REVIEW REVOLUTION

design specification



|  |  |
| --- | --- |
| Student Number | Name |
| 2017314380 | 윤성경 |
| 2014311058 | 김진태 |
| 2014314650 | 유종현 |
| 2013310608 | 정창호 |

Contents

[1. Preface 6](#_Toc23805734)

[1.1. Readership 6](#_Toc23805735)

[1.2. Document Structure 7](#_Toc23805736)

[A. Introduction 7](#_Toc23805737)

[B. System Architecture 7](#_Toc23805738)

[C. Protocol Design 7](#_Toc23805739)

[D. Database Design 7](#_Toc23805740)

[E. Testing Plan 7](#_Toc23805741)

[F. Development Plan 7](#_Toc23805742)

[G. Index 7](#_Toc23805743)

[2. Introduction 8](#_Toc23805744)

[2.1. Objectives 8](#_Toc23805745)

[2.2. Applied Diagram 8](#_Toc23805746)

[2.3. Applied Tool 11](#_Toc23805747)

[2.4. Project Scope 13](#_Toc23805748)

[3. System Architecture – Overall 14](#_Toc23805749)

[3.1. Objectives 14](#_Toc23805750)

[3.2. System Organization 14](#_Toc23805751)

[4. System Architecture – Frontend 18](#_Toc23805752)

[4.1. Objective 18](#_Toc23805753)

[4.2. Subcomponents 18](#_Toc23805754)

[A. Ranking 18](#_Toc23805755)

[5. System Architecture – Backend 22](#_Toc23805756)

[4. Functional Requirements – Frontend 23](#_Toc23805757)

[A. Customized Search result 23](#_Toc23805758)

[B. Item page 24](#_Toc23805759)

[C. Keyword 24](#_Toc23805760)

[D. Review page 25](#_Toc23805761)

[E. Review Post 26](#_Toc23805762)

[F. User Page 26](#_Toc23805763)

[5. Functional Requirements – Backend 27](#_Toc23805764)

[A. Login 27](#_Toc23805765)

[B. Review Analysis 28](#_Toc23805766)

[C. 연관상품 finder 28](#_Toc23805767)

[D. 상품 정보 제공 29](#_Toc23805768)

[E. 개인정보 변경 시 기능 29](#_Toc23805769)

[6. Non-functional Requirements 30](#_Toc23805770)

[A. Product requirements 30](#_Toc23805771)

[B. Organization requirements 31](#_Toc23805772)

[C. External requirements 31](#_Toc23805773)

[7. Scenario Examples 32](#_Toc23805774)

[A. Login and User Page Scenario 32](#_Toc23805775)

[B. Review Comparing Scenario 32](#_Toc23805776)

[B. System Models 34](#_Toc23805777)

[7.1. Context models 34](#_Toc23805778)

[A. Context Diagram 34](#_Toc23805779)

[B. Process Diagram 35](#_Toc23805780)

[7.2. Interaction models 36](#_Toc23805781)

[A. Use case Diagram 36](#_Toc23805782)

[B. Tabular Description of Use case Diagram 37](#_Toc23805783)

[7.3. Behavioral models 41](#_Toc23805784)

[A. 리뷰 분석 시스템 DFD 41](#_Toc23805785)

[B. 연관상품 finder system DFD 41](#_Toc23805786)

[C. 리뷰 작성 in Sequence Diagram. 42](#_Toc23805787)

[C. System Evolution 43](#_Toc23805788)

[1. As Component in Other System 43](#_Toc23805789)

[2. Deep Review Comparison 44](#_Toc23805790)

[3. Advanced Similar Product Finder System 44](#_Toc23805791)

[4. Advanced Review Analysis System 45](#_Toc23805792)

[D. Appendices 46](#_Toc23805793)

[1. Hardware requirements 46](#_Toc23805794)

[2. Database requirements 47](#_Toc23805795)

[A. User 47](#_Toc23805796)

[B. Item 47](#_Toc23805797)

[C. Review 47](#_Toc23805798)

[D. Keyword 47](#_Toc23805799)

[10. Index 48](#_Toc23805800)

[10.1. Tables 48](#_Toc23805801)

[10.2. Figures 49](#_Toc23805802)

[10.3. Diagrams 49](#_Toc23805803)

[11. References 51](#_Toc23805804)

# Preface

이 장에서는 예상 독자를 정의하고, 각 장의 내용을 요약한다. 또한 현재 version과 이전 version의 차이점에 대해 설명한다. 하지만 본 문서는 초기 버전이기 때문에 이 부분을 생략한다.

## Readership

본 문서는 다양한 독자에게 읽힐 것이라고 상정하고 있다. 따라서 각 부분을 서술하는 데 있어 어떠한 독자층을 상정하고 있는지를 설명한다.

## Document Structure

### Introduction

본 문서를 서술하는 데 사용된 다양한 다이어그램과 표현 도구들에 대해 설명하고, 본 소프트웨어 프로젝트가 다루는 시스템의 범위에 대해 서술한다.

### System Architecture

시스템과 각 서브시스템의 구조를 개괄적으로 기술하고, 시스템의 전체 기능이 각 서브시스템과 하드웨어에 어떻게 할당되었는지 설명한다.

### Protocol Design

기본적으로 시스템의 각 컴포넌트, 특히 프론트엔드 시스템과 백엔드 시스템간의 상호작용을 규정하는 인터페이스와 프로토콜을 어떻게 구성하는지에 대해 기술하고, 해당 인터페이스가 어떤 기술에 기반해 있는지 설명한다.

### Database Design

Requirements 문서에서 규정된 데이터베이스 요구 사항을 기반으로, 각 데이터 엔티티의 속성과 관계를 ER diagram을 통해 표현하고 최종적으로 Relational Schema, SQL DDL를 작성한다.

### Testing Plan

미리 작성된 Test를 이용해, verification 과 validation을 시행한다. 이 Test 작성에 대한 계획을 설명한다.

### Development Plan

시스템을 구현하는 데 필요한 개발 도구와 프로그래밍 언어, 라이브러리 등의 개발 환경에 대해 설명하고, 시스템 개발 일정을 기술한다.

### Index

본 문서에서 사용된 그림, 표, 다이어그램 등의 색인을 기술한다.

# Introduction

## Objectives

이 장에서는 본 시스템이 등장하게 된 배경과 필요성에 대해 설명하고, 시스템의 대략적인 기능에 대해 설명한다. 또한 해당 시스템을 개발함으로써 기대되는 효과에 대해 설명한다

## Applied Diagram

1. **UML**

UML is a general purpose and developmental modelling language and technique that combines different aspects of a system to represent relations, processes or results of an overall model or system. It is essential to mention that we have used it thoroughly in this document to visualize the workflow of the system.

Since it provides different modelling techniques and a handful subset of diagrams. It can be efficiently used to provide means of communication between developers and users as it covers wide range of symbols and definitions and it consists of the following diagrams: Package Diagram, Deployment Diagram, Class Diagram, State Diagram, Sequence Diagram and ER Diagram.

1. **Package Diagram**

Package diagrams are kind of structural diagrams which show the arrangement and organization of model elements. Package diagram can show both structure and dependencies between sub-systems or modules in a more abstract way than other types of UML diagrams. This abstraction leads to the use of package diagrams in simplifying complex class diagrams by grouping them in packages.

1. **Deployment Diagram**

The deployment diagram describes the physical deployment of information generated by the software program (artifact)on hardware components.

Deployment diagrams are made up of several UML shapes. The three-dimensional boxes, known as nodes, represent the basic software or hardware elements, or nodes, in the system. Lines from node to node indicate relationships, and the smaller shapes contained within the boxes represent the software artifacts that are deployed.

