

SSH SHOP

Requirement Specification



Team 7

2015310874 강재현

2017311902 김태규

2013311305 유재현

2015312871 차봉건

2016313856 성아영

목차

1. Preface

A. Objective.....	7
B. Readers.....	7
i. User Requirement Readers.....	7
ii. System Requirement Readers.....	7
C. Document Structure.....	7
i. Preface.....	7
ii. Introduction.....	8
iii. Glossary.....	8
iv. User Requirement Definition.....	8
v. System Architecture.....	8
vi. System Requirement Specification.....	8
vii. System Models.....	8
viii. System Evolution.....	8
ix. Appendices.....	8
x. Index.....	9
D. Version of the Document.....	9

2. Introduction

A. Objective.....	10
B. Needs.....	10
C. SSH SHOP.....	12
D. Expected benefit and effect.....	13

3. Glossary

A. Objective.....	14
B. Term definition.....	14

4. User Requirements Definition

A. Objective.....	16
B. Functional Requirements.....	16
i. Sign in / Log in.....	16
ii. My page.....	16
iii. Upload Product.....	16
iv. Product List.....	17
v. Board.....	17
vi. Search.....	17
vii. Messenger.....	17
viii. Streaming.....	17
C. Non-Functional Requirements.....	18
i. Product Requirement.....	18
1. Usability.....	18
2. Efficiency.....	18
3. Dependability.....	18
4. Security.....	19
ii. Organizational Requirement.....	19
1. Operational.....	19
2. Development.....	19

iii.	External Requirement.....	20
1.	External System Policy.....	20
2.	Personal Information Protection.....	20

5. System Architecture

A.	Objective.....	21
B.	System Structure as a whole.....	21
C.	Frontend Architecture.....	22
D.	Backend Architecture.....	23
E.	Messenger System.....	24
F.	User Product Dealing System.....	25
G.	Board System.....	26
H.	Video Streaming System.....	26

6. System Requirement Specification

A.	Objective.....	27
B.	Functional Requirements.....	27
i.	Login Function.....	27
ii.	My page Display Function.....	28
iii.	Upload Product Function.....	29
iv.	Product List Function.....	29
v.	Board Function.....	30
vi.	Search Function.....	30
vii.	Messenger Function.....	31
viii.	Streaming Function.....	31

C. Non-Functional Requirements.....	32
i. Product Requirement.....	32
1. Usability.....	32
2. Efficiency.....	32
3. Dependability.....	32
4. Security.....	32
ii. Organizational Requirement.....	33
1. Operational.....	33
2. Development.....	33
iii. External Requirement.....	34
1. External System Policy.....	34
2. Personal Information Protection.....	34
D. Scenario Examples.....	34
i. Login Scenario.....	34
1. Initial Assumption.....	34
2. Normal flow of events.....	34
3. What can go wrong.....	34
4. Complete state on completion.....	35
ii. Trade Scenario.....	35
1. Initial Assumption.....	35
2. Normal flow of events.....	35
3. What can go wrong.....	35
4. Complete state on completion.....	36

7. System Models

A. Objective.....	36
B. Context Models.....	36
i. Context Diagram.....	36
ii. Process Diagram.....	37
C. Interaction Models.....	37
i. Use case Diagram.....	37
ii. Tabular description of use case Diagram.....	38
1. Login.....	38
2. Register new items.....	38
3. Write posts on board.....	39
4. View recommendation.....	39
5. View bookmark.....	39
6. Search.....	40
7. Chatting.....	40
8. Purchase Item.....	40
iii. Behavior Models.....	41
1. Login.....	41
2. Upload product and posts.....	41
3. Search.....	41
4. Chatting and video streaming.....	42

8. System Evolution

A. Objective..... 42

B. Limitation and Assumption..... 42

C. Evolution of User Requirement..... 43

D. Evolution of Environment..... 44

9. Appendices

A. Objective..... 44

B. Database Requirement..... 44

C. User-System Requirement..... 44

D. Development Process..... 45

10.Index

A. 표 index..... 46

B. 그림, Diagram index..... 46

1. Preface

A. Objective

Preface는 본 요구사항 명세서에 관한 전반적인 설명이 포함된다. 예상되는 독자들과 이 문서의 전반적인 구조, 그리고 문서의 버전에 관해 설명한다. 예상되는 독자에 맞춰 주로 어떤 부분이 어떤 독자를 대상으로 작성되었는지 설명하고 문서의 각 부분에서 어떤 내용을 주로 설명하려고 했는지, 그리고 이 문서는 어떤 과정으로 수정되어 왔는지를 설명하고자 한다.

B. Readers

먼저 이 문서의 예상되는 독자는 크게 두 부류로 나뉠 수 있다. 이 서비스를 사용하는 사용자 (User), 그리고 이 문서를 활용해 서비스를 개발할 개발자 (System Developer)로 나뉘어 진다.

i. User Requirement Readers

사용자의 관점에서 사용자들이 이용할 수 있는 서비스에 관해 설명한다. 이 서비스의 주 이용자인 판매자와 구매자가 우리 서비스를 이용하면 어떤 이점이 있는지, 또 그 이외의 다양한 이해관계자들을 위해 Non-functional Requirement 사항도 관계자들이 이해하기 쉬운 방식을 통해 설명하고자 하였다.

ii. System Requirement Readers

System Requirement에서는 본 프로젝트에서 개발되어야 하는 특정 기능이 어떤 식으로 구현되어야 하는지 설명한다. 이는 실질적으로 개발을 담당하는 Software Developer, 더욱 자세하게는 Client engineers, System architects 그리고 System end-user까지 이 문서의 독자가 될 수 있다,

C. Document Structure

본 요구사항 명세서는 총 10가지 파트로 구성되어 있다. Preface, Introduction, Glossary, User Requirement Definition, System Architecture, System Requirement Specification, System Models, System Evolution, Appendices, 그리고 Index가 그 10가지 파트다. 각 파트의 역할과 내용은 아래와 같다.

i. Preface

Preface에서는 본 문서의 예상되는 독자들과 문서의 전반적인 구조와 각 부분의 역할, 그리고 이 문서의 버전과 각 버전에서 변화된 내용에 관해 설명한다.

ii. Introduction

본 시스템을 구상한 배경과, 그 배경에서 도출해낸 needs를 위해 구현하고자 하는 주요 기능, 그리고 본 서비스를 이용했을 때 서비스 이용자들이 얻을 수 있는 예상되는 기대효과에 관해 설명한다.

iii. Glossary

본 문서를 조금 더 명확하고 이해하기 쉽게 하기 위해 여러 용어를 정의해 두었다.

iv. User Requirements Definition

사용자에게 어떤 서비스를 제공하는지에 대하여 설명한다. 개발자가 아닌 사용자가 이해하는 것을 목적으로 하며 사용자가 잘 이해할 수 있도록 자연어와 시각자료를 활용해 각 요구사항들을 설명한다.

v. System Architecture

시스템의 전반적인 구조를 설명한다. 전체 시스템 구조부터 각 sub-system과 sub-system간의 관계에 대해서 설명한다.

vi. System Requirements Specification

User requirements definition에서 더 나아가 세부적인 것들을 포함한 설명을 functional requirements와 non-functional requirement로 나누어 명시한다.

vii. System Models

System component와 system 그리고 system 주변의 환경 간의 관계를 기술하기 위함이다.

viii. System Evolution

시스템이 기반하고 있는 주요 가정들과 제약사항들에 관해서 서술한다. 또한 앞으로 예상되는 변화에 관해 기술하고 있다.

ix. Appendices

본 시스템에 관해 더욱 자세하고 구체적인 정보를 제공하고자 한다. DB정보 혹은 최소한으로 요구되는 사양정보를 담고있다.

x. Index

본 문서를 작성하는데 사용된 테이블, 다이어그램, 그림 등 여러 자료의 인덱스를 찾기 쉽게 정리한다.

