

# 대학생 중고거래 챗봇

Requirement

Specification

|            |     |
|------------|-----|
| 2016311867 | 김우연 |
| 2017311656 | 윤응구 |
| 2015311849 | 이동환 |
| 2016312107 | 문경진 |
| 2014313932 | 전국민 |

# Contents

|                                      |    |
|--------------------------------------|----|
| 1. Preface -----                     | 4  |
| 1.1 Objective -----                  | 4  |
| 1.2 Readership -----                 | 4  |
| A. User Requirement Readership       |    |
| B. System Requirement Readership     |    |
| 1.3 Document Structure -----         | 5  |
| A. Preface                           |    |
| B. Introduction                      |    |
| C. Glossary                          |    |
| D. User Requirements Specification   |    |
| E. System Architecture               |    |
| F. System requirements Specification |    |
| G. System models & System Evolution  |    |
| H. Appendices                        |    |
| I. Index                             |    |
| 1.4 Version of the Document -----    | 7  |
| 2. Introduction -----                | 8  |
| 2.1 Objective -----                  | 8  |
| 2.2 Needs -----                      | 8  |
| 2.3 플랫폼으로써 챗봇 -----                  | 9  |
| 2.4 중고거래 서비스 -----                   | 10 |
| 2.5 Expected Effect -----            | 11 |

|  |    |
|--|----|
| 3. Glossary -----                          | 12 |
| 3.1 Objective -----                        | 12 |
| 3.2 Term definitions -----                 | 12 |
| 4. User Requirement Definition -----       | 13 |
| 4.1 Objective -----                        | 13 |
| 4.2 Functional Requirements -----          | 13 |
| 4.3 Non-functional Requirements -----      | 15 |
| 5. System Architecture -----               | 18 |
| 5.1 Objective -----                        | 18 |
| 5.2 Overview -----                         | 18 |
| 5.3 User Subscription System -----         | 19 |
| 5.4 Seller System -----                    | 19 |
| 5.5 Buyer System -----                     | 20 |
| 6. System Requirements Specification ----- | 21 |
| 6.1 Objective -----                        | 21 |
| 6.2. Functional Requirements -----         | 21 |
| 6.3. Non-Functional Requirements -----     | 26 |
| 7. System Models -----                     | 28 |

|                               |    |
|-------------------------------|----|
| 7.1 Context Models            | 28 |
| 7.2 Interaction Models        | 29 |
| 7.3 Behavioral Model          | 32 |
| 8. System Evolution           | 33 |
| 8.1 Objective                 | 33 |
| 8.2 Limitation and Assumption | 33 |
| 8.3 Evolution                 | 33 |
| 9. Appendices                 | 35 |
| 10. Index                     | 36 |
| 10.1 Tables                   | 36 |
| 10.2 Figures and Diagrams     | 36 |

# 1. Preface

## 1.1 Objective

Preface는 본 요구사항 명세서의 전체적인 개요에 대해 설명한다. 개요로는 예상 독자와 문서의 전반적인 구조, 버전 별 수정사항과 관리내용이 있다. 이를 통해 문서의 가독성을 높이고, 효율적인 관리를 가능하게 한다.

## 1.2 Readers

본 명세서의 독자는 두 가지 유형이 있다. 그에 따라 User Requirement와 System Requirement로 구성되어 있다.

### A. User Requirement Readers

주요 고객을 정의하고, 사용자의 요구사항에 따라 제공하는 서비스와 제약 사항에 관해 기록한다. Functional System Requirement 뿐만 아니라 Non Functional Requirement까지 설명한다. 예상 독자는 중고거래를 하고자 하는 대학생(판매/구매), 즉 서비스를 이용하고자 하는 고객이다.

### B. System Requirement Readers

고객의 요구사항을 토대로 서비스 개발을 위한 구현 사항과 기능을 전문적으로 기술한다. 주요 독자는 개발자 이외에도 Client engineer, Contractor, End User 까지도 예상 독자가 될 수 있다.

## 1.3 Document Structure

본 문서는 10개의 장으로 구성되어 있다. Preface, Introduction, Glossary, User Requirements Specification, System Architecture, System requirements Specification, System models, System Evolution, Appendices, Index이다. 각 장의 내용은 다음과 같다

### A. Preface

본 요구사항 명세서의 전체적인 개요에 대해 설명한다. 개요로는 예상 독자와 문서의 전반적인 구조, 버전 별 수정사항과 관리내용이 있다. 이를 통해 문서의 가독성을 높이고, 효율적인 관리를 가능하게 한다.

### B. Introduction

본 서비스의 필요성과 시장성, 그리고 시스템이 반영하고자 하는 고객의 needs를 설명한다. 시스템의 주요 기능과 기대효과를 기술한다.

### C. Glossary

본 요구사항 명세서에 등장하는 용어들에 대해 정의한다. 이를 통해 비개발자도 이해할 수 있는 문서를 만들며, 해석에 혼동의 여지가 있는 모든 용어들에 대해 조작적 정의를 한다.

### D. User Requirements Specification

실제 사용자에게 제공되는 서비스에 대해 개괄적으로 설명한다. UML이나 다이어그램, 여러 시각적 요소를 활용하여 독자의 이해를 돕는다. Functional/Non Functional 요구사항을 모두 기술하며, 최종 시스템은 여기에 기술된 모든 조건을 충족해야 한다.

## **E. System Architecture**

Requirement Specification 을 바탕으로 시스템 구조를 작성한다. 시스템의 전체적인 기능과 기능 별로 서브시스템이 어떻게 구성되어 있는지 서술한다.

## **F. System Requirements Specification**

그동안 작성된 Requirement를 기반으로 Functional/Non-functional requirement와 기타 요구사항을 상세하게 서술한다. 개발 과정에서 활용성이 높은 부분이므로 도표 등을 이용하여 체계적으로 기술한다.

## **G. System Models**

시스템 외부/내부, 컴포넌트 간의 관계를 다이어그램으로 표현한다.

## **H. System Evolution**

시스템의 한계에 대해 서술하고, 어떤 변경사항이 있을지 예측하며 그에 대한 대응 방안까지 설명한다.

## **I. Appendices**

문서 작성 과정에서 생략된 참고자료 등을 기술하고 구체적인 정보를 제공한다. 하드웨어, 데이터베이스 요구사항, 개발환경 요구사항 등이 포함되어 있다. 또한 시스템 사용에 필요한 최소의 설정과 환경, 데이터의 관계와 구조에 대해 설명한다.

## **J. Index**

본문에서 사용된 그림, 표, 다이어그램 등의 색인을 기술한다.

