SONGSTERR

-Requirements Specification-



songsterr

2016310936 우승민

Contents

[1. Preface 6](#_Toc40296106)

[1.1 Objective 6](#_Toc40296107)

[1.2 Readership 6](#_Toc40296108)

[A. User Requirements 6](#_Toc40296109)

[B. System Requirements 6](#_Toc40296110)

[1.3 Documents Contents 7](#_Toc40296111)

[A. Preface 7](#_Toc40296112)

[B. Introduction 7](#_Toc40296113)

[C. Glossary 7](#_Toc40296114)

[D. User Requirement Definition 7](#_Toc40296115)

[E. System Architecture 7](#_Toc40296116)

[F. System Requirements Specification 8](#_Toc40296117)

[G. System Models 8](#_Toc40296118)

[H. System Evolution 8](#_Toc40296119)

[I. Appendices 8](#_Toc40296120)

[J. Index 8](#_Toc40296121)

[K. Reference 8](#_Toc40296122)

[2. Introduction 9](#_Toc40296123)

[2.1 Background 9](#_Toc40296124)

[2.2 System Overview 13](#_Toc40296125)

[2.3 Expected Benefits 14](#_Toc40296126)

[3. Glossary 15](#_Toc40296127)

[4. User Requirements Definition 17](#_Toc40296128)

[4.1 Functional Requirements 17](#_Toc40296129)

[A. Sign up / Login 17](#_Toc40296130)

[B. Search 18](#_Toc40296131)

[C. Push Notification 18](#_Toc40296132)

[D. My Page 19](#_Toc40296133)

[E. Compare page 19](#_Toc40296134)

[F. Item ranking 19](#_Toc40296135)

[G. Item detail 20](#_Toc40296136)

[4.2. Non-functional Requirements 20](#_Toc40296137)

[A. Product requirements 20](#_Toc40296138)

[B. Organization requirements 21](#_Toc40296139)

[C. External requirements 21](#_Toc40296140)

[5. System Architecture 22](#_Toc40296141)

[5.1 Objective 22](#_Toc40296142)

[5.2 Overall System Architecture 22](#_Toc40296143)

[5.3 Sign up / Login System 23](#_Toc40296144)

[5.4 Compare System 24](#_Toc40296145)

[5.5 My page System 25](#_Toc40296146)

[5.6 Search System 26](#_Toc40296147)

[6. System Requirements Specification 27](#_Toc40296148)

[6.1 Functional Requirements 27](#_Toc40296149)

[A. Sign up/Login 27](#_Toc40296150)

[B. Search 28](#_Toc40296151)

[C. Push Notification 29](#_Toc40296152)

[D. Compare 29](#_Toc40296153)

[E. My page 30](#_Toc40296154)

[F. Ranking 31](#_Toc40296155)

[G. Item Detail 32](#_Toc40296156)

[6.2. Non-functional Requirements 32](#_Toc40296157)

[A. Product requirements 32](#_Toc40296158)

[B. Organization requirements 33](#_Toc40296159)

[C. External requirements 34](#_Toc40296160)

[6.3 Scenario Examples 34](#_Toc40296161)

[A. Compare 34](#_Toc40296162)

[7. System Models 35](#_Toc40296163)

[7.1 Objective 35](#_Toc40296164)

[7.2 Context model 35](#_Toc40296165)

[A. Process Diagram 35](#_Toc40296166)

[7.3 Interaction model 36](#_Toc40296167)

[A. Use Case Diagram 36](#_Toc40296168)

[B. Tabular Description of Use case Diagram 37](#_Toc40296169)

[C. Behavioral Models 41](#_Toc40296170)

[*A.* *Ranking System* 41](#_Toc40296171)

[*B.* *Compare System* 41](#_Toc40296172)

[8. System Evolution 42](#_Toc40296173)

[8.1.  Multiple Comparison Feature 42](#_Toc40296174)

[8.2.  Voice/Image Search Technique 42](#_Toc40296175)

[8.3.  Customer Review Verification 43](#_Toc40296176)

[8.4.  OAuth implementation 43](#_Toc40296177)

[9. Appendices 44](#_Toc40296178)

[A. Hardware Requirements 44](#_Toc40296179)

[B. Database Requirements 44](#_Toc40296180)

[1) User 44](#_Toc40296181)

[2) Item 45](#_Toc40296182)

[3) Wish List 45](#_Toc40296183)

[4) Review 45](#_Toc40296184)

[10. Index 46](#_Toc40296185)

[10.1 Tables 46](#_Toc40296186)

[10.2 Figures 47](#_Toc40296187)

[10.3 Diagrams 47](#_Toc40296188)

[11. Reference 48](#_Toc40296189)

# 1. Preface

## 1.1 Objective

본 Preface에서는 이 문서의 예상 독자를 설정하며, 문서의 전반적인 내용의 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.

### 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.

### 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.

### Preface

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

### 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.

### 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.

### 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.

### System Architecture

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

### 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.

### System Models

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

### System Evolution

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

### Appendices

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

### Index

본 문서에서 사용된 그림, 표, 다이어그램 등의 색인을 기술한다.

