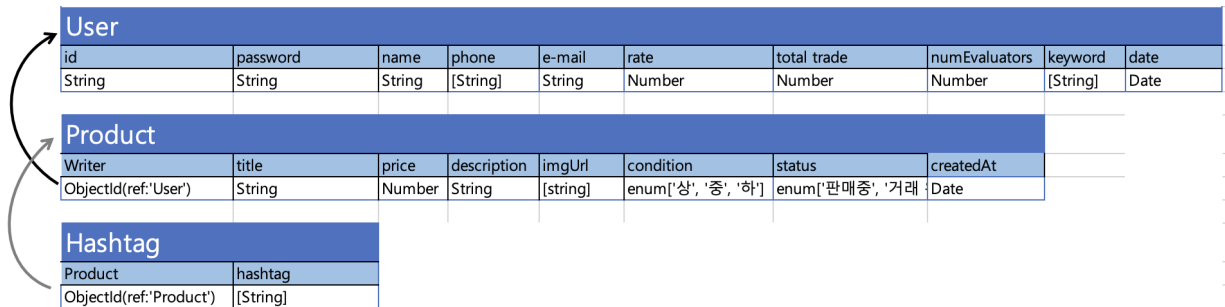


Figure 1 Class Diagram

회원가입 시 **비밀번호는 암호화 알고리즘을 통해 DB에 저장**되며 프로필 페이지에 들어가는 개인 정보 및 프로필 평가 시 반영되는 평점과 구매자 수에 대한 속성 값이 존재한다. 또한, 키워드 알림 등록을 위한 키워드를 문자열 리스트로 설정할 수 있으며 User 클래스의 PK와 Product 클래스의 Writer를 외부키 관계로 설정할 수 있다.

따라서, 특정 유저가 거래글 포스팅을 등록하게 되면 해당 거래글의 PK는 User의 ID로 설정되고 상품에 대한 가격, 제목, 설명, 이미지, 상태, 거래 상태 값을 속성으로 지정할 수 있다.

실제 구매를 희망하는 사람은 구매자 ID와 판매자 ID의 값을 기반으로 유니크하게 채팅방을 생성할 수 있고 거래가 정상적으로 완료되었을 경우 Product 테이블의 status를 거래 완료로 토글 할 수 있게 된다.

2.4 Constraints

성균문고 웹 서비스는 이후 3장에서 자세히 기술되는 내용을 바탕으로 구성되며 BE와 FE가 분리되어 구현된다. 개발 과정 중 아래와 같은 사항들을 우선시한다.

- Git을 통해 버전 관리를 하며 커밋 메시지를 직관적으로 유지한다.
- 2주 단위 스프린트를 통해 작은 단위 릴리즈를 거치고 주기적으로 통합을 수행한다.
- FE 개발에 있어서 React.js를 사용하고 BE 개발에 있어 Node express를 사용한다.
- 페이지 라우팅에 대해 더 높은 성능을 제공하기 위해 swc가 적용된 Next.js를 적용한다.
- 로컬 스토리지 사용을 최대한 지양하고 외부 스토리지를 마련해 유저 경험을 향상시킨다.
- 핵심 기능을 최소한의 페이지에 배치시켜 직관적이고 친숙한 UI를 구성한다.
- develop과 main 브랜치를 나누어 개발 환경과 프로덕션 환경을 구분한다.
- Chrome 117, Firefox 118, Safari 15.2 환경에서 인터랙션과 핵심 기능을 작동하는지 테스트한다.
- Swagger UI를 개발하여 FE와 BE 간의 의사소통에 있어 명확성을 확립한다.
- 페이지, 컴포넌트, API의 추후 유지보수 및 발전을 대비하여 주석을 작성하고 리뷰한다.
- Figma로 초안 디자인을 구성하고 반응형 UI를 구성하여 질 높은 서비스를 제공한다.
- 중복 코드를 줄이고 ESLint를 사용하여 코드 스타일을 일관성있게 유지한다.
- 서비스 전용 메일을 생성하여 알림 기능을 구현한다.
- 성균관대 학생의 편의성을 도모하려는 목적성을 우선시하고 이에 따른 외부 스토리지의 과금이 발생하지 않게 지속적인 모니터링을 수행한다.

2.5 Assumptions and Dependencies

성균문고는 웹 서비스로 브라우저 환경에서 서비스되며 구글 Chrome 브라우저에 최적화되어있다. 추가적으로 Firefox와 Safari 브라우저 환경도 지원한다.

외부 third-party 컴포넌트의 연결 및 유저 이벤트를 최적의 환경에서 처리하기 위해 각 브라우저의 최신 버전과 직전의 최신 버전에서 사용된다는 가정 하에 설계되었으며 이 외의 환경에서는 정상 작동이 되지 않을 수도 있다.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 회원가입 User Interface

SKKU_MUNGO

회원가입 로그인 고객센터

SKKU_MUNGO 회원가입

스무공고 계정으로 성균관대 양 캠퍼스 간 학생들이 전공 중 고 서적을
합리적인 가격으로 빠르게 구매하고 안전하게 거래할 수 있습니다

캠퍼스 선택

전공 선택

다음

Figure 2 회원가입(캠퍼스 선택)의 User Interface

1. 이 플랫폼의 사용자는 성균관대학교에 재학중인 학생으로 설정하며, 캠퍼스(명륜, 율전)를 선택한다.
2. 1의 과정을 마친 후, 사용자는 본인의 전공을 드롭다운 형식으로 선택한다.
3. 사용자가 선택한 캠퍼스와 전공의 정보를 바탕으로 [Figure 3]로 이동한다.

