**Requirement Specification**

**TEAM #2**

2014310355 김현승

2013312665 나상희

2016314771 김지명

2013314364 강성필

목차

[1. Preface 6](#_Toc7803974)

[1.1 Objective 6](#_Toc7803975)

[1.2 Readers 6](#_Toc7803976)

[A. User requirement readers 6](#_Toc7803977)

[B. System requirement readers 6](#_Toc7803978)

[1.3 Document Structure 6](#_Toc7803979)

[A. Preface 6](#_Toc7803980)

[B. Introduction 6](#_Toc7803981)

[C. Glossary 6](#_Toc7803982)

[D. User requirements definition 6](#_Toc7803983)

[E. System architecture 7](#_Toc7803984)

[F. System requirements specification 7](#_Toc7803985)

[G. System models 7](#_Toc7803986)

[H. System evolution 7](#_Toc7803987)

[I. Appendices 7](#_Toc7803988)

[J. Index 7](#_Toc7803989)

[1.4 Version of the document 7](#_Toc7803990)

[A. Version format 7](#_Toc7803991)

[B. Version management 7](#_Toc7803992)

[C. Version update history 7](#_Toc7803993)

[2. Introduction 8](#_Toc7803994)

[2.1 Objectives 8](#_Toc7803995)

[2.2 Needs 8](#_Toc7803996)

[2.3 Shark 8](#_Toc7803997)

[2.4 Signature 8](#_Toc7803998)

[2.5 Expected effect of Shark 8](#_Toc7803999)

[A. 8](#_Toc7804000)

[B. 8](#_Toc7804001)

[C. 8](#_Toc7804002)

[3. Glossary 8](#_Toc7804003)

[3.1 Objectives 8](#_Toc7804004)

[3.2 Term definitions, acronym and abbreviation 8](#_Toc7804005)

[A. Term definition 8](#_Toc7804006)

[4. User requirements definition 9](#_Toc7804007)

[4.1 Objectives 9](#_Toc7804008)

[4.2 Functional requirements 9](#_Toc7804009)

[A. Sign-up 9](#_Toc7804010)

[B. Login 9](#_Toc7804011)

[C. Posting 9](#_Toc7804012)

[D. Categorizing 9](#_Toc7804013)

[E. Search 10](#_Toc7804014)

[F. AR View 10](#_Toc7804015)

[4.3 Non-functional requirements(Product requirements) 10](#_Toc7804016)

[A. Performance 10](#_Toc7804017)

[B. Security 10](#_Toc7804018)

[C. Dependability 10](#_Toc7804019)

[D. Usability 11](#_Toc7804020)

[4.4 Non-functional requirements(Organization requirements) 11](#_Toc7804021)

[A. Environmental requirement 11](#_Toc7804022)

[4.5 Non-functional requirements(External Requirements) 11](#_Toc7804023)

[A. Security requirement 11](#_Toc7804024)

[B. Regulatory requirement 11](#_Toc7804025)

[5. System architecture 11](#_Toc7804026)

[5.1 Objectives 11](#_Toc7804027)

[5.2 System architecture 12](#_Toc7804028)

[A. Sign-up & Login system 12](#_Toc7804029)

[B. Posting system & Categorizing system 13](#_Toc7804030)

[C. Searching system 15](#_Toc7804031)

[D. AR view system 16](#_Toc7804032)

[6. System requirements specification 17](#_Toc7804033)

[6.1 Objectives 17](#_Toc7804034)

[6.2 Functional requirements 17](#_Toc7804035)

[6.3 Non-functional requirements 17](#_Toc7804036)

[6.4 Scenario 17](#_Toc7804037)

[A. 17](#_Toc7804038)

[B. 17](#_Toc7804039)

[C. 17](#_Toc7804040)

[7. System models 17](#_Toc7804041)

[7.1 Objective 17](#_Toc7804042)

[7.2 Context models 17](#_Toc7804043)

[A. Context model 17](#_Toc7804044)

[B. Process diagram 17](#_Toc7804045)

[7.3 Interaction models 17](#_Toc7804046)

[A. Data-driven modeling 17](#_Toc7804047)

[B. Event-driven modeling 17](#_Toc7804048)

[8. System evolution 17](#_Toc7804049)

[8.1 Objectives 17](#_Toc7804050)

[8.2 Assumption and limitation 17](#_Toc7804051)

[8.3 Evolution of hardware 17](#_Toc7804052)

[8.4 Evolution of Software 17](#_Toc7804053)

[8.5 Changes of user requirements 18](#_Toc7804054)

[8.6 Changes of environment 18](#_Toc7804055)

[9. Appendices 18](#_Toc7804056)

[9.1 Objectives 18](#_Toc7804057)

[9.2 Database requirements 18](#_Toc7804058)

[9.3 User-system requirements 18](#_Toc7804059)

[9.4 Development processes 18](#_Toc7804060)

[10. Index 18](#_Toc7804061)

[10.1 Table index 18](#_Toc7804062)

[10.2 Figure index 18](#_Toc7804063)

[10.3 Diagram index 18](#_Toc7804064)

[11. Reference 18](#_Toc7804065)

# 1. Preface

## 1.1 Objective

Preface 에서는 본 문서의 예상되는 독자들과 문서의 전반적인 구조, 각 부분의 역할에 대해 제시한다. 또한 버전 관리, 버전 변경 기록, 문서의 변경사항들과 근거에 대해 서술한다.

## 1.2 Readers

이 파트는 읽는 대상이 누군인지에 따라 사용자를 중심으로 하는 user requirement, 개발자를 중심으로 하는 system requirement 로 2가지로 나뉜다. User requirement는 고객, 또는 계약자의 관점에서 작성되고 그것을 기반으로 개발자가 이해하기 쉽도록 system requirement를 작성한다.

