<Comparewise >

-Design Specification-



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

Contents

[1. Preface 3](#_Toc41100809)

[1.1 Objective 3](#_Toc41100810)

[1.2 Readership 4](#_Toc41100811)

[1.3 Document Structure 4](#_Toc41100812)

[A. Introduction 4](#_Toc41100813)

[B. System Architecture 4](#_Toc41100814)

[C. Protocol Design 4](#_Toc41100815)

[D. Database Design 4](#_Toc41100816)

[E. Testing Plan 4](#_Toc41100817)

[F. Development Plan 5](#_Toc41100818)

[G. Index 5](#_Toc41100819)

[2. Introduction 5](#_Toc41100820)

[2.1 Objective 5](#_Toc41100821)

[2.2 Applied Diagram 5](#_Toc41100822)

[A. UML Diagram 5](#_Toc41100823)

[B. Package Diagram 5](#_Toc41100824)

[C. Class Diagram 6](#_Toc41100825)

[D. Sequence Diagram 6](#_Toc41100826)

[E. ER Diagram 7](#_Toc41100827)

[2.3 Applied Tool 8](#_Toc41100828)

[A. Power point 8](#_Toc41100829)

[B. Draw.io 8](#_Toc41100830)

[2.4 Project Scope 8](#_Toc41100831)

[3. System Overall Architecture 9](#_Toc41100832)

[3.1 Objective 9](#_Toc41100833)

[3.2 System Organization 9](#_Toc41100834)

[3.3 Sign up / Login System 11](#_Toc41100835)

[3.4 Compare System 12](#_Toc41100836)

[3.5 My page System 13](#_Toc41100837)

[3.6 Search System 14](#_Toc41100838)

[8. Testing Plan 14](#_Toc41100839)

[8.1 Objective 15](#_Toc41100840)

[8.2 Testing Policy 15](#_Toc41100841)

[A. Development Testing 15](#_Toc41100842)

[B. Release Testing 15](#_Toc41100843)

[C. User Testing 16](#_Toc41100844)

[D. Testing Case 16](#_Toc41100845)

[9. Development plan 16](#_Toc41100846)

[9.1 Objective 16](#_Toc41100847)

[9.2 Frontend Environment 16](#_Toc41100848)

[A. React Native 16](#_Toc41100849)

[B. PWA 17](#_Toc41100850)

[C. Vue.js 17](#_Toc41100851)

[A. Firebase 18](#_Toc41100852)

[B. Google Cloud Platform 19](#_Toc41100853)

[C. Node.js 19](#_Toc41100854)

[10. Index 20](#_Toc41100855)

[10.1 Tables 20](#_Toc41100856)

[10.2 Figures 20](#_Toc41100857)

[10.3 Diagrams 20](#_Toc41100858)

# 1. Preface

## 1.1 Objective

Preface에는 본 문서의 예상 독자층을 선정하고, 각 Chapter에 대해 간략한 설명을 제공한다.

## 1.2 Readership

본 문서는 Stakeholders를 비롯한 엔지니어, 등 다양한 독자 층을 선정하고 있다.

## 1.3 Document Structure

### A. Introduction

Introduction chapter에서는 본 문서에서 사용된 다양한 다이어그램 및 다이어그램 생성을 위해 사용한 프로그램, 도구들을 소개하고 개발되어질 System이 지향하는 목표 및 전반적인 개발 범위를 기술한다.

### B. System Architecture

System Architecture Chapter에서는 개발되어질 System의 Overall Design과 더불어 각 세부적인 Function, Subsystem의 전반적인 System Architecture을 다이어그램 및 도식과 함께 설명한다.

### C. Protocol Design

Protocol Design Chapter 에서는 Frontend 와 Backend 간의 상호작용 및 통신을 규정하는 Protocol에 대해 기술한다. 또한 해당 상호작용에 사용된 인터페이스와 사용된 기술 및 서식 규정 등을 제시한다.

### D. Database Design

Database Design Chapter에서는 Requirements Specification문서에 작성된 초기버전의 Database을 기반으로 System에 사용되어질 Database들을 ER Diagram으로 표현하고, Relational Schema및 No SQL기반의 Key-value Database Overall을 기술한다.