### 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 소설 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.

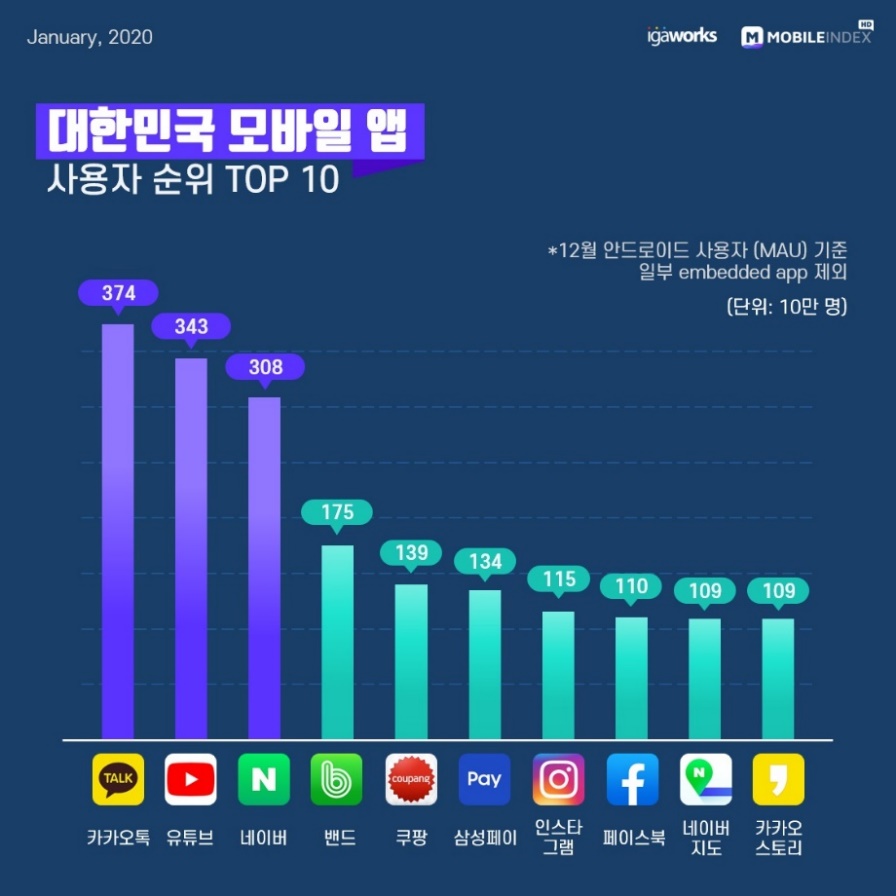


그림 1. 커머스 앱 “쿠팡”은 기존 강세였던 SNS 어플리케이션 (인스타그램, 페이스북) 보다도 많은 사용자수를 보유하고 있다.

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

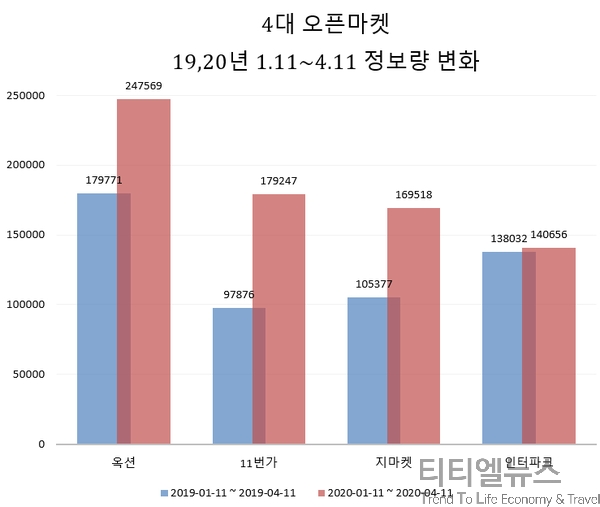


그림 2. 온라인 오픈마켓에 등록된 정보량은 빠른 속도로 증가하는 추세를 보이고 있다.

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, 한가지 상품의 이름인 “갤럭시 탭 s6 lite”를 검색하였을 경우 총 10,419개의 상품이 검색되었다.

