

# <Comparewise >

-Requirements Specification-



Student Number	Name
2015313266	신한솔
2015312305	박원호
2016311821	한승하
2017314628	오승진
2017314738	나정우
2017312582	신정원

# Contents

1. Preface.....	6
1.1 Objective.....	6
1.2 Readership .....	6
A. User Requirements .....	6
B. System Requirements .....	6
1.3 Documents Contents .....	7
A. Preface .....	7
B. Introduction .....	7
C. Glossary .....	7
D. User Requirement Definition .....	7
E. System Architecture .....	7
F. System Requirements Specification.....	8
G. System Models .....	8
H. System Evolution .....	8
I. Appendices.....	8
J. Index .....	8
K. Reference .....	8
2. Introduction .....	9
2.1 Background .....	9
2.2 System Overview.....	13
2.3 Expected Benefits .....	14
3. Glossary .....	15
4. User Requirements Definition .....	17
4.1 Functional Requirements .....	17
A. Sign up / Login .....	17

## Requirements Specification

B. Search.....	18
C. Push Notification .....	18
D. My Page.....	19
E. Compare page.....	19
F. Item ranking .....	19
G. Item detail.....	20
4.2. Non-functional Requirements .....	20
A. Product requirements.....	20
B. Organization requirements .....	21
C. External requirements.....	22
5. System Architecture .....	22
5.1 Objective.....	22
5.2 Overall System Architecture .....	23
5.3 Sign up / Login System.....	24
5.4 Compare System.....	25
5.5 My page System .....	26
5.6 Search System.....	27
6. System Requirements Specification.....	28
6.1 Functional Requirements .....	28
A. Sign up/Login.....	28
B. Search.....	29
C. Push Notification .....	30
D. Compare .....	30
E. My page .....	31
F. Ranking .....	32
G. Item Detail .....	32
6.2. Functional Requirements ( Back End ) .....	33

## Requirements Specification

Site Collecting System .....	33
6.3 Non-functional Requirements .....	34
A. Product requirements.....	34
B. Organization requirements .....	35
C. External requirements.....	35
6.4 Scenario Examples .....	36
A. Compare .....	36
7. System Models.....	37
7.1 Objective.....	37
7.2 Context model .....	37
A. Process Diagram .....	37
7.3 Interaction model .....	38
A. Use Case Diagram.....	38
B. Tabular Description of Use case Diagram .....	39
C. Behavioral Models.....	43
A. Ranking System .....	43
B. Compare System.....	43
8. System Evolution .....	44
8.1. Multiple Comparison Feature .....	44
8.2. Voice/Image Search Technique .....	44
8.3. Customer Review Verification.....	45
8.4. OAuth implementation.....	45
9. Appendices.....	46
A. Hardware Requirements.....	46
B. Database Requirements.....	46
1) User .....	46
2) Item .....	47

## Requirements Specification

3) Wish List.....	47
4) Review .....	47
10. Index .....	48
10.1 Tables.....	48
10.2 Figures.....	49
10.3 Diagrams .....	49
11. Reference .....	50

# 1. Preface

## 1.1 Objective

본 Preface에서는 이 문서의 예상 독자를 설정하며, 문서의 전체적인 구성 및 각 Subtitle에서 서술하고 있는 세부사항들에 대한 간략한 개요를 제공한다.

## 1.2 Readership

본 Readership 파트에서는 해당 문서의 독자를 User Requirements, System Requirements 두 가지 기준에 따라 제시한다. 각 기준에 맞는 서로 다른 예상 독자층을 선정하여 작성함으로써, 이후 해당 문서를 이용하게 되는 독자층에 따라 해당 System에 대한 전반적인, 그리고 상세한 알맞은 정보를 제공하는 것을 목표로 한다.

### A. User Requirements

User Requirements은 해당 System의 사용자를 독자로 하여, 예상 독자층이 해당 문서를 쉽게 이해하는 것을 목표로 기술한다. 따라서 자연어로 작성을 기본으로 하며, 해당 독자층의 Back ground를 고려하여 서술한다. 또한 도표, 이미지 등의 시각자료를 활용한다.

### B. System Requirements

System Requirements는 개발 과정에 참여하는 Engineer들과, 해당 System을 계약할 Client, 등을 예상 독자로 선정한다. 따라서 계약서에 사용되어 질 수 있도록 최대한 상세하게 기술하며, System의 기능 및 제약사항을 개발 현장에서 쓰여지는 용어들로 기술한다.

## 1.3 Documents Contents

본 챕터에서는 해당 문서의 전체적인 Structure과 각 Chapter에서 기술하고 있는 내용의 요약본을 제시한다.

### A. Preface

해당 문서의 예상 독자를 선정하고, 그에 따른 서술 방식을 제시한다. 또한 해당 문서의 전체적인 구조와 각 Chapter에 대한 요약을 기술한다.

### B. Introduction

본 System의 개발 배경과, 목표에 대한 근거 및 주장을 기술한다. 개발될 System의 차별화되는 Service에 대해 서술하며, 해당 System이 사용됨에 따라 기대되어지는 변화들을 제시하고, 기대효과를 성취하게 될 과정에 대해 설명한다.

### C. Glossary

본 문서에서 사용되어질 기술용어들에 대한 정의들을 제시한다. 다양한 Background를 가진 다양한 독자들이 본 문서를 읽는데 불편함이 없도록 하는 것을 목표로 하며, 가능한 상세한 정의들을 제시, 또한 가능한 제시된 모든 용어에 대한 정의들의 들을 수록하는 것을 목표로 한다.

### D. User Requirement Definition

개발되어질 System에 대한 Functional, Non-Functional Requirements들을 실제 System 사용자의 입장에서 설명한다. System 사용자를 예상 독자로 선정함에 따라, 자연어 기술과 다양한 시각자료를 이용하여 서술한다.

### E. System Architecture

해당 System에 대한 전체적인 구조를 서술한다. 해당 문서에서 목표로 하고 있는 System의 세부 Function들이 각 Subsystem에 분배되어질 Outline을 제시한다.

### F. System Requirements Specification

해당 파트는 예상 독자층을 System Requirements들의 예상 독자인 개발에 참여하는 Engineer들과 계약을 진행할 Client를 목표로 기술한다. User Requirements에 대한 기술을 바탕으로 더 상세하고, 전문적인 Functional requirements, Non-functional requirements들을 제시한다. 해당 문서가 개발 과정에서 사용될 수 있도록 가능한 구체적인 내용을 제시한다.

### G. System Models

System의 각 Components들의 관계 및 System의 Domain와의 관계 등을 다이어그램으로 제시한다.

### H. System Evolution

System 개발상의 한계에 대하여 서술하고, 운영 단계에서 발생할 수 있는 예상된 변화를 기술하며, 그에 따른 대처방안을 제시한다.

### I. Appendices

본문에서 생략된 참고자료 등을 기술한다. 하드웨어, 데이터베이스 요구사항, 개발 환경 요구사항 등이 포함되어 있다.

### J. Index

본 문서에서 사용된 그림, 표, 다이어그램 등의 색인을 기술한다.

### K. Reference

본 문서에서 인용된 자료의 출처를 기록한다.



## 2. Introduction

해당 챕터에서는 System개발이 필요하게 된 Background와, System의 전반적인 Overview, 그리고 해당 System이 사용됨에 따라 기대되어지는 기대효과를 제시한다.

### 2.1 Background

소셜 커머스 어플리케이션 및 온라인 쇼핑시장은 날이 갈수록 거대해지고 있다. 또한 2020년 전 세계를 강타한 코로나 바이러스라는 초유의 사태는 한국 특유의 배달시스템과 더불어 온라인 시장 흥행의 원인이 되었다. 그 중 쿠팡과 같은 소셜 커머스 어플리케이션은 기존의 탄탄한 이용자층을 가지고 있는 페이스북, 인스타그램 보다도 많은 사용자를 보유하게 되었다. 또한 소셜 커머스 어플리케이션 이외에도 온라인 쇼핑 시장 전체가 코로나로 인한 상승세를 보이고 있다.

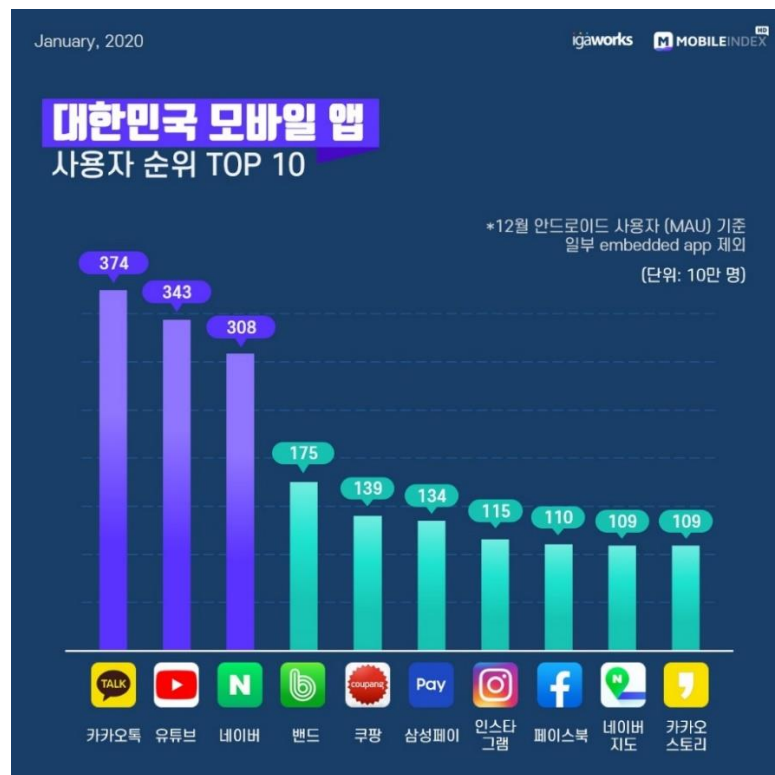


그림 1. 커머스 앱 “쿠팡”은 기존 강세였던 SNS 어플리케이션 (인스타그램, 페이스북) 보다도 많은 사용자수를 보유하고 있다.

온라인 쇼핑 매체 및 소셜 커머스 어플리케이션이 성장하고, 더 많은 판매업자들이 온라인 매체를 이용함에 따라, 온라인 유통망에 등록되는 상품 수 또한 기하급수적으로 증가하고 있다.