### E. Testing Plan

Testing Plan Chapter에서는 본 개발 작업 이후 Testing을 진행하기 앞서 Testing에 대해 수립된 전반적인 방향성 및 계획과, Testing Policy를 제시한다.

### F. Development Plan

Development Plan Chapter에서는 본 개발과정의 전체적인 계획과 앞으로의 개발 계획을 제시한다.

### G. Index

Index는 본 문서에 사용된 도식 및 다이어그램과 그림들의 색인을 제공한다.

# 2. Introduction

## 2.1 Objective

본 Introduction chapter에서는 개발될 System의 Designing process에 사용된 다양한 다이어그램 및 도구 들을 소개하고, System의 전반적인 개발 범위를 기술한다.

## 2.2 Applied Diagram

### A. UML Diagram

UML Diagram은 Unified Modeling Language로, 통합 모델링 언어를 의미한다. 이는 객체지향 Software네 특화되어 있으며, System을 문서화, 및 Architecture를 구축하는데 사용되며, System의 전제적인 Workflow를 나타내는데 효율적이다.

UML Diagram은 User부터 Developer까지의 넓은 범위의 Communication을 위해 사용되며, 상황에 따라 다르게 사용되는 여러 종류의 UML Diagram이 존재한다. 이 문서에서 사용되는 Component Diagram, Sequence Diagram, ER Diagram을 사용 및 소개한다.

### B. Package Diagram

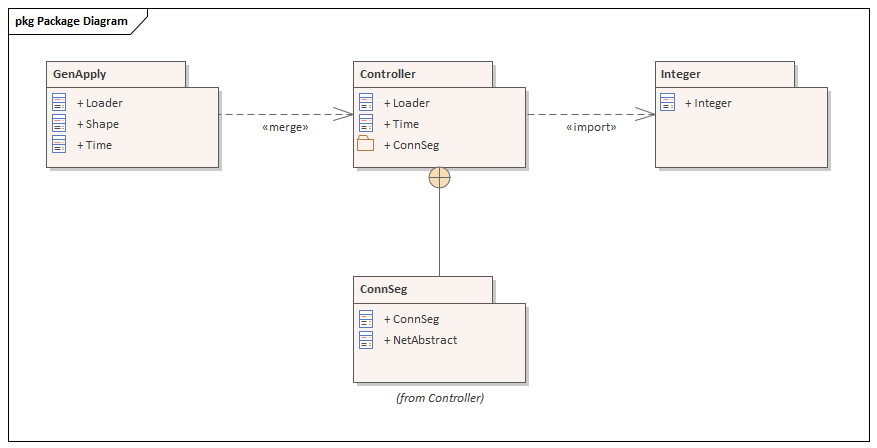


그림 . Package Diagram 예시

Package Diagram은 System의 각 요소들을 package단위로 묶어 표현하는 형식의 Diagram이다. 통합 모델링 언어 UML의 일종으로 System을 여러 개의 Subsystem 혹은 Package로 나누고, 각 Package 간의 상호관계를 표현하는데 유리하다.