D. Version of the Document

Version	Modified Date	Explanation
0.1	2019.10.15	요구사항 명세서 목차 작성 Introduction 작성
0.2	2019.10.17	Preface 작성 User requirement specification 작성 System architecture 작성
0.3	2019.10.18	System requirement specification 작성 System model 작성 System Evolution 작성 Appendices 작성
1.0	2019.10.27	추가 수정사항 점검 및 마무리 수정 문서 양식 통합

<표 1.1 Version of the Document>

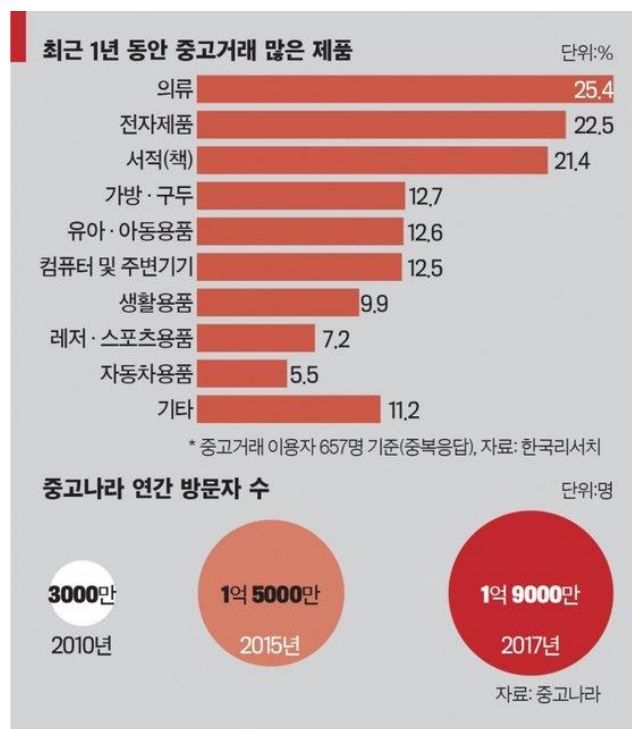
2. Introduction

A. Objective

이 부분에서는 SSH SHOP의 제안 배경과 어떤 요구사항을 반영하고자 하였는지를 설명한다. 또한 이 서비스를 통해 얻을 수 있는 이점과 효과까지 서술하고자 한다.

B. Needs

많은 매체들이 20-30대 소비자층의 키워드로 가성비를 꼽는다. 이를 잘 보여주는 현상이 '중고거래의 증가'이기도 하다. 계속되는 경기 불황, 정보통신기술 (ICT)의 발달 그리고 가성비를 중시하는 20-30대의 합리적인 소비형태가 어우러져 국내 중고거래 시장 규모는 날이 갈수록 커지고 있다. 실제로 2017년 유통업계에서 조사한 자료에 따르면 중고시장의 규모는 약 18조원까지 성장한 것으로 추산된다. 이는 글로벌 금융위기 직후인 2009년 4조원에서 약 4.5배 증가한 수치라고 한다¹. 또 중고거래 서비스 '헬로마켓'에서는 1분에 약 4400개 꼴로 새 매물이 올라오고 있다고 한다.



<그림 2.1 증가하고 있는 중고거래>

¹ "중고시장, 8년만에 4.5배 성장... 18조 규모", 비즈니스리포트, 2017.06.15,

하지만 이처럼 중고거래 규모가 커지면서 따라오는 부작용이 있다. 바로 인터넷 거래 사기이다. 중고거래는 주로 택배거래로 이루어지고, 물건의 상태를 직접 확인하기 어렵다는 점등 여러 이유로 사기의 위험성이 높다는 단점이 있다. 또한 그 물건의 원가를 알기 어렵고, 중고거래는 일반적으로 가격이 싸기 때문에 사기라고 생각하지 못하고 유혹당하기 쉽다는 점도 중고거래의 단점이다.



<그림 2.2 증가하고 있는 사기 발생 건수>

실제로 경찰청 사이버 안전국이 제공한 정보에 따르면 2014년 인터넷 사기 발생 건수가 약 5만건이었던 것에 비해, 2018년에는 약 11만건으로 매우 빠르게 늘고있다.

이런 이유 때문에 우리는 SSH SHOP 서비스를 제공해 중고거래 사기방지를 위한 다양한 방법을 제시하고, 이를 통해 그동안 중고거래에 나쁜 이미지를 가지고 있던 이용자까지 믿고 중고거래를 이용할 수 있게 하고자 하는 것이 우리 서비스의 주목적이다.

C. SSH SHOP

우리 서비스에서 제공하고자 하는 주 기능은 다음과 같다.

1. Messenger system

판매자와 구매자 사이에 번호를 교환해 연락을 취하는 대신 중고거래 서비스 자체에 채팅기능을 넣어 개인정보 보호를 할 수 있게 만들었다.

2. Video Streaming service

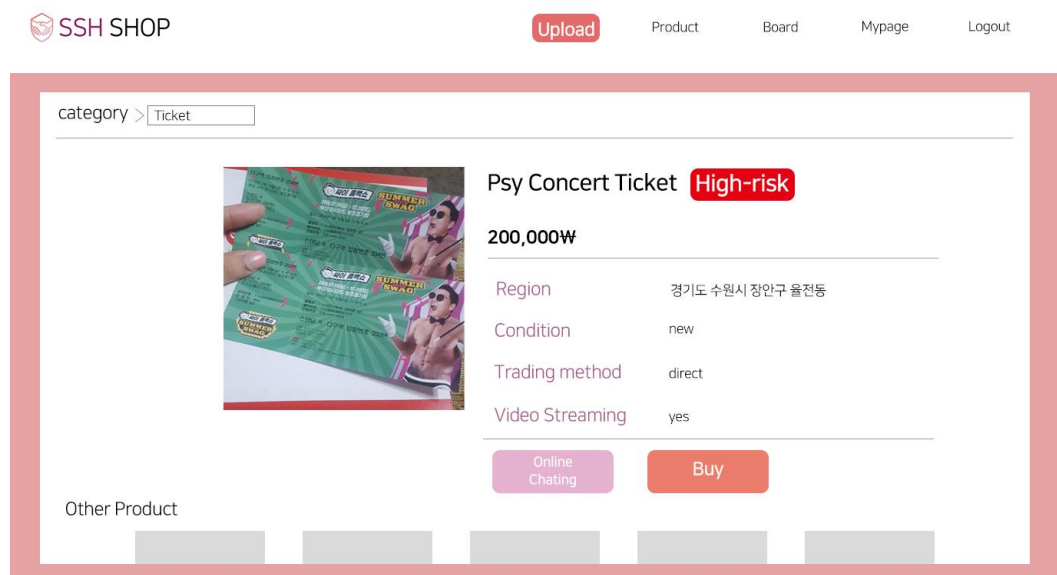
판매하는 물건을 사진으로만 올렸을 때 사기 위험성이 높았다는 점에서 착안해 구매자가 원한다면 판매자가 판매하는 물건을 비디오 스트리밍 기능을 통해 직접 확인할 수 있게 하여 사기 위험성을 낮추고자 하기 위한 기능이다.

3. Indirect-dealing system.

구매자와 판매자 사이에서 직접적으로 돈이 오고 가는게 아니라 중간에 중간관리자를 둔다. 구매자가 물건을 확인하고 잘 받았다는 확인절차 이후 판매자에게 돈을 전송하는 장치를 두어 사기 위험성을 낮추고자 하였다.

4. Alerting a high risk of fraud product

사기 위험성이 높은 품목(쿠폰, 티켓 등 실물 확인을 하더라도 사용가능여부를 알 수 없는 품목)에 한해 구매 페이지에서 다음과 같이 알림을 제공하는 서비스를 만들고자 한다.



<그림 2.3 Alert a high risk of fraud product>

5. Board for reporting fraud

사기를 당했을 때 신고할 수 있는 게시판을 따로 두어 사기를 친 판매자에 제한을 두고 다음에 구매할 때 구매자가 판매자의 신뢰도를 확인할 수 있게 만들고자 한 기능이다.

D. Expected benefit and effect

i. Buyer(구매자)

구매자는 구매 페이지에서 지역, 물건의 상태(new or used), 거래방법, 비디오 스트리밍 가능 여부 등을 직관적으로 확인할 수 있기 때문에 판매자에게 일일이 물어보지 않고도 많은 정보를 알 수 있고 자신이 원하는 물건을 쉽게 구입할 수 있다.

서로 번호를 주고받을 필요 없이 서비스 자체가 제공하는 채팅시스템을 이용해 구매자와 대화를 나눌 수 있어서 개인정보를 보호할 수 있다는 장점이 있다.

만약 안심거래를 원했을 때 판매자가 물건을 보내지 않거나 혹은 이상한 물건을 보내왔을 때 구매자는 쉽게 환불을 받을 수 있고 이에 따라 사기를 당할 위험성이 줄어든다.

만약 원한다면 비디오 스트리밍 기능을 통해 구매하고자 하는 물건의 상태를 더 자세하고 신뢰도 높은 방법으로 확인할 수 있으며 이에 따라 역시 사기 위험성이 줄어들 수 있고 자신이 원하던 물건이었는지 더 쉽게 확인할 수 있다.

ii. Seller(판매자)

판매자는 판매할 물건을 올리기 전에 다양한 항목을 직접 선택할 수 있으며 이를 통해 구매자에게 사전에 다양한 정보를 제공해 허위거래의 가능성을 줄일 수 있다.

구매자와 마찬가지로 서비스 자체에서 제공하는 채팅기능을 통해 개인정보 유출을 방지할 수 있다.