3.1.2 회원가입 인증 User Interface

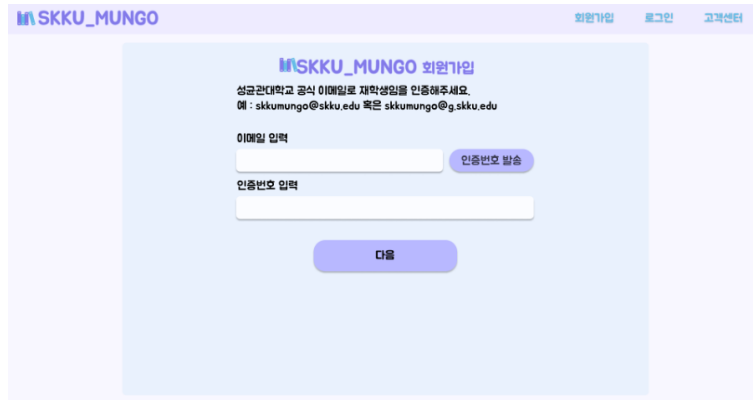
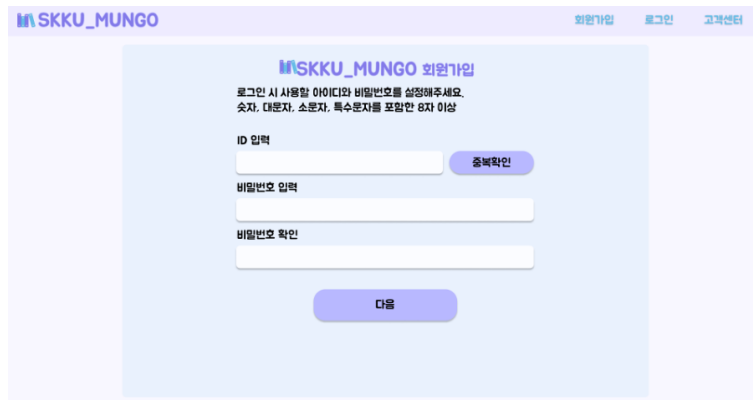


Figure 3 회원가입(이메일 인증)의 User Interface

1. [Figure 2]에서의 과정을 마친 사용자는 이메일 인증을 진행한다. 이메일은 skku 도메인만 이용가능하며, 다른 도메인 사용시 인증번호 발생이 되지 않는다.
2. 사용자는 본인의 skku이메일로 발송된 인증번호를 입력한다. 인증번호가 일치할 시, [Figure 4]로 이동한다. 재시도 횟수에 대한 제한은 없다.

3.1.3 회원가입 (아이디 & 비밀번호) User Interface

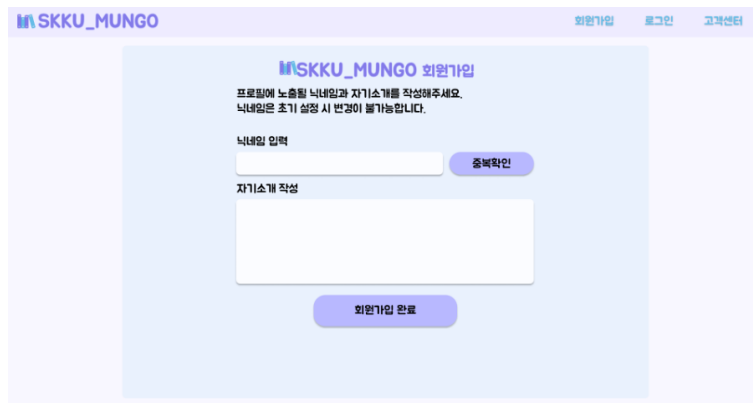


The screenshot shows a web browser window with the URL 'SKKU_MUNGO'. The page has a light blue header with navigation links: '회원가입' (Sign Up), '로그인' (Login), and '고객센터' (Customer Center). The main content area is titled 'SKKU_MUNGO 회원가입' and contains the following text: '로그인 시 사용할 아이디와 비밀번호를 설정해주세요. 숫자, 대문자, 소문자, 특수문자를 포함한 8자 이상'. Below this, there are three input fields: 'ID 입력' (ID Input), '비밀번호 입력' (Password Input), and '비밀번호 확인' (Confirm Password). A '중복확인' (Check Duplicate) button is located next to the ID input field. A '다음' (Next) button is at the bottom of the form.

Figure 4 회원가입(아이디, 비밀번호)의 User Interface

1. 사용자는 ID를 입력하고, 중복여부를 먼저 체크한다. 중복되지 않은 ID에 대해서 사용 가능하다.
2. 원하는 비밀번호를 입력한 후, 다시 한번 본인의 비밀번호를 확인한다. 비밀번호가 일치할 시, [Figure 5]로 이동 가능하다.

3.1.4 회원가입 (닉네임 & 자기소개) User Interface



The screenshot shows a web browser window with the title 'SKKU_MUNGO'. The page has a light blue header with the logo 'SKKU_MUNGO' on the left and navigation links '회원가입', '로그인', and '고객센터' on the right. The main content area is a light blue box with the title 'SKKU_MUNGO 회원가입'. Below the title, there is a message: '프로필에 노출될 닉네임과 자기소개를 작성해주세요. 닉네임은 초기 설정 시 변경이 불가능합니다.' (Please write a nickname and self-introduction that will be displayed on the profile. The nickname cannot be changed when initially set). There are two input fields: '닉네임 입력' (Nickname input) and '자기소개 작성' (Self-introduction writing). The '닉네임 입력' field has a '중복확인' (Check duplicate) button next to it. The '자기소개 작성' field is a larger text area. At the bottom of the form, there is a '회원가입 완료' (Registration complete) button.

Figure 5 회원가입(닉네임, 자기소개)의 User Interface

1. 사용자는 닉네임을 입력하고, 중복되지 않을 시 해당 닉네임을 사용 가능하다.
2. 간단한 자기소개글을 입력하게 되면, 회원가입이 완료된다.

3.1.5 로그인 User Interface



Figure 6 로그인의 User Interface

1. 사용자는 ID와 Password를 입력한다. 로그인에 성공하면 [Figure 7]로 이동한다.
2. 최초 로그인에 성공하면 JWT 토큰이 클라이언트에 전달되어 사용된다.