### C. Class Diagram

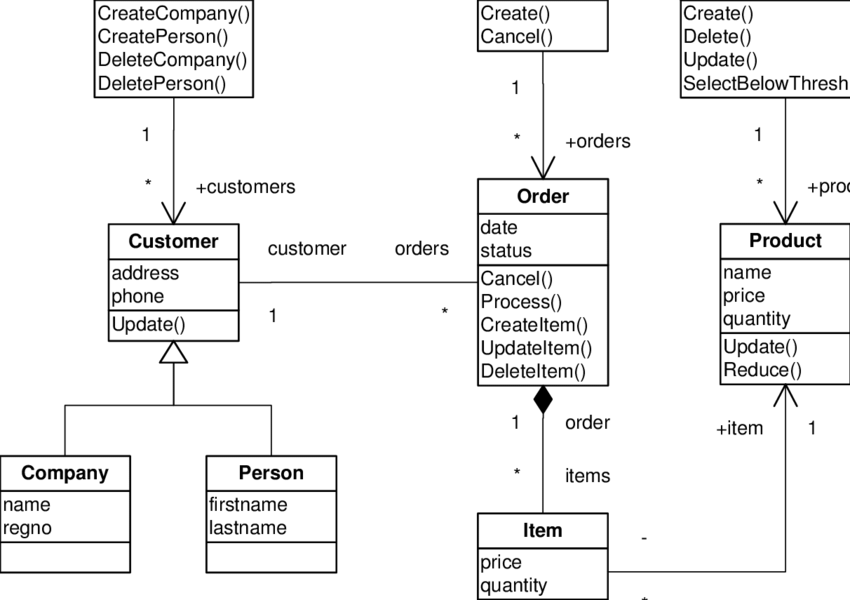


그림 . Class Diagram 예시

Class Diagram은 객체지향 System에서 사용되어진 각 객체 Class들을 나타내고, 각 Class 객체가 가지고 있는 Method및 Information들을 표시하며, 서로 다른 객체들 간의 상호관계 및 객체간 상호작용에 사용되는 Interface를 나타낸다.

### D. Sequence Diagram

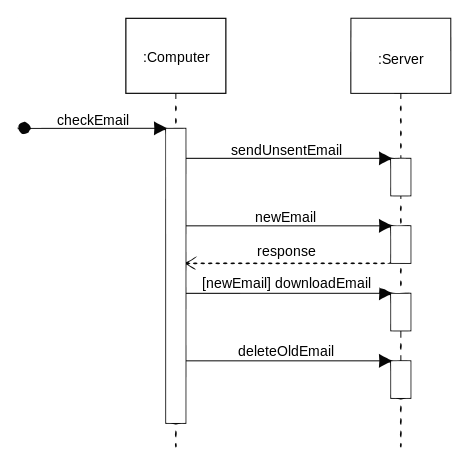


그림 . Sequence Diagram 예시

Sequence Diagram은 Event에 따라 변화되는 State들을 한눈에 이해할 수 있도록 그려진 다이어그램이다. 특정 Function혹은 특정 동작에 대한 전체 Sequence를 표시함으로써, 기능 혹은 동작이 이루어지는 과정을 표현할 수 있다.

### E. ER Diagram

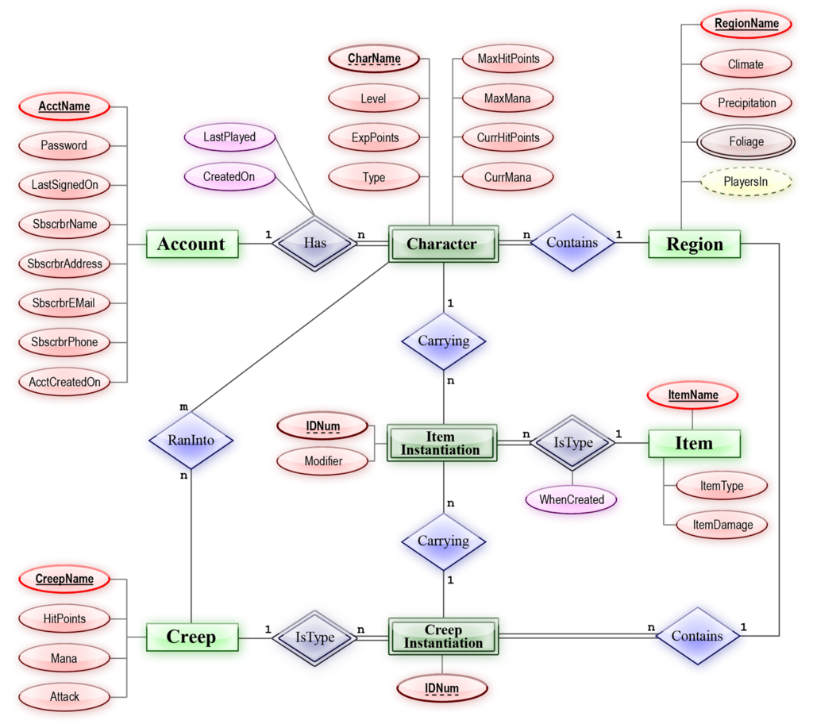


그림 . ER Diagram 예시

ER Diagram은 Entity-Relationship Model으로 Entity와 Entity의 속성의 관계를 나타낸 다이어그램이다. 일반적으로 Database를 설계, 기술할 때 사용되며 Relational Schema를 작성하는 베이스로 사용된다.