3. Glossary

A. Objective

Glossary에서는 문서에 등장하는 기술적인 용어에 관해 정의한다. 중고거래, 혹은 시스템에 관한 배경지식이 없는 독자들도 본 문서를 이해하기 쉽도록 가능한 모든 용어를 3가지 분류로 나눠 정의하고자 한다.

B. Term definition

i. 사용자 관련 용어

용어	정의
구매자(Buyer)	'SSH SHOP'에 등록되어 있는 중고거래 물품을 구매하고, 서비스를 이용하는 모든 사용자를 의미한다.
판매자(Seller)	'SSH SHOP'을 통해 판매하고자 하는 물품을 등록하고, 제공하는 다양한 서비스를 이용하는 모든 사용자들을 의미한다.
사용자(User)	'SSH SHOP' 서비스를 이용하는 구매자와 판매자 모두를 총칭하는 표현이다.
중간관리자	Indirect-dealing 서비스 제공을 위해 구매자와 판매자 사이에서 중재 역할을 해주는 역할을 수행한다.
관리자	'SSH SHOP' 서비스를 제공하는데 필요한 관리역할을 한다. 중간 관리자를 포함하는 개념으로 시스템을 운영하고 서비스의 운용이나 유지보수 및 여러 관리역할을 책임진다.

<표 3.1 사용자 관련용어>

ii. 서비스 관련 용어

용어	정의
상품(Product)	'SSH SHOP'에 판매자가 판매를 위해 등록한 물품을 의미한다.
Sign in / Log in	서비스 이용을 위해 자신만의 계정을 생성하고 자신의 계정으로 서비스를 이용하는 것을 의미한다. 서비스 사용자는 모두 계정 등록 및 로그인을 해야 한다.

Upload	판매자가 판매하고자 하는 상품을 등록하는 과정을 말한다.
Condition	판매자가 상품을 등록할 때 (Upload) 기입해야 하는 사항으로 상품이 새 상품인지, 사용한 상품인지에 관한 정보를 의미한다.
Video streaming	'SSH SHOP'에서 제공하는 서비스로 상품을 실시간으로 제공되는 동영상을 통해 확인할 수 있는 서비스이다.
Messenger	판매자와 구매자가 소통할 수 있는 방법으로 'SSH SHOP'에서 제공하는 대화서비스를 의미한다.
거래(Dealing)	판매자와 구매자, 혹은 중간관리자 까지를 포함해 상품을 주고받는 과정을 거래라고 통칭한다.

<표 3.2 서비스 관련 용어>

iii. 개발 관련 용어

용어	정의
서버(Server)	네트워크를 통해 정보나 서비스를 제공하는 컴퓨터 시스템을 통칭하는 말이다.
프레임워크	원하는 기능 구현에 집중해 개발할 수 있도록 기본적으로 필요한 기능을 갖추고 있는 틀을 의미한다.
인터페이스(Interface)	서로 다른 두개의 시스템, 장치 사이에서 정보나 신호를 주고받을 때 그 접점을 의미한다. 소프트웨어, 하드웨어, 사람 간에 이루어질 수 있는 소통을 통칭한다.
데이터베이스(DB)	서비스 제공을 위해 정보를 저장하고 있는 저장소를 의미한다.
프론트엔드(Frontend)	우리 눈에 직접적으로 보여지는 영역을 말한다.
백엔드(Backend)	눈에 보이지 않는 서버, 프로그램의 기능 등의 영역을 말한다.

<표 3.3 개발 관련 용어>

4. User Requirements Definition

A. Objective

User requirements definition에서는 사용자에게 어떤 서비스를 제공하는지에 대하여 설명한다. 설명은 Functional requirements, Non-functional requirements로 나누어 설명하며 개발자가 아닌 사용자가 이해하는 것을 목적으로 하며 사용자가 잘 이해할 수 있도록 자연어와 시각자료를 활용해 각 요구사항들을 설명한다.

B. Functional Requirements

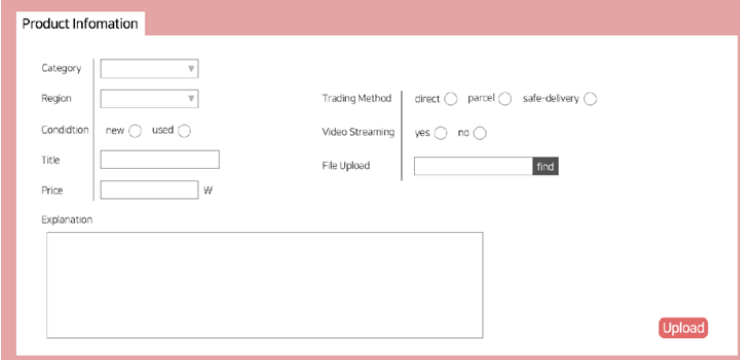
i. Sign in / Log in

회원가입, 로그인 기능이다. Login을 한 사용자들만이 SSH SHOP을 이용할 수 있다. 네이버/카카오톡 아이디를 통해서 회원가입 없이 로그인 가능하고, 이를 원치 않는 사용자들은 회원가입을 통해 계정을 생성하고 이를 통해 로그인 할 수 있다.

ii. My page

User 개인별 페이지이다. Login을 한 후에만 이용할 수 있다. My page에서는 개인별 정보를 조회/수정할 수 있고 자신이 구매한 물품 목록, 자신이 업로드한 물품 목록을 확인/수정할 수 있다.

iii. Upload product



The screenshot shows a web interface for 'SSH SHOP' with a navigation bar containing 'Product', 'Board', 'Mypage', and 'Logout'. The main content area is titled 'Product Information' and contains a form for uploading a product. The form fields are: Category (dropdown), Region (dropdown), Condition (radio buttons for 'new' and 'used'), Title (text input), Price (text input with a '₩' symbol), Trading Method (radio buttons for 'direct', 'parcel', and 'safe-delivery'), Video Streaming (radio buttons for 'yes' and 'no'), File Upload (text input with a 'find' button), and Explanation (text area). An 'Upload' button is located at the bottom right of the form.

<그림 4.1 Product Upload 창>

판매자가 상품을 업로드하는 기능이다. 제품의 category, region, condition(state), title, price, trading method, video streaming, file upload(photo), explanation에 대한 설명을 기술하여 업로드한다.

iv. Product list

SSH SHOP에 upload된 모든 제품들의 list를 보여준다. 가전제품, 전자기기, 의류, 생활용품, 음식, high-risk, 기타 목록으로 구분한다.

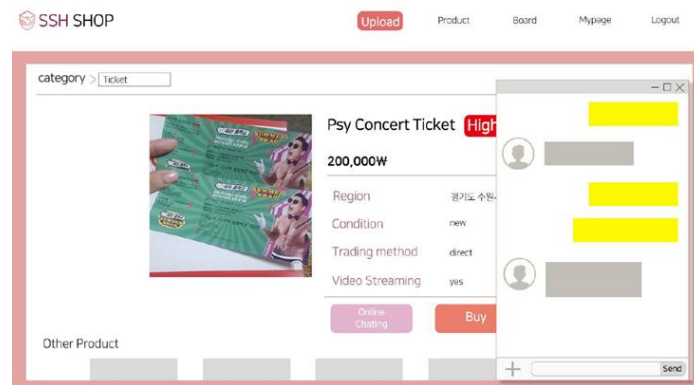
v. Board

사용자들의 게시판이다. 사용자는 궁금한 점이나 사기를 당했을 때 그것에 관련된 내용을 작성하여 업로드할 수 있다. 궁금한 점에 대하여는 관리자가 답변을 작성하고, 사기와 관련된 글은 사용자들이 물건을 구매하기 전에 미리 보면서 반복적인 사기를 방지한다.

vi. Search

제품을 검색하는 기능이다. 돋보기가 그려진 block을 클릭함으로써 검색할 수 있다. Upload된 제품들 중에서 검색어와 일치하는 title을 찾아서 list로 보여준다. 지역을 검색할 경우에는 그 지역의 상품들을 list로 보여준다.

vii. Messenger



<그림 4.2 Messenger 창>

한 제품을 클릭하여 상세정보를 본 경우, Online chatting이라는 버튼을 눌렀을 때 제공하는 기능이다. 이를 통해 제품 seller와 메시지를 주고받으면서 communication을 할 수 있는 messenger창을 제공한다. Buyer는 seller와의 communication을 통해 거래 방법, 거래 가격 조정, 상품을 육안으로 보기 위한 streaming 요청을 할 수 있다.

viii. Streaming

messenger에서 buyer가 요청할 경우 제공받을 수 있는 기능이다. messenger에서 streaming버튼을 누르면 buyer와 seller가 공유할 수 있는 링크를 messenger에서 제공하고 그 링크를 통해 실시간으로 communication할

수 있다.