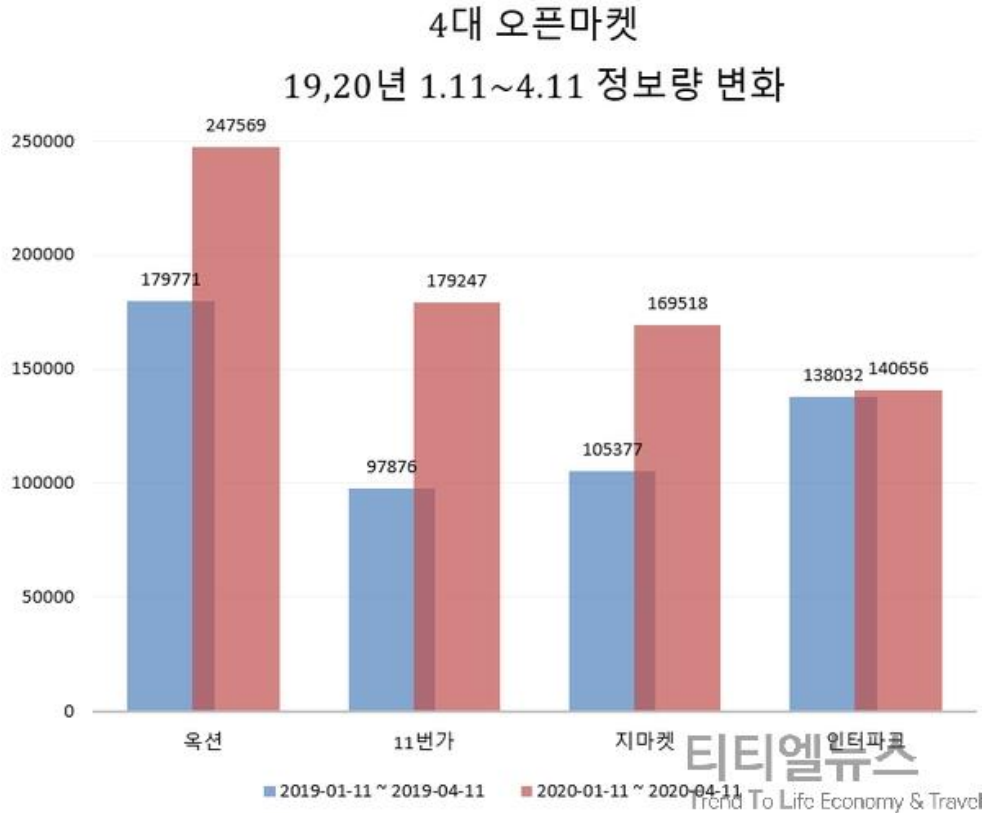


그림 2. 온라인 오픈마켓에 등록된 정보량은 빠른 속도로 증가하는 추세를 보이고 있다.

이러한 많은 정보량은 소비자들에게 더 많은 유용한 정보를 제공하기도 하지만, 지나치게 많은 정보의 제공으로 오히려 소비자들이 자신에게 필요한 상품을 결정하는데 방해요소가 되기도 한다. 실제로 [그림 3]에서 확인할 수 있듯 인기있는 소셜 커머스 어플리케이션인 “쿠팡”에서 대분류 검색어인 “태블릿”을 검색하였을 경우 2,768,776개의 상품이, [그림4]에서처럼 한가지 상품의 이름인 “갤럭시 탭 s6 lite”를 검색하였을 경우 총 10,419개의 상품이 검색되었다.

## Requirements Specification

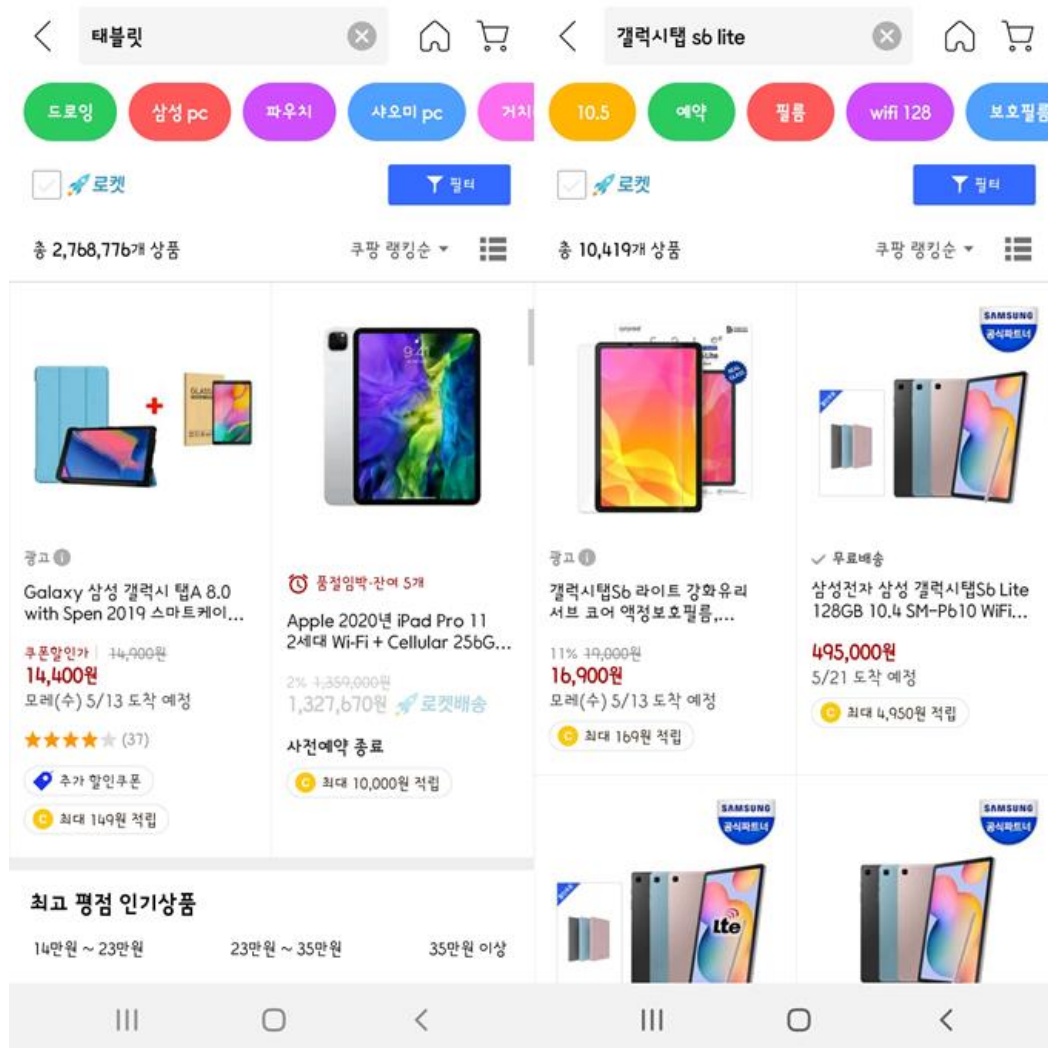



그림 3. 인기 소셜 커머스 어플리케이션 “쿠팡”에서의 “태블릿” 검색결과 (왼쪽), “갤럭시탭 s6 lite” 검색결과 (오른쪽)

또한 개별 상품에서의 소비자에게 주어지는 정보 또한 지나치게 방대해졌다. 배송 일자, 배송비, 할인 정보, 다양한 판매처, 평점, 제품 색상, 등 판매자가 확인하여야 하고, 비교하여야 하는 정보가 지나치게 많아 한가지 상품, 혹은 다양한 상품중에서 구매를 결정하기가 어려워지고 있다.



등록일: 2020.03 | 이미지출처: 제조사제공

**요약정보**    **가격비교** 74    **상세정보**    **상품의견** 81    **추천**

**쇼핑몰별 최저가** [더보기 >](#)

<b>G마켓</b>	474,990원	무료배송
<b>AUCTION.</b>	474,990원	무료배송
<b>전자랜드</b>	475,000원	무료배송
<b>G9</b>	495,000원	무료배송
<b>타임커머스 TMON</b>	495,000원	무료배송
<b>11번가</b>	495,000원	무료배송
<b>인터파크</b>	495,000원	무료배송
<b>위예프</b>	495,000원	무료배송
<b>K쇼핑</b>	506,250원	무료배송
<b>이온컴</b>	현금 495,000원 <a href="#">바로구매</a>	무료배송

[TOP](#)

**G마켓**    무료배송

**최저가 474,990원**    [최저가 구매하기](#)

태블릿PC / 안드로이드 10 / 26.42cm(10.4인치) / TFT-LCD / 2000x1200 / 224ppi / 5:3 / 멀티터치 / 엑시노스9 9611 / 옥타(8)코어 / 2.3GHz+1.7GHz / Mali G72 / 램:4GB / LPDDR4X / 내장:128GB / UFS2.1 / microSD:최대1TB / WiFi전용 / 802.11 a/b/g/n/ac / 블루투스v5.0 / WiFi 듀얼밴드 / GPS / 핫스팟 / 후면:800만화소 / 전면:500만화소 / HDR / 동영상촬영 / 오토포커스 / 스피커: 2개 / 3.5mm단자 / USB타입C / 전용펜 포함 / 전용키보드 미지원 / 7,040mAh / 사용시간:최대12시간 / 가로: 154.3mm / 세로: 244.5mm / 두께: 7.0mm / 465g / 출시가: 495,000원

★★★★★  
92점 (62)

[관심](#)    [공유](#)    [신고](#)

그림 4. 하나의 단일상품에서도 확인하여야 하는 정보가 지나치게 많다. (왼쪽), 하나의 단일상품에 대한 가격비교만으로도 70가지가 넘는 선택지가 존재한다. (오른쪽)

우리는 기존 Ranking 어플리케이션을 바탕으로하여, 소비자들에게 다양한 카테고리 안에서 판매량, 인기순위에 따른 제품군 랭킹을 제공하며, 소비자들에게 비교하고 싶은 제품사이의 더 쉬운 비교가 가능한 UI 및 Interface를 제공하여 소비자들의 구매 결정 및 제품비교를 돕는 어플리케이션 개발을 목표로 하였고, 이를 Comparewise 라고 부르기로 하였다.

## 2.2 System Overview

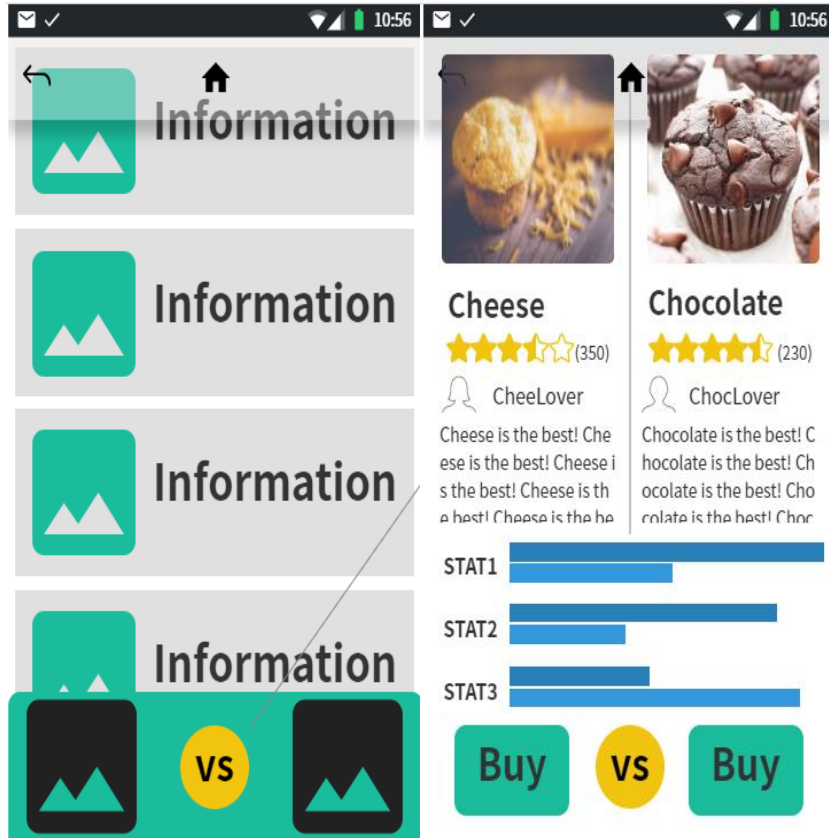


그림 5. Comparewise의 Information UI 초기버전 ( 왼쪽 )

그림 6. Comparewise의 Compare UI 초기버전 ( 오른쪽 )

Comparewise가 제공하고자하는 Compare UI는 위와 같다. Ranking System을 기본으로 하며, 소비자가 구매를 고민하는 제품들을 ( 2가지 제품사이의 비교를 초기값으로 하나, 여러 제품의 비교가 가능한 UI에 대해서도 논의중이다. ) 에 대한 상세 비교 Interface를 제공한다. [그림 8] 이러한 비교기능을 제공함으로써, 구매 고민중인 상품들 간의 상세 비교가 가능하고, 서로 다른 상품을 별도의 페이지 이동 없이 모든 혹은 선별적인 데이터의 비교가 가능하게 하는 것을 목표로 한다. Ranking system을 목표로 하여 제품의 판매는 직접적으로 이루어지지 않으며, 선택한 혹은 구매를 원하는 제품을 선정하고, 해당 제품을 판매하는 사이트로의 연결까지를 제공해주는 System을 구상하였다.