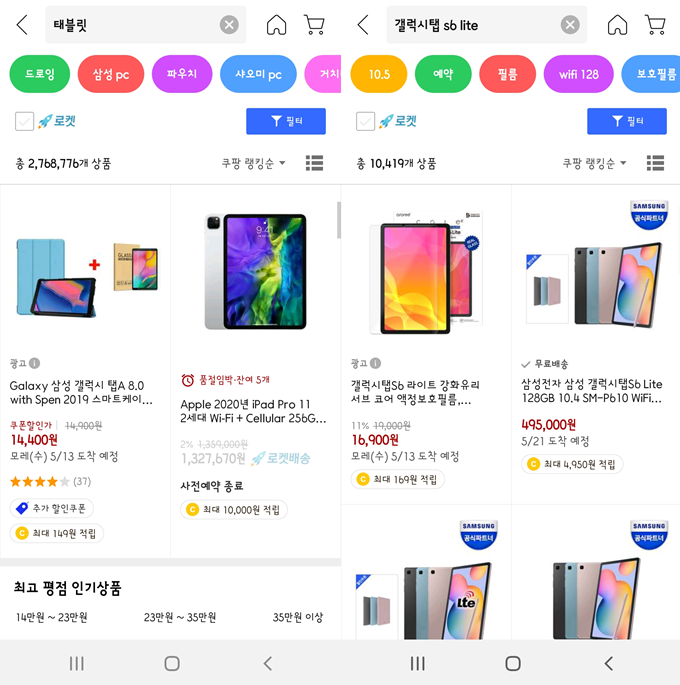
****

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

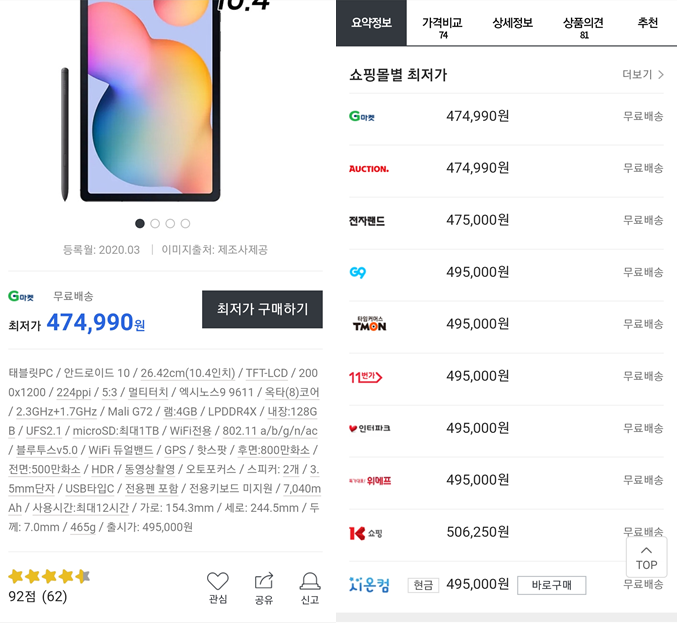
또한 개별 상품에서의 소비자에게 주어지는 정보 또한 지나치게 방대해졌다. 배송 일자, 배송비, 할인 정보, 다양한 판매처, 평점, 제품 색상, 등 판매자가 확인하여야 하고, 비교하여야 하는 정보가 지나치게 많아 한가지 상품, 혹은 다양한 상품중에서 구매를 결정하기가 어려워지고 있다. 

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

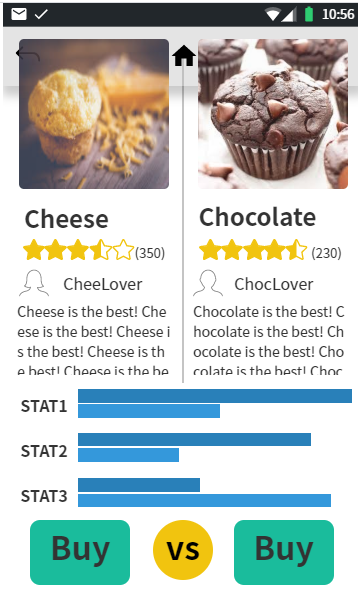
****

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 6] 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** |
| **Compare** | 사용자가 추가한 상품들을 1대1 로 세부정보를 나열하여 비교해주는 page |
| **Login** | 사용자가 application을 개인 기록과 환경에 맞게 사용할 수 있게 해주는 function |
| **Search** | 사용자가 구매하고자 하는 상품 혹은 가장 유사한 상품을 찾아주는 기능 |
| **Ranking** | 기간 별로 가장 많이 팔린 상품을 순위별로 보여주는 page |
| **Wish list** | 사용자가 구매하기를 희망하는 상품을 모아놓은 page |
| **Rating** | 사용자가 구매 이후 해당 상품에 대한 평가를 0~5의 숫자 지표로 나타낸 것 |
| **Review** | 사용자가 구매 이후 해당 상품에 대한 평가를 글(자연어)로 나타낸 것 |
| **Product Crawler** | 여러 웹사이트에서 상품 판매에 대한 정보를 모으게 도와주는 module |
| **Database** | 사용자 및 상품에 대한 정보를 포함하는 backend term |
| **API** | A set of higher-class functions and interfaces developed by advanced programmers that can be imported and use easily in different projects |
| **NLP** | 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 필수 적인 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 한 다. 검증이 된다면 서비스에 접근할 수 있는 권한이 생긴다.