C. Non-functional Requirements

i. Product Requirement

1. Usability

본 시스템은 사용자들에게 최대한 간편하고 익숙한 User Interface(UI)를 제공할 것이다. 물품 등록, 물품 찾기, 거래 진행 등 전반적인 서비스 동작을 처음 사용하는 사용자들도 쉽게 사용할 수 있도록 할 것이며, 사용자 계정 또한 네이버, 카카오 계정으로 연동할 수 있게 만들어서 편리함을 높일 것이다. 마지막으로 언어는 일단 한국어만 사용 가능한 것으로 한다.

2. Efficiency

A. Performance

지역, 물품 종류, 물품을 사용한 시간 등의 항목들을 기준으로 물품을 검색할 수 있게 하여서, 원하는 물품을 찾는데 걸리는 시간을 줄일 수 있도록 한다. 또한 검색 목록을 띄우는 걸리는 시간은 3초 이내이도록 하고, 채팅 지연 시간은 2초 이내이도록 한다.

B. Space

판매자가 물품을 등록할 때 올릴 수 있는 사진들의 총 용량은 10mb 이내이도록 한다.

3. Dependability

본 시스템은 계정 등록 당시 획득한 개인정보를 절대 유출하지 않는다. 하지만, 거래 도중 사기거래가 발생할 경우, 사기자의 개인정보는 사용될 수도 있다. 물품 등록, 거래 진행과 같은 상황에서 시스템적 오류가 발생하여 서비스 연결이 잠시 끊어지더라도, 서버에 최종적으로 저장된 상태로 다시 돌아가게 하여 문제가 없도록 한다. 또한 streaming, 계정 가입시 개인정보 인증, 안전결제 서비스 등을 통하여, 사기 거래 발생률을 방지하도록 한다.

4. Security

본 시스템은 중고거래시 발생하는 사기문제를 방지하기 위해서, 익명과 실명을 적절히 사용한다. 물품 등록과 같은 경우는 익명으로 작성 가능하지만, 거래시작시 상대방의 정보를 알 수 있다. 이를 위해 사용자는 계정 등록을 진행할 때, 자신 명의의 휴대폰 번호, 주민등록증을 인증해야 한다. 당연히 개인관련 정보들은 외부에서 절대로 접근할 수 없도록 보호되어 진다. 또한 안심결제 서비스를 도입하여, 보안기능을 더 높이도록 한다.

ii. Organizational Requirement

1. Operational

배포하는 방법은 비용과 성능 등을 감안하여 최대한 다양하게 고려해야 한다. 물리 서버의 경우에는 구축 이후 비용을 최소화할 수 있지만, 방화벽과 네트워크 설정을 별도로 해야 하는 불편함을 감수해야 한다. 또한, 초반 구축 비용이 부담을 가중할 수 있다, 그래서 대안으로 클라우드 서버로 눈을 돌리기로 했다. 클라우드 서비스를 제공하는 곳 중에서 대표적인 곳은 네이버, 구글과 아마존이다. 구글과 아마존은 가격대가 비슷하다. 다만 구글은 서버 내 vCPU와 RAM의 크기를 자율적으로 조절할 수 있지만, 아마존은 여러 모델을 기반으로 설치에 용이함을 이점으로 가지고 있다. 선두 주자로서의 많은 정보 역시 아마존 클라우드 서비스인 AWS의 장점이다. 따라서, 구글의 클라우드 서비스인 GCP와 아마존의 AWS 사이에서 고민할 예정이다. 실질적인 배포 방식으로는 사용 예정 WAS(Web Application Server) 프레임워크인 Django를 이용할 예정이나 기타 sub-system간의 integration상에서 문제가 발생할 경우 sub-system간의 조율을 위해서 별도의 웹 서버 프로그램(대표적인 예로 Nginx가 있다)를 별도로 활용해서 문제없이 모든 시스템이 작동하도록 할 예정이다.

2. Development

짧은 시간안에 프로젝트를 완성하기 위하여, Parallel Development를 도입한다. 이를 위해 개발 프로세스를 plan-driven으로 설계하며, 개발 단위는 프론트엔드와 백엔드로 나뉘서 개발을 진행한다. 또한, 프론트엔드와 백엔드는 서로 독립적으로 구동할 수 있도록 설계한다.

iii. External Requirement

1. External System Policy

본 시스템은 사용자들 간 안전한 중고거래를 제공해주는 시스템이다. 사용자가 잘못된 물품 사진을 등록하거나, 의도적인 사기를 치는 등의 행위를 할 경우, 법을 위반한 것으로 간주하여 고소 처리 당할 수 있다. 따라서, 사기 행위에 대한 운영정책을 확실히 정하도록 하며, 사용자에게 본 시스템의 운영정책을 면밀히 확인하고 사용하는 것을 권장하도록 한다.

2. Personal Information Protection

본 시스템은 안전한 중고거래 서비스를 제공하기 위해 개인정보를 수집하고 있다. 따라서, 사용자의 개인정보를 제대로 보호하지 못하는 경우, 민형사상의 책임을 질 가능성이 있다. 하지만 중고거래 사기피해를 막기위해선, 일정이상의 개인정보가 필요하기 때문에, 개인정보가 유출되지 않는데 많은 노력을 기울이도록 한다.

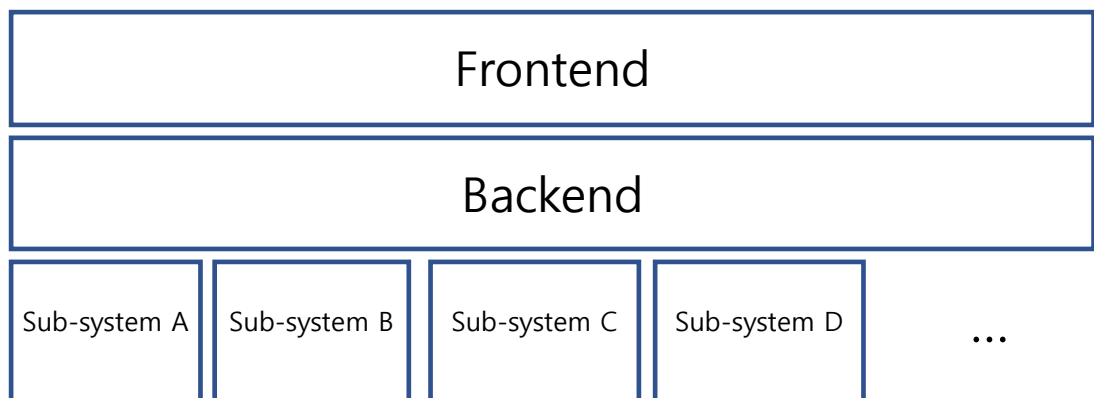
5. System Architecture

A. Objective

이 장에서는, 시스템의 전반적인 구조를 설명한다. 전체 시스템 구조부터 각 sub-system과 sub-system간의 관계에 대해서 설명한다. 보다 정확한 이해를 위하여 구조도를 추가로 첨부한다.

B. System structure as a whole

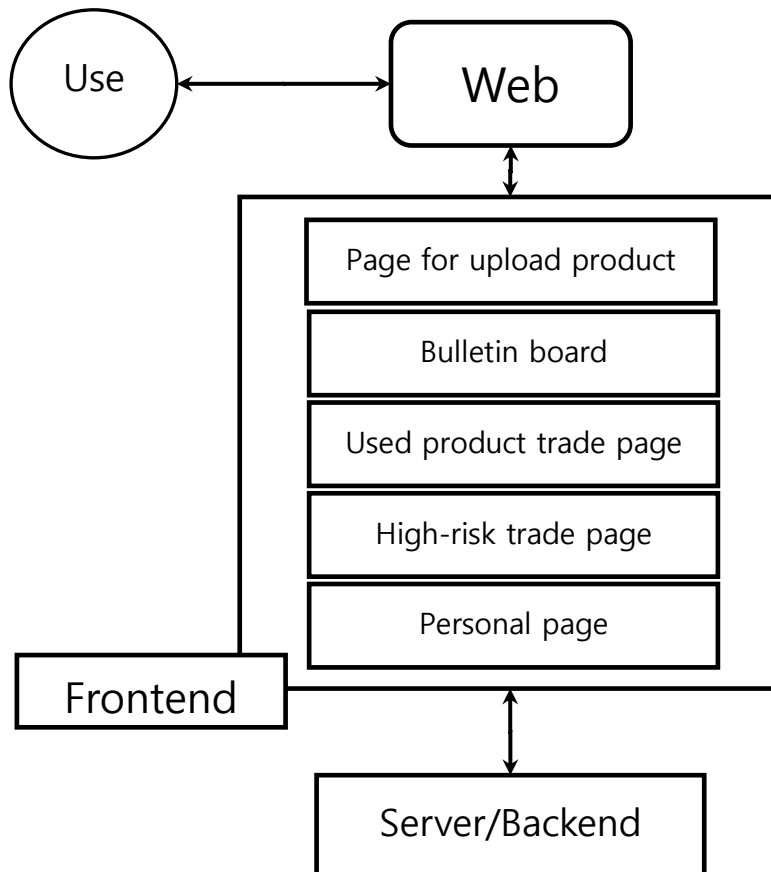
크게 사용자의 요청에 대응하고, 후술할 backend에 사용자의 요청을 넘겨주고 backend의 처리 결과를 기반으로 사용자에게 처리 결과를 전달하는 역할을 주로 하는 frontend와, frontend를 통해 사용자의 요청을 받아 처리하고 그 결과를 frontend로 넘겨주는 sub-system들의 묶음인 backend로 크게 나누어진다.