## 1.4 Version of the Document

| Version | Modified Date | Explanation  |
|---------|---------------|--|
| 0.1     | 2019.10.24    | - 문서 목차 작성   |
| 0.3     | 2019.10.28    | - Preface/Introduction/Glossary 작성   |
| 0.5     | 2019.10.30    | - User Requirements Specification<br>/System Architecture 작성<br>- System Requirements Specification 작성 |
| 1.0     | 2019.10.31    | - System Models/Evolution 작성<br>- Appendices 작성<br>- System Requirements Specification 수정              |
| 2.0     | 2019.11.2     | - Index 작성<br>- 문서 양식 통일 및 최종 검토   |



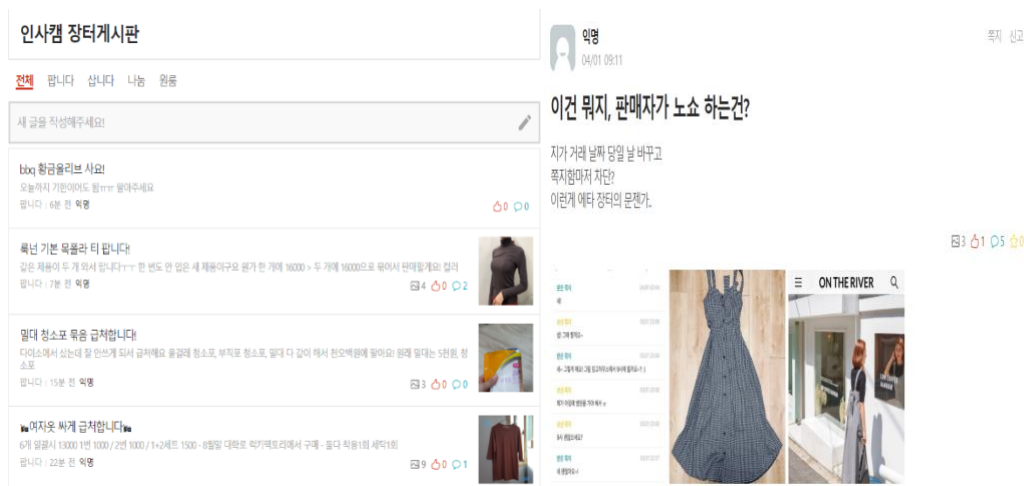
## 2. Introduction

### 2.1 Objective

이 장에서는 서비스 착안 배경과 예상 고객층의 Needs를 서술한다. 그를 바탕으로 최종 시스템이 어떠한 방식으로 Needs를 충족할 수 있는지, 또 서비스를 통한 기대효과를 설명한다.

### 2.2. Needs

대학별 커뮤니티 어플리케이션인 에브리타임의 성균관대 거래 장터 게시판에서 착안했다. 이 장터 게시판은 원룸, 책, 가구, 식료품 등 대학생들의 수요에 따라 구색이 다양하며, 거래 빈도가 높다. 그러나 에브리타임의 특성인 익명성 때문에 거래가 일방적으로 파기되고, 직거래 약속 시간에 나오지 않는 등 판매/구매자간 원활한 소통이 되지 않는 경우가 많다.



<그림 1. 성균관대학교 에브리타임 중고거래 게시물>

이는 직거래가 주가 되는 대학교 내 거래에 치명적인 단점이 될 수 있다. 또한, 에브리타임은 거래가 주목적이 아닌 커뮤니케이션 앱으로 그 기능이 부족하고, 접근성이 떨어진 다. 이러한 문제를 극복하기 위해 전문적으로 대학별 중고거래를 할 수 있는 챗봇을 고안했 고, 학생들이 빠르고 쉽게 거래할 수 있는 플랫폼을 만들어보고자 한다.

## 2.3. 플랫폼으로써 챗봇

현재 중고거래에 가장 많이 사용되는 플랫폼은 웹사이트 중고나라, 번개장터이다. 그 외 스마트폰과 sns의 활성화에 따라 어플리케이션 당근마켓과 페이스북/인스타그램 게시 물을 통한 거래가 있고 특히 대학생을 대상으로 한 대학 커뮤니티 앱 에브리타임의 장터게 시판 등이 있다.

이처럼 중고거래 플랫폼은 중고나라, 번개장터도 전용 앱을 개발하는 등 웹에서 어플리케이션으로 확장되는 추세이다. 게다가 sns 앱에서도 중고거래의 수요공급이 증가하고 있 는 것으로 보아 접근성이 중고거래에 있어 중요한 가치임을 알 수 있다. 판매자는 물건을 쉽 고 빠르게 등록하고, 구매자는 최소한의 클릭을 통해 원하는 물건을 찾을 수 있어야 하며, 판 매자와 구매자가 실시간으로 연락하여 빠른 시간내에 거래할 수 있어야 한다.

나날이 중요해지고 있는 중고거래에서 접근성의 가치를 '챗봇'이라는 플랫폼을 통해 실현한다. 대부분의 사람들이 쓰고 있는 카카오톡의 챗봇 기능을 활용하여 익숙하고 빠른 거래를 유도해 새로운 플랫폼의 가능성을 제시한다.

## 2.4. 중고거래(서비스 이름)

중고거래(서비스 이름)가 제공하는 서비스는 크게 네 가지이다.

### A. 물건 판매

- 판매자는 상품 등록, 상품 정보 수정, 상품 정보 삭제 기능을 통해 판매물품을 관리할 수 있다. 도서는 ISBN인식을 통해 간편하게 등록할 수 있으며, 원룸과 같은 경우 위치를 등록할 수 있는 기능을 제공한다.

### B. 물건 구매

- 구매자는 이미지 검색이나 직접 키워드 입력을 통해 물건을 검색할 수 있으며, 판매자의 신뢰도와 거래기록을 열람할 수 있다. 구매를 원할 경우, 오픈채팅 링크를 통해 문의가 가능하다.

### C. 판매자와 구매자 연결

- 카카오톡의 오픈채팅 기능을 이용하여 판매자와 구매자를 직접 연결해준다. 연결 후 거래를 할 경우 에브리타임의 시간표를 받아와 공강 시간을 비교하여 거래시간을 추천해준다.

### D. 대학별 중고장터 분류

- 본인 소속 대학교를 선택하고, 학교별 장터 커뮤니티를 생성한다. 대학교 인증을 위해 에브리타임 API를 추가하여 신뢰도를 높인다. 또한 에브리타임 시간표를 등록하여 추후 거래를 할 때 거래시간 추천에 이용한다.

## 2.5. Expected Effect

### A. 중고물품을 구입하고자 하는 대학생

학교 교재 처럼 물품의 특수성이 있어 일반 중고 거래 사이트에서 거래가 쉽지 않은 물품이나, 직거래를 하고 싶지만 시간 여유가 없는 대학생이 거래를 할 수 있다.

### B. 중고물품을 판매하고자 하는 대학생

기존 중고거래 사이트(중고나라, 번개장터)에 올리기는 부담스럽고, 시간적 여유가 없는 대학생이 쉽고 빠른 직거래를 가능하게 한다. 같은 학교 학생이기 때문에 어느정도 신뢰감이 생기며, 직거래 시간과 장소 설정이 유연하다는 장점이 있다. 또한 챗봇을 통해 오픈 프로필로 구매자와 판매자가 연결이 되기 때문에 실시간 연락도 가능하다.

### C. 거래의 확장

최근 대학생들 사이에서 넷플릭스 4인 정액제 공유, 식료품 공동구매 등 여러 형태의 공유 구매가 이루어 지고 있다. 이 서비스를 통해 중고 거래 뿐만 아니라 사용자들이 자발적으로 또다른 구매 형태를 창조하기를 기대한다.

## 3. Glossary

### 3.1. Objective

본 요구사항 명세서에 등장하는 용어들에 대해 정의한다. 이를 통해 비개발자도 이해할 수 있는 문서를 만들며, 해석에 혼동의 여지가 있는 모든 용어들에 대해 조작적 정의를 한다.