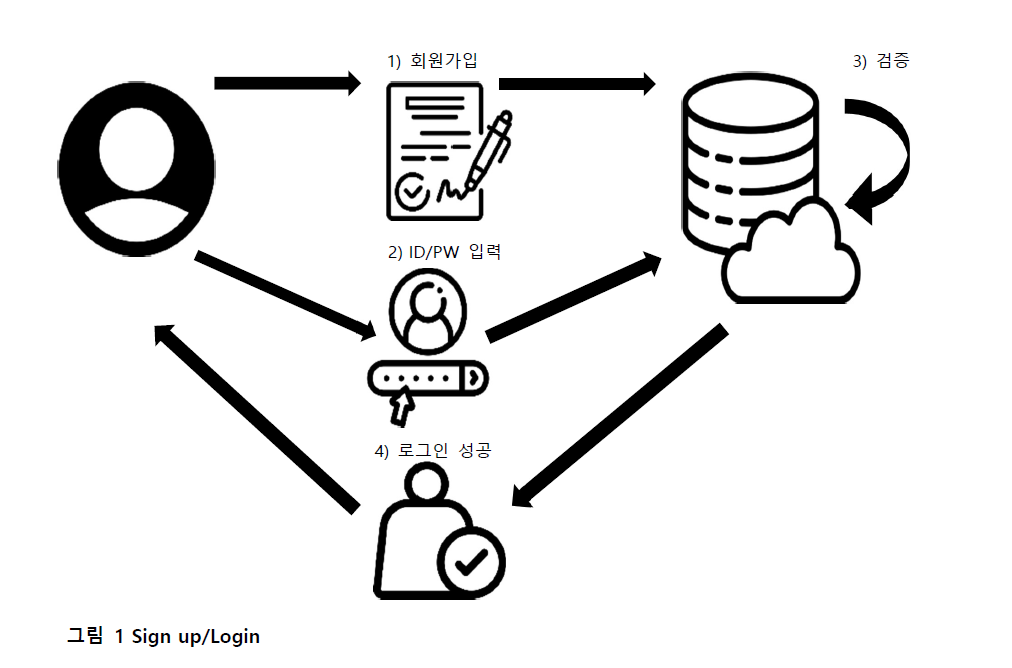


그림 7. Sign up & Login

### **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 Display한다. Display된 상품은 여러가지 기준으로 정렬될 수 있으며, 해당 상품을 좀 더 자세히 보기 위해, 해당 상품을 클릭할 수 있다.

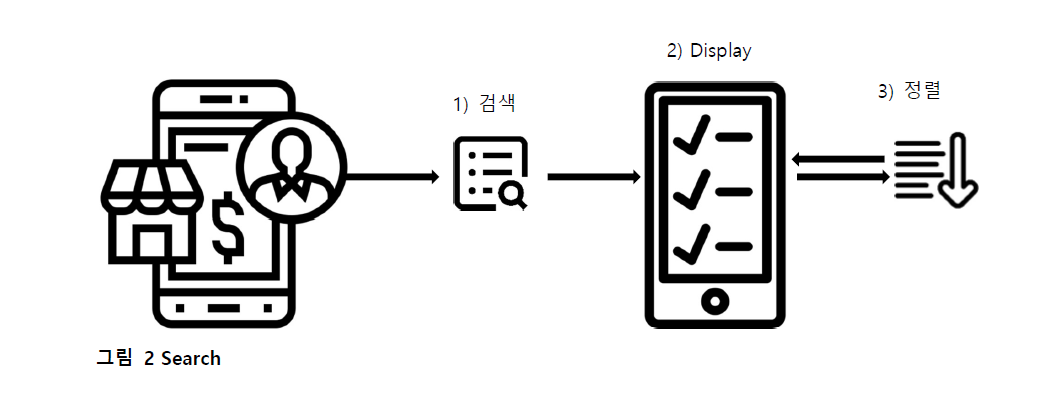


그림 8. Search

### **C. Push Notification**

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

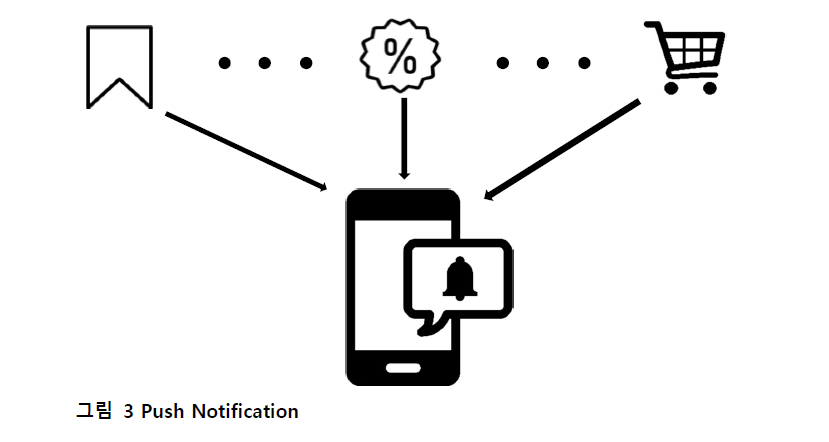


그림 9. Push Notification

### **D. My Page**

사용자의 개인에 관련된 정보를 확인할 수 있는 페이지이다. 사용자 개인 정보부터 시작해서, Wish list에 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 list에 넣은 상품 중 두 가지를 비교하여 살 수 있게 한다. 혹은 Ranking Page에서 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**

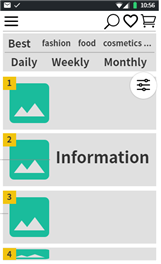


그림 10. Ranking ( App page )

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 interact에 방해가 되지 않도록 효율성을 추구해야한다.