3.1.6 메인페이지 User Interface

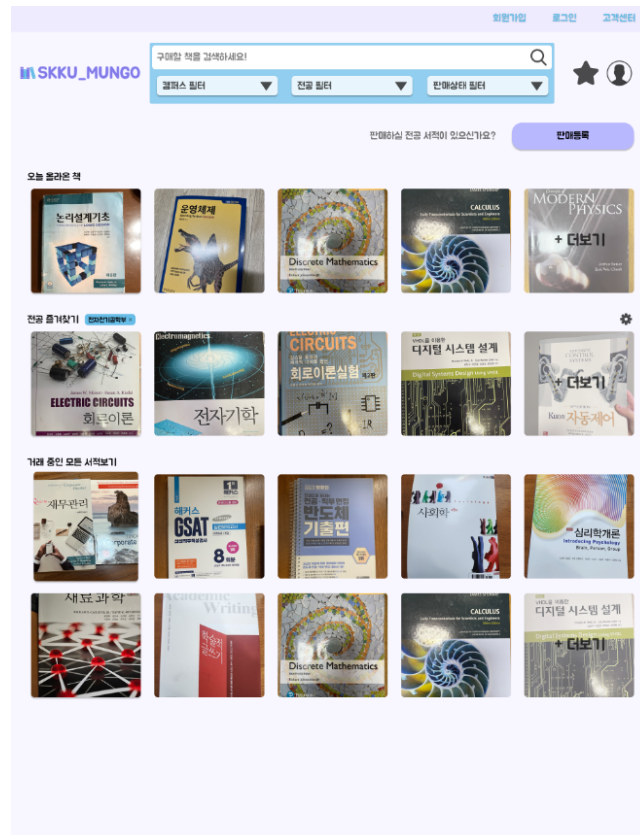


Figure 7 메인페이지의 User Interface

1. 화면 진입시 중앙에는 오늘 올라온책, 사용자가 설정한 전공 즐겨찾기, 거래 중인 모든 서적보기의 섹션으로 나누어 서적들을 확인할 수 있다.
2. 상단에서 캠퍼스(명륜, 율전), 전공, 판매상태를 선택하여 필터 검색을 할 수 있다. 필터 설정 후, 돋보기 아이콘을 누르면 원하는 서적이 화면 중앙에 표시된다.
3. 화면의 전공서적 사진들을 클릭시 해당 판매 게시물로 이동한다.
4. 상단 우측에 존재하는 판매등록 버튼을 누를 시, 게시물 작성 화면으로 이동한다.
5. 상단 우측의 별표 버튼을 누르면, 본인이 관심 설정한 서적들에 대해서 확인 가능하다. 옆에 있는 사람모양 아이콘을 누를 시, 회원정보 페이지로 이동한다.

6. 화면 최상단 우측의 회원가입 버튼을 누를 시, [Figure 1-1]로 이동하여 회원가입 진행이 가능하다. 옆에 존재하는 로그인 버튼을 누를 시, [Figure2]로 이동하여 로그인이 가능하며, 로그인 상태일 시, 로그인 버튼은 로그아웃 버튼으로 바뀌어 로그아웃이 가능하다.

3.2 Hardware Interfaces

Table 3 - Hardware Interfaces

Name	
Description	<ul style="list-style-type: none"> - 성균문고는 웹 브라우저를 사용할 수 있는 모든 디바이스에 대해서 서비스를 지원한다. - 구글 Chrome 브라우저에 최적화
Requirements	<ul style="list-style-type: none"> - 최소 2GB 메모리 공간 - 1.0Hz 프로세스를 지원하는 디바이스

3.3 Software Interfaces

Table 4 - Software Interfaces

Name	
Description	사용자는 Chrome 웹 브라우저, Firefox 웹 브라우저, Safari 웹 브라우저의 최신버전을 통해 서비스를 이용할 수 있다.
Software Used	<ul style="list-style-type: none"> - express(v4.18.2) - mongodb(v6.1.0) - react(v18.0.0)

3.4 Communications Interfaces

Table 5 - Communication Interfaces

Name	
Source / Destination	User Device / Main Computer
Description	사용자는 성균문고 페이지에 로그인하여 중고서적 판매글을 작성하거나, 원하는 중고서적을 구매할 수 있다. 업로드 된 거래글에서 채팅을 통해 거래를 진행할 수 있다. 거래가 종료되면, 사용자는 판매자에 대하여 평점을 매길 수 있다.
Connection form	<ul style="list-style-type: none"> - WIFI 및 LAN - 셀룰러 데이터

4. Functional Requirements

4.1 Use Case

4.1.1 회원가입

Table 6 - 회원가입 Use Case

Use case name	회원가입
Actor	회원가입을 하지 않은 유저
Description	신규 유저가 시스템 사용을 위하여 회원가입을 한다.
Normal Course	<ol style="list-style-type: none"> 1. 신규 유저가 회원가입 페이지에 진입한다. 2. 캠퍼스 및 전공을 설정한다. 3. 성균관대 이메일 인증을 완료한다. 4. 아이디 및 패스워드 설정을 진행한다. 5. 닉네임 및 간단한 설명을 작성한다.
Pre-condition	가입되지 않은 유저여야 한다. 성균관대학교 학생이어야 한다.
Post-condition	등록된 아이디 및 패스워드로 로그인이 가능해야 한다.
Assumptions	-

4.1.2 로그인/로그아웃

Table 7 - 로그인/로그아웃 Use Case

Use case name	로그인/로그아웃
Actor	회원가입이 된 유저
Description	유저가 로그인/로그아웃을 한다.
Normal Course	<로그인> <ol style="list-style-type: none"> 아이디와 패스워드를 입력하여 로그인한다. 아이디와 패스워드가 다를 경우 다시 시도한다. <로그아웃> <ol style="list-style-type: none"> 로그아웃 버튼을 눌러 로그아웃 한다.
Pre-condition	<로그인> <ul style="list-style-type: none"> 회원가입이 된 유저여야 한다. <로그아웃> <ul style="list-style-type: none"> 로그인 상태인 유저여야 한다.
Post-condition	-
Assumptions	-

4.1.3 프로필 수정

Table 8 - 프로필 수정 Use Case

Use case name	프로필 수정
Actor	로그인 상태인 유저
Description	로그인 된 유저가 개인정보를 변경한다.
Normal Course	<ol style="list-style-type: none"> 메인페이지에서 사용자 아이콘을 눌러 프로필 수정 페이지에 접속한다. 전공/닉네임/간단한 설명 등 유저가 원하는 항목을 변경한다. 필요시 서적 즐겨찾기 설정을 한다. 저장 버튼을 눌러 변경된 내용을 저장한다.
Pre-condition	유저가 로그인 되어 있어야 한다.
Post-condition	사용자 아이콘을 눌렀을 때, 변경된 회원정보가 나와야 한다.
Assumptions	-

4.1.4 게시글 작성

Table 9 - 게시글 작성 Use Case

Use case name	게시글 작성
Actor	로그인 상태인 유저
Description	유저가 전공 서적 거래를 위한 게시글을 작성한다.
Normal Course	<ol style="list-style-type: none"> 1. 서적의 앞, 옆, 뒤 등의 최소 3장의 사진을 등록한다. 2. 게시글 제목을 작성한다. 3. 서적의 상태(최하~최상)를 설정한다. 4. 사용되어지는 전공을 설정한다. 5. 게시글 본문을 작성한다. 6. 등록하기 버튼을 눌러 게시글을 등록한다.
Pre-condition	유저가 로그인 되어 있어야 한다.
Post-condition	작성한 게시글이 메인페이지에 나타나야 한다.
Assumptions	-