## 2.3 Applied Tool

### A. Power point



그림 . Power Point Logo

Power Point는 MS사에서 제공하는 MS Office에 속해 있는 Tool으로, 주로 Ppt형식의 Presentation자료를 생성하는데 사용된다. Power Point는 매우 강력한 도형 생성 기능을 가지고 있어 Diagram을 그리는데 유용하게 사용될 수 있다.

### B. Draw.io



그림 . Draw.io Logo

Draw.io는 다이어그램 생성에 특화되어 있는 Tool으로 여러가지 종류의 다이어그램을 생성, 편집하는데 굉장히 편리하다.

## 2.4 Project Scope

Comparewise는 기존의 커머스 어플리케이션들에 지나치게 많은 상품과 각 상품에 대한 지나치게 많은 데이터들이 등록되어 있어, 상품들을 하나하나 비교하기가 까다로운 문제를 해결하기 위해 소비자에게 강력하고 편리한 비교 UI를 제공하는 것이 궁극적인 목표이다. 개발 기간이 짧으며, 사이즈가 큰 System이 아니라는 점을 감안하여, 구글에서 제공하는 강력한 Backend Tool인 Firebase를 Backend로 사용하며 이에 잘 맞는 Google Cloud Platform 및 Node.js로 Backend를 구성한다, 웹 기반 Application임을 감안하여 PWA와 Vue.js 에 더하여 Facebook의 오픈소스 프로젝트인 React Native를 사용하여 Frontend를 구성한다.

Frontend System은 사용자에게 구매에 필요한 정보들을 제공 및 사용자와의 Communication을 위하여 동작한다. Backend system은 Frontend가 사용자에게 제공하기 위해 필요한 Data요청을 처리하며, 실시간으로 바뀌는 판매 사이트들을 Crawling 하여 제공하는 역할을 담당한다.

# 3. System Overall Architecture

## 3.1 Objective

System overall Architecture에서는 전반적인 System의 구조를 소개하며, System의 Sub System들과 각 Sub System간의 관계를 기술한다.

## 3.2 System Organization

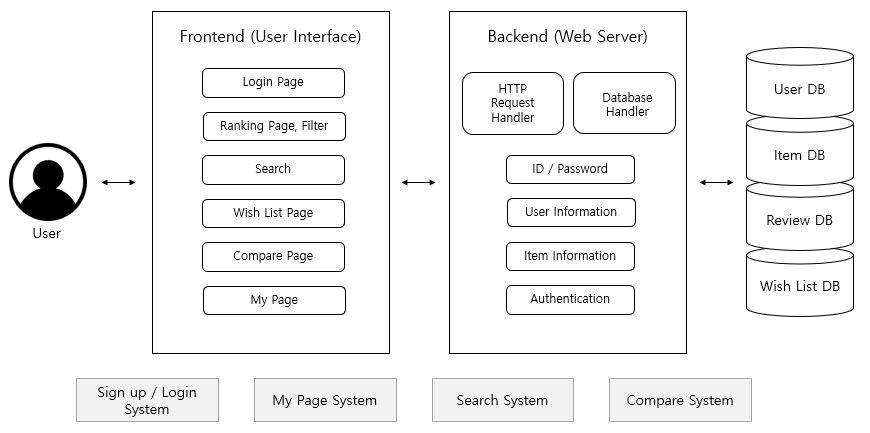


Diagram . Overall System Architecture

Comparewise의 Overall System구조는 위와 같다. Web Server기반의 Application이며, 요청과 요청에 따른 처리가 이어지는 System이기 때문에 Client-server Model을 적용하였다. Frontend와 Backend는 JSON을 사용하는 Http통신으로 Data를 교환하며, Backend는 Request Handler에 의해 요청에 따른 처리를 분배하며, Database Handler를 통해 필요한 정보를 Database에서 수집해 Frontend에게 응답한다.