#### *3. Dependability*

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

#### *4. Security*

사용자의 개인정보는 오직 사용자에 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 Service를 이용하여, 데이터베이스에 서비스 운용에 필요한 정보를 저장하며, 사용자의 정보는 레이아웃이 전환될 때, 기록되어 사용될 수 있다. 또한, React Native를 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 서브 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. 로그인은 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

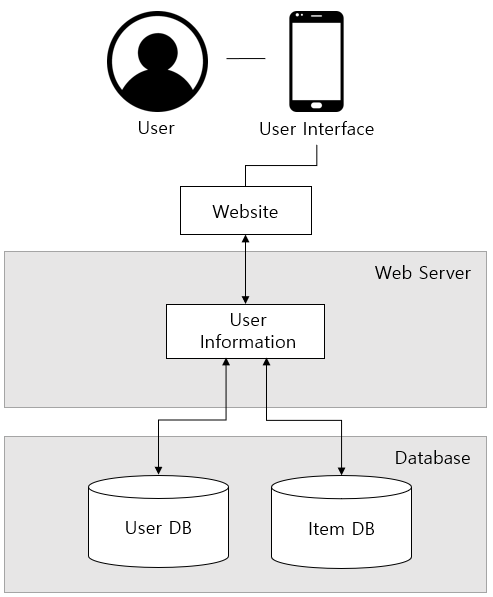


Diagram 4. My Page System Architecture

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

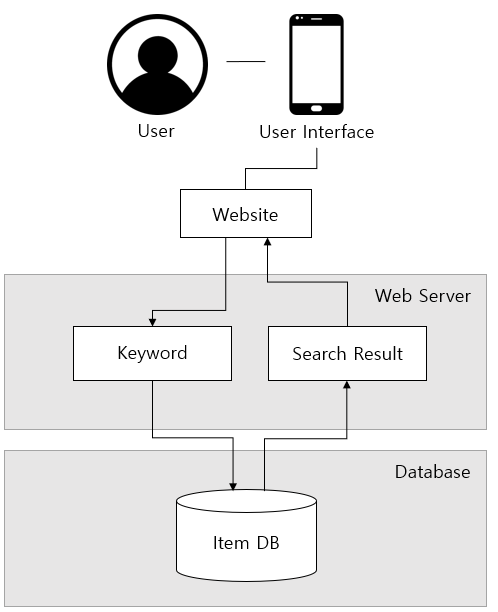


Diagram 5. Search System Architecture

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**

|  |  |  |
| --- | --- | --- |
| Name | Sign up Function | |
| Description | 서버의 Database에 User의 정보를 등록하여, 사용자로서 승인한다. | |
| Inputs | Google / SNS 계정 | Google 계정 혹은 타 SNS 계정 |
| User Information | 사용자의 개인정보 Class |
| Source (Input) | User input | |
| Outputs | 서버의 DB에 사용자 인증 정보 등록 | |
| Destination (Output) | Database | |
| Action | DB에 사용자에 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 | |

#### 1) Sign up

Table 2. Sign up

#### 2) Login

|  |  |  |
| --- | --- | --- |
| Name | Login Function | |
| Description | Authenticate and authorize the user | |
| Inputs | Google / SNS 계정 | Google 계정 혹은 타 SNS 계정 |
| Source (Input) | User input | |
| Outputs | 사용자 인증 정보 토큰 | |
| Destination (Output) | Application UI | |
| Action | DB서버에 등록된 계정들과 비교하여, 해당 계정정보가 존재한다면, 사용자 인증정보를 제공하여, 서비스에 접속할 수 있도록 한다. | |
| Requirements | -사용자의 인증에 대한 정보를 제공해줄 수 있는 타 서비스 계정 | |
| Pre-condition | 사용자는 회원 전용 서비스에 접근 불가 | |
| Post-condition | 사용자는 회원 전용 서비스에 접근 가능 | |
| Side effects | Frontend(User Interface)가 회원 전용 서비스가 제공되는 Main Home으로 전환된다. | |

Table 3. Login

### **B. Search**

|  |  |  |
| --- | --- | --- |
| Name | Search Function | |
| Description | 사용자의 검색어와 조건에 따라 만족하는 상품들을 Display한다. | |
| Inputs | Keyword | 검색어 String |
| Condition | 검색 Filtering에 필요한 조건 Class |
| Source (Input) | User input | |
| Outputs | 검색 결과 | |
| Destination (Output) | Application UI | |
| Action | 사용자로부터 검색 Keyword를 입력 받아, 해당 조건에 만족하는 결과물을 UI를 통해 제공하며, 조건들을 통해 결과물을 Filtering한다. | |
| Requirements | -검색 Keyword  -검색 Filtering Condition | |
| Pre-condition | 검색 이전 Main Home UI | |
| Post-condition | 검색 결과를 Display | |
| Side effects | 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 | 탑-다운 형식 두 제품을 비교하는 페이지 제공 | |
| Inputs | 두 개의 아이템 ID | 사용자가 비교하길 원하는 두가지 아이템의 ID  (자료형 입력) |
| Source (Input) | 기존 즐겨 찾기에 존재하는 아이템 중 사용자가 두가지를 선택 후 비교 버튼을 클릭 | |
| 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. | |
| Requirements | 두 가지 상품을 골라야 한다. | |
| Pre-condition | - 두 상품은 모두 같은 카테고리에 속해야 한다. | |
| Post-condition | - 주문 가능한 상품만 주문하기 버튼이 나타나야 한다. | |
| Side effects |  | |