4.1.5 거래 채팅

Table 10 - 거래 채팅 Use Case

Use case name	프로필 수정
Actor	로그인 상태인 유저
Description	유저가 판매자에게 거래를 위한 채팅을 한다.
Normal Course	<ol style="list-style-type: none"> 1. 구매하고 싶은 게시물 페이지에 접속한다. 2. 게시글 내 채팅 아이콘을 클릭한다. 3. 판매자에게 구매의사를 말하며 채팅을 시작한다.
Pre-condition	유저가 로그인 되어 있어야 한다.
Post-condition	-
Assumptions	-

4.1.6 거래완료

Table 11 - 거래완료 Use Case

Use case name	프로필 수정
Actor	구매자와 거래를 완료한 유저
Description	판매자가 구매자와 거래를 마친 후, 게시글을 거래완료 상태로 바꾼다.
Normal Course	<ol style="list-style-type: none"> 1. 판매자와 구매자가 합의하에 거래를 진행한다. 2. 거래가 완료된 후, 판매자는 채팅 페이지 내 거래 완료 버튼을 누른다.
Pre-condition	판매자와 구매자가 채팅 진행중이어야 한다.
Post-condition	거래완료 후, 판매자와 구매자는 서로에게 평점을 매길 수 있어야 한다.
Assumptions	-

4.1.7 유저평가

Table 12 - 유저평가 Use Case

Use case name	프로필 수정
Actor	로그인 상태인 유저
Description	거래를 마친 후, 유저에게 평점을 부여한다.
Normal Course	<ol style="list-style-type: none"> 1. 거래를 희망하는 유저와 거래를 진행 후, 채팅 페이지 내 거래 완료 버튼을 클릭한다. 2. 판매자는 구매자에게, 구매자는 판매자에게 평점을 부여한다.
Pre-condition	거래가 완료된 유저에게 적용되어야 한다.
Post-condition	-
Assumptions	-

4.1.8 이상 유저 제한

Table 13 - 이상 유저 제한 Use Case

Use case name	프로필 수정
Actor	성균문고 관리자
Description	관리자가 이상 유저에 대해서 제한을 가한다.
Normal Course	<ol style="list-style-type: none"> 1. 관리자 계정으로 로그인한다. 2. 이상 유저에 대한 정보를 확인 후, 제한 조치를 가하는 설정을 한다..
Pre-condition	관리자 계정이어야 한다.
Post-condition	제한을 받은 유저는 서비스 이용 제한 등에 조치가 가해져야 한다.
Assumptions	-

4.1.9 게시물 검색

Table 14 - 게시물 검색 Use Case

Use case name	프로필 수정
Actor	유저
Description	유저가 원하는 게시물을 검색한다.
Normal Course	<ol style="list-style-type: none"> 1. 유저가 원하는 검색어를 화면 상단 검색창에 입력한다. 2. 원하는 검색어를 입력 후, 돋보기 버튼을 눌러 게시물을 조회한다.
Pre-condition	-
Post-condition	유저가 입력한 검색어가 포함된 게시물이 검색되어야 한다.
Assumptions	-

4.2 Use-case Diagram

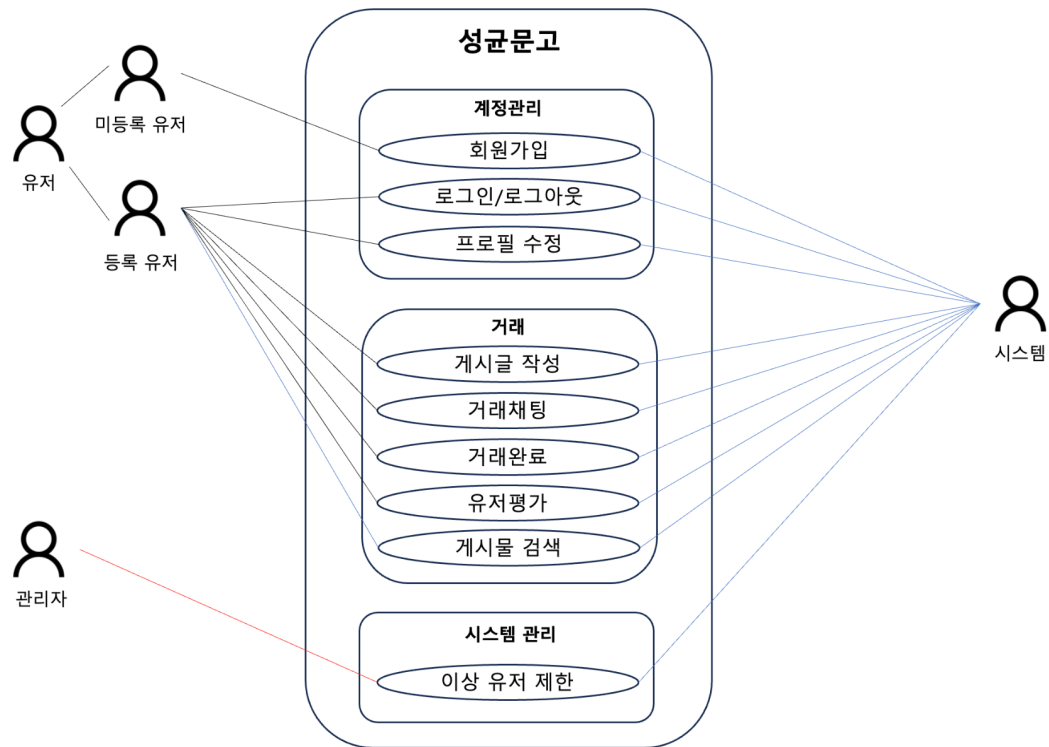


Figure 8 Use Case diagram

4.3 Data Flow Diagram & Dictionary

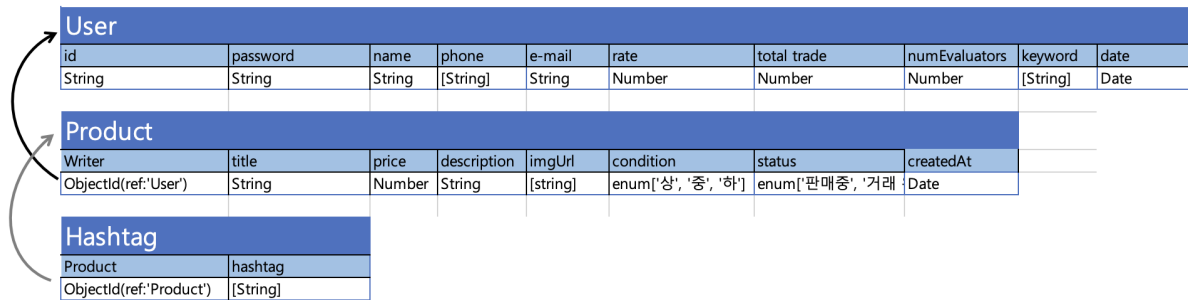


Figure 9 Data diagram

Table 15 - Data dictionary(User)