<그림 5.1 System structure as a whole>

C. Frontend architecture

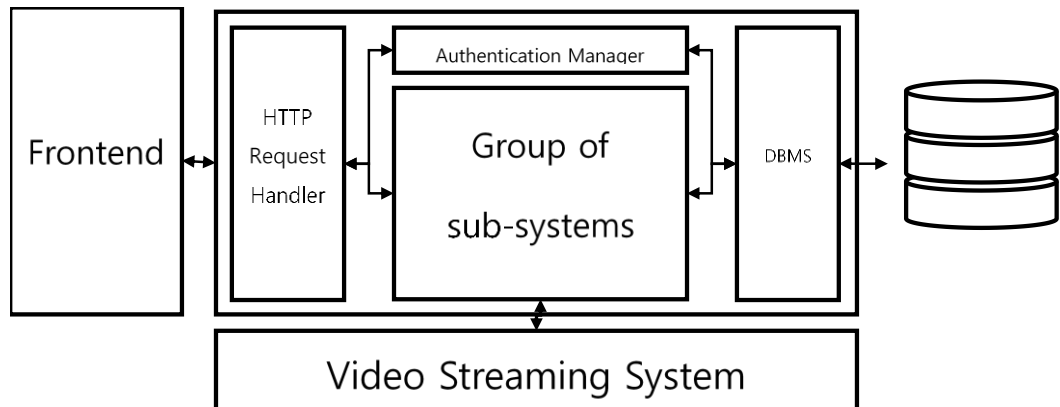
Frontend는 user가 시스템을 이용하면서 확인하는 실질적인 부분이다. 사용자는 backend에 구현되어 있는 정보와 기능들을 frontend를 통해서 이용하며, frontend에서는 사용자의 request를 정리해서 backend를 통해 정보를 탐색하고 respond한다. 따라서 user는 system의 접속 경로인 web browser를 통해서 볼 때 모든 기능을 찾을 수 있어야 한다. Ssh shop이 보여줘야 할 페이지로는 크게 5가지가 있다: 중고 물품 게시 페이지, 중고 거래 페이지, high-risk 거래 페이지, 각종 게시판, 그리고 개인 페이지이다. (로그인 페이지는 backend와의 연동으로 이루어지므로 여기서는 제외하였다.) 이 페이지들은 사용자의 웹 브라우저를 통해서 볼 수 있다.



<그림 5.2 Frontend architecture>

D. Backend architecture

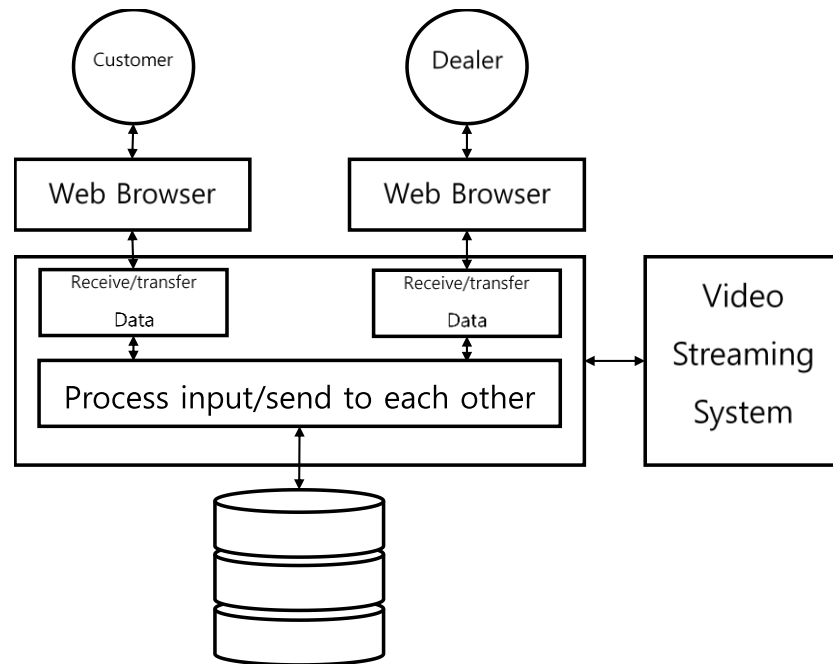
Backend는 frontend에서 받아진 사용자의 요청(request)을 처리하는 sub-system들의 집합체이다. 각 sub-system은 별도로 운용되기보다는 여러 sub-system간에 상호 정보교환을 기반으로 운용된다. Sub-system에는 frontend에서 띄워주는 페이지의 기능들을 처리하는 sub-system과 이런 sub-system들을 지원하는 sub-system들로 구성된다.



<그림 5.3 Backend architecture>

E. Messenger system

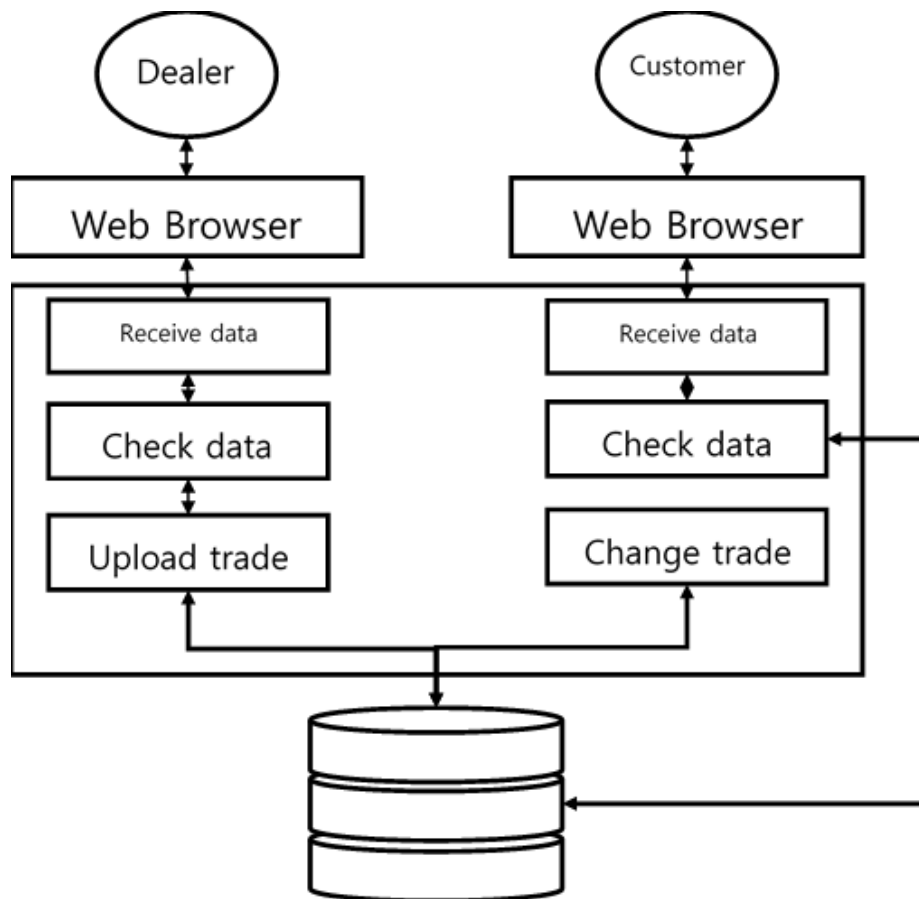
서비스 이용자인 구매자와 판매자간 1:1 메신저 시스템으로, web 브라우저에서 팝업 형태로 나오는 구조이다. DB 시스템과 연동되어 채팅이 진행된다. Video Streaming System과 연동해서 사용자에게 판매자가 촬영한 영상을 메신저 시스템을 통해서 제공할 수 있도록 지원할 계획이다.