### 3.2. Term Definitions

| Terms          | Definition  |
|----------------|---|
| 구매자(Buyer)     | 중고물품을 구매하려는 사용자                                   |
| 판매자(Seller)    | 중고물품을 판매하려는 사용자                                   |
| 도서ISBN         | 국내 출판물의 국제표준도서번호, 도서를 식별하는 용도로 쓰인다.               |
| 자연어            | 사람들이 일상적으로 쓰는 언어                                  |
| QR코드           | 많은 정보를 담고 있는 격자무늬의 2차원 코드                         |
| 에브리타임          | 대학별 커뮤니티 어플리케이션                                   |
| Aws(아마존 웹 서비스) | Amazon 에서 제공하는 웹 서비스, 본 프로젝트에서는 서버와 데이터베이스를 사용한다. |
| 데이터베이스         | 데이터를 저장하고 불러오기 위한 일종의 저장소                         |
| API            | 시스템 자원 또는 다른 서비스에 접속하기 위해 제공되는 인터페이스              |
| 챗봇             | 문자 또는 음성으로 대화하는 기능이 있는 컴퓨터 프로그램 또는 인공지능           |

<표1. Term Definitions>

## 4. User Requirements Definition

### 4.1. Objective

해당 장에서는 본 시스템이 사용자에게 필수적으로 제공해야 하는 서비스에 대하여 서술한다. 또한 제품이나 시스템 프로세스가 따라야 할 제약 사항에 대해서도 서술한다. 해당 장의 내용은 자연어와 다이어그램을 활용하여 독자가 이해할 수 있도록 Functional Requirements 와 Non-Functional Requirements 에 대해서 상세히 기술한다.

### 4.2. Functional Requirements

#### A. Sign up

이 기능은 사용자가 특정 ID 및 비밀번호를 사용하여 플랫폼에 접근 할 수 있도록 만들어 주는 가장 기본적인 기능이다. 사용자는 중고거래 챗봇을 사용하기 위해서 Sign up 을 진행해야 한다. 개인간의 거래를 추구하는 서비스 이기 때문에 판매자, 구매자를 구분 하지 않는다. 중고거래 챗봇은 카카오톡 플랫폼 기반이므로, 회원가입 과정에서는 가입 약관 동의 후 학교, 시간표를 기본정보로 입력하게 되며, 에브리타임 플랫폼을 통해 학교 인증을 진행하게 된다. 인증이 완료 될 경우, Sign up 은 완료되고 정보들은 user DB 에 저장된다.

#### B. Product registration

이 기능은 중고거래 챗봇이 제공하는 기본적인 기능으로, 개개인의 사용자가 자신이 판매하고자 하는 상품에 대해 판매를 하기 위한 등록을 하는 기능이다. 판매할 상품을 등록 할 경우 우선 상품에 대한 오픈 프로필을 생성한 후, 일반, 도서, 원룸에 해당하는 카테고리를 선택한 후 상품의 제목, 가격, 상세정보를 입력한다. 등록된 상품은 상품 DB 에 저장되고, 해당 대학교 내부 사용자는 판매자의 오픈 프로필에 접근하여 거래가 가능하게 한다.

### **C. ISBN recognition**

이 기능은 이미지에서 도서 ISBN 을 인식하는 기능이다. 판매자가 상품 등록 시, 도서 카테고리를 선택하게 되면 자동으로 ISBN 인식 기능이 활성화 되어 구매자가 ISBN 검색을 통해 도서 검색을 할 수 있는 서비스를 제공한다.

### **D. House location registration**

이 기능은 판매하고자 하는 주거공간의 위치를 등록하는 기능이다. 판매자가 상품 등록 시, 원룸 카테고리를 선택하게 되면 자동으로 위치 정보 입력 기능이 활성화 되어 구매자의 위치정보를 통한 구매 의사 결정을 도와줄 수 있는 기능이다.

### **E. Search**

이 기능을 사용자가 원하는 상품을 검색 하는 기능이다. 검색 조건은 이름, 카테고리 등 다양한 조건이 될 수 있으며, 검색하여 나온 결과를 점수, 가격 등 다양한 필터로 재정렬이 가능하도록 적용하면 사용자가 보다 원하는 결과를 편리하게 획득할 수 있을 것이다.

### **F. Image Search**

사용자는 이 기능을 통하여 이미지를 통해 상품을 검색할 수 있다. 기존의 Search 에서 다양한 필터를 통한 검색과 달리 검색의 입력에 이미지를 주고, 그 이미지를 DB 에 저장된 상품들과 비교 분석하여 가장 유사도가 높은 상품들을 목록에 보여주는 기능이다.

### **G. Trading time recommendation**

이 기능은 판매자와 구매자 사이 기본정보로 등록 되어 있는 시간표를 기반으로 거래 가능한 시간을 추천해주는 기능이다. 본 시스템은 택배나 다른 운송수단이 아닌 직거래를 목표로 설계한 시스템으로 판매자와 구매자의 커뮤니케이션을 통한 거래 시간을 정하는 것이 중요한데, 이 기능을 통해 더욱 원활한 커뮤니케이션을 이루어 낼 수 있다.

## H. QR code generator/scanner

이 기능은 판매자와 구매자 사이 거래를 확인 하기 위한 QR 코드기능이다. 거래 성사 여부를 판정하기 위하여 판매자는 QR 코드를 생성하고, 구매자는 해당 QR 코드를 스캔함으로써 안전한 거래를 이루어 낼 수 있고, 추후 거래 기록에 신뢰성을 높일 수 있다.

## 1.2. Non-functional Requirements

### A. Product requirements

#### 1. Usability

Usability 는 전자 상거래 응용 프로그램 또는 웹 사이트의 성공을 위한 핵심 요구 사항중 하나이다. 시스템은 직관적이고, 간단한 유저 인터페이스를 제공해야 한다. 본 시스템은 각 기능에 접근 할 수 있도록 직관적인 UI 를 제공한다. 또한 사용자들이 카테고리를 설정, 검색 방법을 선택하여 사용자들이 원하는 상품을 쉽고 간편하게 검색할 수 있도록 한다.

#### 2. Performance

본 시스템은 실시간 개인간 거래 서비스이므로 유연한 서비스를 제공해야한다. 거래 중의 과정들이 바로 확인이 가능해야 하며 내용의 누락이 없어야 하기 때문에 performance 적인 측면에서 문제가 없도록 해야 한다. 또한 사용자의 판매등록이나 등록된 상품에 대한 추천 필터 적용 등에 시간적인 문제가 발생 해서는 안되고 불필요한 지연에 인해 사용자가 불편함을 겪는 일이 없어야 한다.

#### 3. Dependability

본 시스템은 대학 내부의 사용자간의 거래를 가능하게 하기 위해서 실시간으로 거래가 진행되기 때문에 거래중인 상품이 임의로 변경되거나 삭제 되지 않고 유지 되어야한다. 또한 판매자가 등록 상품에 대해 수정 및 변경 또는 삭제를 할 경우 즉각적으로 반영을 하도록 해야 한다.

#### 4. Security



본 시스템의 개인간의 거래를 추구하기 위해 학교 선택과 시간표 등의 개인정보를 수집한다. 따라서 이를 통해 사용자가 누군지 유추 할 수 있기 때문에 외부의 악의적인 침입으로부터 해당 시스템을 보호하고, 고객의 개인정보가 유출되는 불상사가 없도록 한다. 또한 관리자를 위한 접속 경로는 관리자를 만일 알 수 있도록 설계되어야 한다. 관리자를 제외하고는 사용자만이 고유의 카카오톡 아이디를 이용하여 자신의 데이터베이스에 접근하여 관리할 수 있다.

## B. Organizational requirements

### 1. Operational Requirement

본 시스템은 개인간 실시간 거래 서비스 이므로 판매자와 구매자 간의 거래를 진행하기 위한 커뮤니케이션이 원활해야한다. 또한 이용자에게 상품 업로드, 필터링, 상품 거래를 하는 일련의 과정이 유연하게 이어져야 하고, 해당 기능들을 주력으로 제공 하여야 한다.

### 2. Environment Requirement

본 시스템은 카카오톡 i 챗봇 전용 시스템으로 상품 판매, 상품 구매, 필터링, 커뮤니케이션 수단이 카카오톡 환경에 최적화된 화면과 기능을 제공해야 한다.