Name	Type	Constraint	Description
id	String	NOT NULL	회원의 아이디
password	String	NOT NULL	회원의 비밀번호
name	String	NOT NULL	회원의 이름
phone	String		회원의 전화번호
e-mail	String	NOT NULL	회원의 이메일
rate	Number		회원의 평점
totalTrade	Number		총 거래수
numEvaluators	Number		회원을 평가한 사람의 수
keyword	String		회원이 설정한 즐겨찾기 키워드
date	Date	NOT NULL	회원가입한 날짜

Table 16 - Data dictionary(Product)

Name	Type	Constraint	Description
Writer	ObjectId (ref: User)	NOT NULL	작성자 아이디
title	String	NOT NULL	게시글 제목
price	Number	NOT NULL	중고서적 가격
description	String		게시글 설명
imgUrl	String	NOT NULL	중고서적 이미지 주소
condition	Enum[String]	NOT NULL	중고서적 상태(상, 중, 하)
status	Enum[String]	NOT NULL	판매 상태(판매중, 거래완료)
createdAt	Date	NOT NULL	게시글 생성일

Table 17 - Data dictionary(Hashtag)

Name	Type	Constraint	Description
Product	ObjectId (ref: Product)	NOT NULL	상품 아이디
hashtag	String		해시태그 내용

5. Other Nonfunctional Requirements

5.1 Product Requirements

5.1.1 Usability Requirements (1)

- 별도의 매뉴얼을 필수적으로 숙지하지 않더라도, 편리하게 사이트를 이용할 수 있도록, 직관적이며, 가시적인 UI 구성을 사용해야 한다.
- 페이지 내 텍스트는 특정 기능에 대한 설명을 명확하게 제공해야 한다. 또한 각 아이콘들은 서비스의 기능을 사용자에게 추상적으로 전달할 수 있어야 한다.
- 한 페이지 내에서 최소한의 기능을 제공한다. 단순함을 뛰어넘어, 기능적, 콘텐츠적 UI적으로 필요한 부분만을 제작하여 반영한다.
- 유저 행동에 대한 화면 전환 및 기능 전환이 발생할 경우, 반응 시간 동안 화면 전환 중이라는 특정 UI를 제공한다. 이를 통해서 사용자가 체감하는 화면 전환에 대한 지루함과, 답답함을 해소할 수 있다.
- 전체적인 사이트의 디자인은 간결해야 하며, 서비스 내 상품들의 가시성을 해치지 않아야 한다. 하얀색 바탕에, 보라색(9381FF), 하늘색(67BDE8) 계열을 기반으로 한 디자인을 진행한다.
- 시스템 개발자에게 연락할 수 있는 손쉬운 방법을 명시해야 한다.
- 언어는 한국어를 제공하며, 캠퍼스 및 전공 필터링을 통한 필요한 정보만을 제공받을 수 있다.

5.1.2 Usability Requirements (2)

- 채팅방 내 의사소통 전송 속도는 2sec을 넘지 말아야 하며, 사이트 내에서 유저 행동에 대한 반응은 3sec을 넘지 말아야 한다.
- 거래 서적 관련 페이지의 모든 반응 속도는 3sec을 넘지 말아야 한다. 또한 포스팅 기능 활용 시 신규 포스팅이, 페이지에 표현되는데 2sec을 넘지 말아야 한다.
- 특정 상태 변경이 발견될 경우, 화면 전환을 진행한다. 동일한 화면일 경우에, 화면 전환하지 않으며 이를 통해서, 비효율적인 활동을 방지한다.

5.1.3 Dependability Requirements

- 사용자는 특정 이메일을 통한 하나의 계정을 갖되, 유저가 ID와 Password를 정확히 입력하였을 경우에 상항 해당 사용자의 계정으로 접속할 수 있어야 하며, 정보를 제공해야 한다.
- 게시물 포스팅 시, 사용자가 사진을 업로드할 때 끊김이 없어야 한다.
- 서버의 과부하를 방지하기 위하여, 경로는 최소화하고, 반복적인 활동을 줄인다.
- 서버에 문제가 생기는 빈도를 최소화하기 위하여, 정기적으로 서버를 유지, 보수해야한다.
- 서비스가 제공될 당시 숨겨져 있거나, 비활성화되어 있거나, 문서화되지 않은 기능이 없어야 한다. 모든 기능이 명확하게 제시되어야 한다.
- 특정 기능을 추가하거나 유지 보수 이후, 다음과 같은 여부를 확인해야 한다.
 1. 서비스에 충돌 및 버그가 있는지 테스트 한다.
 2. 모든 서비스 정보와 메타 데이터가 완전하고 정확한지 확인한다.
 3. Efficiency Requirements 기준을 적절히 반영하고 있는지 확인한다.
- 특정 유저 간의 거래가 종료되었을 경우, 더이상 채팅을 보낼 수 없어야 한다.

5.1.4 Security Requirements

- 사용자는 회원가입 시 성균관대학교 메일(skku.edu 또는 g.skku.edu)로 한정하여, 성균관대학교 학생임을 인증을 받아야 하며, 성균관대학교 학생임이 인증된 사용자만이 사이트 사용이 허용된다.
- 중고 서적 거리를 진행하는데 있어서, 부적절한 정보나, 사실과 다른 거짓 거래 게시와 같은 부적절한 행위를 적발할 경우에, 이를 신고 및 차단할 수 있는 기능을 제공한다. 관리자가 해당 신고를 확인하여, 부적절한 행위로 인한 신고가 접수되었을 경우, 벌점을 1점씩 부여, 벌점이 3점 이상이 될 경우, 관리자는 해당 유저를 회원 탈퇴 시킨다.
- 서적 관련 콘텐츠가 아닌, 모욕적이거나, 의도적으로 혐오감을 제공하거나, 저급한 내용이 포함되어있을 경우, 이를 신고 및 차단할 수 있는 기능을 제공한다. 신고, 차단만이 아닌, 정기적인 콘텐츠 피드백을 통하여 필터링한다. 적발될 경우, 즉각적으로 회원 탈퇴시킨다. 부적절한 콘텐츠의 기준은 다음과 같다.