### A. User requirement readers

User requirement는 고객을 중심으로 작성된 문서이다. 기술적인 지식이 없어도 이해하기 쉽도록 자연어로 고객의 요구사항을 작성한다. 또한 그림이나 표를 이용하여 이해를 돕는다. Client manager, contractor manager, customer등이 주 대상이다.

### B. System requirement readers

각각의 user requirement를 상세히 체계화하여 기술한 것이다. 구조화된 형식을 이용하여 개발자가 이해하기 쉽도록 한다. 고객과 계약자 사이에 계약을 위한 중요한 정보가 된다. 기술자들이 주 대상이다.

## 1.3 Document Structure

이 문서는 크게 10가지 단원으로 나뉜다. 각각의 설명을 기술해 놓았다.

### Preface

Preface에서는 문서의 독자를 정의하고 문서의 전체적인 구조를 다룬다. Version History를 통해 버전마다 수정된 내역과 수정의 이유를 볼 수 있다.

### Introduction

1. Introduction에서는 중고판매 중개 앱 SHARK가 왜 필요한지에 대하여 설명한다. SHARK의 개략적인 구조를 설명하고 이들이 어떻게 business model에 부합하는지를 설명하며, SHARK 로고와 이 시스템으로 인해 얻을 수 있는 기대 효과에 대하여 서술한다.

### C. Glossary

Glossary에서는 본 문서에 서술된 기술적인 용어들에 대해 정의한다. 전문 배경 지식이 없는 독자들이 본 문서를 이해할 수 있도록 서술한다.

### D. User requirements definition

User requirement definition에서는 사용자가 직접 사용하게 될 서비스에 관해 다룬다. 보는 사람이 이해하기 쉽도록 자연어와 표, 그림을 사용하여 기술한다.

### E. System architecture

System Architecture은 예상되는 시스템의 최종 구조를 기술한다. 데이터의 흐름, 함수의 흐름 등을 기술한다.

### F. System requirements specification

System requirement specification은 Functional requirement와 nonfunctional requirement를기술한다. Functional requirement는 시스템에서 사용되는 기능적인 부분을 다룬다. Nonfunctional requirement는 시스템의 보안, 안전성 등 비기능적인 부분을 다룬다.

### G. System models

System model은 외부의 요소, 또는 내부 요소들과 시스템이 어떻게 상호작용하는지 알아보기 쉽도록 만들어진다.

### H. System evolution

System evolution은 시스템을 유지보수 하면서 예상되는 여러가지 변화들에 대해서 다룬다. 하드웨어의 변화, 고객의 요구사항 변화와 같은 사항들을 다룬다.

### I. System evolution

Appendices는 시스템에 대한 자세하고 상세한 정보들을 기술한다. 시스템에 필요한 하드웨어나 데이터 구조 등이 해당된다.

### J. Index

Index는 문서에 포함된 표, 그림 등을 표시해준다.

## 1.4 Version of the document

### A. Version format

버전 번호는 0.1을 시작으로 major.minor[.maintenance]로 구성된다.

### B. Version management

Requirement specification을 수정할 때 마다 버전을 업데이트 한다. 새로운 부분을 추가하거나 구성 순서를 바꾸는 등의 큰 변화가 있을 경우 major number를 변경한다. 기존에 완성된 파트에 사양을 추가하거나 변경을 가하는 경우 minor number를 변경한다. 단순 오탈자를 수정하거나 문서 디자인을 보기 좋게 수정할 때엔 maintenance 번호를 추가하여 변경한다.

### C. Version update history

|  |  |  |
| --- | --- | --- |
| Version | Modified Date | Modified Contents |
| 0.1 | 2019-04-30 | 목차 생성 |
| 1.0 | 2019-05-02 | User requirements 및 System architecture 작성 |
| 2.0 | 2019-05-04 | 6장 정리 |
| 2.1 | 2019-05-04 | 1장정리, 6장 추가 |
| 3.0 | 2019-05-04 | 1장 version management 수정, 11.Reference 양식 추가,  2.Introduction 추가 |

# 2. Introduction

## 2.1 Objectives

Introduction에서는 중고판매 중개 앱 SHARK가 왜 필요한지에 대하여 설명한다. SHARK의 개략적인 구조를 설명하고 이들이 어떻게 business model에 부합하는지를 설명하며, SHARK 로고와 이 시스템으로 인해 얻을 수 있는 기대 효과에 대하여 서술한다.

## 2.2 Needs

### A. IT기반 중고 시장의 폭발적인 성장

불황이기에 오히려 성장하는 시장이 있다. 장기적인 경기침체에 대한 우려가 좀처럼 사라지지 않는 지금 시대에 전통적인 제조업 일종은 어려움에 허덕이고 있지만, 이로 인해 바뀐 소비 트렌드를 노린 각종 판매/보험 서비스는 역설적이게도 호황을 구가하고 있다. 이러한 ‘불황의 역설’에 수혜를 받은 분야는 가성비를 내세운 DIY 가구, 도시락, 저가 생활용품점, 렌탈 업체와 보험업 등이 있다. 중고 시장 역시 소비자들의 ‘가성비’ 소비 트렌드에 힘입어 큰 성장을 보이고 있는 대표적 분야로, 이 분야의 1위 업체인 중고나라에서만 2017년 회원 수 1600만명을 달성하고 2018년에 2조의 거래액을 달성하였다. 비단 중고나라 뿐만 아니라 웹사이트/모바일 앱 기반의 당근마켓, 번개장터, 헬로 마켓 등의 서비스가 다수 런칭, 빠른 성장세를 이어가고 있어 2018년 유통업계 추산 중고시장은 중고차 시장을 제외하고도 20조에 육박하는 것으로 알려졌다.

### B. 모바일 비중이 높은 온라인 쇼핑 시장

온라인 쇼핑 시장에서 모바일의 위상은 다른 매체 (PC, 유선전화, 태블릿)등에 비해 우위에 서있다. KISDI 보고서에 따르면 TV 홈쇼핑이나 해외 직구 같은 분야를 제외하고 오픈마켓, 소셜 커머스, 개인간 직거래(중고거래) 등에서 모바일이 차지하는 비중이 절반 이상이었고 특히 소셜커머스(72%)와 개인간 직거래(61%)에서의 이용 비율이 두드러졌다. 중고 거래 관련 모바일 앱들이 성공적으로 순항중인 이유도 여기에서 기인한 것으로 보인다.