## C. External requirements

### 1. Regulatory Requirement

본 시스템은 전자 상거래를 제공하는 시스템이므로, 데이터의 무단 전재를 금지하는 시스템의 경우 데이터 수집 방식에 따라 운영 정책을 위반할 소지가 있다. 따라서 정보 수집 대상 시스템의 운영 정책을 면밀히 확인하고, 해당 정책을 위반하지 않는 선에서 데이터를 수집해야 한다.

## 2. Personal Information Protection

본 시스템은 사용자의 기존 카카오톡 아이디에 개인정보를 입력 받는 회원 가입을 해야 서비스를 이용할 수 있는 시스템이다. 그러므로 사용자 식별에 필요한 최소한의 정보만을 수집하고, 사용자가 거래 및 결제, 그리고 추천 시스템을 이용하는데 필수적인 정보의 경우에는 회원가입 과정에서 이용약관에 명시하여 동의하도록 한다.

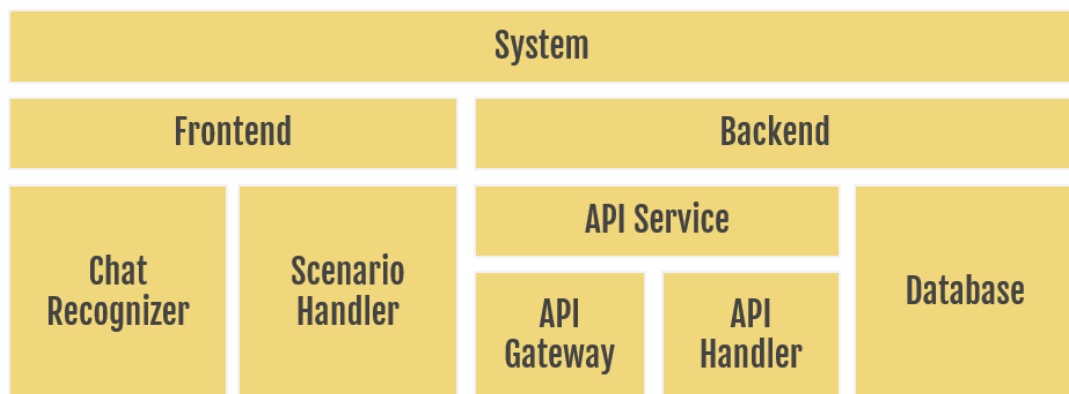
## 5. System Architecture

### 5.1. Objective

목표 시스템의 전반적인 구조의 개괄을 나타낸다. 또한 시스템 기능의 전반적인 분포를 보여준다.

### 5.2. Overview

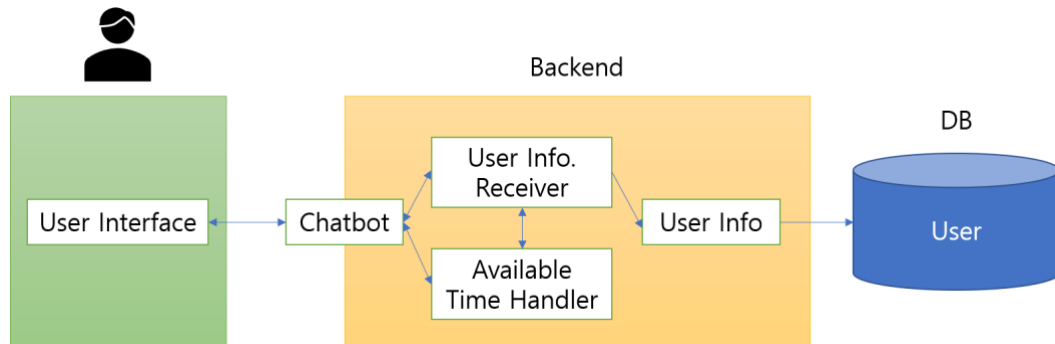
카카오 i와 아마존 웹 서비스(AWS)에서 제공하는 플랫폼을 사용한다. 프론트엔드 관련 플랫폼은 카카오 i에서, 백엔드 관련 플랫폼은 AWS에서 제공하는 것을 사용한다. API 호출 이벤트가 발생했을 때에만 코드를 실행하고, 서버리스로 API 요청을 처리할 수 있도록 한다.



<그림 2 전반적인 시스템 구조>

### 5.3. User Subscription System

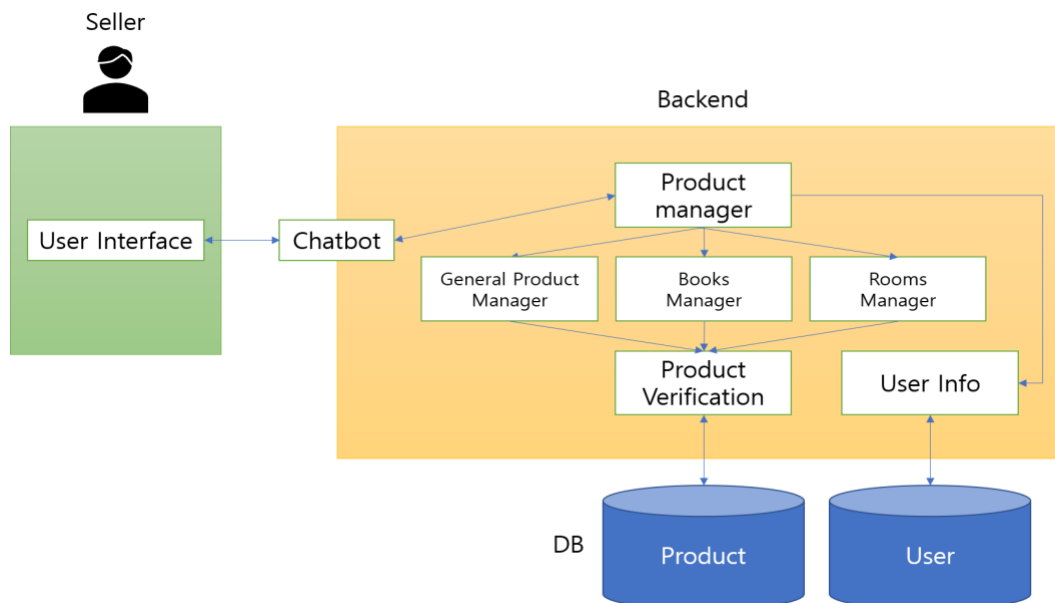
판매자와 구매자 모두 서비스를 이용하기 위해 가입하여야 한다. 사용자로부터 대학 정보를 받아 DB에 저장하며, 여기에는 사용자의 거래 가능 시간 정보가 포함된다.