1. 게시물에 정치적, 종교, 인종, 성적 지향성, 성별과 같은 기타 특정 범주에 대한 언급 또는 설명을 포함할 경우, 명예 훼손의 소지가 있거나, 악의적인 목적이 분명한 내용이 포함되어 있을 경우
 2. 게시물에 특정 교수 및, 수업에 대한 비하와 같이 특정 단독 대상을 비난하는 내용이 포함되어 있을 경우
 3. 서적과 관련되지 않은 정보를 포함할 경우, 중고 서적 거래와 벗어난 문장을 의도적으로 포함하였을 경우
 4. 편법을 위해 메타데이터에 상표 등록된 용어, 인기 있는 앱 이름, 가격 정보나 기타 관련 없는 어구를 포함했을 경우
- 사용자가 회원 가입한 이후, 사용자의 개인정보는 안전하게 보호되어야 한다. 비밀번호의 경우 암호화하여 관리자도 접근할 수 없어야 한다.
 - DB 연동에 사용되는 DB URI는 배포물들과 함께 공유되지 않는다. 관리자 내 따로 제공되며, 이를 통해서 서비스 내 정보들을 안전하게 접근하고 관리한다.

5.2 Organizational Requirements

5.2.1 Environmental Requirements

- 해당 사이트는 React v18.0.0, typescript v5.0.0, Node-express v4.18.2, MongoDB v6.1.0을 기반으로 개발되며, Web browser(Chrome v80, Edge v12, Firefox v34 등) 이상의 버전에서 사용할 수 있도록 개발되어야 한다.
- 위의 웹 브라우저를 설치할 수 있을 만한 hardware 사양(64bit CPU, 512MB of RAM, 200MB of drive space)을 가지고 있어야 한다.
- 반응형 웹앱을 통한 모바일 및 다양한 OS 환경에서 접속하더라도 항상 같은 정보와 기능을 제공해야 한다.

5.2.2 Development Requirements

- Window 10, Linux 18.04 이상의 버전의 환경에서 개발을 진행하도록 한다.
- 웹 UI/UX 의 전체적인 디자인은 Figma 통해 진행하여야 하며 아이콘, 텍스트를 직관적으로 표기한다.
- React 를 사용해 해당 사이트의 Front End 개발을 진행하도록 한다.
- Node.js와 Express를 사용해 해당 사이트의 Back End 개발을 진행하도록 한다.
- Non-SQL 기반 MongoDB를 사용하여 개발을 진행하도록 한다.
- Docker 컨테이너 활용을 통하여 항상 동일한 보장된 환경 내에서 개발을 진행하도록 한다.
- 데이터베이스는 ID, password, 프로필 등의 사용자 정보와 거래되는 중고 서적 상품을 관리하며 각 데이터들은 사용자의 요청이 있지 않는 한 변경되지 않아야 한다. 또한 데이터 무결성이 항상 충족해야 한다.

5.3 External Requirements

5.3.1 Regulatory Requirements

- 저작권 보호법: 시스템은 모든 콘텐츠 및 자산의 저작권 보호법을 엄격히 준수해야 한다. 시스템에 포함된 모든 콘텐츠는 원작자의 동의 없이 사용되어서는 안 되며, 원작자의 요청 시 해당 콘텐츠를 삭제해야 한다. 저작권 보호법을 위반하지 않도록 사용자가 업로드한 콘텐츠에 대한 정책 및 절차를 마련하고 이를 시행해야 한다.
- 개인정보 보호법: 개인정보 보호법 및 규정을 준수하여 개인정보 수집, 저장 및 처리에 대한 모든 요구 사항을 준수해야 한다. 사용자의 개인정보는 안전하게 보호되어야 하며, 사용자 동의를 받은 경우에만 처리가 가능하다. 사용자의 개인정보에 대한 업무 규정 및 안전 조치를 마련하고 이를 시행해야 한다.
- 오픈소스 프로그램 보호 : 본 시스템은 오픈소스 라이선스를 존중하며, 오픈소스 프로그램을 사용할 때 해당 라이선스의 조건을 엄격히 준수해야 한다. 오픈소스 소프트웨어의 사용, 수정, 배포 및 공개에 대한 모든 규정을 준수해야 하며, 모든 오픈소스 라이선스에 대한 규정 준수 문서를 작성하고 유지해야 한다.
- 업계 규정(business regularoty) : 시스템이 해당 업계의 특정 규정 및 요구 사항을 준수해야 한다.
- 접근 제어 및 보안 : 시스템은 사용자 및 관리자의 권한을 관리하고, 무단 접근으로부터 보호해야 한다. 이에 따라 액세스 제어, 데이터 보안, 인증 및 권한 관리를 구현하고 보안 규정을 준수가 필수적이다.
- 보고,감사: 시스템 활동을 모니터링하고, 이를 감사 및 보고해야 한다. .
- 비상 대응: 비상 대응 계획과 재해 복구 계획을 마련하고 시행해야 합니다. 항상 백업 서버를 구축하며, 이를 통해서 재빠른 대응과, 손실을 방지한다.

5.3.2 Ethical Requirements

- 해당 사이트를 운영함에 따라, 부적절한 방법을 통한 금전적 이득을 취하면 안된다.
- 시스템 사용자가 제공되는 서비스를 통하여, 부적절한 금전적 이익을 취하면 안된다.
- 시스템 사용자는 서비스를 가입한 사용자들의 DB를 부적절하게 접근하여 이를 통한 부적절한 금전적 이익을 취하면 안된다.
- 시스템 사용자는 모두에게 동등한 서비스 기회를 부여하여, 누구나 서비스에 대한 동등한 수혜를 받도록 돕는다.
- 시스템 사용자는 개발 의도에서 벗어나, 악용되는 여지를 차단해야 한다.

5.3.3 Legislative Requirements

- 게시물에 대한 공개범위는 서비스 사용자로 한정한다. 성균관대학교 인증을 통해 가입된 서비스 사용자만이 게시물 정보에 접근할 수 있으며, 이를 통한 서비스 사용이 가능하다.
- 사용자들과 소통에 있어서 직접적인 개인 정보를 활용하지 않는다. 사용자 인증 이후, 특정 계정을 이용하여, 서비스 내 타인들과 소통할 수 있다.
- 사용자가 포스팅한 게시물은 모두 공개범위로 올라가게 되며, 사용자가 원할 경우 즉각적으로, 삭제 가능하다.
- S3 스토리지와, MongoDB를 활용한 자료 저장을 진행한다. 비밀번호와 같은 개인 정보 접근에 필요한 자료들은 암호화 하며, 이외의 공개적인 자료들은 암호화하지 않는다.
- S3 스토리지와 MongoDB 는 한 달 간격으로 백업을 진행하여, 혹시나 모를 재난이나, 외부 침입으로부터 대비한다.