## 3.3 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.

## 3.4 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.

## 3.5 My page System

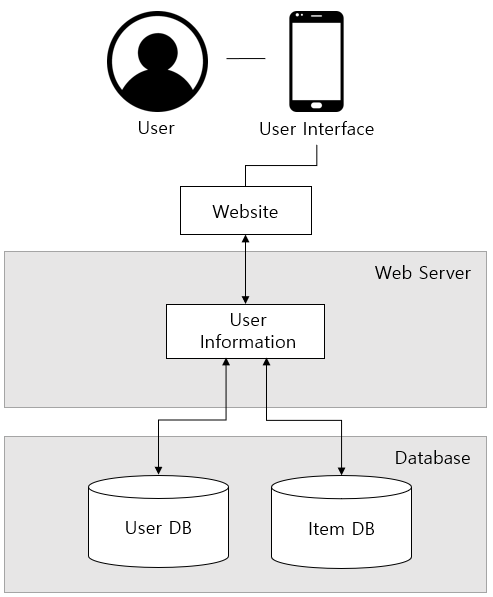


Diagram 4. My Page System Architecture

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.

## 3.6 Search System

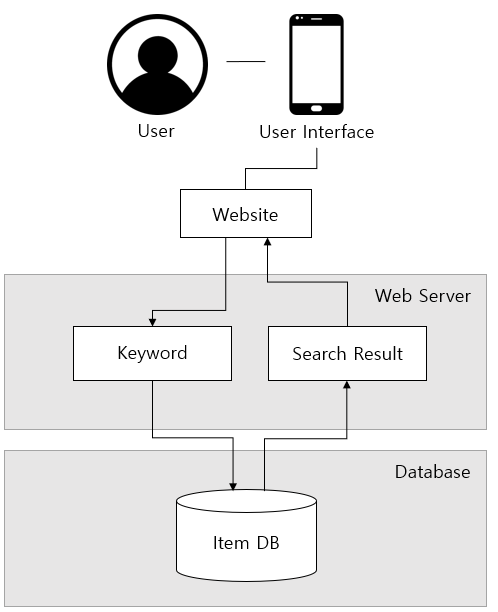


Diagram 5. Search System Architecture

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

# 8. Testing Plan

## 8.1 Objective

본 장에서는 개발되어질 System의 Testing 계획을 서술하고, 각 항목에 대한 Testing의 방향성을 제시한다.

## 8.2 Testing Policy

### A. Development Testing

Development Testing은 개발되어질 system의 전체적인 Synchronization 및 오류 발견과 예방에 초점을 맞추어 개발 과정에서 일어날 수 있는 오류들을 방지하는 것을 목표로 한다. 이를 위해 코드 검토, Data flow 및 사용되어지는 Metrics에 대한 검토와, Peer Code Review등을 계획한다. 또한 Reliability, Security, Performance과 같은 Non functional requirements에 초점을 맞추어 Testing을 진행한다.

#### Reliability

Reliability는 전체적인 System의 신뢰성으로 Comparewise의 경우 판매처 및 어플리케이션에서 제공되어지는 Product에 대한 정보가 늘 최신 정보로 Up to date 되어 있어야 한다. 어플리케이션에 적용되는 정보들이 신뢰 가능한 사이트에서 기인되었는지에 대한 검토와, 실시간으로 올바르게 Crawling 되어지고 있는지에 대한 검토가 요구된다.

#### Security

Comparewise는 실질적인 구매가 진행되는 어플리케이션이 아니기에 실질적인 구매를 지원하는 어플리케이션만큼의 높은 보안수준은 요구되지 않으나, 고객들이 가입시 작성한 개인정보의 보안에 신경을 써야 한다. DB에 대한 접근 권한과, 고객의 ID/PW에 대한 암호화에 대한 검토가 요구된다.

### B. Release Testing

System의 배포는 점진적인 방향으로 진행되어야 한다. 하지만 개발 및 Testing 기간이 짧은 것을 감안하여 초기 배포 버전에 대한 Testing만을 진행하는 것을 목표로 한다.

### C. User Testing

실질적인 System사용에 있어 사용자의 입장에서 일어날 수 있는 오류들을 검토하여야 한다. 개발 이후 사용자의 입장에서 어플리케이션 사용시 일어날 수 있는 시나리오들을 가정하고, 각 시나리오들의 진행과정에서 오류가 없는지 검토하는 방식으로 이루어진다.

### D. Testing Case

개발 기간이 촉박한 관계로 많은 Testing Case를 적용하는 것엔 무리가 있다. 본 Testing Plan은 실제보다 많이 축소된 크기인 50명의 유저와 카테고리별 3개의 제품을 가정하여 Testing을 진행한다.