## 2.3 Expected Benefits

### A. 소비자들의 더 쉽고, 편리한 제품비교

Comparewise가 제안하는 UI는 소비자들의 더욱 편리한 제품 비교를 돕는다. 이러한 간편한 UI는 기존의 페이지를 이동해가며 하나하나 비교해야 하는 수고를 덜고, 더 빠르고 더 간편하게 원하는 상품간 비교가 가능하다. 하나하나 확인하기에는 지나치게 많은 데이터들을 한눈에 더 쉽게 비교가 가능하게 도움으로써, 서로 다른 제품 중 소비자에게 알맞은 제품을 선택하는 것을 돕고, 구매결정을 돕는다.

### B. 다양한 온라인 유통 시스템의 소비 촉진.

Comparewise는 직접적인 제품의 판매가 이루어지지않으며, 비 수익성 어플리케이션을 목표로 한다. 제품에 대한 비교 및 다양한 온라인 판매 사이트와의 연결을 목표로 함에 따라, 기존의 광고에 의해 몸집이 커진 대형 온라인 상권. 뿐만 아닌, 다양한 상권과의 연결을 통해 전반적인 온라인 상권의 소비 촉진제 역할을 할 수 있다.

### 3. Glossary

Term	Description
<b>Compare</b>	사용자가 추가한 상품들을 1 대 1 로 세부정보를 나열하여 비교해주는 page
<b>Login</b>	사용자가 application 을 개인 기록과 환경에 맞게 사용할 수 있게 해주는 function
<b>Search</b>	사용자가 구매하고자 하는 상품 혹은 가장 유사한 상품을 찾아주는 기능
<b>Ranking</b>	기간 별로 가장 많이 팔린 상품을 순위별로 보여주는 page
<b>Wish list</b>	사용자가 구매하기를 희망하는 상품을 모아놓은 page
<b>Rating</b>	사용자가 구매 이후 해당 상품에 대한 평가를 0~5 의 숫자 지표로 나타낸 것
<b>Review</b>	사용자가 구매 이후 해당 상품에 대한 평가를 글(자연어)로 나타낸 것
<b>Product Crawler</b>	여러 웹사이트에서 상품 판매에 대한 정보를 모으게 도와주는 module

<b>Database</b>	사용자 및 상품에 대한 정보를 포함하는 backend term
<b>API</b>	A set of higher-class functions and interfaces developed by advanced programmers that can be imported and use easily in different projects
<b>NLP</b>	Natural Language Processing

Table 1 Glossary



## 4. User Requirements Definition

### 4.1 Functional Requirements

#### A. Sign up / Login

Sign up/Login(이하, 회원가입 및 로그인 기능)은 본 서비스(app이름으로 대체)를 이용하기 위한 필수적인 기능적 요구사항이다. 서비스의 제공자는 이용자의 정보를 저장하고, 인증함으로써, 사용자 관리에 용이함을 도모한다. 서비스의 이용자는 데이터베이스에 저장된 정보들을 기반으로, 자신의 서비스 이용 활동을 관리할 수 있다. 로그인을 하기 위해서는, 우선적으로 데이터베이스에, 사용자 본인의 개인정보를 저장하고, unique한 ID와 암호화된 PW를 등록함으로써, 회원가입 절차를 이행해야 한다. 회원가입은 최초 1회만 가능하다. 회원가입 절차가 완료되면, 로그인 기능을 통해, 서비스에 접근할 수 있다. 등록된 자신의 ID와 PW를 통해, 데이터베이스에 등록된 사용자인지 검증을 한다. 검증이 된다면 서비스에 접근할 수 있는 권한이 생긴다.

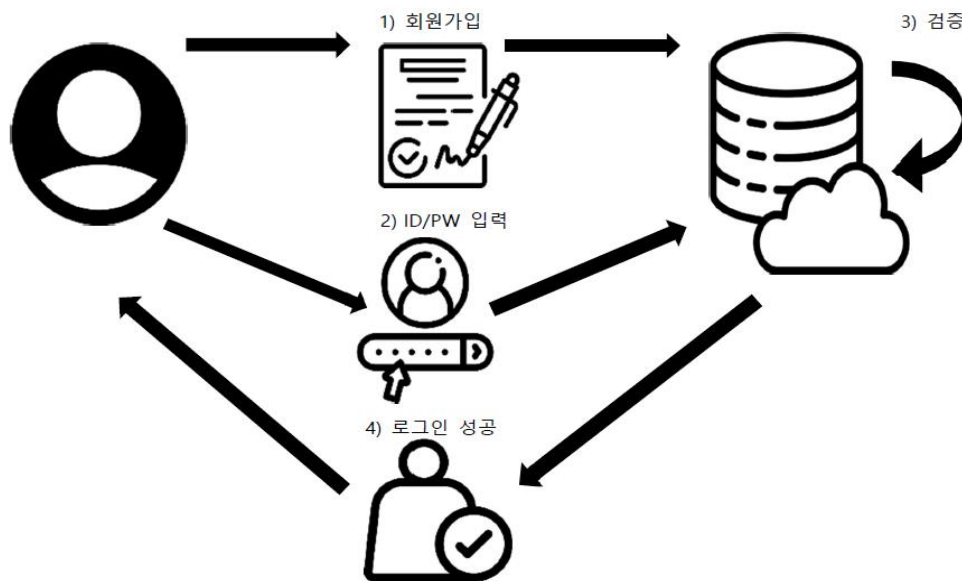


그림 7. Sign up & Login

## B. Search

**Search**(이하, 상품 검색 기능)기능은 사용자가 관심이 있거나, 구매 대상이 되는 상품을 여러가지 조건(이름, 키워드, 특정 스펙, 등)으로 검색할 수 있으며, 그에 대한 결과를 사용자 인터페이스를 통해 **Display** 한다. **Display** 된 상품은 여러가지 기준으로 정렬될 수 있으며, 해당 상품을 좀 더 자세히 보기 위해, 해당 상품을 클릭할 수 있다.

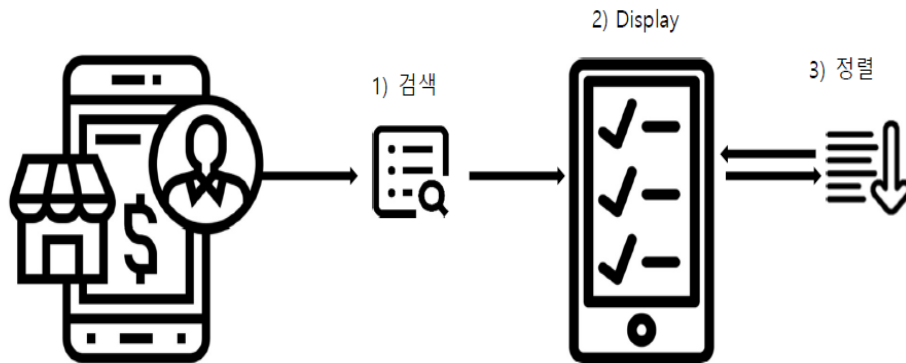


그림 8. Search

## C. Push Notification

사용자의 설정에 따라 Push Notification(이하 푸시 알림 기능)을 받을 수 있다. 사용자의 관심 카테고리, 할인 정보 혹은 제휴 상품 등 여러가지 조건의 업데이트시, 사용자는 푸시 알림을 통해, 더욱 더 쉽게 자신의 관심 항목에 접근할 수 있다. 푸시 알림 기능을 통해 사용자의 접근성을 도모하여, 제공자는 사용자의 더 많은 접근을 이용하여, 수익을 창출 할 수 있다.

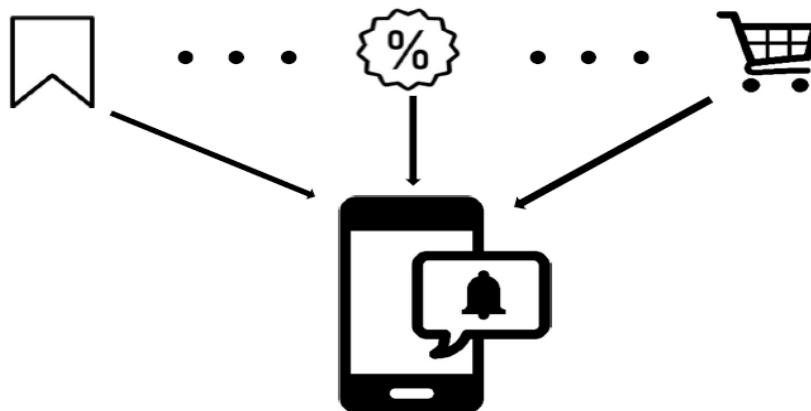


그림 9. Push Notification

#### D. My Page

사용자의 개인에 관련된 정보를 확인할 수 있는 페이지이다. 사용자 개인 정보부터 시작해서, **Wish list** 에 추가한 상품목록 및, **review**, 그리고 문의 내역, 어플리케이션 설정 등을 확인할 수 있다. 이 페이지는 사용자 고유의 것이며, 로그인을 한 사용자만 이용 가능하다.

#### E. Compare page

**Compare** 페이지는 **Ranking page**, **Wish list** 페이지에서부터 접근 가능하다. 사용자가 **Wish list** 에 넣은 상품 중 두 가지를 비교하여 살 수 있게 한다. 혹은 **Ranking Page** 에서 두가지 상품을 선택함으로써 접근할 수 있다. 사용자는 물품의 유형마다 정해진 각 항목별 세부정보를 쉽게 비교할 수 있으며, 전체적인 리뷰를 확인할 수 있다. 추가로 밑의 구매 버튼을 통해 판매 사이트로 바로 이동할 수 있게 한다.