Table 6. Compare

### **E. My page**

|  |  |  |
| --- | --- | --- |
| Name | My page Function | |
| Description | 사용자의 고유한 정보를 표시하는 페이지 | |
| Inputs | ID | 사용자의 ID (string) |
| Password | 사용자의 Password (string) |
| Source (Input) | Login 이후 Mypage 클릭 시 | |
| Outputs | 로그인 된 사용자의 Mypage 화면으로 이동 | |
| Destination (Output) | My page 화면 (웹, 어플리케이션) | |
| Action | 사용자 교유의 페이지이다. 이 페이지를 통해서 개인 정보부터 시작해서, 즐겨 찾기에 추가한 상품목록 및 장바구니, 구매내역, 배송정보, 리뷰, 그리고 상품문의에 대한 내용을 확인할 수 있다. | |
| Requirements | 사용자의 ID와 Password가 필요하다. | |
| Pre-condition | - 로그인하지 못하면 화면을 표시하지 않는다. | |
| Post-condition | - 로그인한 사용자에게만 표시된다. | |
| Side effects |  | |

Table 7. My Page

### **F. Ranking**

Table 8. Ranking

|  |  |  |
| --- | --- | --- |
| Name | Ranking | |
| Description | 전체 및 특정 랭킹 페이지를 보여준다 | |
| Inputs | 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) |
| Source (Input) | 랭킹 버튼 클릭 및 자료형 입력 | |
| Outputs | 업데이트된 랭킹 리스트 정보 | |
| Destination (Output) | 사용자 사용화면 (웹, 애플리케이션) | |
| Action | 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. | |
| Requirements | 랭킹 카테고리 및 기간이 명확해야 한다. | |
| Pre-condition | - | |
| Post-condition | - | |
| Side effects | - | |

|  |  |  |
| --- | --- | --- |
| Name | Item detail | |
| Description | 특정 상품의 세부 정보를 보여준다. | |
| Inputs | 아이템 ID | 사용자가 보고 싶은 아이템의 ID(자료형 입력) |
| Source (Input) | Ranking, Search 페이지에서 사용자가 특정 아이템을 선택한다. | |
| Outputs | 특정 아이템의 세부 정보 목록 | |
| Destination (Output) | 사용자 사용화면(웹, 애플리케이션) | |
| Action | 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. | |
| Requirements | - | |
| Pre-condition | - | |
| Post-condition | - | |
| Side effects | - | |

### **G. Item Detail**

Table 9. Item Detail

## 6.2. 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 interact에 방해가 되지 않도록 효율성을 추구해야한다.

#### *3.Dependability*

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

#### *4.Security*

사용자의 개인정보는 오직 사용자에 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 Service를 이용하여, 데이터베이스에 서비스 운용에 필요한 정보를 저장하며, 사용자의 정보는 레이아웃이 전환될 때, 기록되어 사용될 수 있다. 또한, React Native를 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.

## 6.3 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, 즐겨 찾기 및 장바구니, 구매내역, 배송정보, 리뷰, 그리고 상품문의에 대한 내용을 확인할 수 있다. 사용자는 즐겨 찾기로 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**