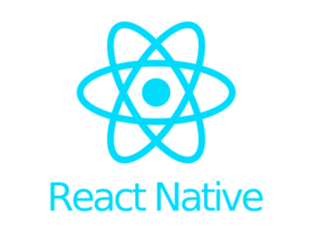
# 9. Development plan

## 9.1 Objective

본 장에서는 실 계발 단계에서 사용되어질 개발 환경 및 Tool들을 소개하고, 개발 일정과 진행상황을 기록한다.

## 9.2 Frontend Environment

### A. React Native



React는 굉장히 광범위한 웹 및 웹 기반 어플리케이션에서 사용되는 강력한 Frontend 라이브러리이다. React Native는 Facebook에서 제공하는 Opensource Mobile Application Framework로서, 기존 Platform기능에 더하여 React를 사용할 수 있도록 만들어진 Framework이다. Android및 IOS 어플리케이션 개발에 사용된다.

### B. PWA



PWA ( Progressive Web Applications ) 는 Web을 기반으로 하는 어플리케이션들에 사용되는 어플리케이션들 이다. Native Application들과 다르게 URL을 통한 쉬운 접근이 가능하므로 사용자가 사용하기에 훨씬 편리하다는 장점이 있다.

### C. Vue.js



Vue.js는 Web기반 어플리케이션들의 UI를 만들기 위해 사용되는 Java Script기반의 오픈소스 Framework이다. 개발자들이 익히기가 쉽다는 장점을 가지고 있으며 PWA및 Node.js와의 연동성이 좋다.

9.3 Backend Environment

### A. Firebase



Firebase는 구글에서 제공하는 Native및 PWA 개발 플랫폼이다. Firebase는 Application의 Backend를 구성하는 Authentication및 NoSQL기반 Key-value Database등을 제공하고 있어, JSON형식의 Data교환이 편리하다. Server측 Infrastructure를 제공하여, 소규모 개발작업에서 Server구성에 드는 부하를 줄여주며, Realtime Database를 제공한다.

### B. Google Cloud Platform



Google에서 제공하는 Cloud Computing Platform이다. Data상호작용 및 Categorization 상세화 및 Entity Analyze에 사용될 계획이다.

### C. Node.js



Java Script 기반의 Server Framework이다. 네트워크 기반 어플리케이션 개발에 특화되어 있는 Software Framework으로 Event-base Non-blocking I/O 와 Single thread Event loop을 통한 높은 처리성능을 가지고 있다.

9.4 Schedule

|  |  |
| --- | --- |
| Proposal | 5/3 |
| Requirement Specification | 5/13 |
| Design Specification | 5/24 |
| Frontend Development | 5/28 |
| Backend Development | 6/3 |
| Testing and Review | 6/7 |

Table . Testing Schedule

# 10. Index

## 10.1 Table

[Table 1. Testing Schedule 21](#_Toc41101181)

## 10.1 Figure

[그림 1. Package Diagram 예시 6](#_Toc41101190)

[그림 2. Class Diagram 예시 7](#_Toc41101191)

[그림 3. Sequence Diagram 예시 7](#_Toc41101192)

[그림 4. ER Diagram 예시 8](#_Toc41101193)

[그림 5. Power Point Logo 9](#_Toc41101194)

[그림 6. Draw.io Logo 9](#_Toc41101195)

## 10.1 Diagrams

[Diagram 1. Overall System Architecture 11](file:///C:\Users\syngj\Desktop\Comparewise_Design%20Specification.docx#_Toc41101197)

[Diagram 2. Sign up / Login System Architecture 12](file:///C:\Users\syngj\Desktop\Comparewise_Design%20Specification.docx#_Toc41101198)

[Diagram 3. Compare System Architecture 13](file:///C:\Users\syngj\Desktop\Comparewise_Design%20Specification.docx#_Toc41101199)

[Diagram 4. My Page System Architecture 14](file:///C:\Users\syngj\Desktop\Comparewise_Design%20Specification.docx#_Toc41101200)

[Diagram 5. Search System Architecture 15](file:///C:\Users\syngj\Desktop\Comparewise_Design%20Specification.docx#_Toc41101201)