<그림 3. 가입 시스템 구조>

### 5.4. Seller System

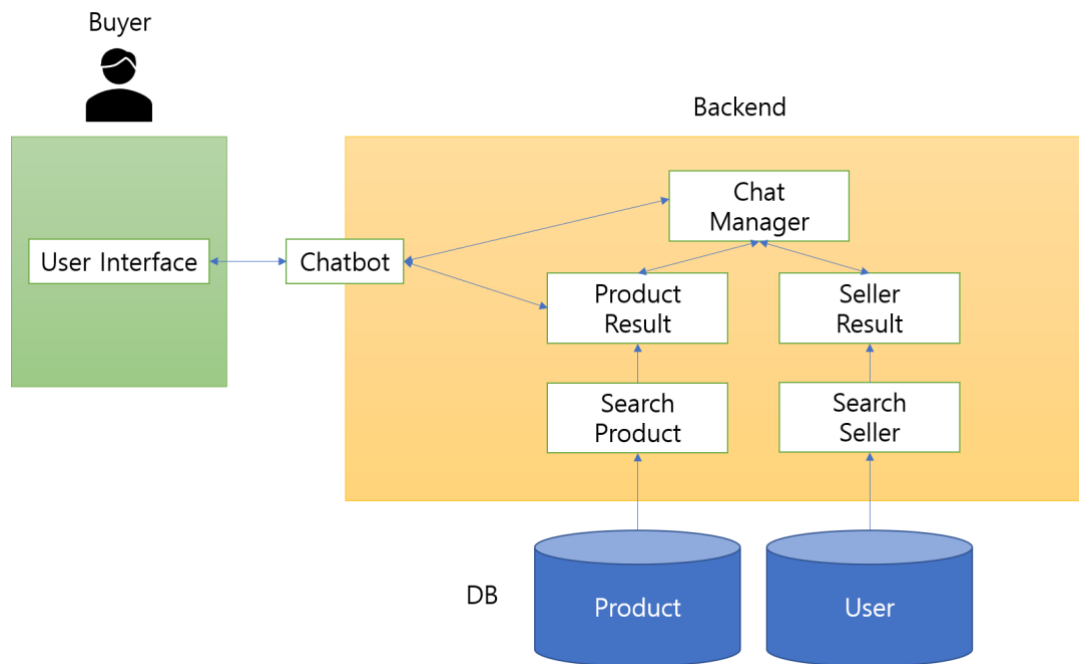
판매자가 물품을 등록할 때에는 Product Manager를 거친다. 판매자가 입력한 정보는 종류에 따라 다르게 처리되며, 검증 절차를 거쳐 DB에 저장된다.



<그림 4. 판매 시스템 구조>

## 5.5. Buyer System

구매자는 상품 검색을 통해 원하는 물품을 찾을 수 있다. DB에서 검색 결과를 표현하여 구매자에게 보여주도록 한다.



<그림5 . 구매 시스템 구조>

## 6. System Requirements Specification

### 6.1 Objective

User Requirement Definition에 정의된 요구사항을 바탕으로 세부적인 Functional Requirements와 Non-functional Requirements를 기술한다.

### 6.2 Functional Requirements

#### A. 회원가입

|       |   |                       |         |
|-------|---|-----------------------|---------|
| 이름    | 회원가입  |                       |         |
| 설명    | 처음 서비스를 이용하는 사용자의 정보를 데이터베이스에 저장한다.                                     |                       |         |
| 입력 소스 | Kakao i   | 출력 대상                 | Kakao i |
| 입력 내용 | NickName  | 사용자가 설정한 별명           |         |
|       | KakaoID   | 사용자의 카카오톡 고유번호        |         |
|       | UnivName  | 사용자의 대학교 이름           |         |
|       | TermAgree   | 서비스 약관 동의 여부          |         |
| 처리 내용 | 데이터베이스에서 동일한 카카오톡 아이디를 가진 사용자가 있는지 검사한 후, 없으면 입력된 사용자 정보를 데이터베이스에 저장한다. |                       |         |
| 출력 내용 | Message   | 처리 성공 여부에 대한 카카오톡 메시지 |         |
| 요구사항  | 사용자가 설정한 대학교에 대한 인증이 완료되어 있어야 한다.                                       |                       |         |
| 영향    | -   |                       |         |

<표 2. 회원가입>

## B. 시간표 등록

|       |   |                       |         |
|-------|---|-----------------------|---------|
| 이름    | 시간표 등록  |                       |         |
| 설명    | 사용자의 에브리타임 시간표 사진을 입력 받아서 사용자가 거래 가능한 시간을 데이터베이스에 저장한다.         |                       |         |
| 입력 소스 | Kakao i   | 출력 대상                 | Kakao i |
| 입력 내용 | KakaoID   | 사용자의 카카오톡 고유번호        |         |
|       | TimetableImage  | 사용자의 에브리타임 시간표 사진     |         |
| 처리 내용 | 사용자의 에브리타임 시간표 사진을 이미지 처리하여 여유 시간을 텍스트 형태로 추출한 후, 데이터베이스에 저장한다. |                       |         |
| 출력 내용 | Message   | 처리 성공 여부에 대한 카카오톡 메시지 |         |
| 요구사항  | 입력된 이미지가 에브리타임에서 생성한 시간표 사진이어야 한다.                              |                       |         |
| 영향    | -   |                       |         |

&lt;표 3. 시간표 등록&gt;

## C. 상품 등록

|       |                                 |                       |         |
|-------|---------------------------------|-----------------------|---------|
| 이름    | 상품 등록                           |                       |         |
| 설명    | 사용자가 판매하고자 하는 상품을 데이터베이스에 저장한다. |                       |         |
| 입력 소스 | Kakao i                         | 출력 대상                 | Kakao i |
| 입력 내용 | KakaolD                         | 사용자의 카카오톡 고유번호        |         |
|       | ItemCategory                    | 상품의 분류                |         |
|       | ItemName                        | 상품의 제목                |         |
|       | ItemPrice                       | 상품의 가격                |         |
|       | ItemImages                      | 상품의 사진들               |         |
|       | ItemDetail                      | 상품의 상세 설명             |         |
|       | ItemISBN                        | 상품의 ISBN (도서)         |         |
|       | ItemLocation                    | 상품의 위치 (원룸)           |         |
| 처리 내용 | 상품의 정보를 데이터베이스에 저장한다.           |                       |         |
| 출력 내용 | Message                         | 처리 성공 여부에 대한 카카오톡 메시지 |         |
| 요구사항  | -                               |                       |         |
| 영향    | -                               |                       |         |

&lt;표 4. 상품 등록&gt;

## D. 도서 ISBN 인식

|       |  |                  |         |
|-------|--|------------------|---------|
| 이름    | 도서 ISBN 인식                                     |                  |         |
| 설명    | 입력된 도서의 ISBN을 인식하고 그 값을 반환한다.                  |                  |         |
| 입력 소스 | Backend  | 출력 대상            | Backend |
| 입력 내용 | ISBNImage                                      | 도서의 ISBN이 포함된 사진 |         |
| 처리 내용 | 도서의 ISBN이 포함된 사진을 이미지 처리하여 ISBN을 텍스트 형태로 추출한다. |                  |         |
| 출력 내용 | ISBN   | 도서의 ISBN         |         |
| 요구사항  | 사진에 ISBN이 포함되어 있어야 한다.                         |                  |         |
| 영향    | -  |                  |         |

&lt;표 5. 도서 ISBN 인식&gt;

## E. 상품 조회

|       |   |               |         |
|-------|---|---------------|---------|
| 이름    | 상품 조회                                       |               |         |
| 설명    | 요청한 기준에 따라 추천된 상품의 목록을 보여준다.                |               |         |
| 입력 소스 | Kakao i                                     | 출력 대상         | Kakao i |
| 입력 내용 | Category                                    | 카테고리          |         |
|       | Name  | 제품명           |         |
|       | ISBN  | ISBN          |         |
|       | Location                                    | 위치            |         |
| 처리 내용 | 입력으로 들어온 내용을 기준으로 사용자에게 적합한 상품 상위 N개를 추천한다. |               |         |
| 출력 내용 | ItemList                                    | 추천된 N개 상품의 내용 |         |
| 요구사항  | 4가지 기준 중 1가지만이 입력 되어야 한다.                   |               |         |
| 영향    | -   |               |         |

&lt;표 6. 상품 조회&gt;