1. **Class Diagram**

It is a diagram that is used to showcase the object classes of a system and the relationship between classes. One of the most fundamental reasons we are using it is because it provides a clear distinction between each class and show the hierarchy and dependency between them.

As far it goes for the inner structure of Class diagram, it consists of some fields indicating some variables, class methods and links or associations between classes.

1. **State Diagram**

State Diagram is a technique to represent different states of a system and all possible next states based on some particular stimuli which triggers the change of the state.

This kind of diagram is very important to analyze different scenarios of the system as the states are represented as nodes and events as arcs which helps in identifying the behavior of the object classes defined in class diagram.

1. **Sequence Diagram**

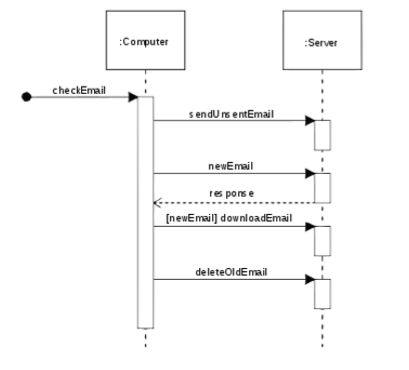


Figure : Example of Sequence Diagram

A sequence diagram to represent the interactions between the actors and objects of the system. To be more specific, the goal of this diagram is present the sequence of interaction and processes that take place in a specific use case instance so that a result could be generated. It is important to notice that the direction of the arrows here is essential to indicate the correct flow of actions.

1. **ER Diagram**



Figure : Example of ER Diagram

ER Diagram 은 Entity 가 가지고 있는 속성과 Entity 간의 관계를 나타낸 다이어그램이다. 이 다이어그램은 주로 데이터베이스를 설계하는 데 사용되며, 해당 다이어그램을 기반으로 Relational Schema, SQL DDL을 작성하게 된다.

## Applied Tool

* 1. **Draw.io**



Figure : Draw.io Logo

Draw.io는 온라인 모델링 툴로서 많은 기본 템플릿과 도형을 제공하기 때문에 사용자가 직접 다이어그램에 사용하기 위해 도형을 만들 필요가 없다. 또한 도형 간 연결선을 간단하게 만들 수 있고, 격자에 위치를 맞출 수 있기 때문에 도형을 정렬하기 편리하다. 이 문서에서 사용된 대부분의 다이어그램은 본 도구로 작성되었다.

* 1. **PowerPoint**



Figure : Powerpoint logo

Powerpoint는 그래픽 프레젠테이션 툴이다. 주로 발표용으로 사용되지만 내장된 도형 작성 기능이 매우 강력하기 때문에 draw.io에서 만들기 힘든 복잡한 다이어그램을 작성하기 위해 사용하였다.

* 1. **ERDPlus**



Figure : ERDPlus Logo

ERDPlus 는 ER Diagram 을 간단한 버튼 클릭으로 생성할 수 있게 해 주는 온라인 툴이다. 적은 노력으로 ER DIagram을 draw.io에 비해 간단하게 만들 수 있기 때문에 ER Diagram을 작성하는 데 사용하였다.

## Project Scope

본 시스템은 상품 평점의 신뢰도가 낮은 상황에서 사용자들이 일일이 직접 리뷰를 읽고 신뢰성을 측정해야 했던 기존의 오픈마켓 서비스들의 단점을 극복하기 위해 리뷰에 자연어 처리 시스템을 도입하여 리뷰의 신뢰성을 판단할 수 있는 기반 지식 없이도 사용자들이 믿을 수 있는 리뷰에 기반한 상품 선택을 도와주는 시스템이다. 본 시스템의 핵심 기능은 리뷰의 분석 기능이며, 해당 기능을 중심으로 각 서브시스템들이 상호작용하도록 설계하였다.

먼저 Frontend System 은 시스템과 사용자와의 상호작용을 담당하며, Backend System Frontend System 에서 오는 데이터 요청에 응답하고 Review Collecting System, Review Analysis System 를 실행시키는 역할을 담당한다. 사용자가 리뷰를 작성하거나 Review Collecting System 에서 정기적으로 오픈마켓에서 리뷰의 목록을 가져오면, Review Analysis System 은 리뷰를 분석해 리뷰의 긍정 평가 정도와 중요도를 측정한다. Review Analysis System이 분석한 리뷰를 데이터베이스에 전달하면, Item Ranking System과 Recommendation System은 추가된 정보를 바탕으로 각 상품의 평가 수치와 추천 카테고리를 수정한다.

# System Architecture – Overall

## Objectives

이번 챕터에서는 본 시스템의 전체적인 구조를 설명한다. 시스템 전체의 구조와 각 서브시스템의 개략적인 구조, 서브시스템 간의 관계를 서술한다.

## System Organization

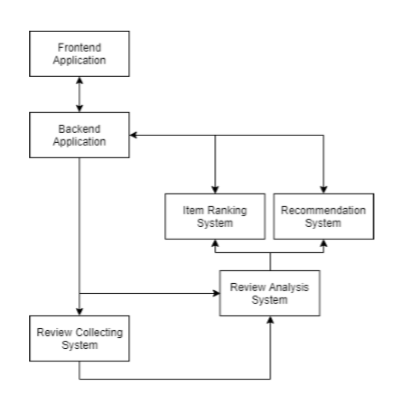


Diagram : Overall System Organization

본 서비스는 클라이언트-서버 모델을 적용해 설계했으며, Frontend Application이 사용자와의 모든 상호작용을 맡고, 프론트엔드 애플리케이션과 백엔드 애플리케이션은 JSON을 기반으로 한 HTTP 통신으로 데이터를 송수신한다. 백엔드 애플리케이션은 프론트엔드로부터 들어오는 각 요청을 컨트롤러로 분배하고, 필요한 객체 정보를 데이터베이스로부터 가져와 JSON 포맷으로 가공한 뒤 전달한다.

백엔드 애플리케이션은 일정 시간마다 Review Collecting System을 실행하는데, 요청을 받은 Review Collecting System 은 저장되어 있는 목표 크롤링 사이트의 목록과 목표 상품 목록을 이용해 해당 사이트에서 리뷰의 정보를 수집한다. 리뷰 정보 수집이 완료되면, 최소한의 가공을 거친 리뷰 목록을 Review Analysis System 에게 전달하고, Review Analysis System 은 전달받은 리뷰를 Google Natural Language API 를 이용해 분석 후 필요한 정보를 데이터베이스에 저장한다. 이후 Item Ranking System 과 Recommendation System 이 업데이트된 리뷰 데이터베이스를 이용해 각 상품의 점수 순위와 추천 카테고리를 업데이트한다. 이후 다시 사용자가 정보를 요청할 경우 업데이트된 정보를 전달한다.

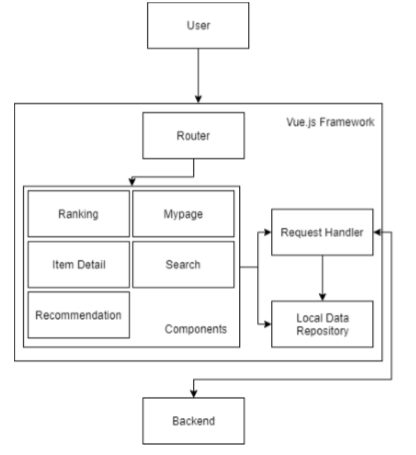
* 1. **Frontend Application** 

Diagram : System Architecture – Frontend

사용자와의 상호작용을 전담하는 시스템으로, Vue.js 프레임워크를 통해 각 컴포넌트를 관리한다. Ranking, Mypage, Item Detail, Search, Recommendation 컴포넌트가 존재하며, 컴포넌트간 공유하는 Shared Resource 의 경우 Local Data Repository 서브시스템에서 관리한다. 각 컴포넌트가 백엔드와의 통신을 전담하는 Request Handler에게 필요한 데이터를 요청하면, Request Handler 는 미리 정해 둔 적절한 프로토콜로 요청을 변형해 백엔드에 전달하게 된다.