<그림 5.4 Messenger System>

F. User product dealing system

판매자가 거래할 물품을 올리는 것부터 구매자가 구입을 결정하고 결재를 하는 일련의 과정을 처리하는 sub-system이다. 마찬가지로 DB와 연동되어 작동될 예정이다. 판매자는 Frontend의 웹 페이지를 통해서 거래를 위해 물품을 올릴 때 필요한 정보들을 기입해 웹 서버로 전송하고 frontend를 통해서 넘겨진 각종 정보들은 이 시스템을 통해서 거래 상태 정보와 함께 DB에 저장된다. 이후, 구매자가 검색을 할 경우 DB에 있는 정보들을 탐색해서 frontend를 거쳐 검색 결과를 전달해주며, 구매자가 구매 결정을 하고 구매에 필요한 각종 정보들을 웹 서버로 전송하면, 마찬가지로 이 시스템이 정보를 처리해서 DB에 저장함과 동시에 결과를 구매자에게 알려준다.



<그림 5.5 User product dealing system>

G. Board system

신고 게시판은 포함한 모든 게시판들의 기반이 되는 하위 시스템이다. 신고 게시판은 추후 중고 거래 관련 데이터 확보를 위하여 별도로 정보를 처리하고, User product dealing system과 연동해서 사기 사건 관련해서 진행해야 할 각종 후속조치들을 진행하고 결과를 DB에 저장함과 동시에 신고자에게 알려준다. 이 정보들은 추후에도 활용될 수 있기에 정리를 해서 저장하면 좋을 수 있다.

H. Video streaming system

Messenger system과 연동되어 소비자에게 판매자가 올린 영상을 바로 볼 수 있도록 지원하는 시스템이다. 소비자에게 판매자가 신뢰를 쌓을 수 있는 시스템으로서 마련한 방법이다. DB를 이용해서 저장하고 시스템에서 제공하는 방법을 통해서 최대한 편리한 방법으로 신뢰를 쌓을 수 있는 방법을 제공하는 것을 목적으로 한다.

6. System Requirements Specification

A. Objective

System requirements specification에서는 User requirements definition에서 더 나아가 세부적인 것들을 포함한 설명을 functional requirements와 non-functional requirement로 나누어 명시한다. Requirements는 각 function의 이름, 기능, 설명, 입력, 출력, 처리, 조건 6가지로 구분하여 설명한다.

B. Functional Requirements

i. Login function

이름	Login function
기능	회원가입이 된 ID/PW 또는 네이버/카카오톡을 통한 로그인 기능.
설명	네이버/카카오톡을 통한 간편 로그인 기능을 지원한다. 간편 로그인을 원하지 않는 경우에는 자체적인 회원가입으로 얻은 ID/PW로 로그인할 수 있다.
입력	네이버/카카오톡 로그인 버튼 클릭, 또는 자체 ID/PW입력 후 로그인 버튼 클릭.
출력	정상적으로 로그인이 된 경우 로그인 된 홈 화면, 로그인이 안된 경우 다시 로그인 화면 출력.
처리	네이버/카카오톡 로그인 버튼을 누른 경우 네이버/카카오톡 API를 이용하여 정보를 받아 DB에 겹치는 것이 없는 경우 저장하고, 자체 회원가입을 한 ID/PW라면 DB에 겹치는 것이 있는지 확인한다. 네이버/카카오톡 로그인의 경우 모든 과정이 잘 완료된다면 홈 화면 출력, 완료되지 않았다면 로그인 화면 출력. 자체 ID/PW가 DB에 있다면 홈 화면 출력, 없다면 로그인 화면 출력.
조건	네이버/카카오톡 로그인의 경우 홈 화면을 띄우는 동시에 사용자의 정보 제공 동의를 얻어야 한다.

<표 6.1 Login function>

ii. My page display function

이름	My page display function
기능	My page를 화면으로 보여준다.
설명	사용자가 My page 버튼을 클릭 했을 때 사용자의 정보를 관리할 수 있게 하고, 사용자가 산 물품 리스트 출력, 사용자가 upload한 물품을 관리할 수 있게 한다.
입력	홈 화면에서 My page 버튼 클릭.
출력	My page 화면 출력.
처리	사용자가 My page 버튼을 클릭 했을 때 DB에서 사용자의 정보를 찾아서 사용자의 개인정보, 사용자가 산 물품 리스트, 사용자가 upload한 물품에 대한 정보를 display하고 사용자가 관리할 수 있게 한다.
조건	Login된 상태에서만 가능하다.

<표 6.2 My page display function>

iii. Upload product function

이름	Upload product Function
기능	새로운 상품을 업로드한다.
설명	사용자가 자신이 팔고 싶은 상품을 제품의 category, region, condition(state), title, price, trading method, video streaming, file upload(photo), explanation에 대한 설명을 기술하여 업로드한다.
입력	홈 화면에서 Upload 버튼 클릭 후 제품의 category, region, condition(state), title, price, trading method, video streaming, file upload(photo), explanation에 대한 정보.
출력	제품 Upload가 업데이트 된 My page 화면 출력.
처리	제품에 관한 정보들을 모아 하나의 Data를 만들고, Upload한 사용자의 data에 추가하여 DB에 저장한다. 그 후에 User의 My page화면을 출력한다. (저장 후 화면 출력이므로 자동 업데이트)
조건	Login된 상태에서만 가능하다.

<표 6.3 Upload product function>

iv. Product list function

이름	Product list function
기능	전체 product list를 출력한다.
설명	홈 화면에서 product list를 눌렀을 때 전체 product를 가전제품, 전자기기, 의류, 생활용품, 음식, high-risk, 기타로 카테고리를 나누어 각각의 list를 보여준다.
입력	홈 화면에서 product list 버튼 클릭.
출력	각각의 카테고리로 나뉘어진 product list 화면 출력.
처리	Product list 버튼이 클릭되면 각 카테고리별로 DB에 있는 정보들을 list로 보여준다.
조건	None.

<표 6.4 Product list function>

v. Board function

이름	Board function
기능	게시판 화면을 출력한다.
설명	홈 화면에서 Board를 클릭했을 때 게시판을 출력한다. 사용자가 자신의 게시물을 게시할 수 있고 다른 게시물을 조회할 수 있다. 게시물은 사기 신고, FAQ 등을 주 목적으로 한다.
입력	홈 화면에서 Board 버튼 클릭.
출력	전체 게시물을 최신 순서로 display하는 화면 출력.
처리	홈 화면에서 Board가 클릭 된 경우 DB에서 게시물 정보들을 읽어와 list로 보여준다. 새로운 게시물은 DB에 시간 순서로 차례차례 저장되게 하여 list로 출력할 때에 순서대로 읽도록 한다.
조건	None.

<표 6.5 Board function>

vi. Search function

이름	Search function
기능	물품 명, 지역 명으로 물품들을 검색한다.
설명	사용자가 원하는 물품의 물품 명이나, 거래하기 가까운 지역을 검색하여 물품을 검색한다.
입력	사용자가 입력한 검색어.
출력	물품 명 혹은 지역 명에 따른 물품 list.
처리	검색어와 일치하는 data가 DB에 있는지 확인한다. 제품의 region과 겹치는 경우 지역을 우선으로 물품을 보여준다. 제품의 title과 겹치는 경우의 물품은 region 다음으로 보여준다.
조건	None.

<표 6.6 Search function>

vii. Messenger function

이름	Messenger function
기능	Buyer와 seller의 messenger를 생성한다.
설명	물품에 대해 궁금한 점이나 거래에 대하여 궁금한 점을 seller에게 직접 물어보고 싶을 때 대화를 할 수 있는 messenger를 생성한다.
입력	제품의 상세 정보 화면에서의 Online chatting 버튼 클릭.
출력	사용자와 Seller의 1:1 messenger 화면 출력.
처리	현재 로그인 되어있는 사용자와 물품을 업로드한 회원의 messenger를 생성, 화면으로 출력한다.
조건	Login한 사용자에게만 제공됨.

<표 6.7 Messenger function>

viii. Streaming function

이름	Streaming function
기능	Buyer가 원하는 상품의 측면을 보게 한다.
설명	Buyer와 seller가 communication할 수 있는 공간을 streaming을 통해 만들어서 buyer가 원하는 상품의 측면을 볼 수 있게 한다. (제품의 결함 확인 용도)
입력	messenger에서의 streaming 버튼 클릭.
출력	Streaming을 제공하는 URL
처리	messenger에서 streaming 버튼이 클릭된 경우 seller와 buyer가 소통할 수 있는 streaming URL을 제공한다.
조건	Login하여 messenger를 이용하는 사람에게 제공됨.

<표 6.8 Streaming function>

C. Non-Functional Requirement

i. Product Requirement

1. Usability

본 시스템은 사용자들에게 최대한 간편하고 익숙한 User Interface(UI)를 제공할 것이다. 물품 등록, 물품 찾기, 거래 진행 등 전반적인 서비스 동작을 처음 사용하는 사용자들도 쉽게 사용할 수 있도록 할 것이며, 사용자 계정 또한 네이버, 카카오 계정으로 연동할 수 있게 만들어서 편리함을 높일 것이다. 마지막으로 언어는 일단 한국어만 사용 가능한 것으로 한다.