## F. 이미지 검색

|       |  |                  |         |
|-------|--|------------------|---------|
| 이름    | 이미지 검색   |                  |         |
| 설명    | 입력된 이미지와 가장 유사한 이미지를 포함한 상품들의 목록을 반환한다.                      |                  |         |
| 입력 소스 | Backend  | 출력 대상            | Backend |
| 입력 내용 | Image  | 검색 대상이 되는 이미지    |         |
| 처리 내용 | 데이터베이스에 저장된 모든 이미지들과 검색 대상 이미지와의 유사도를 계산한 후, 상위 N개 상품을 반환한다. |                  |         |
| 출력 내용 | ItemList   | 유사도 상위 N개 상품의 내용 |         |
| 요구사항  | -  |                  |         |
| 영향    | -  |                  |         |

&lt;표 7. 이미지 검색&gt;



**G. 판매자 신뢰도 조회**

|       |   |                          |         |
|-------|---|--------------------------|---------|
| 이름    | 판매자 신뢰도 조회  |                          |         |
| 설명    | 판매자의 신뢰도를 조회한다.                                       |                          |         |
| 입력 소스 | Kakao i   | 출력 대상                    | Kakao i |
| 입력 내용 | KakaoID   | 판매자의 카카오톡 고유번호           |         |
|       | ItemID  | 상품의 고유번호                 |         |
| 처리 내용 | 판매자의 거래 기록과 기타 정보를 이용해 설정된 공식에 따라 신뢰도 점수를 계산한 후 반환한다. |                          |         |
| 출력 내용 | Message   | 판매자 신뢰도 점수가 포함된 카카오톡 메시지 |         |
| 요구사항  | 판매자의 카카오톡 고유번호 또는 상품의 고유번호가 반드시 입력되어야 한다.             |                          |         |
| 영향    | -   |                          |         |

&lt;표 8. 판매자 신뢰도 조회&gt;

**H. 거래 시간 추천**

|       |  |                        |         |
|-------|--|------------------------|---------|
| 이름    | 거래 시간 추천   |                        |         |
| 설명    | 판매자와 구매자 간의 거래시간을 등록된 시간표를 바탕으로 추천한다.                    |                        |         |
| 입력 소스 | Kakao i  | 출력 대상                  | Kakao i |
| 입력 내용 | KakaoID1   | 판매자의 카카오톡 고유번호         |         |
|       | KakaoID2   | 구매자의 카카오톡 고유번호         |         |
| 처리 내용 | 판매자와 구매자의 등록된 시간표를 기준으로 공통적인 여유시간을 찾아서 상위 N개 거래시간을 추천한다. |                        |         |
| 출력 내용 | Message  | 추천된 거래시간이 포함된 카카오톡 메시지 |         |
| 요구사항  | -  |                        |         |
| 영향    | -  |                        |         |

&lt;표 9. 거래 시간 추천&gt;

## I. 판매자 거래 확인 QR 발급

|       |  |                     |         |
|-------|--|---------------------|---------|
| 이름    | 판매자 거래 확인 QR 발급                                    |                     |         |
| 설명    | 판매자-구매자 간의 거래 확인을 위한 QR코드를 발급한다.                   |                     |         |
| 입력 소스 | Kakao i  | 출력 대상               | Kakao i |
| 입력 내용 | KakaolD  | 판매자의 카카오톡 고유번호      |         |
|       | ItemID   | 상품의 고유번호            |         |
| 처리 내용 | 판매자의 상품에 대한 거래 확인 일회용 토큰을 발급하고 이를 QR 코드로 만들어 반환한다. |                     |         |
| 출력 내용 | QRCode   | 거래 확인 토큰이 포함된 QR 코드 |         |
| 요구사항  | -  |                     |         |
| 영향    | -  |                     |         |

&lt;표 10. 판매자 거래 확인 QR 발급&gt;

## J. 구매자 거래 확인 QR 인식

|       |  |                       |         |
|-------|--|-----------------------|---------|
| 이름    | 구매자 거래 확인 QR 인식  |                       |         |
| 설명    | 판매자의 거래 확인 QR을 인식하여 판매자와 구매자의 거래를 연결한다.                                      |                       |         |
| 입력 소스 | Kakao i  | 출력 대상                 | Kakao i |
| 입력 내용 | KakaolD  | 구매자의 카카오톡 고유번호        |         |
|       | TransactionID  | 거래 확인 토큰              |         |
| 처리 내용 | 거래 확인 토큰을 이용하여 판매자의 상품 목록에서 거래된 상품을 판매 완료 처리하고 판매자와 구매자의 거래 기록에 거래를 각각 기록한다. |                       |         |
| 출력 내용 | Message  | 처리 성공 여부에 대한 카카오톡 메시지 |         |
| 요구사항  | 거래 확인 토큰이 유효해야 한다.   |                       |         |
| 영향    | 판매자의 상품이 판매 완료 처리된다.   |                       |         |

&lt;표 11. 구매자 거래 확인 QR 인식&gt;

## 6.3 Non-functional Requirements

### 6.3.1 Product Requirements

#### A. Usability

본 시스템은 간편한 중고거래 서비스를 제공해야 한다. 따라서 서비스의 각 기능에 간편하게 접근할 수 있는 방법이 존재해야 하고 사용에 필요한 입력 단계를 최대한 줄여야 한다. 또한 챗봇의 특성상 사용자가 직접 입력하는 단계를 최대한 줄이고 다음 입력 내용을 추천하거나 자동으로 입력 내용을 생성할 수 있어야 한다.

#### B. Performance

본 시스템은 여러 사용자가 동시에 사용하는 것을 전제로 하고 있다. 따라서 사용자의 수에 따라 시스템이 스케일링 되어 사용자의 수에 관계없이 일정한 속도로 서비스를 제공해야 한다.

#### C. Dependability

본 시스템에서 발생하는 모든 등록, 수정, 삭제, 거래 작업은 원자성을 지켜야 한다. 즉, 프로세스가 진행되는 도중에는 다른 프로세스에 의해 영향을 받지 않아야 하며 문제의 발생으로 프로세스가 중단되는 경우 모든 진행 상황이 폐기되어야 한다. 또한 시스템 중단 문제가 발생했을 때 시스템을 재시작하여 바로 이전 상태로 되돌아갈 수 있어야 한다.

#### D. Security

모든 작업은 접근 권한이 있는 사용자와 외부 시스템에 의해서만 실행되어야 한다. 따라서 시스템은 작업을 실행하기 전에 접근 권한을 확인하여 자격이 확인된 대상이 작업 실행을 요청하는지 확인해야 한다.

### 6.3.2 Organizational Requirements

#### A. Operation

본 시스템은 사용자 수에 따라 스케일링이 가능해야 하며 유휴 상태에는 비용을 최소화 해야 한다. 따라서 클라우드 서비스를 이용하여 Serverless 구조로 시스템을 설계해야 한다.

다. 또한 클라우드 서비스에 의존적인 부분을 최소화 해야 한다.

### **B. Development**

병렬적인 개발이 가능하도록 모든 하위 시스템의 인터페이스를 미리 정의해야 한다. 또한 개발팀이 개발 환경에 완전히 적응한 상태가 아니므로 하위 시스템 규모에서는 점진적 개발을 통해 불명확한 부분을 제거해 나가면서 개발해야 한다.

## **6.3.3 External Requirements**

### **A. Regulation**

본 시스템은 전자 거래 서비스를 제공하므로 법률에 위배되는 서비스를 제공하지 않아야 하며 문제 발생시 수사 기관에 협조할 수 있도록 시스템이 설계되어야 한다.