#### F. Item ranking

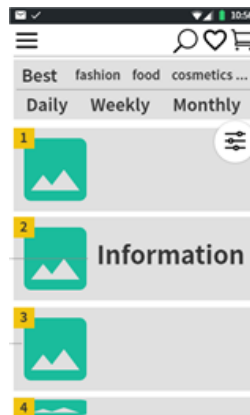


그림 10. Ranking ( App page )

Ranking은 데이터베이스에 기록된 상품들의 판매 횟수를 바탕으로 작성된다. 랭킹 정보를 통해 고객은 인기 있는 제품의 이미지, 간단한 세부정보, 평점을 한눈에 확인할 수 있다. Ranking은 전체 랭킹, 카테고리별 랭킹, 기간별 랭킹으로 세분화되어 제공된다. Ranking은 소비자가 원하는 정보를 쉽게 얻을 수 있도록 돕고, 판매자가 경쟁을 통해 제품 및 판매 품질을 향상시키는 동기를 갖게 한다.

## G. Item detail



그림 11. Item Detail ( App page )

Ranking, Search 기능의 상품 목록에서 선택된 상품의 세부 정보 및 판매 사이트 리스트를 표시한다. 세부 정보는 상품 이름 및 이미지, 평점, 정보, 가격, 상품이 포함된 카테고리 리스트와 세부 정보 등을 표시된다. 판매 사이트 리스트는 상품 판매 사이트를 웹 크롤링하여 리스트 형식으로 만들어 제공된다. 이를 통해 사용자는 아이템 관련 정보들을 비교하고자 하는 다른 아이템의 정보를 각 카테고리 별로 확인할 수 있고, 이용하고자 하는 판매사이트를 직접 선택할 수 있다.

## 4.2. Non-functional Requirements

### A. Product requirements

#### 1. Usability

본 서비스의 시스템은 모바일 디바이스에서 실행되는 어플리케이션이다. 그렇기에, 여러가지 플랫폼에서의 디스플레이, 운영체제 등과 같은 제약이 원활한 서비스 운용에 방해가 되지 않도록, 다양한 환경을 고려하고, 불편함을 느끼지 않고, 그에 상응하는 UI/UX 제공할 수 있어야한다.

#### 2. Efficiency requirement

1) 상품들의 정보가 업데이트 될 때, 데이터베이스에서 데이터를 로드 할 때, 소요되는 시간  
2) Compare 페이지와 같은 서비스 제공 화면의 요청에 대한 응답하는데 소요되는 시간

위와 같은 시간들이 일정 수준 이상이 되어, user와의 interact 에 방해가 되지 않도록 효율성을 추구해야한다.

### *3. Dependability*

사용자의 환경설정 및 정보가 제대로 데이터베이스에 저장되어있어, 서비스 제공의 범위가 개별 사용자의 의도와 상응하게 한정되어야 하고, 여러가지 기능들이 수행되는데, 오류가 발생해서는 안된다.

### *4. Security*

사용자의 개인정보는 오직 사용자에게 의해 등록/수정이 가능하며, 사용자의 요청이 있을 경우에만, 제공자가 해당 사용자의 정보를 수정할 수 있으며, 해당 정보들은 사용자 식별에 사용되며, 외부에 유출될 수 없도록, 데이터베이스의 보안성이 갖추어 져야한다.

## **B. Organization requirements**

### *1. Environmental requirement*

본 서비스는 모바일 디바이스에서 제공되며, 운영체제가 다르더라도, 제공되는 서비스의 제약이 크지 않아야 한다.

### *2. Operational requirement*

개발은 Node.js, React Native, Firebase 등을 이용하여 개발하며, 사용언어는 Java, JavaScript 등이다.

### *3. Development requirement*

Firebase Google Cloud Service 를 이용하여, 데이터베이스에 서비스 운용에 필요한 정보를 저장하며, 사용자의 정보는 레이아웃이 전환될 때, 기록되어 사용될 수 있다. 또한, React Native 를 이용하여, 모바일 디바이스 운영체제 간의 제약을 줄일 수 있도록 한다.

## C. External requirements

### *1. Regulatory/legislative requirement*

사용자의 개인정보는 제공자와 해당 사용자만 열람이 가능하며, 정보의 수정은 선택적으로 관리자가 할 수 있지만, 원칙적으로는 해당 사용자만 수정 권한이 있다. 중개서비스의 특성상, 사용자의 정보일부를 외부로 제공할 경우에는 사용자의 동의 하에, 제공할 수 있다.

### *2. Ethical requirement*

연령제한은 없다.

### *3. Safety/security requirement*

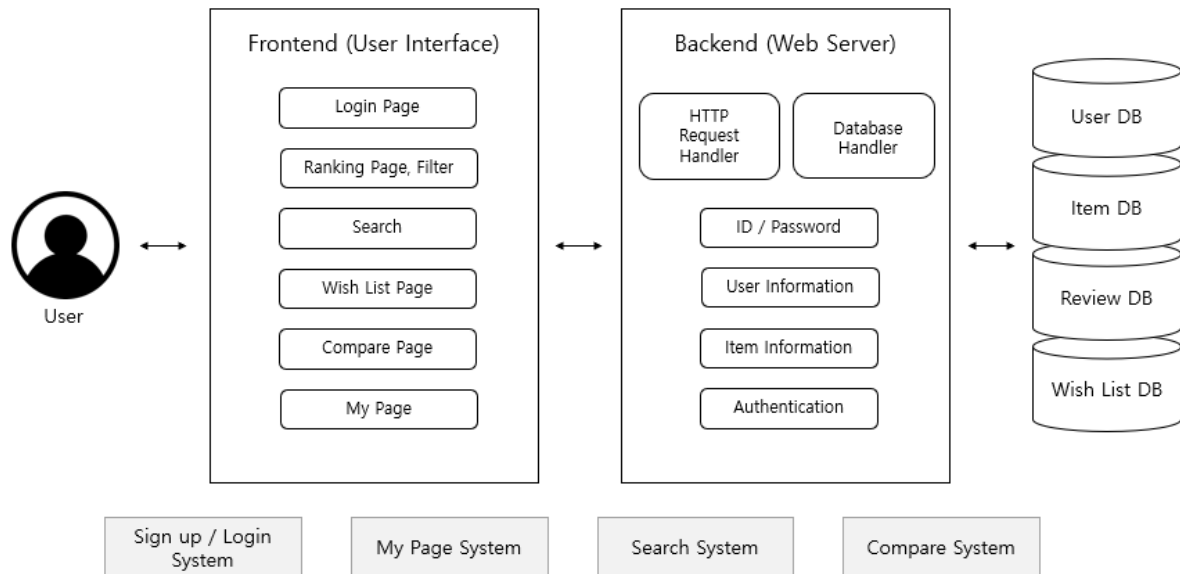
개인정보보호법에 따라 사용자의 개인정보를 외부/내부 적으로 안전하게 저장하고, 보호해야한다.

## 5. System Architecture

### 5.1 Objective

System Architecture에서는 시스템의 구조에 대해 설명한다. 전체적인 시스템의 구조와 서버 시스템에 대한 자세한 개요를 보여준다. 시각화 및 다이어그램을 이용해 이해를 돕는다.

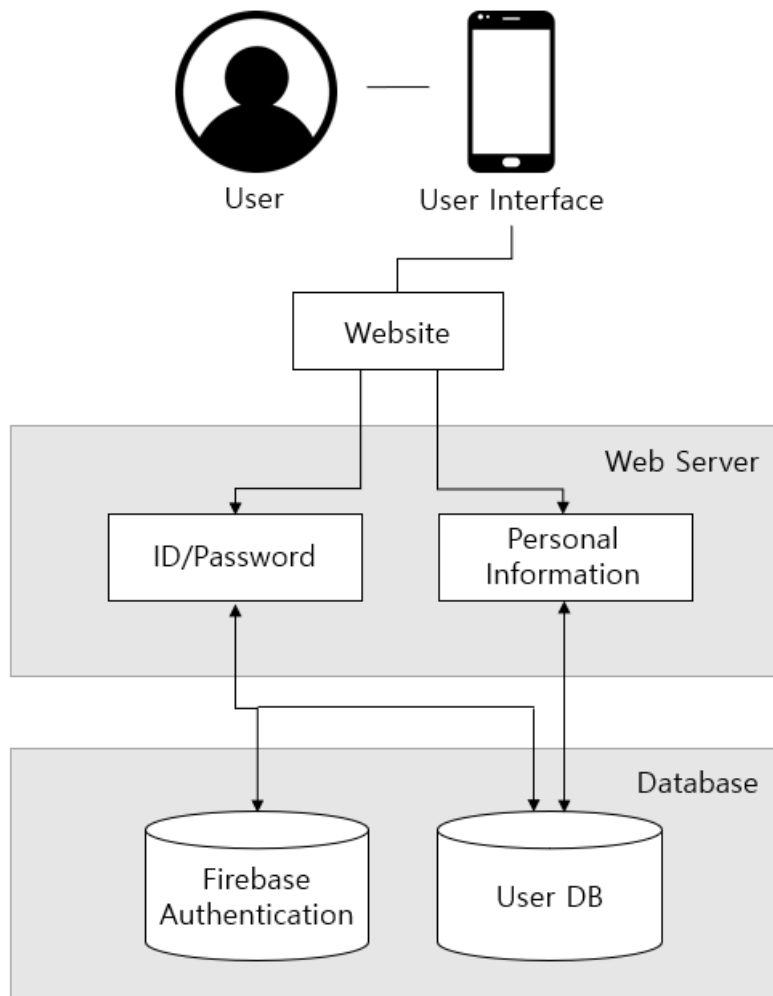
## 5.2 Overall System Architecture



**Diagram 1. Overall System Architecture**

Sign up/Login System, My Page System, Search System, Compare System 네 개의 시스템으로 전체적인 시스템이 구성된다. Frontend에서 어플리케이션이 제공하는 페이지로부터 사용자가 정보를 전달하면 Backend에서 Request와 Database Handler를 통해 이를 처리하고, 결과를 다시 Frontend로 보내 사용자에게 보여준다.