2. Efficiency

A. Performance

지역, 물품 종류, 물품을 사용한 시간 등의 항목들을 기준으로 물품을 검색할 수 있게 하여서, 원하는 물품을 찾는 데 걸리는 시간을 줄일 수 있도록 한다. 또한 검색 목록을 띄우는 걸리는 시간은 3초 이내이도록 하고, 채팅 지연 시간은 2초 이내이도록 한다.

B. Space

판매자가 물품을 등록할 때 올릴 수 있는 사진들의 총 용량은 10mb 이내이도록 한다.

3. Dependability

본 시스템은 계정 등록 당시 획득한 개인정보를 절대 유출하지 않는다. 하지만, 거래 도중 사기거래가 발생할 경우, 사기자의 개인정보는 사용될 수도 있다. 물품 등록, 거래 진행과 같은 상황에서 시스템적 오류가 발생하여 서비스 연결이 잠시 끊어지더라도, 서버에 최종적으로 저장된 상태로 다시 돌아가게 하여 문제가 없도록 한다. 또한 streaming, 계정 가입시 개인정보 인증, 안전결제 서비스 등을 통하여, 사기 거래 발생률을 방지하도록 한다.

4. Security

본 시스템은 중고거래시 발생하는 사기문제를 방지하기 위해서, 익명과 실명을 적절히 사용한다. 물품 등록과 같은 경우는 익명으로 작성이

가능하지만, 거래시작시 상대방의 정보를 알 수 있다. 이를 위해 사용자는 계정 등록을 진행할 때, 자신 명의의 휴대폰 번호, 주민등록증을 인증해야 한다. 당연히 개인관련 정보들은 외부에서 절대로 접근할 수 없도록 보호되어 진다. 또한 안심결제 서비스를 도입하여, 보안기능을 더 높이도록 한다.

ii. Organizational Requirement

1. Operational

배포하는 방법은 비용과 성능 등을 감안하여 최대한 다양하게 고려해야 한다. 물리 서버의 경우에는 구축 이후 비용을 최소화할 수 있지만, 방화벽과 네트워크 설정을 별도로 해야 하는 불편함을 감수해야 한다. 또한, 초반 구축 비용이 부담을 가중할 수 있다, 그래서 대안으로 클라우드 서버로 눈을 돌리기로 했다. 클라우드 서비스를 제공하는 곳 중에서 대표적인 곳은 네이버, 구글과 아마존이다. 구글과 아마존은 가격대가 비슷하다. 다만 구글은 서버 내 vCPU와 RAM의 크기를 자율적으로 조절할 수 있지만, 아마존은 여러 모델을 기반으로 설치에 용이함을 이점으로 가지고 있다. 선두 주자로서의 많은 정보 역시 아마존 클라우드 서비스인 AWS의 장점이다. 따라서, 구글의 클라우드 서비스인 GCP와 아마존의 AWS 사이에서 고민할 예정이다. 실질적인 배포 방식으로는 사용 예정 WAS(Web Application Server) 프레임워크인 Django를 이용할 예정이나 기타 sub-system간의 integration상에서 문제가 발생할 경우 sub-system간의 조율을 위해서 별도의 웹 서버 프로그램(대표적인 예로 Nginx가 있다)를 별도로 활용해서 문제없이 모든 시스템이 작동하도록 할 예정이다.

2. Development

짧은 시간안에 프로젝트를 완성하기 위하여, Parallel Development를 도입한다. 이를 위해 개발 프로세스를 plan-driven으로 설계하며, 개발 단위는 프론트엔드와 백엔드로 나뉘서 개발을 진행한다. 이 때, 프론트엔드 개발은 Nux와 Vue를 사용하도록 하고, 백엔드 개발은 Firebase, Django를 사용하도록 한다. 또한, 프론트엔드와 백엔드는 서로 독립적으로 구동할 수 있도록 설계한다.

iii. External Requirement

1. External System Policy

본 시스템은 사용자들 간 안전한 중고거래를 제공해주는 시스템이다. 사용자가 잘못된 물품 사진을 등록하거나, 의도적인 사기를 치는 등의 행위를 할 경우, 법을 위반한 것으로 간주하여 고소 처리 당할 수 있다. 따라서, 사기 행위에 대한 운영정책을 확실히 정하도록 하며, 사용자에게 본 시스템의 운영정책을 면밀히 확인하고 사용하는 것을 권장하도록 한다.

2. Personal Information Protection

본 시스템은 안전한 중고거래 서비스를 제공하기 위해 개인정보를 수집하고 있다. 따라서, 사용자의 개인정보를 제대로 보호하지 못하는 경우, 민형사상의 책임을 질 가능성이 있다. 하지만 중고거래 사기피해를 막기위해선, 일정이상의 개인정보가 필요하기 때문에, 개인정보가 유출되지 않는데 많은 노력을 기울이도록 한다.

D. Scenario Examples

i. Login Scenario

1. Initial Assumption

자체 ID/PW로 login하는 경우에는 해당 ID/PW로 사용자가 회원가입을 한 상태이다. 네이버/카카오톡으로 로그인하는 경우에는 그에 해당하는 계정을 가지고 있어야 한다.

2. Normal flow of events

자체 ID/PW로 login하는 경우 ID/PW 입력 후 로그인 버튼을 누른다. 버튼이 눌러지면 DB에 해당하는 ID/PW를 가진 사용자가 있는지 확인한다. 있다면 그 사용자의 계정으로 로그인한다. 네이버/카카오톡으로 로그인하는 경우 간편 로그인 버튼을 눌렀을 때 해당되는 API가 호출되어 계정을 확인하고 로그인 하도록 한다. Login이 완료되면 해당 계정에서의 홈 화면으로 이동한다.

3. What can go wrong

자체 ID/PW가 DB에 없는 경우, 간편로그인을 했는데 해당하는 계정이 없는 경우, login을 할 수 없고 login화면으로 되돌아간다.

4. Complete state on completion

사용자는 자신의 계정으로 login이 된 홈 화면으로 이동하고, upload/board/buy 등 제공되는 서비스들을 이용 가능해진다.

ii. Trade Scenario

1. Initial Assumption

사용자는 자신이 원하는 카테고리에 충족하는 물품을 찾은 상태이다. 사용자는 자신의 개인정보가 다 인증된 상태의 계정으로 login 되어 있어야 한다.

2. Normal flow of events

사용자는 원하는 물품을 클릭하여 자세한 설명을 볼 수 있다. 판매자가 원하는 금액, 물품의 상태, 물품을 사용한 기간 등을 확인할 수 있으며, 판매자가 전에 한 거래들이 존재하고, 그 거래에 대한 상대방의 리뷰가 존재한다면, 이를 확인할 수 있다. 사용자가 실제 이 물품을 거래하길 원한다면, 판매자와 채팅을 진행할 수 있다. 채팅은 Firebase에서 제공하는 Database를 통해서 진행되며, 그 내용은 다 저장되어진다. 만약 구매자가 streaming 서비스를 원하고, 판매자가 이를 제공해줄 수 있다면 판매자는 streaming video가 담긴 링크를 구매자에게 보내줄 수 있다. Streaming video는 물품의 상태, 판매자 신원 인증 등을 가능케 하는 내용을 담고 있어야 한다.

구매자와 판매자간 합의가 성공되었다면, 거래를 진행할 수 있다. 구매자는 먼저 판매자가 올린 금액만큼 입금을 한다. 이 때, 입금된 금액은 본 시스템에서 관리하고 있다. 구매자의 입금을 하면, 판매자에게 알림이 가고, 알림을 받은 판매자는 물품을 구매자에게 보낸다. 최종적으로 구매자가 물품을 받으면, 구매자는 거래확정 버튼을 눌러야 한다. 이를 누르면, 본 시스템에서 관리하고 있던 금액을 판매자에게 입금해주면서 거래는 완료된다.

3. What can go wrong

사용자가 입금하려는 금액이 판매자가 등록한 물품 가격과 다르면, 입금은 취소된다. 만약 구매자가 입금을 하였는데도, 판매자가 계속해서 물품을

보내지 않는다면, 구매자는 일정 절차 확인 후, 입금한 금액을 100% 환불 받을 수 있다. 만약 고의적으로 구매자가 물품을 받은 후에도 거래확정 버튼을 누르지 않는다면, 판매자는 이를 신고할 수 있고, 일정 절차 확인 후, 구매자는 고소처리 당할 수 있다.

4. Complete state on completion

사용자와 판매자는 각각 거래한 내역이 마이페이지에 등록이 되며, 구매자는 거래에 대한 리뷰를 남길 수 있으며, 이는 판매자의 거래 리뷰 사항에 남게 된다. 이는 나중에 거래를 진행하는 다른 구매자들도 볼 수 있다.