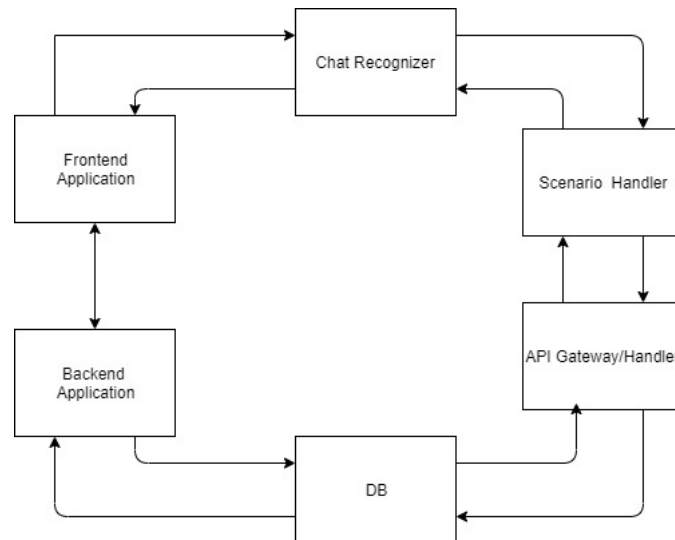
### **B. Personal Information**

거래에 필요한 필수적인 정보를 제외한 개인정보를 외부에 노출해서는 안되며, 최소한의 개인정보만을 수집하여야 한다.

## 7. System Models

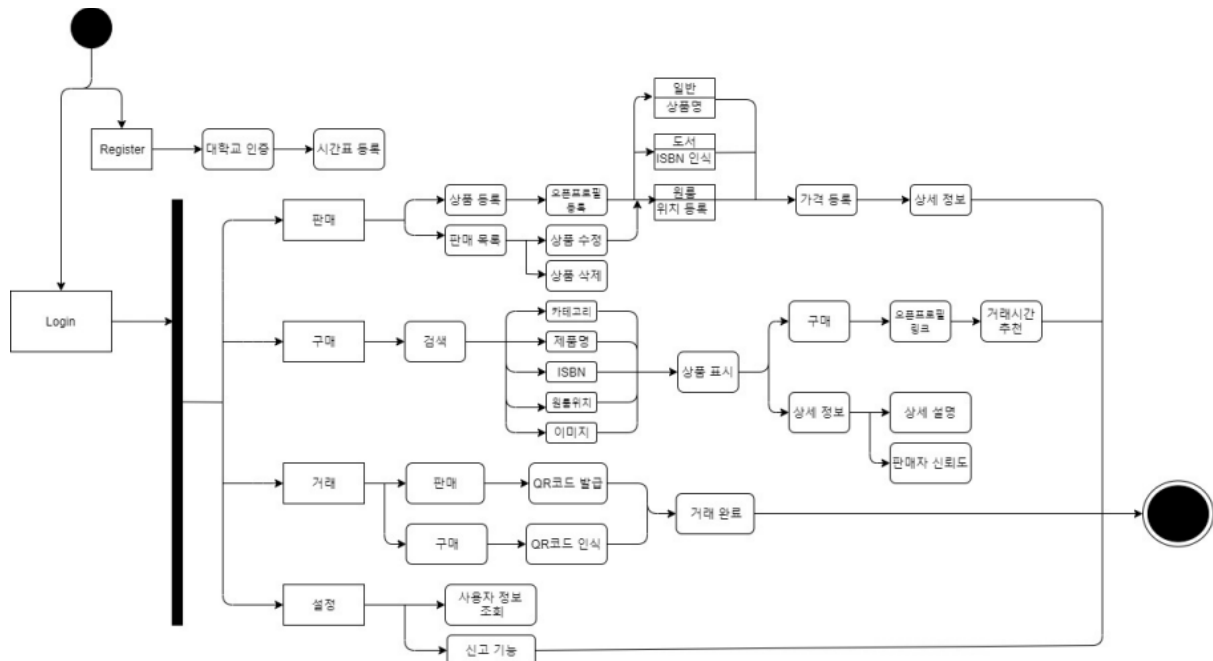
### 7.1 Context Models

#### A. Context Diagram



<Diagram 1 Context Diagram>

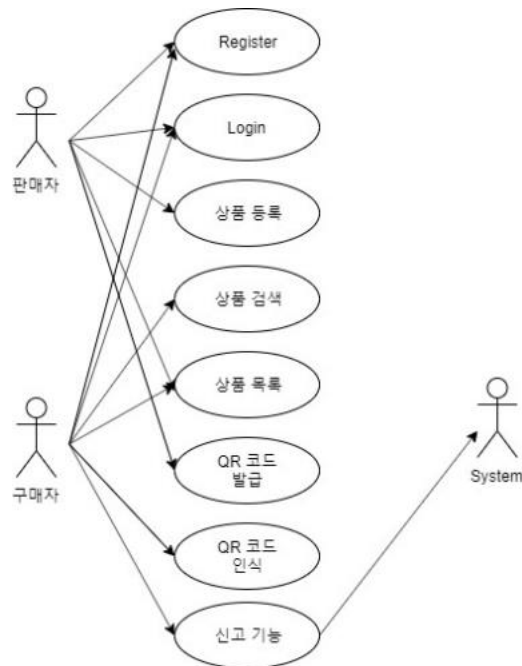
#### B. Process Diagram



<Diagram 2 Process Diagram>

## 7.2 Interaction Models

### A. Use case Diagram



<Diagram 3 Use case Diagram>

### B. Tabular Description of Use Case Diagram

#### 1. Register

| Use case         | Register                         |
|------------------|----------------------------------|
| Actor            | 판매자, 구매자, DB                     |
| Description      | 사용자가 DB 에 회원 등록을 한다.             |
| Trigger          | 사용자가 대학 인증, 시간표 등록과 함께 가입한다.     |
| Success Response | 해당 대학교의 장터 커뮤니티로 등록되어 로그인 가능해진다. |
| Failure Response | 인증에 실패할 경우 가입할 수 없다.             |

## 2. Login

| Use case         | Login   |
|------------------|---|
| Actor            | 판매자, 구매자, DB  |
| Description      | 사용자가 DB 에 등록된 회원 정보와 일치하는지 판단한다.                      |
| Trigger          | 사용자가 ID, 비밀번호를 입력한다.                                  |
| Success Response | ID 와 비밀번호가 DB 의 정보와 일치할 경우, 사용자의 정보와 접속 권한과 함께 로그인된다. |
| Failure Response | DB 에 없거나 일치하지 않을 경우 로그인할 수 없다.                        |

## 3. 상품 등록

| Use case         | 상품 등록  |
|------------------|--|
| Actor            | 판매자, DB  |
| Description      | 판매자가 판매를 원하는 상품을 등록한다.   |
| Trigger          | 판매를 원하는 상품을 해당 카테고리 와 상품 정보와 함께 등록한다.  |
| Success Response | 상품이 도서일 경우 ISBN 인식을, 원룸일 경우 원룸의 위치를 등록하며 그 외엔 제품명을 통해 등록한다. 상세 정보, 가격을 입력한다. |
| Failure Response | 누락된 정보가 있을 경우 등록할 수 없다.  |

## 4. 상품 검색

| Use case    | 상품 검색   |
|-------------|---|
| Actor       | 구매자, DB                                       |
| Description | 사용자가 구매를 원하는 상품을 검색한다.                        |
| Trigger     | 사용자가 카테고리 및 제품명, ISBN, 원룸위치, 이미지 검색을 통해 검색한다. |

|                         |   |
|-------------------------|---|
| <b>Success Response</b> | 검색 조건에 따른 쿼리를 DB 에서 검색해 일치하는 상품들을 보여준다. |
| <b>Failure Response</b> | 일치하는 상품이 없을 경우, 등록된 상품이 없다는 메시지를 보여준다.  |