* 1. **Backend Application**

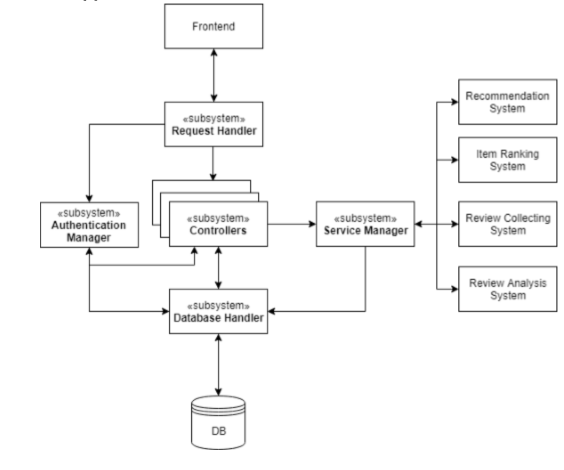


Diagram : System Architecture – Backend

# System Architecture – Frontend

## Objective

전체 시스템 아키텍처 중 사용자와의 상호작용을 담당하는 프론트엔드 시스템의 구조와 각 컴포넌트의 구성, 컴포넌트간의 관계를 서술한다.

## Subcomponents

### Ranking

1. Class Diagram

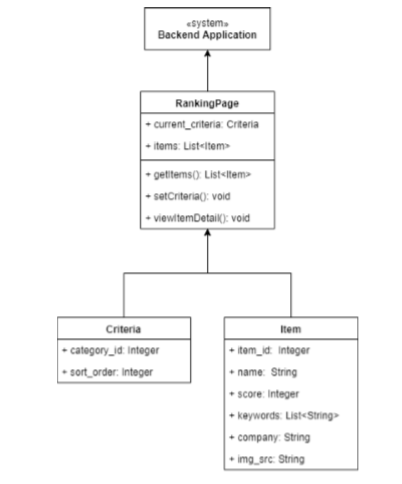


Diagram : System Architecture - Frontend - Ranking

* 1. RankingPage – 랭킹 페이지 객체

1. attributes

* current\_criteria: 랭킹 페이지의 상품 목록을 불러오는 데 사용하는 검색기준
* items: 랭킹 페이지에서 로드되어 있는 상품 목록

1. methods

* getItems(criteria: Criteria): 검색 조건에 해당하는 상품 목록을 백엔드 서버에서 가져온다.
* setCriteria(criteria: Criteria): 검색 조건을 설정한다.
* viewItemDetail(item\_id: Integer): 특정 상품의 자세한 정보를 조회한다. (Item Detail 컴포넌트로 이동)
  1. Criteria – 검색 조건 객체(DTO)

1. attributes

* category\_id: 상품 분류 카테고리 ID
* sort\_order: 정렬 순서(오름차순, 내림차순)
  1. Item – 상품 객체(DTO)1

1. attributes

* item\_id: 상품 ID
* name: 상품 이름
* score: 상품 평가 점수
* keywords: 상품 연관 키워드 목록
* company: 상품 제조사
* img\_src: 상품 썸네일 사진 주소

1. Sequence Diagram

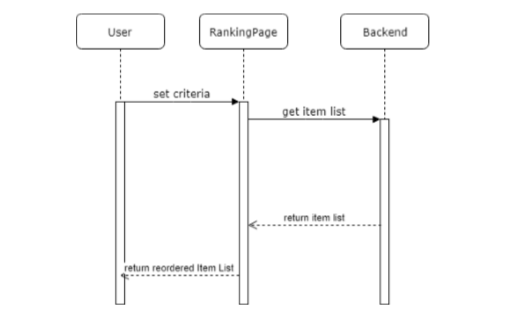


Diagram : System Architecture – Frontend - Ranking - Sequence Diagram

# System Architecture – Backend

이 장에서는User Requirements Definition에 간략히 서술되어 있는 요구사항을 기반으로, 기능적 요구사항, 비기능적 요구사항, 기타 요구사항을 더 detail하게 설명한다. 필요하다면 user requirements에 없는 요구사항을 추가할 수 있다. 이때 본 장은 개발 프로세스 전반에 사용되는 장이므로 structured natural language, graphical notations등을 활용해 체계적이고 명확하게 서술한다.

## Functional Requirements – Frontend

### Customized Search result

|  |  |
| --- | --- |
| Name | Customized Search Result |
| Description | 사용자가 검색한 물품의 리스트를 보여준다. |
| Inputs | 사용자의 검색 키워드 |
| Source (Input) | 사용자의 입력 |
| Outputs | 검색 키워드와 관련관 상품 리스트 |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 사용자가 구매하고자 하는 상품을 검색했을 때 상품들을 리스트로 보여준다. 보여주는 상품은 검색 키워드와 관련이 있는 상품 중에서 얼마나 많이 판매된 상품인지, 사용자가 관심 있을 법한 상품인지의 정도를 종합적으로 판단해 점수를 매기고 순서대로 나열한다. 각 상품은 이미지, 이름, 키워드와 평가 정보 등의 간단한 설명과 함께 보여지고 클릭했을 때 상품의 상세 페이지로 넘어갈 수 있다. |
| Requirements |  |
| Pre-condition |  |
| Post-condition |  |

Table : Customized Search result

### Item page

|  |  |
| --- | --- |
| Name | Item Page |
| Description | 사용자가 클릭한 상품의 상세 페이지를 보여준다. |
| Inputs | 사용자가 클릭한 상품 ID |
| Source (Input) | 사용자의 입력 |
| Outputs | 상품의 상세 페이지 |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 사용자가 상품을 클릭했을 때 상품의 상세 페이지를 보여준다. 상세 페이지 안에는 판매자가 게시한 제품의 상세 정보가 들어가게 되고 그 외에도 리뷰 정보가 들어가게 된다. 이 리뷰 정보에 대한 자세한 사항은 Review Page에서 다룰 것이다. |
| Requirements |  |
| Pre-condition |  |
| Post-condition |  |

Table : Item Page

### Keyword

|  |  |
| --- | --- |
| Name | Keyword |
| Description | 상품과 리뷰의 키워드를 보여준다. |
| Inputs | 상품 ID, 리뷰 ID |
| Source (Input) |  |
| Outputs | 상품, 리뷰의 상세 페이지 |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 상품과 리뷰는 키워드와 함께 표시된다. 리뷰의 키워드는 리뷰를 분석하여 키워드를 찾아내고 이를 초록색부터 빨간색까지 색으로 표시한다. 키워드가 상품에 긍정적인 의미로 사용된 경우 초록색, 부정적인 의미로 사용되는 경우 빨간색으로 표시된다. 상품의 키워드는 리뷰 키워드를 종합하여 키워드 빈도나 구매 영향력에 따라 선정한다. |
| Requirements |  |
| Pre-condition |  |
| Post-condition |  |