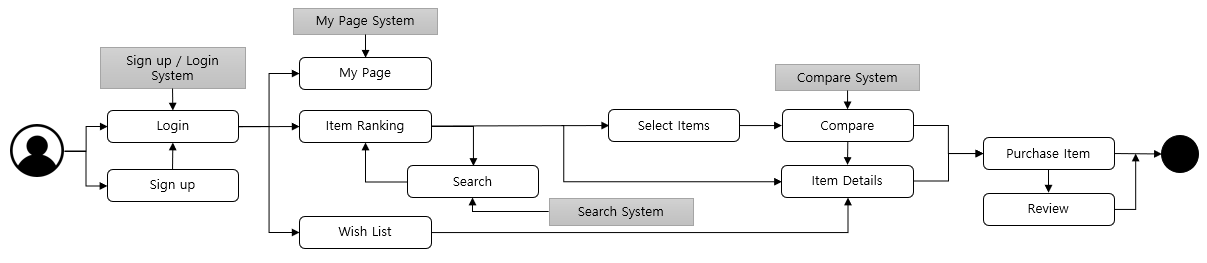


Diagram 6: Overall Process Diagram

## 7.3 Interaction model

### **A. Use Case Diagram**

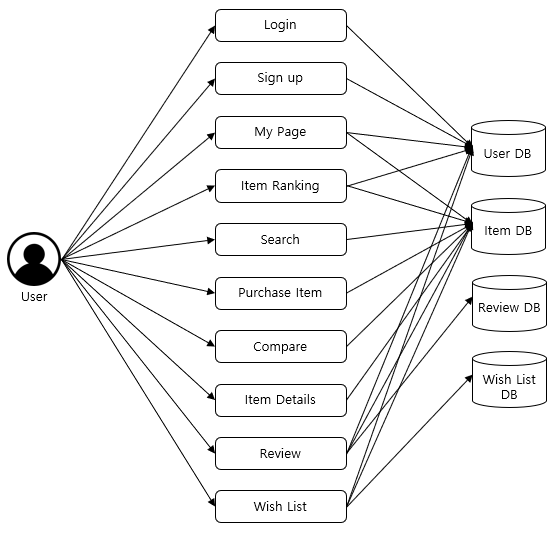
****

Diagram 7: Use Case Diagram

### **Tabular Description of Use case Diagram**

#### *1**. Login*

Table 10. <Use case Diagram> Login

#### *2. Sign up*

Table 11. <Use case Diagram> Sign Up

#### *3. My page*

Table 12. <Use case Diagram> My Page

#### *4. Item Ranking*

Table 13. <Use case Diagram> Item Ranking

#### *5. Search*

Table 14. <Use case Diagram> Search

#### *6. Purchase Item*

Table 15. <Use case Diagram> Purchase Item

#### *7. Compare*

Table 16. <Use case Diagram> Compare

#### *8. Item Details*

Table 17. <Use case Diagram> Item Details

#### *9. Review*

Table 18. <Use case Diagram> Review

#### *10. Wish List*

Table 19. <Use case Diagram> Wish List

## **C. Behavioral Models**

이 장에서는 본 시스템이 중점을 두고 있는 하위 시스템인 ‘Ranking System’과 ‘Compare System’을 behavioral model로 표현하였다.

### *Ranking System*

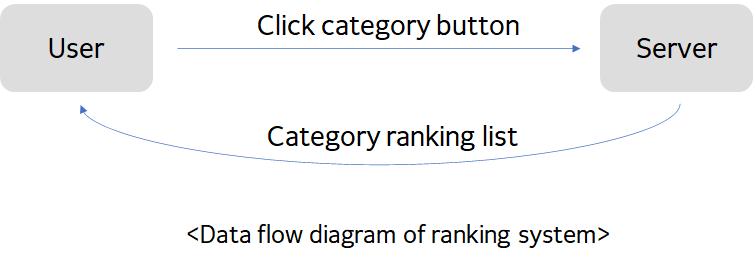


Diagram 8. <Data Flow> Ranking System

### *Compare System*

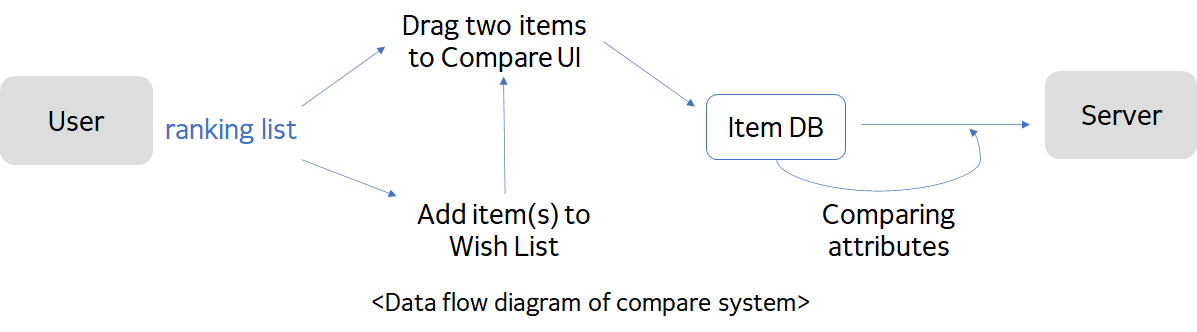


Diagram 9. <Data Flow> Compare System

# 8. System Evolution

본 시스템이 개발과정을 거치고 user들에게 배포되었을 때 모든 개발단계가 끝났다고 할 수 없다. 왜냐면 고객들이 사용하면서 더 원하는 요구사항이 생기는 것은 물론이고 미래에 모바일 디바이스를 포함하여 본 시스템을 둘러싼 환경이 어떠한 변화과정이 생길지 모르기 때문이다. 따라서 이번 챕터에서는 그러한 변화사항을 미리 예상하고, 어떻게 대응할지에 대해 서술한다.

## 8.1.  Score modifying

현재 등록된 악보들은 사용자들이 보고 노래를 들으면서 따라하여 연습할 수 있게 만들었다. 하지만 무척 연주하고 싶지만 악보의 난이도가 너무 높거나, 중간에 오히려 없으면 좋을 것 같은 구간이 있을 수 있다. 따라서 그러한 점을 해결하기 위해 사용자가 악보를 자신만의 style대로 수정이 가능하게 만들면 더 만족성이 높아질 것이라 기대된다.

## 8.2.  Playing external score

현재 ‘Songsterr’ 애플리케이션은 DB에 등록된 악보만을 재생할 수 있다. 하지만 사용자가 원하는 악보가 모두 애플리케이션에 등록되어 있다고 보장할 수 없다. 따라서 기존에 등록된 악보만이 아닌 애플리케이션 외부의 악보(gp 파일)도 재생할 수 있으면, 타 애플리케이션 필요 없게 되어 ‘Songsterr’의 사용량이 늘어날 것이라 기대된다.