5.4 Organizing the Specific Requirements

5.4.1 Context Model

Context Model을 통해서 시스템 내 프로세스의 관계를 파악한다

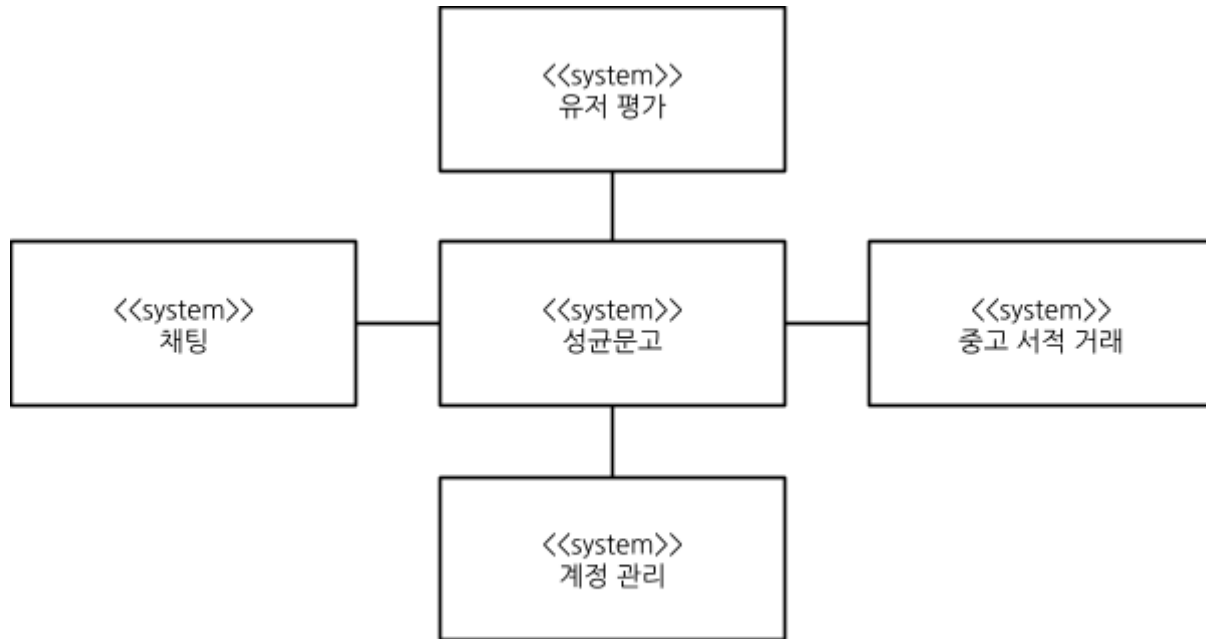


Figure 10 Context Model

5.4.2 Activity Diagram

Activity Diagram을 통해서 activity와 data의 흐름을 단계별로 이해한다

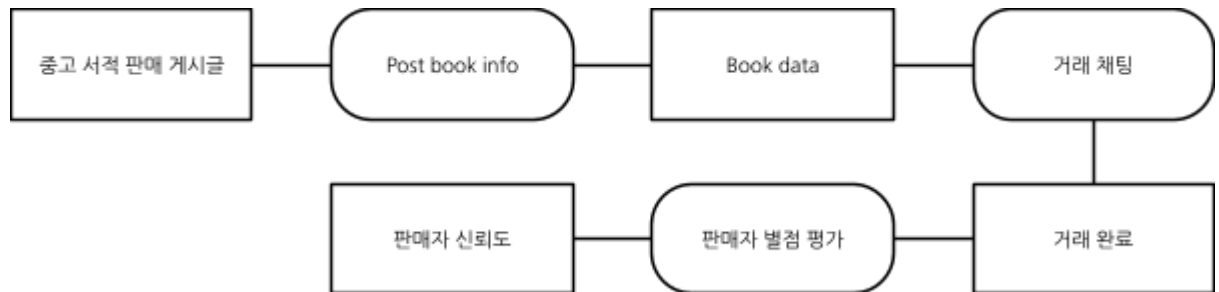


Figure 11 Activity diagram

5.4.3 Sequence Diagram

Sequence diagram을 통해서 사용자와 object의 관계를 나타낸다

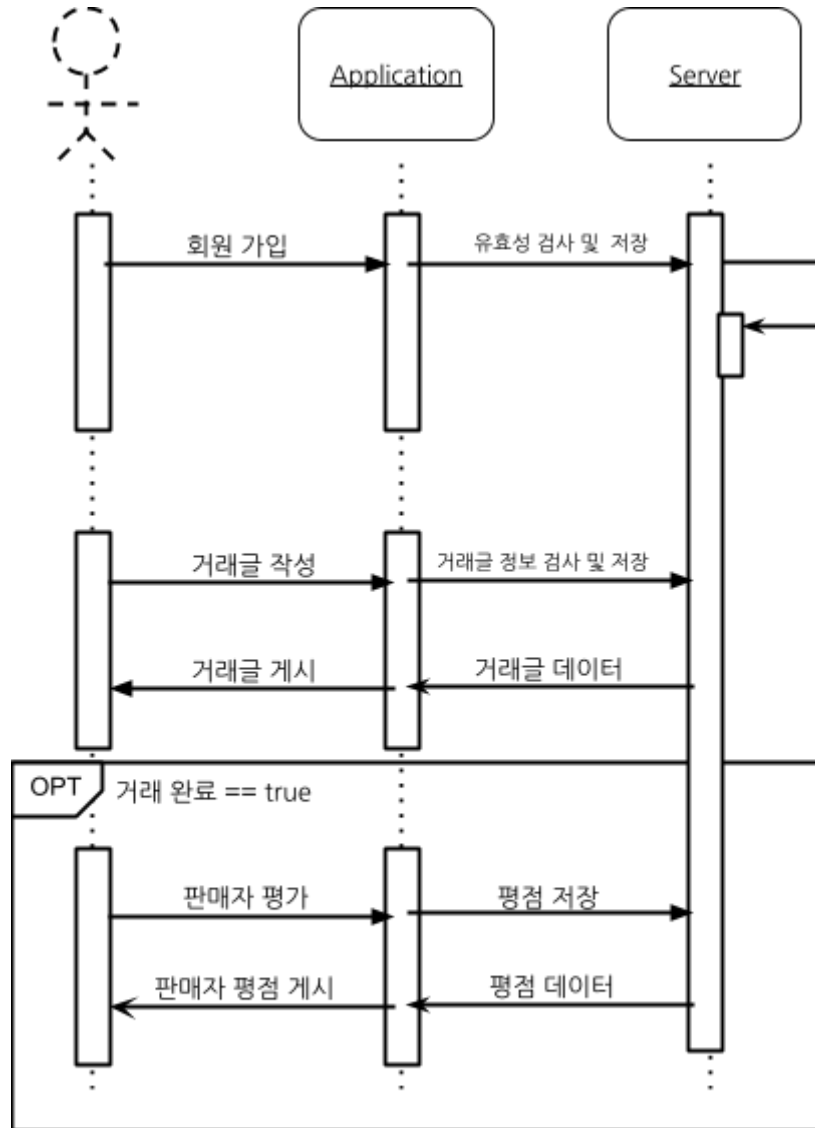


Figure 12 Sequence diagram

5.5 Design Constraints

- 모바일(안드로이드 및 IOS) 와 MacOS, Window, Linux 운영체제를 사용하는 Web browser에서 접속이 가능해야 한다. 이를 위해 웹앱의 형태로 제공되도록 한다.
- HTTP 통신이 가능하도록 설계되어야 한다.
- FE 개발을 위해서 Next.JS (React) 13.5.4 와 Node.js 20 이상 을 필요로 한다.
- BE 개발을 위해서 MongoDB 6.1.0 , Mongoose 7.5.2 Express 4.18.2를 필요로 하며, Swagger 6.2.8을 필요로 한다.
- 프로그램 Git 관리 시스템으로 Github을 사용한다.
- 시스템 개발을 위해 Windows, macOS 혹은 Linux기반의 OS 모두 가능하며 64bit CPU 와 최소 512MB이상의 RAM 이 필요하다.
- 접속 브라우저 환경은 구글 크롬 브라우저에 최적화 되어있으며 애플 Safari, Firefox 브라우저에서도 작동이 가능하다.