Table : Key word

### Review page

|  |  |
| --- | --- |
| Name | Review Page |
| Description | 상품의 리뷰 관련 정보를 보여준다. |
| Inputs | 상품 ID |
| Source (Input) | 사용자가 상품 페이지 안에서 리뷰 페이지를 보게 된다. |
| Outputs | 유사 상품들과의 리뷰 비교 정보, 해당 상품의 전체 리뷰 정보 |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 사용자가 상세 페이지 안에서 상품정보를 보기 전에 리뷰 페이지를 발견할 수 있다. 선택한 상품과 유사한 상품을 보여주고 각 상품의 아래에는 상품의 키워드, 대표 리뷰를 표시한다. 유사한 상품을 선정할 때에는 카테고리, 가격, 고객들의 검색기록, 키워드 정보 등 여러 가지 특성을 고려한다. 대표 리뷰는 사용자의 추천 수에 따라 3~5개를 선정한다. 리뷰 비교 아래에는 해당 상품의 전체 리뷰를 자세히 보여준다. 리뷰를 한 화면에 보여줄 수 없는 경우 navigation방식을 사용해서 일정 단위로 보여주고, 로드 하는 방식을 사용한다. 사용자는 리뷰에 추천, 비추천을 할 수 있으며 (추천 수-비추천 수)값에 따라 리뷰가 보여지는 순서가 결정된다. |
| Requirements |  |
| Pre-condition |  |
| Post-condition |  |

Table : Review Page

### Review Post

|  |  |
| --- | --- |
| Name | Review Post |
| Description | 상품의 리뷰를 등록하고 삭제한다. |
| Inputs | 상품 리뷰 문자열 |
| Source (Input) | 사용자가 리뷰 등록 혹은 삭제 클릭 |
| Outputs |  |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 사용자가 상품에 리뷰를 등록할 수 있는 기능이다. 리뷰를 등록할 수 있는 사용자는 상품을 구매한 사용자로 한정하며, 한 명의 사용자는 한 개의 리뷰만 등록할 수 있다. 사용자는 자신이 등록한 리뷰를 삭제할 수 있다. |
| Requirements | Login user, buyer. |
| Pre-condition |  |
| Post-condition |  |

Table : Review Post

### User Page

|  |  |
| --- | --- |
| Name | Review Evaluation |
| Description | 사용자가 자신의 정보를 관리할 수 있다. |
| Inputs | 로그인 된 계정의 비밀번호 |
| Source (Input) | 사용자가 my page로 들어가는 버튼을 누르고 자신의 비밀번호를 입력한다. |
| Outputs | 사용자의 계정 정보 |
| Destination (Output) | 사용자의 웹 브라우저 |
| Action | 사용자는 자신의 정보를 확인하고 관리할 수 있다. 비밀번호를 바꾸거나 자신의 상품 구매 정보를 확인할 수 있다. 맞춤 검색을 위해 자신이 선호하는 키워드를 미리 설정할 수 있다. 또 자신이 등록한 리뷰를 찾아보고 수정 삭제할 수 있다. |
| Requirements | Login user. Correct password |
| Pre-condition |  |
| Post-condition |  |

Table : User Page

## Functional Requirements – Backend

### Login

|  |  |  |
| --- | --- | --- |
| Name | Login Function | |
| Description | 사용자 확인 및 인증 | |
| Inputs | ID | 사용자 Id (string type) |
| Password | 사용자 Password(string type) |
| Source (Input) | HTTP request handler | |
| Outputs | 해당 사용자 정보 객체 | |
| Destination (Output) | 사용자의 웹 브라우저 | |
| Action | 입력된 ID를 바탕으로 Database에서 사용자 정보를 찾는다. 입력된 Password를 hash한 값과 Database에 저장된 hash값을 비교한다. 일치할 경우 생성한session의 session Id 와 함께 해당 사용자가 필요로 하는 정보 객체를 User의 웹 페이지로 response에 담아 전송한다. | |
| Requirements | ID와 Password 모두 일치해야만 함 | |
| Pre-condition | 사용자는 로그인 되어 있지 않고 고객 인증이 필요한 서비스와 고객 맞춤형 서비스를 사용할 수 없음 | |
| Post-condition | 사용자는 고객 인증이 성공적으로 되어서 기존 서비스와 더불어 고객 맞춤형 서비스를 사용할 수 있음 | |
| Side effects | 고객이 보는 웹 페이지의 일부에 고객의 이름 등의 정보를 보여주게 됨 | |

Table : Login Function

### Review Analysis

|  |  |
| --- | --- |
| Inputs | Posting 된 Review, Review Data Set |
| Source (Input) | HTTP request handler, Review analyzing system |
| Outputs | 분석된 Review |
| Destination (Output) | Database |
| Action | 포스팅 된 리뷰를 자연어 처리 API 등을 사용해서 분석한다. 이 분석을 통해 keyword, critique 점수를 얻어낸다. 얻어진 정보들은 Review와 함께 Database에 저장된다. Database에 update되는 과정에서 전체 리뷰 overview, summary 정보도 같이 update된다. |
| Requirements | 리뷰에서 키워드를 올바르게 추출할 수 있는 효율적인 NLP model or system |
| Pre-condition | 해당 리뷰가 포함되지 않은 상품 리뷰 목록과 리뷰 정보 |
| Post-condition | 해당 리뷰가 포함된 상품 리뷰 목록과 리뷰 정보 |
| Side effects | Review 포스팅이 적용될 때까지 시간이 조금 더 걸린다. |

Table : Review Analysis System

### 연관상품 finder

|  |  |
| --- | --- |
| Inputs | 상품 리스트 |
| Source (Input) | Database |
| Outputs | 각 상품에 따른 연관 상품 리스트 |
| Destination (Output) | Database |
| Action | 리뷰 분석 시스템에서 분석을 완료한 리뷰의 리스트를 저장한다. 리뷰 제목, 작성자, 출처 사이트, 내용, 키워드, 긍정 평가 정도, 중요도를 포함해 데이터베이스에 기록한다. |
| Requirements | 상품에 비교를 위한 정보들이 존재할 것 |
| Pre-condition | 상품의 연관상품이 존재 또는 존재하지 않음 |
| Post-condition | 상품의 연관상품이 존재. 낮은 확률로 존재하지 않을 수 있음. |
| Side effects | 해당 프로세스가 진행된 후에 연관 상품이 달라질 수 있다. 따라서 고객이 보는 연관 상품이 더 이상 연관상품이 아니게 될 수 있다. |

Table : 연관상품 finder system

### 상품 정보 제공

|  |  |
| --- | --- |
| Inputs | 사용자의 상품 정보 Request, 상품 정보 리스트 |
| Source (Input) | HTTP request handler, Database |
| Outputs | 해당 상품 정보 |
| Destination (Output) | 사용자의 Web page |
| Action | 상품 정보의 요청이 들어오게 되면 HTTP request handler가 해당 상품을 Database에서 찾는다. 존재할 경우 해당 상품 정보를 Response에 담아 보낸다. 존재하지 않을 경우 해당 예외를 위한 예외처리 페이지를 보낸다. |
| Requirements | 해당 상품이 데이터베이스에 존재할 것 |
| Pre-condition | - |
| Post-condition | - |
| Side effects | - |

Table : 품 정보 제공

### 개인정보 변경 시 기능

|  |  |
| --- | --- |
| Inputs | 사용자의 개인정보 변경 Request, 개인정보 리스트 |
| Source (Input) | HTTP request handler, Database |
| Outputs | 변경된 개인정보, 사용자에게 보낼 Response |
| Destination (Output) | DataBase, 사용자의 웹페이지 |
| Action | 개인정보 변경 Request가 오게 되면 HTTP request handler가 해당 ID을 포함하는 정보 객체를 Database에서 찾는다. 존재할 경우 변경 Request에 담긴 정보로 객체를 업데이트한 후 Database에 저장한다. 그 후 사용자의 웹페이지로 변경되었음을 알리는 Response 페이지를 보낸다. |
| Requirements |  |
| Pre-condition | 변경 전 사용자 정보가 데이터베이스에 존재 |
| Post-condition | 변경 후 사용자 정보가 데이터베이스에 존재 |
| Side effects |  |

Table : 개인정보 변경 시 기능

## Non-functional Requirements

### Product requirements

1. Usability

사용자들이 키워드로 정리된 리뷰로 직관적인 이해에 긍정적 영향을 받아야 한다. 이해를 측정하는 방식은 사용자의 체감을 설문조사를 통해서 test한다. 우리 시스템을 쓴 사용자들이 실제 설문에서 80퍼센트 이상 기존 시스템보다 이해에 도움이 된다고 답해야 하며, 5퍼센트 미만이 기존 시스템보다 비효율적이라고 대답해야 한다.