### C. 기존 중고시장 서비스의 문제점

#### C.1 복잡한 카테고리

중고나라에서 판매 물품을 등록할 때엔 물품이 정확한 카테고리에 들어가야 한다. 매물을 둘러보려는 구매자 역시 자신이 관심을 가진 카테고리를 골라 게시물들을 찾아본다. 그러나 사이트/앱에서 허용하는 물건이 다양하면 다양할수록 올릴 수 있는 카테고리 또한 세분화되며, 현재는 지나치게 복잡화 되어있다. 중고나라 카페 왼쪽에 쌓여 있는 카테고리 링크만 해도 수백이고, 다른 앱들은 사정이 그나마 괜찮지만 그래도 복잡한 건 변함없다. 이렇듯 판매자는 자신이 가진 물건을 어디에 올려야 할 지에 대해 시간을 소모하고, 구매자 역사 자기에게 맞는 게시판을 찾아가는 비용이 생긴다. 추가적인 문제도 있다. 판매자는 자신의 물건을 많은 사람이 봐줬으면 하는 마음에 비슷한 분류의 카테고리에 자신의 물건을 중복해서 등록할 수도 있다. 이는 판매자 입장에서도 수고롭고, 구매자 입장에서도 정확하지 않은 중복된 정보를 볼 수 있는 개연성이 생긴다.

#### C.2 물건의 사이즈 정보 부족

직접적으로 물건을 보고 사지 않으면 겪는 문제는 무엇이 있을까? 여러 관점이 있겠지만 일단은 실제 사이즈를 사진만으로 가늠하기 힘들다는 단점이 있을 것이다. 방 안에 중고 미니냉장고를 두고 싶은데, 집이 복잡해 둘 자리는 한정되어 있고 여기에 냉장고가 들어갈 지 거래사이트의 사진만을 보고 확인하기는 힘들다. 판매자가 가로, 세로, 높이 길이를 넣어 놓았거나 모델명을 확인해서 제품 판매 사이트에 검색해 제원을 알아냈다고 하더라도, 내 방의 공간 크기를 줄자 등으로 측정해 봐야 하는 수고가 추가로 든다. 하물며 책상 위에 올려놓을 스탠드를 사더라도 사이즈가 어느정도 일지 가늠해보고 싶은 수요는 생길 수 있다.

### 해결방법

#### D.1 이미지 자동 인식을 이용한 판매품 태그 부착

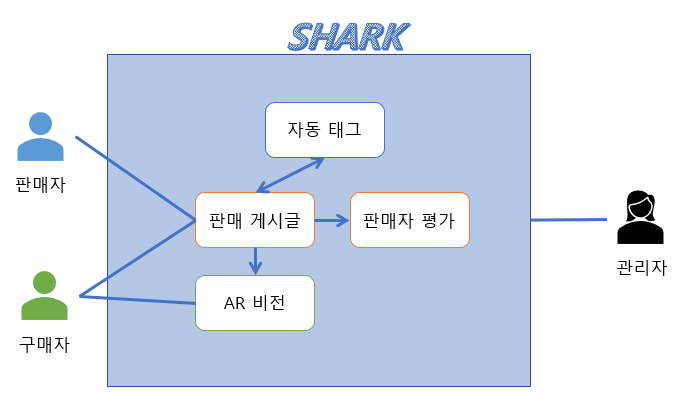
판매자가 판매를 위해 올린 사진을 보고 시스템이 알아서 카테고리를 추가해 준다면 판매자의 수고를 덜어줄 수 있다. 사진을 이미지 인식 및 분석 AI를 이용해 이미 지정 되어있는 태그들을 추천해 준다면 판매자는 태그를 추가할지, 삭제할지 결정만 하면 되도록 의사결정이 단순해지고, 구매자 역시 엉뚱한 태그를 달고 있는 물건을 보지 않아도 될 것이다. 구매자는 자기 입맛에 맞는 태그를 골라 검색한다면 세분화된 게시판 없이도 보다 큰 만족도를 얻을 수 있다.

#### D.2 AR 기술을 이용한 크기 정보 전달

AR로서 자신의 방 안에 가상의 물건을 집어넣어 볼 수 있다면 크기정보 전달을 말끔하게 해낼 수 있다. 실제로 이케아의 모바일 앱에는 실제 가구 모델링을 AR을 이용하여 배치해 볼 수 있는 기능이 성공적으로 쓰이고 있다. 우리의 중고판매 중개 앱에선 비록 판매자가 실제 비율의 판매품 모델을 올릴 수는 없겠지만, 가로 세로 높이 정도의 정보만 입력한다면 구매자의 입장에서는 직육면체 모델만으로도 그 크기를 가늠하는 데에 문제가 없을 것이다.

## 2.3 Shark

SHARK는 C2C 플랫폼으로 핵심 기능은 이미지 분석기반 자동 태그 추천, AR을 이용한 크기 정보 제공이다. 계좌번호, 전화번호, 사용자 평가 등의 반복해서 사용해야 하는 개인정보가 있기에 회원제를 채택하였다.



<Figure 2.1> SHARK의 서비스 구조

판매 게시글은 판매자, 구매자 모두 접근할 수 있는 기능으로

## 2.4 Signature

## 2.5 Expected effect of Shark

### A.

### B.

### C.

# 3. Glossary

## 3.1 Objectives

## 3.2 Term definitions, acronym and abbreviation

### A. Term definition

# 4. User requirements definition

## 4.1 Objectives

User requirements는 User가 원하는 요구사항에 대해 서술하며 크게 기능적 요구사항(Functional requirements)와 비기능적 요구사항(Non-functional requirements)로 나누어 서술한다.