### 5.3 Sign up / Login System

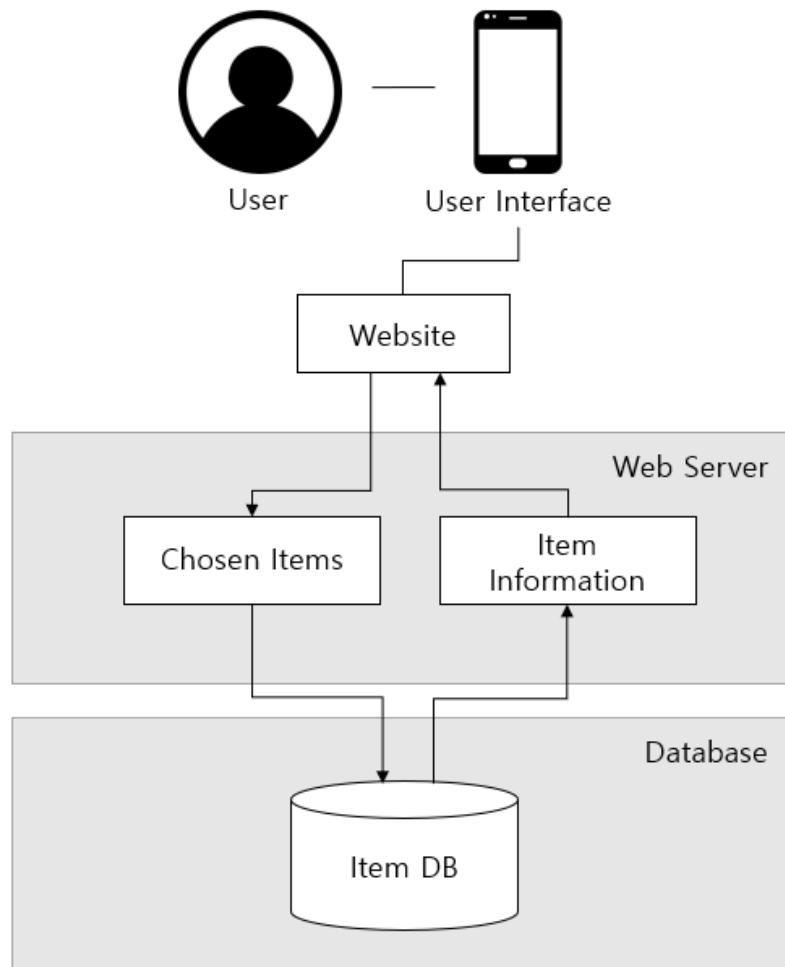


**Diagram 2. Sign up / Login System Architecture**

사용자의 회원가입부터, 로그인, 로그아웃을 지원하는 시스템이다. 회원가입은 사용자에게 정보를 입력 받아 DB에 가입 이력이 있으면 이를 알리고, 없으면 새로운 계정을 만들도록 지원하는 기능이다. 로그인도 사용자에게 받은 정보를 DB에 검색하여 ID와 Password가 모두 일치하는지 확인하는 기능이다. Firebase의 Authentication SDK를 통해 Google, Facebook, Twitter 등의 사이트와 연동하여 로그인이 가능하다.



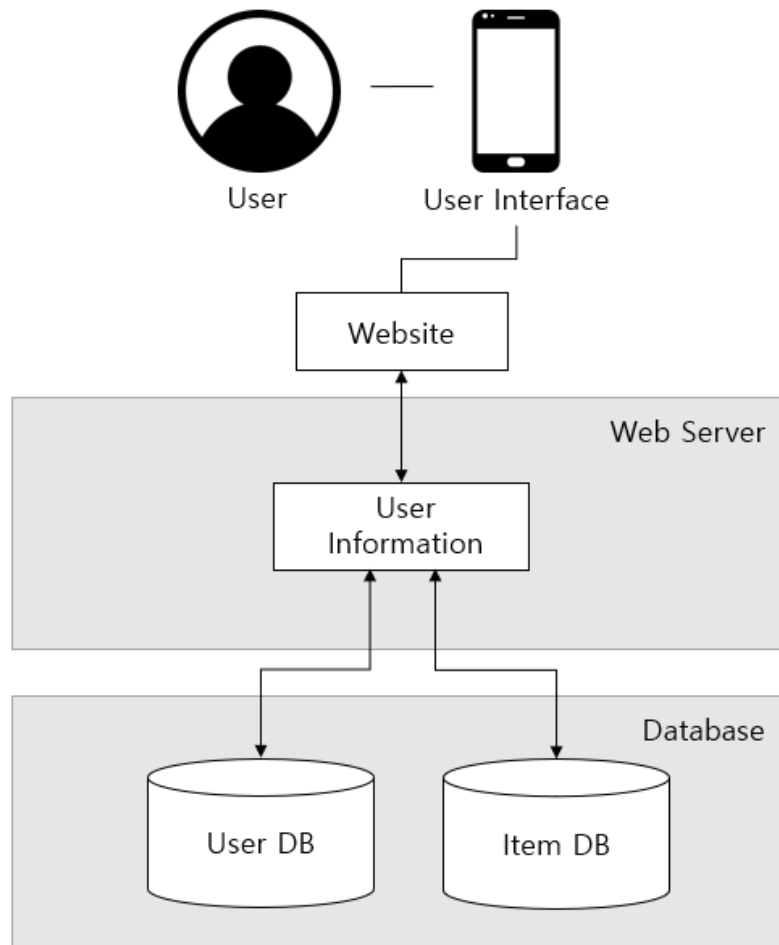
## 5.4 Compare System



**Diagram 3. Compare System Architecture**

사용자가 선택한 두 개의 아이템의 정보를 한 눈에 비교할 수 있도록 제공하는 시스템이다. 사용자가 두 아이템을 클릭하여 선택하면, 카테고리 별로 정해진 지표들에 대해 데이터베이스에서 두 아이템의 정보를 찾아 제공한다.

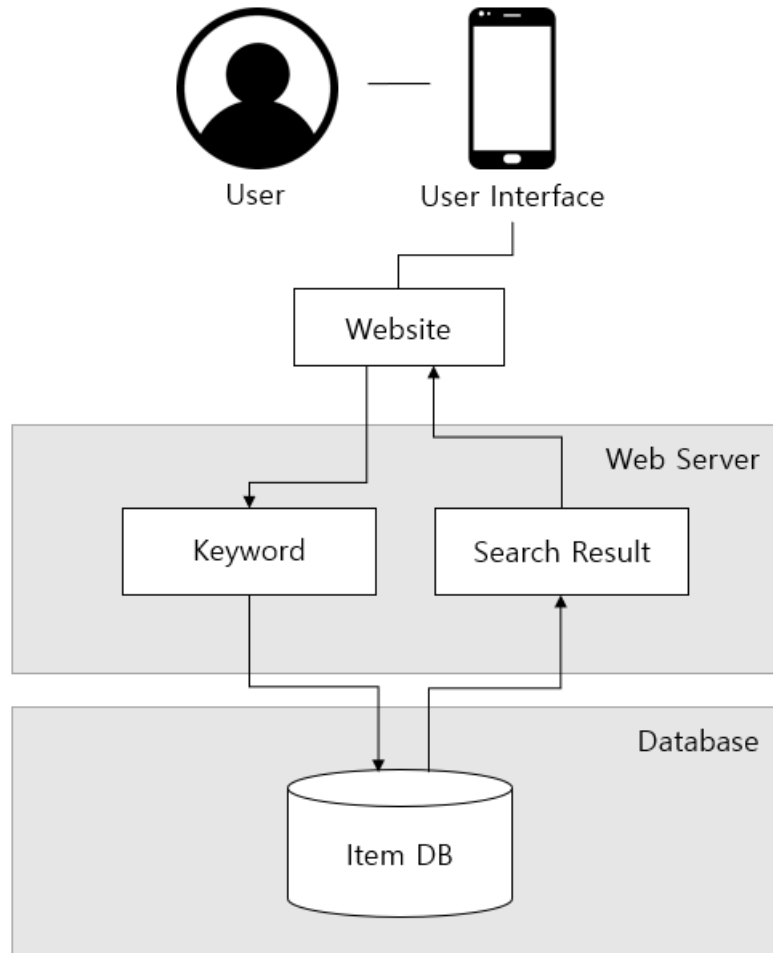
## 5.5 My page System



**Diagram 4. My Page System Architecture**

사용자의 개인 정보와 어플리케이션을 사용하며 유용한 기본적인 정보를 수정 및 확인할 수 있는 시스템이다. 사용자의 로그인 정보를 통해 데이터베이스에서 사용자의 개인 정보와 여러 아이템의 정보를 찾아 제공한다.

## 5.6 Search System



**Diagram 5. Search System Architecture**

사용자가 원하는 아이템을 검색할 수 있는 시스템이다. 검색하길 원하는 키워드를 입력하면 데이터베이스에서 그에 대한 정보를 찾아 검색 결과를 보여준다.

## 6. System Requirements Specification

### 6.1 Functional Requirements

#### A. Sign up/Login

**Table 2. Sign up**

Name	Sign up Function	
Description	서버의 Database에 User의 정보를 등록하여, 사용자로서 승인한다.	
Inputs	Google / SNS 계정	Google 계정 혹은 타 SNS 계정을 이용하여 Sign up을 수행할 수 있다.
	User Information	사용자의 개인정보 Class
Source (Input)	User input	
Outputs	서버의 DB에 사용자 인증 정보 등록	
Destination (Output)	Database	
Action	DB에 사용자에게 대한 인증 정보와 사용자 개인정보를 저장하고, 후에 Login Function에서 사용될 정보들을 관리한다.	
Requirements	-사용자의 인증에 대한 정보를 제공해줄 수 있는 타 서비스 계정 -사용자 데이터 관리에 필요한 개인정보	
Pre-condition	서버에 등록/승인되지 않은 사용자	
Post-condition	서비스를 이용할 수 있는, 서버에 등록/승인된 사용자	
Side effects	추후 Login Function을 통해, 사용자가 등록된 계정을 통해 서비스에 접근할 수 있다.	

#### 1) Sign up

#### 2) Login

Name	Login Function	
Description	Authenticate and authorize the user	
Inputs	Google / SNS 계정	Google 계정 혹은 타 SNS 계정으로 Login이 가능하며, 그 외의 방법으로도 사용자의 ID/PW를 통해 Login 가능

<b>Source (Input)</b>	User input
<b>Outputs</b>	사용자 인증 정보 토큰
<b>Destination (Output)</b>	Application UI
<b>Action</b>	DB서버에 등록된 계정들과 비교하여, 해당 계정정보가 존재한다면, 사용자 인증정보를 제공하여, 서비스에 접속할 수 있도록 한다.
<b>Requirements</b>	-사용자의 인증에 대한 정보를 제공해줄 수 있는 타 서비스 계정
<b>Pre-condition</b>	사용자는 회원 전용 서비스에 접근 불가
<b>Post-condition</b>	사용자는 회원 전용 서비스에 접근 가능
<b>Side effects</b>	Frontend(User Interface)가 회원 전용 서비스가 제공되는 Main Home으로 전환된다

Table 3. Login

## B. Search

<b>Name</b>	<b>Search Function</b>	
<b>Description</b>	사용자의 검색어와 조건에 따라 만족하는 상품들을 Display한다.	
<b>Inputs</b>	Keyword	검색어 String
	Condition	검색 Filtering에 필요한 조건 Class
<b>Source (Input)</b>	User input	
<b>Outputs</b>	검색 결과	
<b>Destination (Output)</b>	Application UI	
<b>Action</b>	사용자로부터 검색 Keyword를 입력 받아, 해당 조건에 만족하는 결과물을 UI를 통해 제공하며, 조건들을 통해 결과물을 Filtering한다.	
<b>Requirements</b>	-검색 Keyword -검색 Filtering Condition	
<b>Pre-condition</b>	검색 이전 Main Home UI	
<b>Post-condition</b>	검색 결과를 Display	
<b>Side effects</b>	검색 결과를 Display하는 UI로 전환되며, 사용자의 선택에 따라 상품을 나열할 수 있다.	