다양한 크기의 기기에 맞춰 동적으로 크기를 변경하는 반응형 웹 디자인을 채택해야 한다.

1. Performance

본 시스템의 자원 사용량과 그에 해당하는 비용이 예상되는 매출 증가액의 50%를 넘어가면 안 된다. 요청의 개수가 많아지더라도 충분히 합리적인 수준으로 자원 사용량을 유지해야 한다(exponential complexity 미만).

1. Dependability

서버와 데이터베이스 연결이 잘 되어야 한다. 많은 요청이 들어오더라도 안정적인 설계를 구성해야 한다. 서버에 동시에 n건 이상의 request가 들어오더라도 오류 없이 안정적인 처리를 할 수 있도록 test한다.

1. Security

리뷰 시스템은 모든 user input을 주요 문자인 <,>,’,”,\,/ 등에 대해서escape해야 하며, 모든 비밀번호는 Database에 hash된 값으로 저장해서 악의적인 공격이나 비밀번호 유출에 대비한다.

### Organization requirements

1. Operational  
   다양한 구조의 웹 시스템에 적용될 것을 가정하므로, 최대한 architecture dependency가 적도록 overhead를 줄인 간결한 시스템으로 구성한다. 그렇게 해야 복잡한 구조의 시스템에 병합되더라도 바로 적용하고 변경하기 쉽기 때문이다.
2. Development  
   짧은 기간 내에 안정적인 구조로 개발해야 하므로 Parallel Development가 가능하고, architecture design이 보다 완성도 있는 plan driven의 개발 프로세스를 사용한다. 프론트엔드와 백엔드를 나눠서 독립적으로 구현하며, 각각 독립적으로 구동할 수 있게 설계해야 한다.

### External requirements

1. Ethical review management  
   현 시스템은 리뷰, 즉 자유로운 의견 표출을 다루기 때문에 다른 사용자가 읽기에 윤리적으로 문제가 되는 내용이 게시되었을 때, 많은 사람들에게 피해를 줄 수 있기에 그런 내용을 자체적으로 검증하고 차단하기 위한 장치가 필요하다. 이 시스템에서는 그런 장치를 직접 개발하지는 않지만 review posting code에 filtering component가 들어갈 수 있는 자리를 명시해서 실제 시스템에서 사용될 경우에 쉽게 filtering 기능을 사용할 수 있도록 지원할 것이다.

## Scenario Examples

### Login and User Page Scenario

1. Initial Assumption

유저가 이미 회원가입을 마친 상황에서 로그인을 하려고 하고, 로그인 후 유저 페이지에서 비밀번호를 바꾸려고 하거나 기본 키워드 set을 바꾸려고 한다.

1. Normal flow of events

유저가 아이디와 비밀번호를 제출하면 서버는 비밀번호를 hash한 뒤 데이터베이스 안의 비밀번호 hash값과 비교한 뒤 같으면 login을 허용한다. 로그인 후에 유저는 유저 페이지로 들어가기 위해 한번 더 비밀번호를 입력하고, 유저 페이지에서 바꿀 비밀번호나 바꿀 기본 키워드 set을 제출한다. 서버에서는 제출한 것이 형식에 맞는지 확인한 후 그에 맞게 데이터베이스를 업데이트한다.

1. What can go wrong

유저가 제출한 아이디, 비밀번호가 데이터베이스의 값과 대응되지 않거나, 제출한 새 비밀번호나 기본 키워드 set이 허용되는 형식이 아닌 경우에는 그에 맞는 에러 메시지를 띄운다.

1. System state on completion

유저는 로그인 된 상태에서, 원하는 새 비밀번호 혹은 새 기본 키워드 set을 갖게 된다.

### Review Comparing Scenario

1. Initial Assumption

사용자는 특정 종류의 상품을 사기 위해서 쇼핑몰에 들어왔고, 이미 회원가입이 되어 있으며, 로그인을 마쳤다. 사용자는 특정 종류의 상품과 그 유사 상품들에 대해 어떤 것을 살 지 아직 정하지 못한 상황이다.

1. Normal flow of events

사용자가 상품 검색 창에 상품의 종류를 입력하면, 그 결과 페이지엔 사용자의 기본 선호 키워드 set에 따라서 키워드를 고려해 정렬된 상품 목록이 나온다. 상품 목록에는 각각 상품의 리뷰 총점, 주요 리뷰 키워드, 각 리뷰 키워드 별 긍정도가 표시되며, 사용자는 별도로 마련된 공간에서 선호 키워드를 바꿔가며 원하는 키워드에 맞는 상품을 찾아볼 수 있다. 사용자가 관심 있는 상품을 클릭하면 해당 상품의 상세 페이지로 들어가며, 그 페이지에는 상품의 세부정보와 함께 유사 상품의 리뷰를 다같이 summary로 비교할 수 있는 공간이 존재한다. 총 리뷰 점수, 리뷰의 긍정도, 주요 키워드, 주요 키워드 별 리뷰의 긍정도, 최다 추천 리뷰 상위 5개 정도의 리뷰 summary가 있고, 그 아래에는 상품의 자세한 리뷰가 있다. 각 키워드를 클릭하면 키워드 별로 긍정도, 키워드 별 리뷰를 모아서 볼 수 있다. 또한 각 리뷰에도 키워드가 붙어 있어서 리뷰를 한눈에 요약해서 보기 편하다. 유사 상품의 summary를 누르면 해당 상품의 상세 페이지로 넘어가고, 사용자는 이를 통해서 원하는 상품을 고를 수 있다.

1. What can go wrong

키워드 추출의 품질과 리뷰 숫자가 부족해서 충분한 키워드를 만들어내지 못한 경우 리뷰로 상품을 정의하는 특성 상 리뷰 중심 분석 시스템의 의미가 사라질 수 있다. 이런 경우는 키워드를 표시하는 자리에 [첫 번째 리뷰를 쓸 기회!] 라는 키워드를 달아둔다. 쇼핑몰 정책적으로 첫 30개의 리뷰에 incentive를 주거나, 추천이 많이 달린 리뷰어에게 혜택이나 리뷰의 색 변경 등의 긍정적 피드백을 제공하는 방식으로 리뷰를 장려할 수 있을 것이다.

1. System state on completion

사용자가 원하던 종류의 유사 상품들을 모두 비교한 뒤 원하는 상품을 결정할 수 있게 된다.

결제 서비스는 이 시스템에서 따로 구현하는 것이 시스템의 유연성을 망치고, overhead일 것 같아서 구현하지 않고, 리뷰 서비스에 집중하려고 한다. (결제 서비스는 적용될 웹 사이트에 이미 구현되어 있을 것이다. 만약 없다면 결제 모듈을 새로 가져오는 방식으로 구현할 수 있다.)

# System Models

이번 장에서는 각 시스템 컴포넌트와 전체 시스템, 그리고 시스템을 둘러싼 환경 사이의 관계를 여러 가지 Diagram을 통해 기술한다.