기능적 요구사항은 주로 User가 system에서 필요한 기능에 대해서 정의하고 비기능적 요구사항은 Product requirements, Organization requirements, External requirements로 나누어 서술할 것이며 Product requirements의 경우 System에 관련된 Performance, Dependability, Security, Usability와 같은 내용에 대해 서술한다.

## 4.2 Functional requirements

### A. Sign-up

Sign-up 기능은 User로부터 가입에 필요한 정보를 받아 계정을 생성하는 기능이다. 모든 게시물의 등록, 열람 및 기타 기능 사용에는 Login이 필요하므로 모든 기능을 사용하기 위해서는 Sign-up을 통해 계정을 생성하는 것이 필요하다.

### B. Login

User는 게시물의 등록, 열람 및 기타 모든 기능의 사용을 하기 위해 Login을 해야 한다. User의 아이디와 패스워드가 일치하는 경우에 정상적으로 Login할 수 있고, 제공하는 모든 기능을 사용할 수 있다.

### C. Posting

Posting 기능은 판매자가 판매하고자 하는 물품을 게시물에 등록하는 기능이다. 이 기능을 통해 판매자는 모든 사용자들에게 판매하고자 하는 물품에 대한 정보를 전달할 수 있으며 물품의 사진, 세부 정보, 판매 가격 등을 게시물에 표시한다.

### D. Categorizing

Categorizing 기능은 판매자가 물품을 등록할 때 판매하고자 하는 물품의 사진을 업로드하면 자동으로 물건의 카테고리를 정해주는 기능이다. 판매자는 추천 태그 목록을 보고 원하는 태그를 선택할 수 있으며 원하는 태그가 없을 경우에는 판매자가 직접 태그를 입력할 수 있다. 판매자가 물품의 사진을 등록하지 않는 경우에는 판매자가 태그를 직접 입력하도록 한다.

### E. Search

Search 기능은 판매자가 올린 게시물을 구매자가 검색할 수 있는 기능이다. 카테고리로 분류되어 있는 물품을 찾을 수 있으며 판매자가 선택한 태그를 검색할 수도 있다. 구매자가 쉽게 자신이 원하는 물품을 찾을 수 있도록 도와줄 수 있다.

### F. AR View

AR view 기능은 구매자가 게시물에 등록된 물품을 AR을 사용하여 확인할 수 있는 기능이다. 판매자가 올린 물품의 크기 정보를 바탕으로 구매자가 AR로 물품의 크기를 대략적으로 확인할 수 있는 기능을 제공한다. 크기정보가 중요한 물품의 경우에는 구매자가 직접 물품을 확인하지 않고도 대략적인 크기를 확인할 수 있으므로 번거로움이 줄어들 수 있다.

## 4.3 Non-functional requirements(Product requirements)

### A. Performance

시스템의 주요한 기능인 Categorizing이나 AR view 기능을 포함하여 모든 기능에 있어서 문제가 없어야 한다. 특히 이 시스템의 주요한 기능인 Categorizing이나 AR view 기능은 이 시스템의 주요한 기능이므로 원활하게 작동해야 한다. 뿐만 아니라 웹 서비스에 관련된 시스템인 만큼 원활한 서버와의 통신이 필요할 것이다.

### B. Security

시스템은 사용자의 ID, 비밀번호, 개인 주소 등 민감한 개인정보를 유출하지 않도록 하여야 한다. 한 사용자가 시스템을 사용할 때 불필요한 개인 정보는 다른 사용자에게는 보이지 않도록 하며 데이터베이스 관리를 철저히 해야 한다.

### C. Dependability

시스템은 물품을 등록할 때 등록된 사진에 알맞은 카테고리를 추천해주어야 하며 물품을 검색할 때에는 사용자가 검색한 내용에 대해 적합한 게시물을 제공해야 한다.

### D. Usability

구매자는 물품의 등록을 쉽게, 판매자는 물품의 검색을 쉽게 하기 위한 UI와 UX를 구축하여 User가 편리하게 사용할 수 있도록 한다. 최대한 불필요한 기능을 없애고 간단한 UI를 사용하여 처음 사용하는 User도 어려움 없이 사용할 수 있도록 한다.

## 4.4 Non-functional requirements(Organization requirements)

### A. Environmental requirement

시스템은 안드로이드 앱에서 정상적으로 작동해야 한다. AR기능의 경우 특정 버전이상의 안드로이드 모바일에서 작동하도록 한다. 모바일 기기에서 화면 비율이나 글자가 제대로 나타나도록 해야 한다.

## 4.5 Non-functional requirements(External Requirements)

### A. Security requirement

User가 제공한 사용자의 비밀번호, 휴대폰 번호와 같은 개인정보가 외부로 유출되지 않도록 데이터를 관리하는 것이 필수적이다. 다른 User는 본인의 개인정보 이외에 다른 User의 개인정보 데이터에는 접근하지 못하도록 한다.

### B. Regulatory requirement

User의 개인정보에는 휴대폰 번호나 이름 등이 포함될 수 있는데 User가 제공한 개인정보를 외부에 제공하고자 할 경우에는 사용자의 동의를 받아 사용하도록 한다.