Table 4. Search

## C. Push Notification

Name	Push Notification Function	
Description	사용자의 환경설정에 따라 Push 알림 기능을 수행한다.	
Inputs	User Select	푸시 알림 기능 사용 여부 boolean
	User Information	사용자의 개인정보 Class
Source (Input)	User input	
Outputs	사용자 Application을 통한 Device 알림	
Destination (Output)	User Device	
Action	DB의 사용자 정보(북마크, 관심항목 등)을 통해, 해당 카테고리에 관한 알림 정보를 Device로 전송한다.	
Requirements	-사용자 알림 정보에 필요한 정보	
Pre-condition		
Post-condition		
Side effects	사용자 Device로 notification을 전송하여, 해당 알림을 선택하면, 해당 항목 화면을 UI를 전환한다.	

Table 5. Push Notification

## D. Compare

Name	Compare Function	
Description	탭-다운 형식 두 제품을 비교하는 페이지 제공	
Inputs	두 개의 아이템 ID	사용자가 비교하길 원하는 두가지 아이템의 ID (자료형 입력)
Source (Input)	기존 Wish list 에 존재하는 아이템 중 사용자가 두가지를 선택 후 비교 버튼을 클릭, Ranking 페이지에서 Compare페이지를 거쳐서 이동	
Outputs	두 상품을 비교해주는 페이지로 이동	
Destination (Output)	두 상품을 비교하는 사용자 사용 화면 (웹, 애플리케이션)	
Action	사용자가 즐겨 찾기에 존재하는 아이템 중에서 비교하고 싶은 두가지 아이템을 선택하여 드래그 해서 비교 버튼을 누르면 비교페이지로 이동하며, 두 가지 상품에 관련된 항목별 세부 정보와 리뷰를 제공한다.	

<b>Requirements</b>	두 가지 상품을 골라야 한다.
<b>Pre-condition</b>	- 두 상품은 모두 같은 카테고리에 속해야 한다.
<b>Post-condition</b>	- 주문 가능한 상품만 주문하기 버튼이 나타나야 한다.
<b>Side effects</b>	

Table 6. Compare

## E. My page

<b>Name</b>	<b>My page Function</b>	
<b>Description</b>	사용자의 고유한 정보를 표시하는 페이지	
<b>Inputs</b>	ID	사용자의 ID (string)
	Password	사용자의 Password (string)
<b>Source (Input)</b>	Login 이후 Mypage 클릭 시	
<b>Outputs</b>	로그인 된 사용자의 Mypage 화면으로 이동	
<b>Destination (Output)</b>	My page 화면 (웹, 애플리케이션)	
<b>Action</b>	사용자 교유의 페이지이다. 이 페이지를 통해서 개인 정보부터 시작 해서, Wish list에 추가한 상품목록 및 review, 애플리케이션 설정, 그리고 문의에 대한 내용을 확인할 수 있다.	
<b>Requirements</b>	사용자의 ID와 Password가 필요하다.	
<b>Pre-condition</b>	- 로그인하지 못하면 화면을 표시하지 않는다.	
<b>Post-condition</b>	- 로그인한 사용자에게만 표시된다.	
<b>Side effects</b>		

Table 7. My Page

## F. Ranking

<b>Name</b>	Ranking	
<b>Description</b>	전체 및 특정 랭킹 페이지를 보여준다	
<b>Inputs</b>	카테고리별 랭킹	기본적으로 제시된 카테고리 버튼 클릭 or 특정 카테고리 검색 및 설정(string)
	기간별 랭킹	Daily, Weekly, Monthly 버튼 클릭 or 특정 기간 설정(자료형 입력)
<b>Source (Input)</b>	랭킹 버튼 클릭 및 자료형 입력	
<b>Outputs</b>	업데이트된 랭킹 리스트 정보	
<b>Destination (Output)</b>	사용자 사용화면 (웹, 애플리케이션)	
<b>Action</b>	사용자가 랭킹 버튼을 클릭하면 선택한 기간 및 카테고리 정보가 백엔드로 전송된 후, 데이터베이스로부터 Ranking 정보를 얻는다. 이후 Ranking 정보는 사용자 사용화면(웹, 애플리케이션)으로 보내진다.	
<b>Requirements</b>	랭킹 카테고리 및 기간이 명확해야 한다.	
<b>Pre-condition</b>	-	
<b>Post-condition</b>	-	
<b>Side effects</b>	-	

Table 8. Ranking

## G. Item Detail

<b>Name</b>	Item detail	
<b>Description</b>	특정 상품의 세부 정보를 보여준다.	
<b>Inputs</b>	아이템 ID	사용자가 보고 싶은 아이템의 ID(자료형 입력)
<b>Source (Input)</b>	Ranking, Search 페이지에서 사용자가 특정 아이템을 선택한다.	
<b>Outputs</b>	특정 아이템의 세부 정보 목록	
<b>Destination (Output)</b>	사용자 사용화면(웹, 애플리케이션)	
<b>Action</b>	사용자가 Ranking, Search의 아이템 목록에서 특정 아이템을 클릭하거나 검색을 하면, 아이템 ID가 백엔드로 전송된다. 이후 데이터베이스로부터 얻은 특정 아이템 세부 정보 목록을 사용자 사용화면(웹,	



	애플리케이션)으로 보낸다.
<b>Requirements</b>	-
<b>Pre-condition</b>	-
<b>Post-condition</b>	-
<b>Side effects</b>	-

## 6.2. Functional Requirements ( Back End )

### Site Collecting System

**Table 9. Item Detail**

<b>Name</b>	<b>Site Collecting Function</b>	
<b>Description</b>	판매 사이트 리스트를 크롤링한다..	
<b>Inputs</b>	Target Site list	크롤링한 목표 사이트의 목록
<b>Source (Input)</b>	백엔드 애플리케이션 서버, 크론 스케줄러	
<b>Outputs</b>	파싱된 판매 사이트 리스트	
<b>Destination (Output)</b>	판매 사이트 데이터베이스 Updater	

<b>Action</b>	타겟 사이트의 리스트를 순회하며, 목표 상품을 검색해 나오는 상품 페이지의 이름, URL 등을 파싱하고 목록화한다.
<b>Requirements</b>	-
<b>Pre-condition</b>	-
<b>Post-condition</b>	-
<b>Side effects</b>	-

Table 10. Site Collecting system

## 6.3 Non-functional Requirements

### A. Product requirements

#### 1. Usability

본 서비스의 시스템은 모바일 디바이스에서 실행되는 어플리케이션이다. 그렇기에, 여러가지 플랫폼에서의 디스플레이, 운영체제 등과 같은 제약이 원활한 서비스 운용에 방해가 되지 않도록, 다양한 환경을 고려하고, 불편함을 느끼지 않고, 그에 상응하는 UI/UX 제공할 수 있어야한다.

#### 2. Efficiency requirement

1) 상품들의 정보가 업데이트 될 때, 데이터베이스에서 데이터를 로드 할 때, 소요되는 시간, 2) Compare 페이지와 같은 서비스 제공 화면의 요청에 대한 응답하는데 소요되는 시간

위와 같은 시간들이 일정 수준 이상이 되어, user 와의 interact 에 방해가 되지 않도록 효율성을 추구해야한다.

#### 3. Dependability

사용자의 환경설정 및 정보가 제대로 데이터베이스에 저장되어있어, 서비스 제공의 범위가 개별 사용자의 의도와 상응하게 한정되어야 하고, 여러가지 기능들이 수행되는데, 오류가 발생해서는 안된다.

### *4.Security*

사용자의 개인정보는 오직 사용자에게 의해 등록/수정이 가능하며, 사용자의 요청이 있을 경우에만, 제공자가 해당 사용자의 정보를 수정할 수 있으며, 해당 정보들은 사용자 식별에 사용되며, 외부에 유출될 수 없도록, 데이터베이스의 보안성이 갖추어 져야한다.

## **B. Organization requirements**

### *1.Environmental requirement*

본 서비스는 모바일 디바이스에서 제공되며, 운영체제가 다르더라도, 제공되는 서비스의 제약이 크지 않아야 한다.

### *2.Operational requirement*

개발은 Node.js, React Native, Firebase 등을 이용하여 개발하며, 사용언어는 Java, JavaScript 등이다.

### *3.Development requirement*

Firebase Google Cloud Service 를 이용하여, 데이터베이스에 서비스 운용에 필요한 정보를 저장하며, 사용자의 정보는 레이아웃이 전환될 때, 기록되어 사용될 수 있다. 또한, React Native 를 이용하여, 모바일 디바이스 운영체제 간의 제약을 줄일 수 있도록 한다.

## **C. External requirements**

### *1.regulatory/legislative requirement*

사용자의 개인정보는 제공자와 해당 사용자만 열람이 가능하며, 정보의 수정은 선택적으로 관리자가 할 수 있지만, 원칙적으로는 해당 사용자만 수정 권한이 있다. 중개서비스의 특성상, 사용자의 정보일부를 외부로 제공할 경우에는 사용자의 동의 하에, 제공할 수 있다.

### *2. ethical requirement*

연령제한은 없다.

### *3. safety/security requirement*

개인정보보호법에 따라 사용자의 개인정보를 외부/내부 적으로 안전하게 저장하고, 보호해야한다.

## 6.4 Scenario Examples

### **A. Compare**

#### *1.Initial Assumption*

사용자는 본 시스템에 등록되어 있으며, 구매를 희망하는 상품의 후보군들을 즐겨 찾기에 추가해 두었다. 하지만 아직 정확히 어떤 상품을 구매할 지는 결정하지 못한 상황이다.

#### *2. Normal flow of events*

사용자는 먼저 시스템에 로그인을 한다. 그 이후에 자신의 마이 페이지에 접근이 가능하게 된다. 마이 페이지로 이동하면, 하위 메뉴인 개인 정보 관리, Wish list 에 추가한 상품목록 및 review, 애플리케이션 설정, 그리고 문의로 이루어져 있다. 사용자는 Wish list 로 이동하여 기존에 자신이 구매를 희망하고자 하는 상품들의 목록을 확인할 수 있다. 여기서 특정 카테고리를 설정하면 해당하는 상품들의 목록이 필터링 되어 나타난다. 사용자는 구매를 결정하기 위해서 상품을 하단에 존재하는 compare 메뉴로 드래그 할 수 있다. 그렇게 같은 카테고리의 두개의 상품을 드래그 하게 되면 compare 페이지로 이동할 수 있게 되고, compare

페이지에서는 두 상품의 항목별 세부 정보를 확인할 수 있고, 그에 대한 리뷰 또한 같이 비교할 수 있다. 어떤 상품을 구매할 것인지 선택하면, 결제 페이지로 이동한다.

### 3. What can go wrong

특정 카테고리를 선택했을 때, 해당하는 상품이 두 가지 이상 존재하지 않을 가능성이 있다. 이 경우에는 해당하는 상품이 있으면 보여주되, 비교할 수 있는 상품이 없다는 메시지를 출력한다.

### 4. System state on completion

사용자가 상품을 구매하면 구매 기록을 저장하고, 정해진 기간 내에 구매 기록을 참고할 수 있도록 한다.

## 7. System Models

### 7.1 Objective

System Models에서는 시스템과 시스템 컴포넌트, 시스템 환경 간의 관계를 보여준다. 여러 Diagram을 통해 Context Model, Interaction Model을 나타낸다.