## 7.1. Context models

### A. Context Diagram

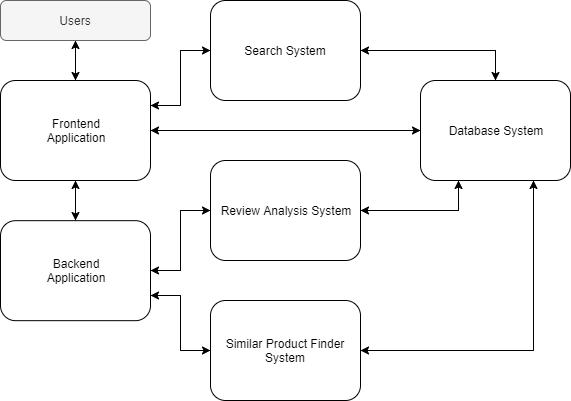


Diagram : Overall context diagram

### B. Process Diagram

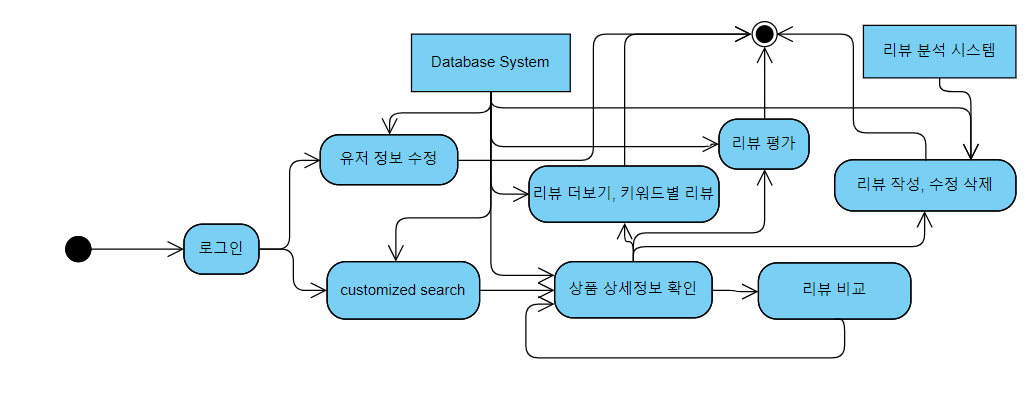


Diagram : Overall process diagram

## 7.2. Interaction models

### Use case Diagram

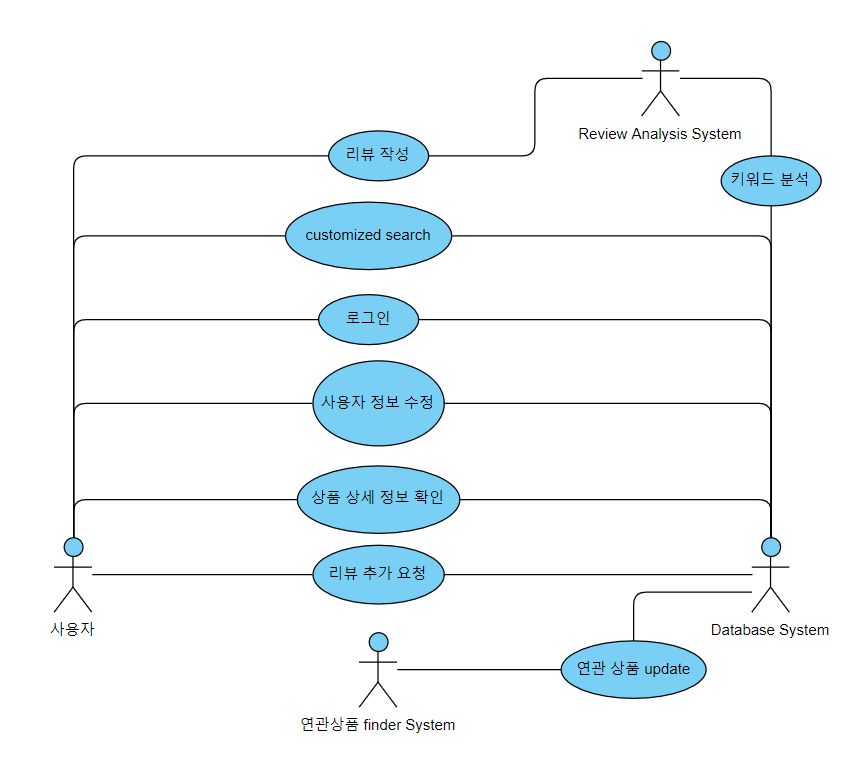


Diagram : Use case diagram

### Tabular Description of Use case Diagram

1. 로그인

|  |  |
| --- | --- |
| Use case | 로그인 |
| Actor | User, DB system |
| Description | 사용자가 시스템에 등록된 유저인지 확인하는 과정이다. |
| Trigger | 사용자가 ID와 PW를 입력하고 로그인 버튼을 클릭한다.. |
| Success Response | 입력된 ID가 DB system에 존재하고, ID의 PW가 입력된 PW와 같은 경우 로그인 성공 메시지를 출력하고 홈페이지로 이동한다. |
| Failure Response | 입력된 ID가 DB system에 존재하지 않거나 ID의 PW와 입력된 PW가 다른 경우 “가입하지 않은 아이디이거나, 잘못된 비밀번호입니다” 라는 메시지를 출력한다. |

Table : 로그인

1. 사용자 정보 수정

|  |  |
| --- | --- |
| Use case | 사용자 정보 수정 |
| Actor | User, Database system |
| Description | 사용자가 선호하는 키워드를 수정한다. |
| Trigger | 사용자가 선호하는 키워드를 수정하고 수정 버튼을 클릭한다. |
| Success Response | 사용자가 수정한 키워드를 DB system에 저장하고, “정상적으로 저장되었습니다.” 메시지를 출력한다. |
| Failure Response | - |

Table : 사용자 정보 수정

1. 리뷰 작성, 키워드 분석

|  |  |
| --- | --- |
| Use case | 리뷰 작성, 키워드 분석 |
| Actor | User, Review Analysis System, DB system |
| Description | 사용자의 작성된 리뷰에서 키워드를 추출하여 함께 저장한다. |
| Trigger | 사용자가 상품의 리뷰를 작성하고 입력 버튼을 클릭한다.. |
| Success Response | Review Analysis System가 리뷰를 분석해 키워드를 추출하고 DB system에 함께 저장한다. 등록한 리뷰가 포함되도록 상품 페이지를 갱신한다. |
| Failure Response | 로그인 되지 않은 사용자, 상품을 구매하지 않은 사용자, 이미 리뷰를 등록한 사용자는 리뷰를 등록할 수 없으며 각 경우에 맞는 알림 메시지를 출력한다. |

Table : 리뷰 작성, 키워드 분석

1. Customized Search-1

|  |  |
| --- | --- |
| Use case | Customized Search-1 |
| Actor | User, DB system |
| Description | 사용자가 선호 키워드를 선택하고 상품을 검색하면 관련된 상품의 리스트를 보여준다. |
| Trigger | 사용자가 선호 키워드 설정 후, 검색 키워드를 입력하고 검색 버튼을 클릭한다. |
| Success Response | 선호 키워드, 검색 키워드와 관련이 높은 상품의 리스트를 보여준다. |
| Failure Response | 사용자가 요청한 정보와 관련된 상품이 없는 경우, “(검색 키워드)에 대한 검색 결과가 없습니다”라는 메시지를 출력한다. |

Table : Customized Search-1

1. Customized Search-2

|  |  |
| --- | --- |
| Use case | Customized Search-2 |
| Actor | 사용자, Client system (생략) |
| Description | 사용자가 사전에 또는 검색 후 상품 리스트가 띄워진 창에서 설정한 키워드를 바탕으로 검색 시 상품 및 키워드의 순서를 결정한다.. |
| Trigger | 상품 검색 후, 상품 리스트가 띄워진 창에서 다시 선호하는 키워드 순서를 정하고 확인 버튼을 누른다. |
| Success Response | 키워드에 대한 점수가 높은 상품을 먼저 보여준다.  설정된 키워드를 다른 키워드에 앞서 보여준다. |
| Failure Response | - |