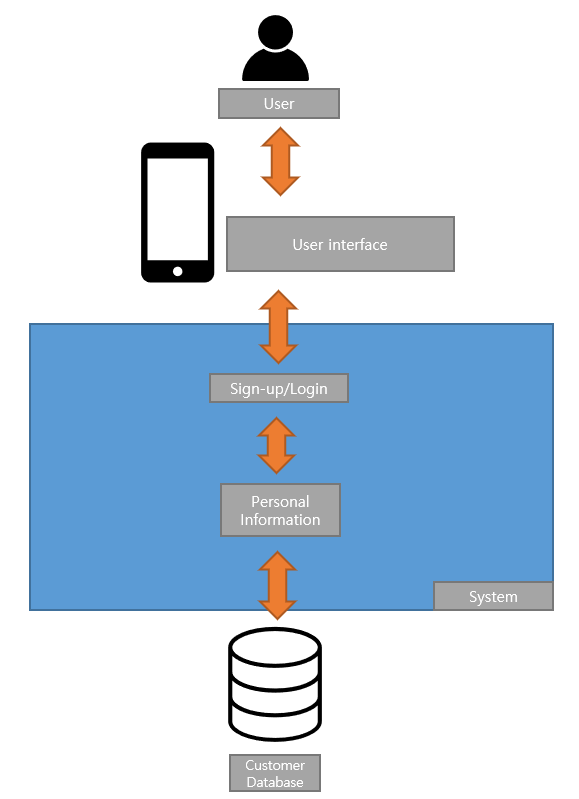
# 5. System architecture

## 5.1 Objectives

System architecture에서는 시스템의 기능 별로 대략적인 흐름에 대해 서술한다. 또한 시스템이 제공하는 기능의 전체적인 분포를 보여준다.

## 5.2 System architecture

### A. Sign-up & Login system

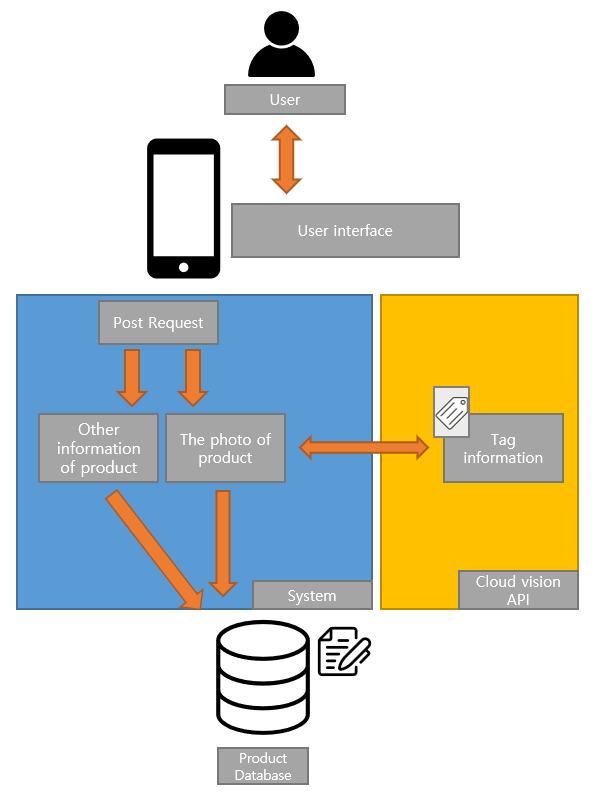


<Figure 5.2.A Sign-up and login system architecture>

Sign up과 Login 기능은 SHARK에 접속했을 때 회원 가입이나 로그인 기능을 수행한다. Sign-up 기능은 사용자가 사용자 정보를 입력하면 Customer database에 저장한다.

Login 기능은 User가 입력한 아이디가 Customer database에 존재하는지 확인하며 존재한다면 User가 입력한 비밀번호가 아이디에 해당하는 비밀번호와 일치하는지 확인한다. 만약 아이디가 존재하지 않거나 비밀번호가 틀린 경우에 다시 입력하도록 요구한다.

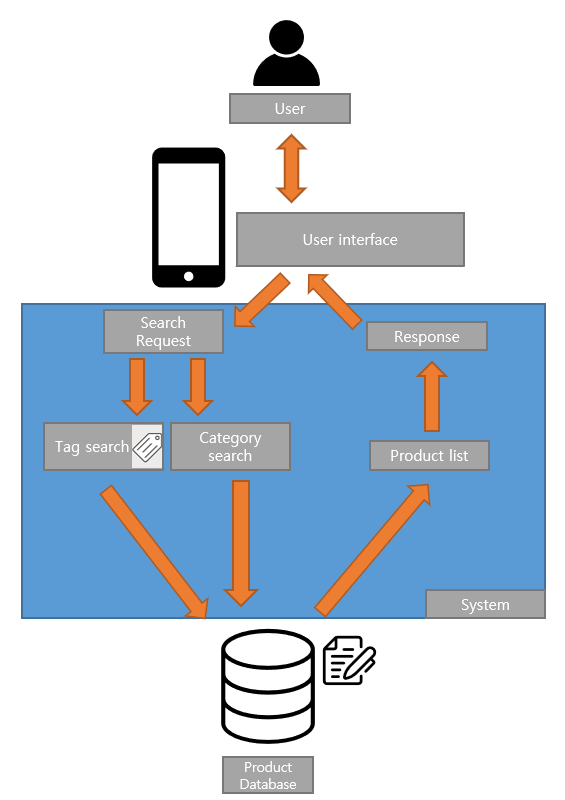
### B. Posting system & Categorizing system



<Figure 5.2.B Posting system & Categorizing system architecture>

Posting 기능은 판매자가 물품을 판매하기 위해 게시물을 올리는 것을 도와주는 기능이다. 판매자는 물품의 이름, 크기 정보, 가격, 사진, 기타 정보 등을 입력할 수 있다. 만약 판매자가 사진을 등록하는 경우에는 Categorizing 기능을 사용할 수 있는데 Categorizing 기능은 판매자는 물품의 사진과 정보를 입력하게 되고 물품의 사진을 바탕으로 Cloud Vision API를 사용하여 tag information을 추출해내는 기능이다. 만약 tag information이 부정확하거나 사용자가 원하지 않는 tag라면 사용자가 원하는 tag를 직접 작성한다. 사용자는 Posting 기능과 Categorizing 기능을 사용하여 판매하고자 하는 물품을 게시판에 올릴 수 있게 된다. 물품에 대한 사용자의 입력 정보, 사진, 태그 등이 Product database에 저장된다.