### 7.2 Context model

#### A. Process Diagram

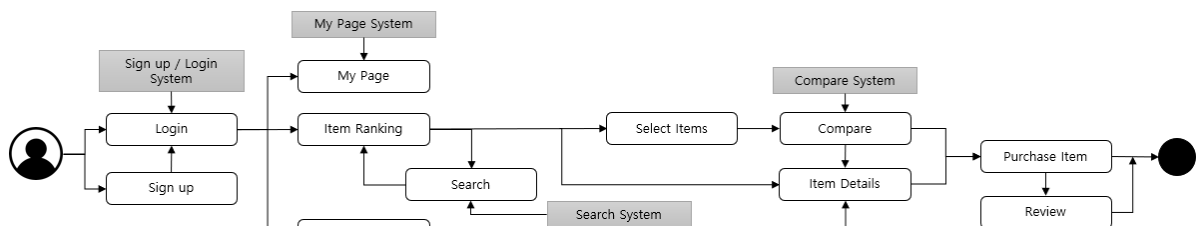


Diagram 6: Overall Process Diagram

## 7.3 Interaction model

### A. Use Case Diagram

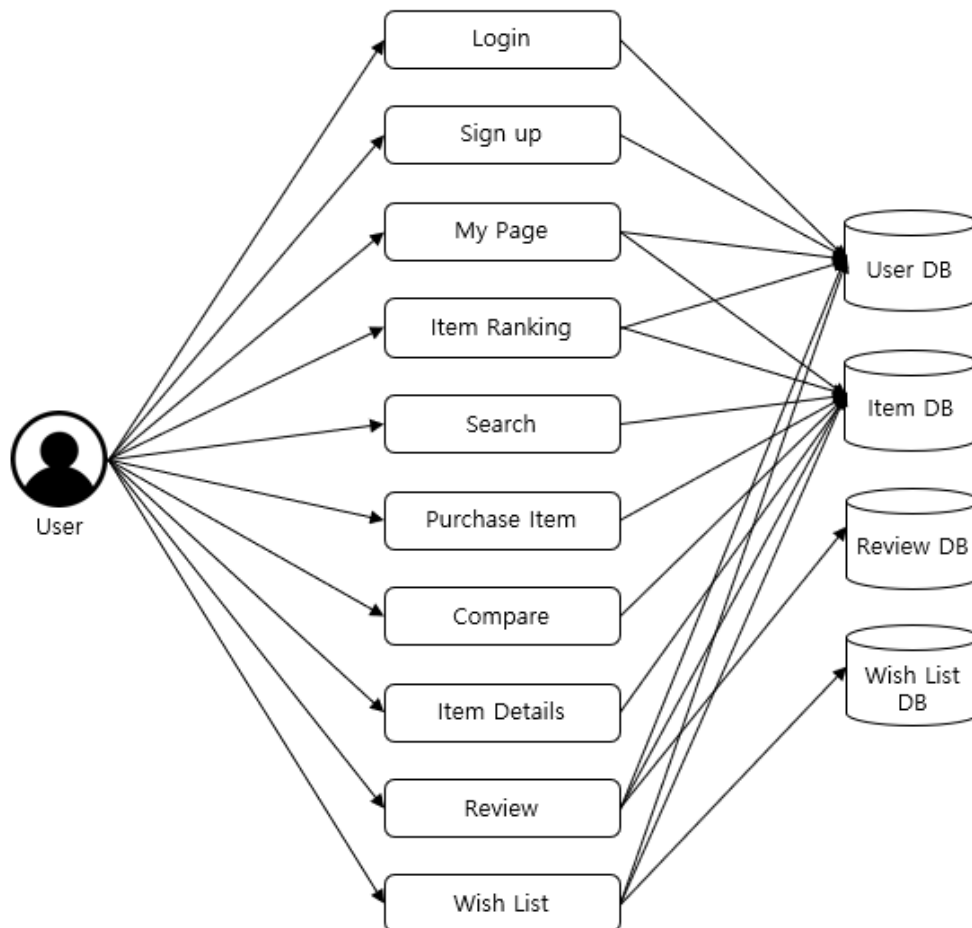


Diagram 7: Use Case Diagram

## B. Tabular Description of Use case Diagram

### 1. Login

Use Case	Login
Actor	User, Database(User)
Description	User로부터 ID와 Password를 입력 받아 시스템에 등록된 User 정보와 일치하는지 확인한다. SNS 계정으로 로그인할 수 있다.
Trigger	User가 ID와 Password를 입력하고 Login을 클릭한다.
Success Response	User의 ID가 시스템 데이터베이스에 존재하고 Password가 일치한다면 User의 정보를 반환하고 로그인에 성공한다.
Failure Response	User의 ID가 시스템 데이터베이스에 존재하지 않으면 ID가 존재하지 않음을 알리고, Password가 일치하지 않으면 Password가 틀림을 알린다.

**Table 11. <Use case Diagram> Login**

### 2. Sign up

Use Case	Sign up
Actor	User, Database(User)
Description	User의 정보를 시스템 데이터베이스에 등록하여 로그인에 사용한다. SNS 계정이 있는 경우 따로 가입하지 않을 수 있다.
Trigger	User가 양식에 맞추어 개인 정보를 입력하고 Sign up을 클릭한다.
Success Response	User의 개인 정보를 시스템 데이터베이스에 등록하고 이를 알린 후 로그인 페이지로 돌아간다.
Failure Response	User가 이미 시스템 데이터베이스에 등록되었거나 잘못된 형식의 정보를 입력하였다면 이를 알린다.

**Table 12. <Use case Diagram> Sign Up**

### 3. My page

Use Case	My Page
Actor	User, Database(Item, User)
Description	User가 자신의 개인 정보 및 시스템 이용에 관한 정보를 My Page를 통해 수정 및 조회한다.
Trigger	User가 로그인 된 상태에서 My Page 버튼을 클릭한다.
Success Response	User의 정보를 시스템 데이터베이스에서 검색해 이를 화면에 출력한다.
Failure Response	-

Table 13. <Use case Diagram> My Page

### 4. Item Ranking

Use Case	Item Ranking
Actor	User, Database(Item, User)
Description	User가 전체 및 특정 카테고리의 Item의 순위를 조회한다. 기간별 랭킹을 확인할 수 있다.
Trigger	User가 원하는 Item의 카테고리 및 기간을 선택한다.
Success Response	데이터베이스에서 해당 카테고리 및 기간의 Ranking 정보를 얻어 이를 화면에 출력한다.
Failure Response	-

Table 14. <Use case Diagram> Item Ranking

### 5. Search

Use Case	Search
Actor	User, Database(Item)
Description	User가 키워드와 카테고리 등의 필터링을 통해 원하는 Item들을 데이터베이스에서 조회한다.
Trigger	User가 키워드와 필터링을 입력한다.
Success Response	User가 입력한 정보를 데이터베이스에서 검색해서 일치하는 Item들을 나열한다.
Failure Response	키워드와 필터링에 일치하는 상품이 없으면 User에게 빈 결과를 보여주고 오류 메시지를 출력한다.

Table 15. <Use case Diagram> Search



## 6. Purchase Item

Use Case	Purchase Item
Actor	User, Database(Item)
Description	구입을 원하는 Item의 판매 사이트로 접속하여 Item을 구매한다.
Trigger	구입을 원하는 Item의 Item Detail 페이지에서 구매하기를 누른다.
Success Response	Item을 판매하는 사이트로 접속된다.
Failure Response	재고가 없거나 판매가 중지되었다면 User에게 이를 알리고 원래 페이지로 돌아간다.

Table 16. &lt;Use case Diagram&gt; Purchase Item

## 7. Compare

Use Case	Compare
Actor	User, Database(Item)
Description	같은 카테고리의 두 Item을 선택하면 카테고리 별로 정해진 항목에 따라 세부 정보를 보기 쉽게 조회한다.
Trigger	User가 비교를 원하는 같은 카테고리의 두 Item을 선택한다.
Success Response	User에게 카테고리 별로 정해진 몇 가지의 항목으로 세부 정보를 데이터 베이스에서 검색해 제공한다. 여기에는 리뷰 및 가격정보도 포함된다.
Failure Response	선택한 두 Item이 같은 카테고리가 아니면 이를 알리고 오류 메시지를 출력한다.

Table 17. &lt;Use case Diagram&gt; Compare

## 8. Item Details

Use Case	Item Details
Actor	User, Database(Item)
Description	User가 선택하는 특정 상품의 세부 정보를 보여준다.
Trigger	User가 자세히 보기 원하는 Item을 선택한다.
Success Response	데이터베이스에서 User가 선택한 Item의 세부 정보를 검색해 출력한다.
Failure Response	-

Table 18. &lt;Use case Diagram&gt; Item Details

## 9. Review

Use Case	Review
Actor	User, Database(Item, Review)
Description	User가 Item을 구매한 후 Review를 남길 수 있다. My Page에서 User가 남긴 Review를 확인할 수 있다.
Trigger	User가 Item을 구매한 후 평점과 Review를 작성하고 등록한다.
Success Response	User가 작성한 리뷰가 데이터베이스에 저장된다. 평점은 Item의 평균 평점에 즉시 적용된다.
Failure Response	-

Table 19. &lt;Use case Diagram&gt; Review

## 10. Wish List

Use Case	Wish List
Actor	User, Database(Item, Wish List)
Description	User가 자신이 등록한 Wish List 목록을 조회한다.
Trigger	User가 로그인 된 상태에서 Wish List 버튼을 클릭한다.
Success Response	데이터베이스에서 검색해 User가 등록한 Wish List 목록의 Item을 보여준다.
Failure Response	-

Table 20. &lt;Use case Diagram&gt; Wish List

### C. Behavioral Models

이 장에서는 본 시스템이 중점을 두고 있는 하위 시스템인 'Ranking System'과 'Compare System'을 behavioral model 로 표현하였다.

#### A. Ranking System

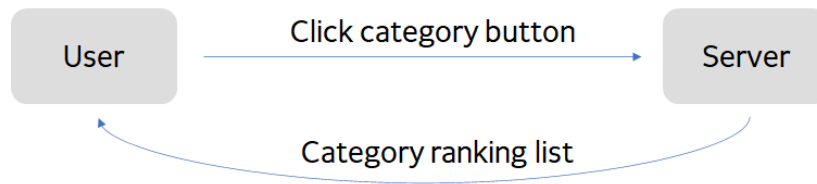


Diagram 8. <Data Flow> Ranking System

#### B. Compare System

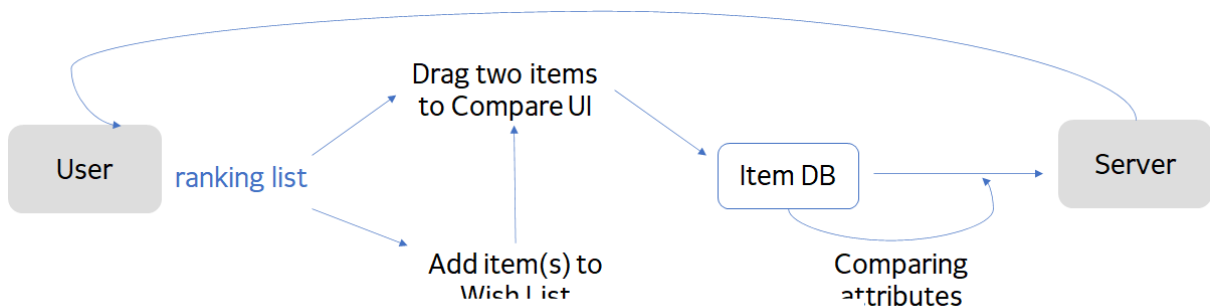


Diagram 9. <Data Flow> Compare System

## 8. System Evolution

시스템이 배포되었다는 것이 시스템 개발의 종지부를 의미하지 않는다. 시스템이 배포된 이후에도 시스템을 둘러싼 사업적/기술적 측면의 환경 변화, 해당 시스템을 사용하는 사용자들의 변화 혹은 사용자들의 요구사항 변화 등 여러 변화들에 민감하게 반응해야 한다. 따라서 이번 장에서 어떤 방면에서 변화가 발생할지 예측하고 이를 기존 시스템에 어떠한 방식으로 통합을 시킬지에 대해 논의해보고자 한다. 이를 통해 시스템 변화에 대한 기민한 대응과 효율적인 비용 절감을 기대할 수 있다.