## 5. 상품 목록

|                         |   |
|-------------------------|---|
| <b>Use case</b>         | <b>상품 목록</b>  |
| <b>Actor</b>            | 판매자, 구매자, DB  |
| <b>Description</b>      | 판매자의 현재 판매 목록을 보여준다.                                  |
| <b>Trigger</b>          | 판매자 정보의 판매 목록을 조회한다.                                  |
| <b>Success Response</b> | 판매자의 유저 정보로 DB 에 등록된 상품들을 user_id 를 이용한 쿼리로 검색해 보여준다. |
| <b>Failure Response</b> | 판매 중인 상품이 없을 경우 상품이 없다는 메시지를 보여준다.                    |

## 6. QR 코드 발급, QR코드 인식

|                         |  |
|-------------------------|--|
| <b>Use case</b>         | <b>QR 코드 발급, QR 코드 인식</b>                    |
| <b>Actor</b>            | 판매자(발급), 구매자(인식)                             |
| <b>Description</b>      | 거래 시에 판매자가 사용할 QR 코드를 발급한다.                  |
| <b>Trigger</b>          | 거래 시에 구매자가 판매자의 QR 코드를 인식한다.                 |
| <b>Success Response</b> | 실제 직거래가 이루어질 경우 QR 코드 인식을 통해 거래 완료 처리할 수 있다. |
| <b>Failure Response</b> | 거래가 이루어지지 않을 경우, 인식할 수 없다.                   |

## 7. 신고 기능

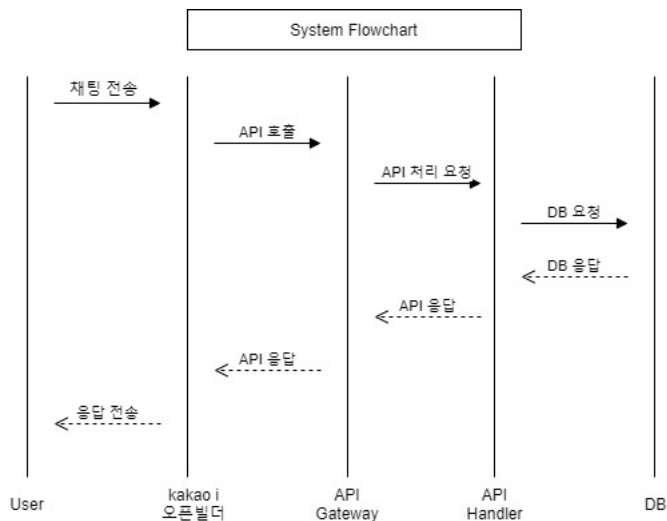
|                 |              |
|-----------------|--------------|
| <b>Use case</b> | <b>신고 기능</b> |
| <b>Actor</b>    | 판매자, 구매자, DB |



|                         |  |
|-------------------------|--|
| <b>Description</b>      | 거래 중 피해를 입은 구매자가 판매자를 신고할 수 있다.  |
| <b>Trigger</b>          | 판매자의 아이디를 신고한다.  |
| <b>Success Response</b> | 신고 횟수가 시스템에 누적되어, 일정 횟수 이상의 신고를 받은 사용자는 신고 횟수가 정보란에 기입되거나, 이용 정지를 당한다. |
| <b>Failure Response</b> | 허위 신고를 막기 위해 하루에 신고 가능 횟수를 제한한다.                                       |

## 7.3 Behavioral Model

### A. Sequence Diagram



## 8. System Evolutions

### 8.1. Objective

시스템의 한계에 대해 서술하고, 어떤 변경사항이 있을지 예측하며 그에 대한 대응 방안까지 설명한다. 이러한 과정을 통해 시스템 변경 사항에 대한 가이드라인을 정리하며, 변화 비용을 최소화함으로써 시스템이 장기적으로 유지될 수 있도록 하는 것에 목표를 둔다.

### 8.2. Limitation and Assumption

챗봇에서 서비스를 구현해야 하는 특성상 많은 데이터를 요구하거나, 높은 사양을 기반으로 하는 기능을 구현하기 어렵다. 사용자에게 편의성을 제공하기 위해 기능의 발전은 이루어져야 할 것이며, 안전성과 혁신성을 위해 해결책이 필요할 것으로 보인다.

또한 기능 중 이미지 검색, QR코드 인식, 시간표 이미지를 인식하여 약속시간을 추천해주는 서비스가 있다. 이 모든 기능은 정확성이 요구된다. 그러나 사용자가 늘어날수록 예외 상황이 많을 것으로 예측되며, 초기 코드를 수정하고 정확성을 높일 수 있는 방안을 마련해야 한다.

### 8.3. Evolution

#### A. 플랫폼 확장

챗봇의 기능적 한계를 극복하기 위해 웹사이트, 어플리케이션과 연동하여 확장한다. 현재 목표는 웹사이트로 먼저 확장하는 것을 염두에 두고 있으며, 이 경우 동기화가 중요할 것이다. 플랫폼 확장을 통해 원룸 GPS기반 검색, 여러 개 물건 한번에 등록하는 기능 등 편의성을 극대화하는 것에 효과적일 것이다.

## **B. 향상된 이미지 처리 기능**

사진의 품질에 구애받지 않고, ISBN코드를 올바르게 인식하도록 지속적으로 코드 수정을 한다. 오픈소스 기반으로 pair programming을 통해 최대한 많은 사람이 코드 유지보수를 가능하게 한다.

## 9. Appendices

### 9.1. Objective

문서 작성 과정에서 생략된 참고자료 등을 기술하고 구체적인 정보를 제공한다. 하드웨어, 데이터베이스 요구사항, 개발환경 요구사항 등이 포함되어 있다. 또한 시스템 사용에 필요한 최소의 설정과 환경, 데이터의 관계와 구조에 대해 설명한다.

### 9.1. Hardware Requirements

서비스가 카카오톡 챗봇을 통해 제공되므로 유저의 환경에 카카오톡이 존재해야 한다. 또한 백엔드 서버와 데이터베이스에서 상품이 등록되고 수정되므로, 동기화를 위해 네트워크에 항상 연결되어 있어야 한다.

### 9.2. Database Requirements

#### A. Seller

판매자 정보를 담고 있는 엔티티

#### B. Buyer

구매자 정보를 담고 있는 엔티티

#### C. Goods

상품 정보를 담고 있는 엔티티

## 10. Index

### 10.1 Table

|                         |    |
|-------------------------|----|
| <표 1. Term Definitions> | 12 |
| <표 2. 회원가입>             | 21 |
| <표 3. 시간표 등록>           | 22 |
| <표 4. 상품 등록>            | 22 |
| <표 5. 도서 ISBN 인식>       | 23 |
| <표 6. 상품 조회>            | 23 |
| <표 7. 이미지 검색>           | 23 |
| <표 8. 판매자 신뢰도 조회>       | 24 |
| <표 9. 거래 시간 추천>         | 24 |
| <표 10. 판매자 거래 확인 QR 발급> | 25 |
| <표 11. 구매자 거래 확인 QR 인식> | 25 |

### 10.2 Figure and Diagram

|                               |    |
|-------------------------------|----|
| <그림 1. 성균관대학교 에브리타임 중고거래 게시글> | 8  |
| <그림 2 전반적인 시스템 구조>            | 18 |
| <그림 3. 가입 시스템 구조>             | 19 |
| <그림 4. 판매 시스템 구조>             | 19 |
| <그림 5 . 구매 시스템 구조>            | 20 |

<Diagram 1 Context Diagram> 28

<Diagram 2 Process Diagram> 28

<Diagram 3 Use case Diagram> 29