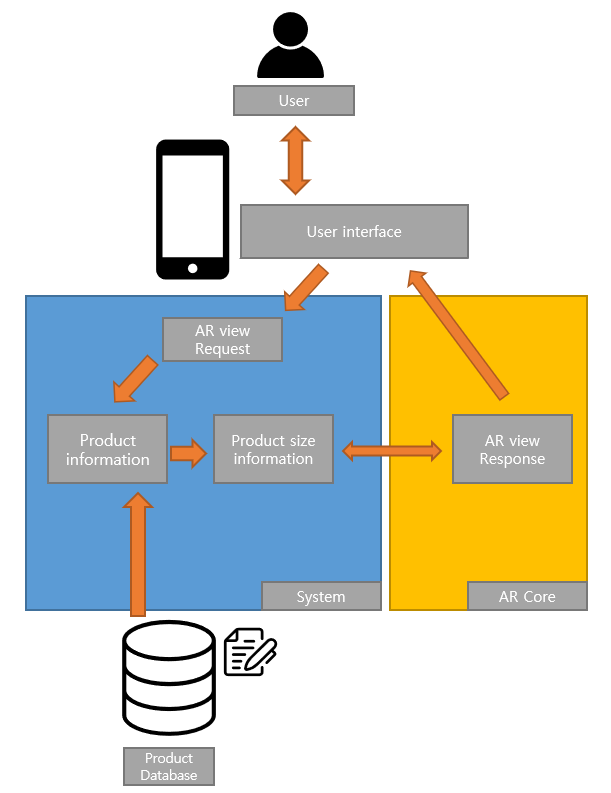
### C. Searching system



< Figure 5.2.C Searching system architecture>

Searching 기능은 사용자가 원하는 물품을 검색할 수 있도록 도와주는 기능이다. 사용자는 검색하고자 하는 Tag나 Category를 입력한다. Product database에 사용자가 원하는 정보가 존재하면 Product database로부터 사용자에게 물품의 정보와 사진들의 리스트를 보여준다. 물품의 리스트는 정확도 순으로 나열하며 정확도가 같으면 최신에 등록된 게시물 순서로 보여준다. 사용자가 검색한 Tag나 Category가 Product database에 존재하지 않는 경우에는 사용자에게 다시 입력을 요구한다.

### D. AR view system



< Figure 5.2.D AR view system architecture>

AR view 기능은 구매자가 물품의 크기를 가상으로 확인할 수 있는 기능이다. 이는 판매자가 Posting기능을 사용할 때 크기 정보를 입력한 경우에만 확인할 수 있다. 만약 물품의 크기정보가 Product database에 존재한다면 사용자가 확인하기 원하는 물품의 크기 정보를 AR Core에게 제공한다. AR Core는 크기 정보를 바탕으로 가상의 물체를 만들어 내고 사용자는 화면에서 AR view를 제공받는다.

AR view기능은 android 7.0버전 이상의 모바일 기기에서만 지원되므로 그 이하 버전은 AR view 기능은 사용할 수 없고 단순한 물품 검색만 가능하다.

# 6. System requirements specification

## 6.1 Objectives

시스템의 기능적 요구사항과 비기능적 요구사항을 명시하고, 시스템을 사용하는데 예상되는 시나리오를 다룬다.

## 6.2 Functional requirements

A. Sign up

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 회원가입 |
| 설명 | 사용자가 사이트를 이용하기 위해 회원정보를 등록하고 회원가입을 할 수 있도록 하는 기능이다. 후에 로그인을 위한 아이디, 비밀번호, 그리고 이름, 전화번호, 주민등록번호 등의 약간의 개인정보를 요청한다. 돈과 물건을 거래하는 사이트인 만큼 개인정보 입력은 필수이다. |
| 입력 | 로그인을 위한 사용자의 아이디, 비밀번호, 이름 전화번호, 주민등록번호 등이 요청된다. |
| 출력 | 가입이 완료된 경우, 가입 완료를 출력한다. 가입을 하는 도중 문제가 생긴 경우 가입이 완료되지 않았다는 신호를 보낸다. |
| 처리 | 가입이 완료된 경우 시스템의 데이터베이스에 사용자의 정보가 저장된다.  비밀번호 같은 경우에는 암호화를 하여 저장한다. |

B. Sign in

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 로그인 |
| 설명 | 사전에 사이트에 회원정보를 등록한 회원이 사이트를 이용할 수 있도록 로그인을 도와주는 기능이다. |
| 입력 | 사용자의 아이디와 비밀번호를 입력으로 받는다. |
| 출력 | 사용자의 입력정보가 데이터베이스상에 존재할 경우, 로그인 완료 신호를 보내준다. 데이터베이스상에 존재하지 않을 경우 가입을 실패하였다는 신호를 보낸다. |
| 처리 | 사용자가 입력한 비밀번호가 사용자가 입력한 아이디에 해당되는 비밀번호가 일치하지 않는 일이 여러 번 나타날 경우 경고 메시지를 띄워준다. |

C. Posting

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 판매 글 작성 |
| 설명 | 판매자가 자신의 물건을 판매하기 위해 글을 등록할 수 있도록 하는 기능이다. 후에 변경사항이 있을 경우 이전에 작성한 글을 수정하여 변경할 수 있다. |
| 입력 | 필수적으로 물건의 사진, 이름, 가격정보, 거래정보, 태그 등을 받는다.  추가적으로 판매자가 물건에 대한 추가정보를 입력한다.  상품의 사이즈정보를 입력하여 Ar testing 기능을 이용할 수 있다. |
| 출력 | 필요한 정보들이 모두 들어갔을 경우 글을 성공적으로 포스트 했다는 신호를 보낸다. 필요한 정보가 빠져 있을 경우 정보가 충분하지 않다는 문구를 띄워준다. |
| 처리 | 판매자가 판매를 위한 글을 작성하면 데이터베이스에 정보가 들어가게 되고 그것을 다시 게시판에서 볼 수 있도록 올려준다. 글을 수정하게 된다면 해당 데이터베이스의 내용 또한 수정해준다. |