## 8.3.  Category

사용자들이 선호하는 음악의 장르, 연주실력 등 각자 찾는 음악의 부류가 다를 것이다. 따라서 ‘Songsterr’에 category를 만들어 사용자가 특별히 찾는 노래가 없을 경우에도 사용자가 지정한 category별로 악보를 보여주면 쉽게 선별할 수 있을 것이다.

# 9. Appendices

## Hardware Requirements

본 시스템은 Progressive Web App 기술을 적용하여 웹의 접근을 아이콘 터치만으로 가능하게 하였다. 해당 기술을 이용하기 위해서 Progressive Web App을 지원하는 최신 운영체제와 브라우저가 필요하다.

## B. Database Requirements

본 시스템은 RDBMS 데이터베이스 구조를 활용해 데이터를 저장한다.

# 10. Index

## 10.1 Tables

[Table 1 Glossary 16](#_Toc40296064)

[Table 2. Sign up 27](#_Toc40296065)

[Table 3. Login 28](#_Toc40296066)

[Table 4. Search 28](#_Toc40296067)

[Table 5. Push Notification 29](#_Toc40296068)

[Table 6. Compare 30](#_Toc40296069)

[Table 7. My Page 30](#_Toc40296070)

[Table 8. Ranking 31](#_Toc40296071)

[Table 9. Item Detail 32](#_Toc40296072)

[Table 10. <Use case Diagram> Login 37](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296073)

[Table 11. <Use case Diagram> Sign Up 37](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296074)

[Table 12. <Use case Diagram> My Page 38](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296075)

[Table 13. <Use case Diagram> Item Ranking 38](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296076)

[Table 14. <Use case Diagram> Search 38](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296077)

[Table 15. <Use case Diagram> Purchase Item 39](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296078)

[Table 16. <Use case Diagram> Compare 39](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296079)

[Table 17. <Use case Diagram> Item Details 39](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296080)

[Table 18. <Use case Diagram> Review 40](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296081)

[Table 19. <Use case Diagram> Wish List 40](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296082)

## 10.2 Figures

[그림 1. 커머스 앱 “쿠팡”은 기존 강세였던 SNS 어플리케이션 (인스타그램, 페이스북) 보다도 많은 사용자수를 보유하고 있다. 9](#_Toc40296083)

[그림 2. 온라인 오픈마켓에 등록된 정보량은 빠른 속도로 증가하는 추세를 보이고 있다. 10](#_Toc40296084)

[그림 3. 인기 소셜 커머스 어플리케이션 “쿠팡”에서의 “태블릿” 검색결과 (왼쪽), “갤럭시탭 s6 lite” 검색결과 (오른쪽) 11](#_Toc40296085)

[그림 4. 하나의 단일상품에서도 확인하여야 하는 정보가 지나치게 많다. (왼쪽), 하나의 단일상품에 대한 가격비교만으로도 70가지가 넘는 선택지가 존재한다. (오른쪽) 12](#_Toc40296086)

[그림 5. Comparewise의 Information UI 초기버전 ( 왼쪽 ) 13](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296087)

[그림 6. Comparewise의 Compare UI 초기버전 ( 오른쪽 ) 13](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296088)

[그림 7. Sign up & Login 17](#_Toc40296089)

[그림 8. Search 18](#_Toc40296090)

[그림 10. Push Notification 18](#_Toc40296091)

[그림 11. Ranking ( App page ) 19](#_Toc40296092)

[그림 12. Item Detail ( App page ) 20](#_Toc40296093)

[그림 13. Multiple Comparison Feature 42](#_Toc40296094)

[그림 14. Voice/Image Search Technique 43](#_Toc40296095)

[그림 15. OAuth Implementation 43](#_Toc40296096)

## 10.3 Diagrams

[Diagram 1. Overall System Architecture 22](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296097)

[Diagram 2. Sign up / Login System Architecture 23](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296098)

[Diagram 3. Compare System Architecture 24](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296099)

[Diagram 4. My Page System Architecture 25](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296100)

[Diagram 5. Search System Architecture 26](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296101)

[Diagram 6: Overall Process Diagram 35](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296102)

[Diagram 7: Use Case Diagram 36](file:///C:\Users\gkstm\Desktop\Requirement%20Specification.docx#_Toc40296103)

[Diagram 8. <Data Flow> Ranking System 41](#_Toc40296104)

[Diagram 9. <Data Flow> Compare System 41](#_Toc40296105)

# 11. Reference

[1] 한국어 APA 웹페이지 / 내주

(티티엘뉴스, 2020)

한국어 APA 웹페이지 / 내주-참고문헌

코로나19 여파 오픈마켓 관심도 급등…옥션 TOP•11번가 상승 . (2020). <http://ttlnews.com/article/travel_report/8055>.

[2] 한국어 APA 웹페이지 / 내주

(IT DONGA, 2020)

한국어 APA 웹페이지 / 내주-참고문헌

'카톡', '유튜브', '네이버', '쿠팡'… 2019년 사용자가 찾은 앱은? . (2020). https://it.donga.com/29974/.