Table : Customized Search-2

1. 연관 상품 Update

|  |  |
| --- | --- |
| Use case | 연관 상품 Update |
| Actor | Database, 연관 상품 finder system |
| Description | 상품에 연관이 있는 유사 상품을 매칭해준다. |
| Trigger | 일정 시간마다 DB를 업데이트하기 위해 실행한다. |
| Success Response | 상품에 연관이 있는 유사 상품을 매칭하여 사용자가 연관 상품을 보고자 할 때 매칭된 상품이 보이게 한다. |
| Failure Response | 찾지 못한 경우 매칭된 상품이 없음을 사용자가 알 수 있도록 예외 처리한다. |

Table : 연관 상품 Update

1. 리뷰 추가 요청

|  |  |
| --- | --- |
| Use case | 리뷰 추가 요청 |
| Actor | User, Database |
| Description | 사용자가 리뷰 더 보기 요청을 보낸 경우 요청에 알맞은 리뷰를 보내준다. |
| Trigger | 사용자가 리뷰 더 보기 버튼이나, 키워드 별로 리뷰 보기 버튼을 눌렀을 때 실행된다. |
| Success Response | 사용자가 원하는 리뷰를 DB에서 찾아서 클라이언트에서 보여준다. |
| Failure Response | 네트워크 연결이 잘 되지 않거나 해당 리뷰가 없는 경우에는 그에 따른 오류 메시지를 출력한다. |

Table : 리뷰 추가 요청

1. 상세정보 확인

|  |  |
| --- | --- |
| Use case | 상세정보 확인 |
| Actor | User, Database |
| Description | 사용자가 원하는 상품의 상세정보를 보여준다. |
| Trigger | 사용자가 상품을 검색한 후 원하는 상품을 클릭한다. 또는 상세정보 안의 리뷰 비교 페이지에서 유사 상품 중 원하는 상품을 클릭한다. |
| Success Response | 사용자가 원하는 상품의 상세정보와 함께 유사 상품의 간략한 정보들을 DB에서 가져와서 보여준다. |
| Failure Response | 필요한 정보가 없는 경우에는 그 정보가 들어갈 자리에 알맞은 오류 메시지를 출력한다. |

Table : 상세정보 확인

## 7.3. Behavioral models

이 장에서는 본 시스템에서 가장 중점으로 두는 하위 시스템인 ‘리뷰 분석 시스템’ 과 ‘연관 상품 finder 시스템’에 초점을 두고 behavioral models을 작성하였다.

### 리뷰 분석 시스템 DFD

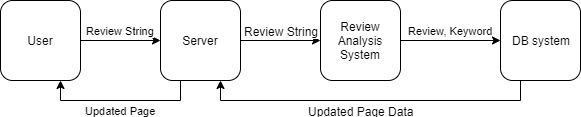


Diagram : DFD of Review Analysis System

### 연관상품 finder system DFD

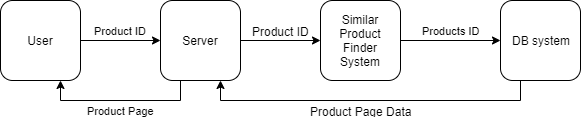


Diagram : DFD of Similar Product Finder System

### 리뷰 작성 in Sequence Diagram.

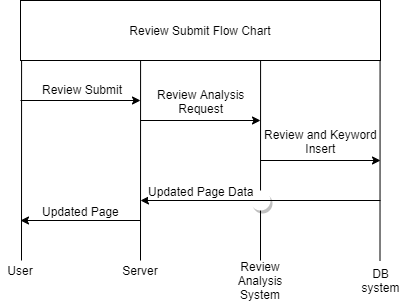


Diagram : Sequence Diagram of Review Submit

# System Evolution

이번 장에서는 시스템의 기반이 되는 fundamental한 가정에 대해 서술하고, 시스템 운영 과정에서 발생할 수 있는 하드웨어의 변화, client의 요구사항 변화 등에 대해 예측하고 그에 대한 해결 방안을 설명한다. 이를 통해 시스템의 추후 수정 시 발생할 수 있는 부수적인 설계 변경의 여지를 없애고 수정 비용을 낮춘다.

### As Component in Other System

Review Revolution은 다른 시스템에서 유사 상품의 리뷰를 비교할 수 있는 기능으로 사용자의 결정을 돕는 Component로 설계되었다. 이에 맞게 핵심 sub-system인 Review Analysis System, Similar Product Finding System위주로 Focus해서 requirement definition이 이뤄졌으며, 나머지 시스템들은 이미 대부분의 웹 사이트에서 자체적으로 구현된 시스템이기 때문에 Review Revolution과의 직접적인 interaction을 하는 시스템 외에는 Overhead로 여기고 제외하여 System boundary를 설정하였다. 실제 운영 시에는 다양한 시스템과 상호작용을 할 것이기 때문에 그에 맞게 flexible하고 간결하게 interaction을 정의해서 Component 적용 시의 추가 작업을 최소화하려고 노력했다.

### Deep Review Comparison

현재 Review Revolution이 적용한 리뷰 비교 서비스는, 유사 상품에 대해서 키워드와 대표 리뷰, 키워드 별 종합 평점 등을 비교해서 보여주는 Shallow Review Comparison이라고 정의할 수 있다. 하지만 유저는 유사 상품을 비교하는 시점에서, 동시에 한 키워드를 선택 시에 유사 상품들의 해당 키워드와 관련된 리뷰만 모아서 비교할 수 있는 서비스 등을 이용해서 더 높은 수준의 리뷰 비교를 지향할 것이다. 이처럼 각 상품별로 제공하는 리뷰 탐색 서비스를 유사상품 리뷰 비교시점에서 이용할 수 있게 한 기능을 Deep Review Comparison이라고 정의한다. 이 기능은 추후에 추가 개발을 통해 적용할 수 있는 기능이다.

### Advanced Similar Product Finder System

현재 Review Revolution의 핵심 시스템 중 하나인 Similar Product Finder System은 이미 정의된 카테고리와 상품 가격 등을 이용해서 사용자가 이 상품 외에 비교할 예상 후보를 찾아서 유사 상품으로 보여준다. 이 기능이 리뷰 비교의 성능을 결정할 정도로 중요하다. 유사 상품이 사용자가 정말 원하던 것이 아닐 경우, 사용자의 신뢰가 떨어지고 다른 온라인 쇼핑몰로 넘어갈 가능성이 높기 때문이다. 그렇기 때문에 더 많은 정보를 통해서 정확한 유사상품을 고르는 것이 중요하다. 이를 위해서 사용자의 검색 기록을 이용한 상품별 연관성 추적으로 시시각각 변하는 소비자의 요구에 맞는 더 정확한 유사상품 모델을 구현할 수 있다.

### Advanced Review Analysis System

현재 Review Revolution System의 핵심 시스템 중 하나인 Review Analysis System은 사용자가 리뷰를 올린 시점에서 키워드를 분석한 후 리뷰의 주요 키워드를 추출하고, 리뷰의 전체적인 긍정도, 리뷰의 키워드 별 긍정도를 추출한 후 이런 정보를 함께 Database에 저장해서 사용자가 리뷰를 더 빠르게 파악할 수 있게 도와준다. 이 기능도 리뷰 비교 시스템에 대한 사용자의 신뢰도와 관련이 높아서 사용자 만족도와 시스템의 가치에 매우 큰 영향을 주는 요소이다. 리뷰에서 올바른 키워드를 추출하고, 그 키워드 측면, 종합적 측면에서 긍정도를 정확하게 파악하는 것이 중요하다. 현재는 Google Natural Language Processing 시스템에 이 성능을 의존하고 있지만, 추후 더 만족스런 서비스를 위해서 직접 상황에 맞는 맞춤 NLP model, Analysis System을 개발하거나 미래의 고성능 NLP 모듈을 도입해서 사용자에게 더 좋은 서비스를 제공할 수 있을 것이다.