D. Add tag

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 판매자의 물건에 태그를 추가해주는 기능이다. |
| 설명 | 구매자가 물건을 구매하고자 할 때, 물건을 찾기 쉽도록 판매자가 물건을 등록할 때, 태그를 달아서 분류를 해주는 기능이다. 판매자가 물건을 판매하고자 할 때, 일일이 카테고리를 분류하는 작업은 귀찮은 작업이라고 판단이 되어 이미지 분류 인공지능인 Google vision을 사용하여 물건에 태그를 추천해준다.  판매자가 판매할 물건의 사진을 올리면 Google Vision을 통해 그 사진과 매칭되는 태그를 추천해준다. 판매자는 추천된 태그를 보고 자신의 물건에 잘 매칭되는 태그를 선택하게 된다. 만약 잘 매칭된다고 생각되는 태그가 존재하지 않는다면 직접 태그를 입력하여 추가할 수 있다. |
| 입력 | 판매할 물건의 사진을 입력 받아서 태그를 추천해주면 판매자는 물건과 잘 매칭된다고 판단되는 태그를 선택한다. 없을 경우 직접 입력하여 추가한다. |
| 출력 | 물건을 분류해줄 태그가 생성된다. |
| 처리 | 물건에 태그가 생성이 된다면 데이터베이스안에 해당 태그에 해당되는 테이블에 물건과 게시글을 추가해준다. |

E. AR testing

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 제품의 사이즈를 체감할 수 있도록 Ar기능을 제공한다. |
| 설명 | 구매자가 물건을 구매하는데 있어서 사이즈 정보는 굉장히 중요하다. 또한 사이즈의 수치만 본다고 해서 그것이 실제로 어느 정도인지 가늠하는 일은 굉장히 어려운 일이다. 그래서 판매자가 제공하는 사이즈 정보를 가지고 실제로 그것의 사이즈를 체감할 수 있도록 AR기능을 제공한다.  판매자가 사이즈정보를 입력하게 되면 그 정보를 가지고 3D 큐브를 만들고 사용자가 AR로 실제 크기를 확인할 수 있도록 도와준다. |
| 입력 | 물건의 사이즈 정보를 받는다. |
| 출력 | 판매자가 입력한 사이즈를 가지고 3D 큐브를 만들어서 AR로 보여준다. |
| 처리 | 물건의 사이즈정보를 받는다. 사이즈 정보를 가지고 3D 큐브를 만든다.  만들어진 큐브를 AR로 띄워준다. |

F. Item search

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 구매자가 구매하고자 하는 물건을 찾을 수 있도록 도와주는 기능이다. |
| 설명 | 구매자가 물건을 빠르게 찾을 수 있도록 도와준다. 또한 여러 비슷한 물건을 보고 싶을 때는 태그를 이용하여 비슷한 물건을 한번에 묶어서 볼 수 있도록 지원해준다. |
| 입력 | 사용자의 검색어 또는 태그를 입력으로 받는다. |
| 출력 | 입력 받은 검색어에 해당하는 물건을 보여준다. 태그가 들어왔을 경우에는 해당 태그로 분류 되어있는 물건들을 보여준다. |
| 처리 | 사용자가 검색어를 입력했을 경우에는 데이터베이스안에 있는 게시글 중 제목이나 본문에 검색어를 포함하고 있는 게시글을 보여준다. 사용자가 태그를 검색했을 경우에는 데이터베이스에서 해당 태그 테이블 안에 있는 물건들을 보여준다. |

G. Recommend Item

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 구매자의 구매 내역, 또는 설정한 관심 태그를 기반으로 물건을 추천해준다. |
| 설명 | 구매자는 자신의 관심태그를 설정할 수 있다. 구매자가 서비스를 사용할 때, 구매자가 지정한 관심 태그를 기반으로 올라와 있는 물건을 추천해준다. |
| 입력 | 사용자의 관심태그 |
| 출력 | 관심태그를 갖는 게시글, 또는 물건 |
| 처리 | 사용자가 지정한 관심태그에 해당되는 데이터베이스 안의 게시글, 또는 물건을 첫 화면에 보여준다. |

H. Seller Grade

|  |  |
| --- | --- |
| Functional Requirement | |
| 기능 | 구매자가 판매자의 평점을 매기는 기능이다. |
| 설명 | 구매자가 물건을 구매 후 물건의 상태 또는 거래 방식 등을 기반으로 판매자를 평가한다. 구매자의 평가는 판매자의 프로필에 남게 되며 이후 판매자와 거래를 원하는 사람은 프로필에서 판매자의 평가를 보고 거래를 좀더 안전하게 할 수 있다. |
| 입력 | 구매자의 별점과 리뷰 |
| 출력 | 구매자의 별점과 리뷰를 포함한 글 |
| 처리 | 구매자가 작성한 별점과 리뷰는 서버를 통해 판매자의 프로필 데이터베이스안에 들어가게 되고 판매자의 프로필에 올려준다. 누구나 그것을 조회할 수 있다. |

6.3 Non-functional requirements

1. Performance

- 판매자가 입력한 사이즈 정보를 가지고 사이즈를 체감할 수 있도록 하는 AR testing 기능을 제공하는 데 있어서 어느정도 정확한 사이즈감을 제공하지 못한다면 오히려 역효과를 불러올 수 있다. AR 기능으로 물건의 사이즈를 본 후 구매를 완료하였는데, 실제 사이즈와 차이가 있어 사용할 수 없는 경우, 사용자는 더 이상 우리 사이트를 신뢰하지 못할 것이다.