### 8.1. Multiple Comparison Feature

현재 비교 기능은 사용자가 같은 카테고리 내에서 비교하고 싶은 두개의 물품을 선택하여 해당 카테고리의 비교 기준들을 토대로 물품에 관한 상세 내역들을 한눈에 비교 가능하도록 보여주는 형식이다. 사용자 입장에서 비교하고 싶은 물품들이 여러 개가 있을 경우를 고려하여 다중 비교 기능을 지원하는 디스플레이 기능 추가를 고려할 수 있다.

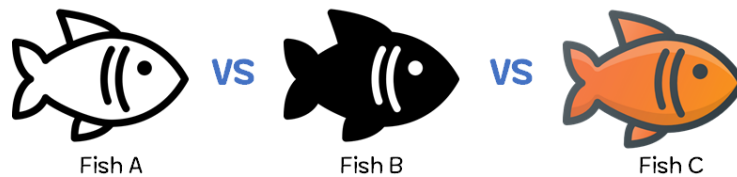


그림 12. Multiple Comparison Feature

### 8.2. Voice/Image Search Technique

사용자들이 원하는 물품을 찾기에는 문자만 지원하는 검색 엔진이 제한적일 수 있다. 문자를 입력하기 어려운 상황에서는 손쉽게 육성으로 원하는 물품을 검색하게 할 수 있다. 더 나아가 인터넷 브라우징을 하다가 찾은 갖고 싶은 사진만을 가지고도 원하는 물품을 검색할 수 있도록 한다면 검색 엔진의 효용성이 더 높아질 것이라고 기대한다.

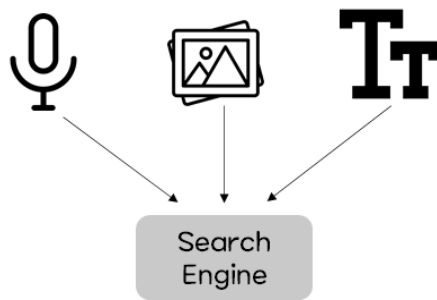


그림 13. Voice/Image Search Technique

### 8.3. Customer Review Verification

비교 기능에서 두 개의 물품을 비교해줄 때 물품에 대한 평균적인 점수를 매기는 차원에서 소비자들의 평점 평균이 하나의 척도로서 활용된다. 수치적으로 비교 가능한 척도일수록 수치에 대한 신뢰성이 중요하다. 따라서 소비자들에게 보다 신뢰 가능한 비교 수치를 제공하기 위해 상품을 소비한 소비자들이 매긴 리뷰를 검토하는 작업이 필요하다.

### 8.4. OAuth implementation

회원가입 과정을 거치지 않고도 구글 로그인(API)이나 네이버, 페이스북 등의 로그인(API)과 연동시키면 손쉽게 가입이 가능하게 될 것이다.

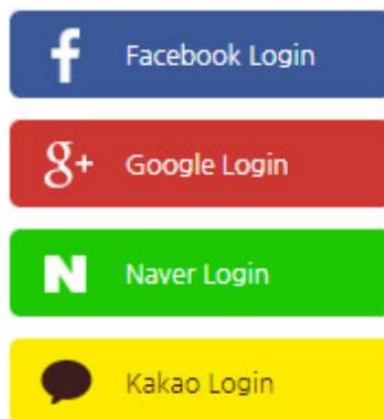


그림 14. OAuth Implementation

## 9. Appendices

이번 장에서는 시스템 운영 환경에 필요한 하드웨어 요구사항 및 시스템에 사용되는 데이터 베이스에 대해 대략적으로 설명한다.

### A. Hardware Requirements

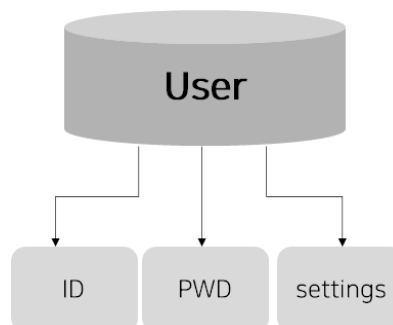
본 시스템에서는 네이티브 앱과 웹의 장점을 결합하기 위해 Progressive Web App 기술을 적용하여 웹의 접근을 네이티브 앱처럼 아이콘 터치만으로 가능하게 하였다. 해당 기술을 이용하기 위해서는 우선 Progressive Web App을 지원하는 최신 운영체제와 브라우저가 필요하다. 또한 Backend Server에서 카테고리별 상품 랭킹, 사용자 리뷰를 반영한 상품 평점 등을 지속적으로 업데이트하므로 네트워크와의 연결이 필수적이다.

### B. Database Requirements

본 시스템은 RDBMS 데이터베이스 구조를 활용해 데이터를 저장한다. 데이터베이스를 이루는 각 엔티티는 다음과 같다.

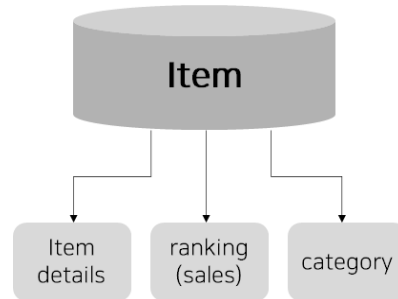
#### 1) User

사용자의 개인 정보 및 세부적인 환경 설정 내역을 담고 있는 엔티티



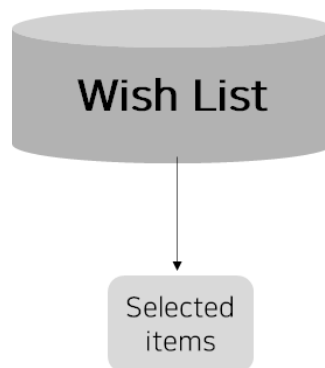
## 2) Item

상품 자체에 대한 세부적인 정보와 해당 상품이 위치해 있는 카테고리 종류와 상품의 순위(판매량)를 담고 있는 엔티티



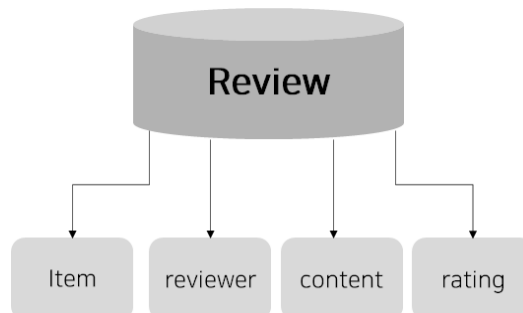
## 3) Wish List

사용자가 Wish List에 담은 상품 정보를 담고 있는 엔티티



## 4) Review

사용자가 특정 상품에 대해 남긴 리뷰 내용 및 평점을 담고 있는 엔티티



# 10. Index

## 10.1 Tables

Table 1 Glossary .....	16
Table 2. Sign up .....	28
Table 3. Login.....	29
Table 4. Search .....	29
Table 5. Push Notification .....	30
Table 6. Compare .....	31
Table 7. My Page.....	31
Table 8. Ranking .....	32
Table 9. Item Detail.....	33
Table 10. Site Collecting system .....	34
Table 11. <Use case Diagram> Login .....	39
Table 12. <Use case Diagram> Sign Up .....	39
Table 13. <Use case Diagram> My Page .....	40
Table 14. <Use case Diagram> Item Ranking .....	40
Table 15. <Use case Diagram> Search.....	40
Table 16. <Use case Diagram> Purchase Item.....	41
Table 17. <Use case Diagram> Compare.....	41
Table 18. <Use case Diagram> Item Details.....	41
Table 19. <Use case Diagram> Review .....	42
Table 20. <Use case Diagram> Wish List.....	42



## 10.2 Figures

그림 1. 커머스 앱 "쿠팡"은 기존 강세였던 SNS 어플리케이션 (인스타그램, 페이스북) 보다도 많은 사용자수를 보유하고 있다.....	9
그림 2. 온라인 오픈마켓에 등록된 정보량은 빠른 속도로 증가하는 추세를 보이고 있다. ....	10
그림 3. 인기 소셜 커머스 어플리케이션 "쿠팡"에서의 "태블릿" 검색결과 (왼쪽), "갤럭시 탭 s6 lite" 검색결과 (오른쪽).....	11
그림 4. 하나의 단일상품에서도 확인하여야 하는 정보가 지나치게 많다. (왼쪽), 하나의 단일상품에 대한 가격비교만으로도 70가지가 넘는 선택지가 존재한다. (오른쪽).....	12
그림 5. Comparewise의 Information UI 초기버전 ( 왼쪽 ).....	13
그림 6. Comparewise의 Compare UI 초기버전 ( 오른쪽 ).....	13
그림 7. Sign up & Login .....	17
그림 8. Search .....	18
그림 9. Push Notification.....	18
그림 10. Ranking ( App page ) .....	19
그림 11. Item Detail ( App page ).....	20
그림 12. Multiple Comparison Feature .....	44
그림 13. Voice/Image Search Technique .....	45
그림 14. OAuth Implementation .....	45

## 10.3 Diagrams

Diagram 1. Overall System Architecture .....	23
Diagram 2. Sign up / Login System Architecture .....	24
Diagram 3. Compare System Architecture .....	25

Diagram 4. My Page System Architecture.....	26
Diagram 5. Search System Architecture .....	27
Diagram 6: Overall Process Diagram.....	37
Diagram 7: Use Case Diagram .....	38
Diagram 8. <Data Flow> Ranking System.....	43
Diagram 9. <Data Flow> Compare System.....	43

## 11. Reference

[1] 한국어 APA 웹사이트 / 내주  
(티티엘뉴스, 2020)

한국어 APA 웹사이트 / 내주-참고문헌  
코로나 19 여파 오픈마켓 관심도 급등...옥션 TOP•11 번가 상승 . (2020).  
[http://ttlnews.com/article/travel\\_report/8055](http://ttlnews.com/article/travel_report/8055).

[2] 한국어 APA 웹사이트 / 내주

(IT DONGA, 2020)  
한국어 APA 웹사이트 / 내주-참고문헌  
'카톡', '유튜브', '네이버', '쿠팡'... 2019 년 사용자가 찾은 앱은? . (2020).  
<https://it.donga.com/29974/>.