- 반응형 웹 디자인을 적용하기 위해 디자인 프레임워크를 Figma를 사용한다.

6. System Features

6.1 System Architecture

해당 장에서는 MVC패턴에 기반하여 시스템의 구조를 설명한다.

MVC는 애플리케이션의 데이터를 처리하는 모델 (Model) , 사용자에게 보여지는 사용자 인터페이스를 담당하는 뷰(View), 모델과 뷰 사이에서 흐름을 제어하는 인터페이스를 담당하는 컨트롤러(Controller)로 분리하는 설계방식이다.

모델은 컨트롤러에서 호출을 통해 데이터베이스에 저장된 데이터를 저장, 수정, 삭제, 가져오기 등의 데이터 비즈니스 로직을 처리한다. 이후에 뷰는 컨트롤러를 통해 반환된 모델 반환값을 가지고 화면에 출력하여 사용자 인터페이스를 구성한다.

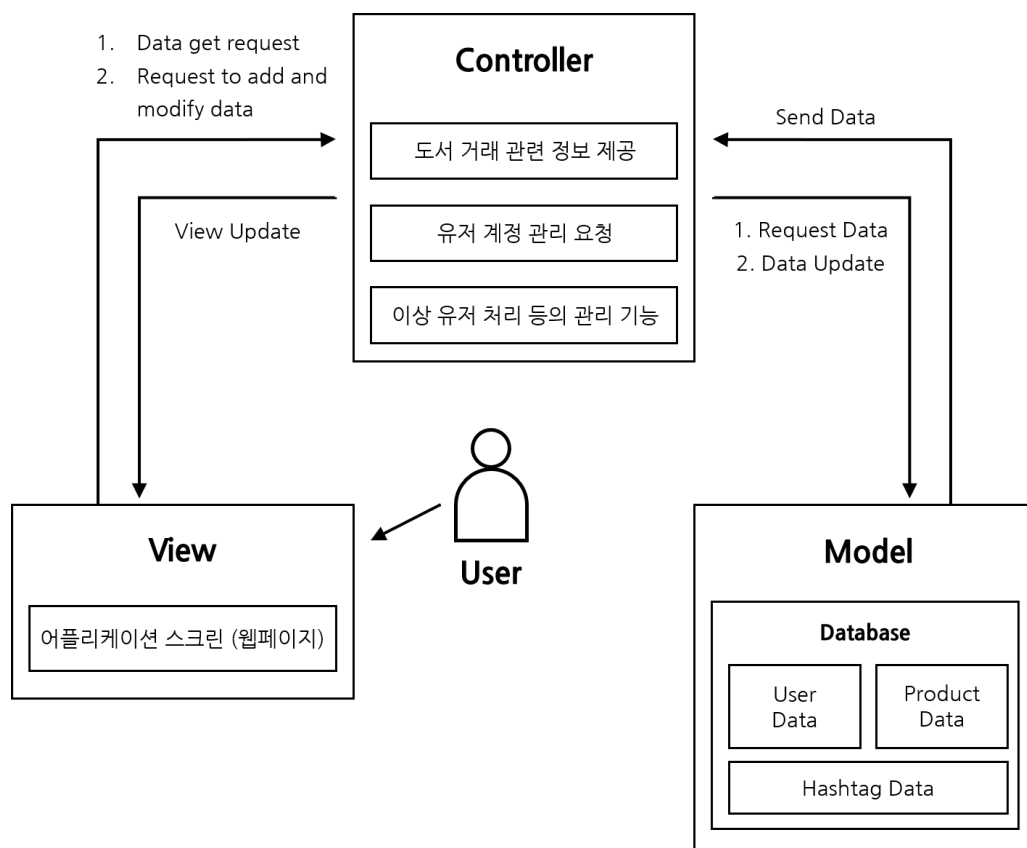


Figure 13 System Architecture

6.2 System Evolution

6.2.1 Limitation and Assumption

판매자의 평점이 소수의 구매자에게 달려있을 가능성이 높다. 교내 중고 거래 서적의 특성 상 판매자로서 거래 횟수가 많지 않을 것이다. 만약 한 두명의 구매자로 인해 판매자의 평점이 결정된다면 데이터의 가치가 굉장히 떨어진다는 뚜렷한 한계점이 있다.

6.2.2 Anticipated Changes

하드웨어의 성능이 나날이 발전함에 따라 시스템을 구동하는 하드웨어의 최소 사양이 증가할 수 있다. 또한, Text book의 전자책(e-book) 지원이 활발히 되거나 pdf file을 조금 더 쉽게 다운 받을 수 있다면 비용과 휴대성 측면에서 학생들의 중고 서적에 대한 수요가 현저히 줄어 들 수 있다. 그러므로 때에 따른 학생들의 요구에 적합한 기능을 추가적으로 제공할 필요가 있다.