- 판매자가 판매 정보를 입력하고 사진을 추가했을 때, 우리 시스템에서 추천해주는 태그정보가 계속해서 형편없을 경우, 판매자를 편하게 해주기 위해 추가된 이 기능이 오히려 판매자가 시스템에 거부감을 느낄 수 있도록 할 수 있다. 그러므로 충분히 정확하고 잘 매칭되는 정보를 주는 것이 중요하다. 또한 태그를 추천하기 위해 걸리는 시간이 너무 길어진다면 더 이상 그 기능을 사용하고 싶지 않을 수 있다. 충분히 빠른 시간내에 정확한 추천을 해주는 것이 중요하다.

- 구매자가 상품 구매를 위해 검색기능을 사용하였을 때, 구매자가 원하는 결과를 띄워주지 못하거나 너무 많은 시간을 필요로 한다면 시스템에 거부감을 느낄 수 있다. 빠른 시간내에 정확한 정보를 띄워주는 것이 중요하다.

2. Security & Safety

- 사이트를 이용하기 위해 사용자들은 개인정보를 입력하여 회원가입을 하기 때문에 개인정보가 외부로 유출되지 않도록 관리하는 것이 중요하다. 또한 거래 도중 사기 등의 문제가 발생하였을 경우 대처를 하기 위해, 사용자가 입력한 개인정보가 신뢰할 수 있는 정보인지 확인하는 것도 중요하다.

3. Usability

- 간편한 UI를 제공하여 사용자가 사용하기 쉽도록 한다. 등록된 게시글, 물건이 많이 질수록 UI가 복잡해질 수 있으므로 등록된 글과 물건들을 잘 분류하여 보기 쉽도록 하는 것이 중요하다.

4. Dependency

- 모든 글과 물건 정보들은 데이터베이스에 저장이 된다. 데이터베이스를 잘 정리하여 후에 검색기능이나 태그 분류기능을 하는데 있어서 불편함이 없도록 한다. 항상 서버와 통신을 해야 하기 때문에 안정적인 서버 구축이 필수적이다.

## 6.4 Scenario

### A. 회원가입 및 로그인

사이트를 이용하기 위해서는 로그인을 해야 한다. 로그인을 할 수 있는 아이디가 존재하지 않는다면 회원가입을 통해 아이디를 생성해야 한다. 회원가입을 하기 위해서는 아이디와 비밀번호 및 몇가지의 개인 정보를 입력해야 한다. 아이디는 다른 아이디들과 중복되어서는 안되고 비밀번호는 몇 가지 조건을 충족해야 한다. 또한 입력한 개인정보는 유효해야 성공적으로 회원가입을 마칠 수 있다.

회원가입을 마친 사용자는 로그인을 하여 사이트를 이용할 수 있다. 로그인을 위해 사용자는 아이디와 비밀번호를 입력한다.

### B. 판매 글 올리기

B.1. 판매 정보 작성

물건을 판매하고자 하는 판매자는 글쓰기를 이용하여 판매 글을 작성한다. 필수적으로 물건 이름, 물건사진, 가격정보, 거래정보 등을 작성하여야 한다. 추가적으로 판매자는 자신의 물건에 대한 정보를 쓸 수 있다.

B.2. 물건 Tagging

판매자가 판매할 물건의 사진을 업로드한다. 업로드 된 사진을 서버로 보내 Google Vision을 적용시킨다. Google Vision을 통해 추출된 태그 중 어느 정도의 일치도 이상의 태그를 불러온다. 해당 태그들을 더욱 적합한 형태로 만들어준다. 만들어진 태그를 판매자에게 추천해준다. 판매자는 추천된 태그를 보고 자신이 원하는 태그를 고른다. 필요하다면 직접 태그를 추가할 수 있다. 판매자가 글을 등록하면서 게시글은 해당 태그들로 분류가 된다.

B.3. AR

판매자는 판매할 물건의 사이즈 정보를 입력하여 AR 기능을 이용할 수 있다. 판매자가 판매 물품 사이즈 정보를 입력하면 시스템은 사이즈 정보를 이용하여 3D 모델을 만든다. 고객은 AR서비스가 제공되는 판매 물품일 경우 AR버튼을 눌러 AR기능을 사용할 수 있다. AR버튼을 누르면 카메라가 켜지면서 카메라 속 3D 모델을 통해 실제 사이즈를 체감할 수 있다.

### 검색 및 구매하기

C.1. 검색 및 구매하기

물건을 구매하고자 하는 고객은 자신이 원하는 물건에 대한 정보를 검색하거나 해당 태그로 들어가서 물건을 찾아본다. 원하는 물건과 관련된 검색어를 이용하여 검색을 할 경우에는 등록된 게시글 중 제목이나 본문에 검색어를 포함한 게시글을 보여준다. 태그를 검색할 경우 해당 태그로 분류된 게시글을 보여준다. 원하는 물건을 발견했다면 해당 물건의 판매자의 정보를 보고 연락을 하여 구매를 완료한다.

# 7. System models

## 7.1 Objective

## 7.2 Context models

### A. Context model

### B. Process diagram

## 7.3 Interaction models

### A. Data-driven modeling

### B. Event-driven modeling

# 8. System evolution

## 8.1 Objectives

## 8.2 Assumption and limitation

## 8.3 Evolution of hardware

## 8.4 Evolution of Software

## 8.5 Changes of user requirements

## 8.6 Changes of environment

# 9. Appendices

## 9.1 Objectives

## 9.2 Database requirements

## 9.3 User-system requirements

## 9.4 Development processes

# 10. Index

## 10.1 Table index

## 10.2 Figure index

## 10.3 Diagram index

# 11. Reference

|  |  |
| --- | --- |
| [1] | 이경은, “조선비즈,” 28 8 2018. [온라인]. Available: http://biz.chosun.com/site/data/html\_dir/2018/08/28/2018082800303.html. [액세스: 4 5 2019]. |
| [2] | 임현우, “불황·IT가 키운 中古시장…중고나라 올해 거래액 2兆,” *한국경제,* p. A1, 24 11 2018. |
| [3] | 최지혜, “온라인쇼핑 이용 현황과 구매 행태 분석,” KISDI, 2017. |