# Appendices

이 장에는 하드웨어 요구사항, 데이터베이스 요구사항, 개발환경 요구사항 등 개발 중인 시스템과 관련된 항목의 상세하고 구체적인 정보가 서술된다.

## Hardware requirements

본 시스템은 웹 기반의 어플리케이션을 전제로 하고 있으며 최신 웹 기술을 사용하기 위해서 HTML 5.0 이상이 제공되는 웹 브라우저 환경을 가정한다. 또한 휴대기기를 포함한 다양한 하드웨어 환경에서도 호환되어 사용될 수 있게 반응형 웹 디자인을 사용한다. 서버 환경에 대해서도 다양한 서버 환경에 쉽게 적용되어 사용될 수 있도록 최대한 간결하고 이해가 쉬운 코드로 작성되어야 하며, 이 시스템은 컴포넌트의 시연환경으로 웹 서버 중 firebase를 사용한 가볍고 간단한 구성으로 HW/platform dependence 측면의 overhead를 줄이고 interaction과 process 구조를 빠른 시간에 이해할 수 있도록 만들었다.

## Database requirements

본 시스템은 NoSQL인 Firebase를 시연 Database로 사용한다. 실제 외부 시스템 활용환경은 RDBMS가 될 수도 있고, NoSQL이 될 수도 있지만 유연하고 간단한 자료 구조를 통해서 효과적으로 Data의 상호작용과 처리 프로세스를 보여줄 수 있다고 생각했기 때문에 NoSQL을 사용했다. 이에 따른 데이터베이스의 자료구조와 requirement를 다음과 같이 정의한다.

### User

시스템에 가입한 유저들의 개인 정보를 담고 있는 엔티티이다. key값으로는 id, password, 기본 키워드 설정, 구매 상품 목록, 남긴 리뷰 목록을 가지고 있다. 이때 password는 보안을 위해 해싱한 값을 저장한다.

### Item

시스템에 등록된 상품의 정보를 담고 있는 엔티티이다. key값으로는 id, 상품 정보, 가격, 유사 상품 목록, 대표 키워드 목록, 대표 리뷰 목록, 전체 리뷰 목록, 전체 평점을 가지고 있다. 대표 리뷰 목록과 전체 리뷰 목록 key를 따로 생성한 이유는 대표 리뷰를 가져올 때 마다 연산을 수행하면 전체적인 cost가 올라가기 때문에 이를 방지하기 위함이다.

### Review

상품 리뷰의 정보를 담고 있는 엔티티이다. key값으로는 id, 리뷰 내용, 작성자, 작성 일시, 상품 평점, 키워드 목록, 리뷰 평점을 가지고 있다. 키워드 목록은 유저가 직접 작성하는 거이 아닌 review analysis 작업에 의해 추가된다.

### Keyword

상품이나 리뷰에서 나타난 키워드들의 정보를 담고 있는 엔티티이다. key값으로는 id, 키워드 내용, 키워드 긍정도를 가지고 있다.

# 10. Index

## 10.1. Tables

[Table 1: Glossary 17](#_Toc23624792)

[Table 2: Customized Search result 30](#_Toc23624793)

[Table 3: Item Page 31](#_Toc23624794)

[Table 4: Key word 31](#_Toc23624795)

[Table 5: Review Page 32](#_Toc23624796)

[Table 6: Review Post 33](#_Toc23624797)

[Table 7: User Page 33](#_Toc23624798)

[Table 8: Login Function 34](#_Toc23624799)

[Table 9: Review Analysis System 35](#_Toc23624800)

[Table 10: 연관상품 finder system 35](#_Toc23624801)

[Table 11: 품 정보 제공 36](#_Toc23624802)

[Table 12: 개인정보 변경 시 기능 36](#_Toc23624803)

[Table 13: 로그인 44](#_Toc23624804)

[Table 14: 사용자 정보 수정 44](#_Toc23624805)

[Table 15: 리뷰 작성, 키워드 분석 45](#_Toc23624806)

[Table 16: Customized Search-1 45](#_Toc23624807)

[Table 17: Customized Search-2 46](#_Toc23624808)

[Table 18: 연관 상품 Update 46](#_Toc23624809)

[Table 19: 리뷰 추가 요청 47](#_Toc23624810)

[Table 20: 상세정보 확인 47](#_Toc23624811)

## 10.2. Figures

[Figure 1: 온라인쇼핑 거래액 전년동월대비 증감률 10](#_Toc23624812)

[Figure 2: 온라인쇼핑 거래액 동향 11](#_Toc23624813)

[Figure 3: 1회 평균 쇼핑 시간 13](#_Toc23624814)

[Figure 4: Review Revolution - 모아 보기 기능 14](#_Toc23624815)

[Figure 5: Review Revolution – 상품 정보 화면 15](#_Toc23624816)

[Figure 6: Customized Search 예시 19](#_Toc23624817)

[Figure 7: item detail page 20](#_Toc23624818)

[Figure 8: review comparison page 21](#_Toc23624819)

[Figure 9: review analysis를 통한 키워드 추출 에시 22](#_Toc23624820)

## 10.3. Diagrams

[Diagram 1: Simple login process 18](#_Toc23624821)

[Diagram 2: Simple sign up process 19](#_Toc23624822)

[Diagram 3: Frontend Architecture 25](#_Toc23624823)

[Diagram 4: Backend Architecture 26](#_Toc23624824)

[Diagram 5: Review Analysis System Architecture 27](#_Toc23624825)

[Diagram 6: Recommendation System Architecture 28](#_Toc23624826)

[Diagram 7: Overall context diagram 41](#_Toc23624827)

[Diagram 8: Overall process diagram 42](#_Toc23624828)

[Diagram 9: Use case diagram 43](#_Toc23624829)

[Diagram 10: DFD of Review Analysis System 48](#_Toc23624830)

[Diagram 11: DFD of Similar Product Finder System 48](#_Toc23624831)

[Diagram 12: Sequence Diagram of Review Submit 49](#_Toc23624832)

# 11. References

KaemingkDiana. (2019). “20 online review stats to know in 2019”. qualtrics: https://www.qualtrics.com/blog/online-review-stats/에서 검색됨

LucaMichael. (2016). “Reviews, Reputation, and Revenue: The Case of Yelp.com”. Harvard Business School: https://www.hbs.edu/faculty/Pages/item.aspx?num=41233에서 검색됨

김준현최지혜, 허준범, 모하메드유도영,. (2019). “requirement.” github: (https://github.com/skkuse/2019spring\_41class\_team5/blob/master/docs/requirement.docx)에서 검색됨

미디어DMC. (2017). “2018 인터넷 쇼핑 행태와 쇼퍼 그룹 및 쇼핑몰 분석 보고서\_요약본.” digieco: https://www.digieco.co.kr/KTData/Board/FILE/PDF/2018%20%EC%9D%B8%ED%84%B0%EB%84%B7%20%EC%87%BC%ED%95%91%20%ED%96%89%ED%83%9C%EC%99%80%20%EC%87%BC%ED%8D%BC%20%EA%B7%B8%EB%A3%B9%20%EB%B0%8F%20%EC%87%BC%ED%95%91%EB%AA%B0%20%EB%B6%84%EC%84%9D%20%EB%B3%B4%EA%B에서 검색됨

통계청. (2019). “2019년 8월 온라인쇼핑 동향.” kostat: http://kostat.go.kr/portal/korea/kor\_nw/3/index.board?bmode=download&bSeq=&aSeq=377803&ord=2에서 검색됨