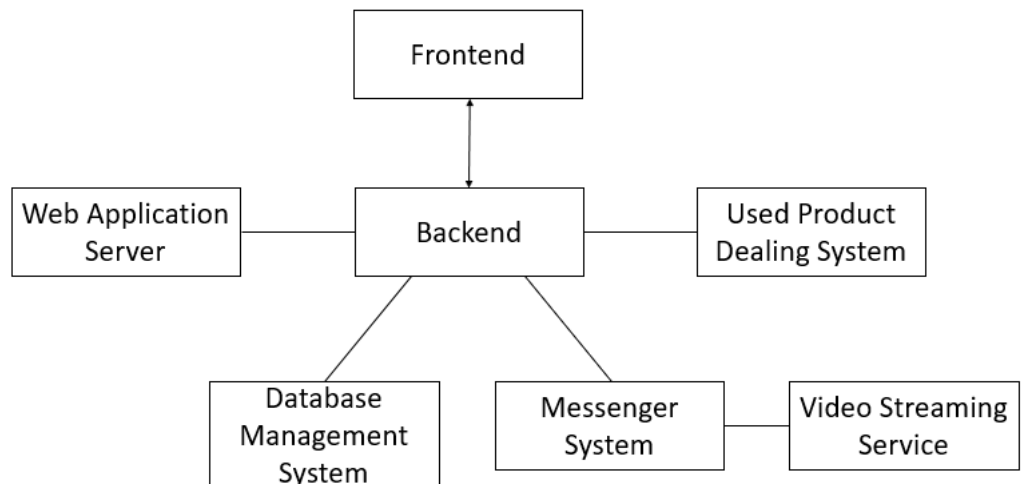
7. System Models

A. Objective

이 장에서는 System component 와 System 그리고 시스템 주변의 환경 간의 관계를 기술한다.

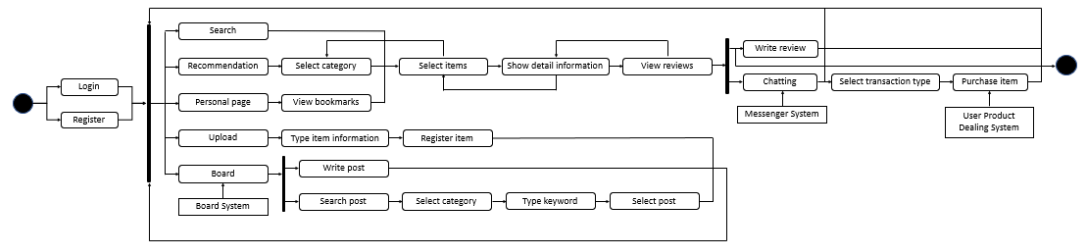
B. Context Models

i. Context Diagram



<그림 7.1 Context Diagram>

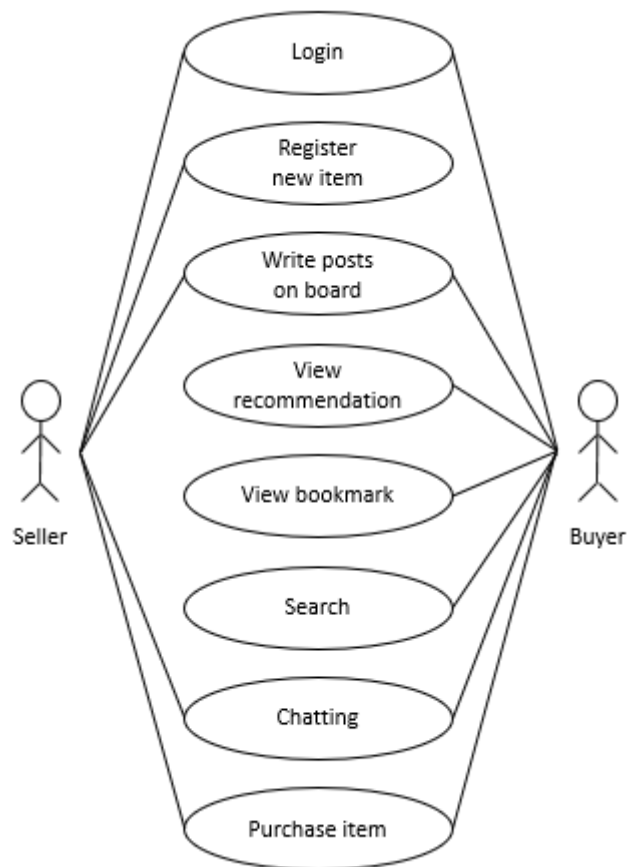
ii. Process Diagram



<그림 7.2 Process Diagram>

C. Interaction Models

i. Use case diagram



<그림 7.3 Use case diagram>

ii. Tabular description of use case diagram

1. Login

Use case	Login
Actor	Seller, Buyer, Database(User)
Description	시스템에 등록되어 있는 유저 정보와 로그인 하려는 사용자가 일치하는지 확인한다.
Trigger	사용자가 ID와 Password(PW)를 입력한다.
Success Response	사용자가 ID와 PW를 입력했을 때 DB에서 해당 ID와 PW의 Hash값이 존재하면 그 사용자에게 해당하는 정보를 반환한다.
Failure Response	사용자의 ID가 Database에 없는 경우 해당 계정이 없다는 메시지를 출력하고 회원가입에 대한 정보를 출력한다. 또한, PW의 Hash가 Database에 존재하는 값과 일치하지 않으면 PW 오류 메시지를 출력한다.

<표 7.1 login>

2. Register new items

Use case	Register new items
Actor	Seller, Database(Item)
Description	판매하려는 상품이 있는 경우 판매자는 제품의 정보를 입력하고 판매글을 등록한다.
Trigger	판매자가 판매 버튼을 클릭한다.
Success Response	상품에 대한 카테고리, 상태, 가격, 거래 방식 등을 입력하고 상품등록 버튼을 누른다. 해당 상품에 대한 글이 게시되고 구매자들로부터 검색된다.
Failure Response	신고가 많이 받는 등의 이유로 신용이 떨어진 판매자의 경우 거래를 제한하는 메시지를 출력하여 상품 등록을 제한한다.

<표 7.2 Register new items>

3. Write posts on board

Use case	Write posts on board
Actor	Seller, Buyer, Database(Board)
Description	사용자는 자유게시판, 신고게시판 등에 글을 작성하고 검색한다.
Trigger	글쓰기 버튼을 클릭하고 내용을 입력한 뒤 확인 버튼을 누른다.
Success Response	사용자는 등록하고자 하는 글을 작성하고 게시물 리스트에 추가한다. 글을 Board Database에 등록되고 다른 사용자로부터 검색 및 확인 가능하게 된다.
Failure Response	부적절한 용어를 사용하거나 부적절한 행위를 하는 사용자의 경우 그에 대한 경고 메시지를 출력하고, 게시글 등록을 제한한다.

<표 7.3 Write posts on board>

4. View recommendation

Use case	View recommendation
Actor	Buyer, Database(Item)
Description	사용자가 원하는 상품에 대한 카테고리를 선택하고 추천 상품을 조회한다.
Trigger	추천 메뉴에서 카테고리를 선택하고 키워드를 입력한다.
Success Response	사용자는 카테고리를 선택하고 키워드를 입력하여 해당 정보와 관련있는 상품을 Item Database에서 검색한다. Item Database에서 해당 정보와 관련있는 상품을 정렬하여 반환하고, 사용자에게 보여준다.
Failure Response	해당 정보와 관련있는 상품이 없는 경우 없다는 메시지와 함께 다른 정보를 입력하도록 한다.

<표 7.4 View recommendation>

5. View bookmark

Use case	View bookmark
Actor	Buyer, Database(Bookmark)
Description	사용자가 관심이 있다고 등록했던 상품 리스트를 조회한다.
Trigger	사용자는 personal page에서 bookmark 조회 버튼을 누른다.
Success Response	해당 사용자의 정보(ID)를 이용하여 Bookmark Database에서 그 사용자에게 대한 Bookmark 리스트를 조회한다. Bookmark Database는 정렬된 리스트를 반환하고 사용자에게 출력된다.
Failure Response	Bookmark된 상품이 없는 경우 아무 내용도 없는 빈 리스트를 출력한다.

<표 7.5 View bookmark>

6. Search

Use case	Search
Actor	Buyer, Database(Item)
Description	사용자는 이름, 카테고리, 키워드 등을 입력하고, 판매글 목록에서 관련 상품을 검색한다.
Trigger	사용자는 검색하려는 상품에 대한 정보를 입력하고 검색 버튼을 누른다.
Success Response	검색 관련 정보는 Item Database에 전달되고, Item Database에서 관련있는 상품 목록을 정렬하여 반환한다.
Failure Response	해당 검색 조건과 관련있는 상품이 없는 경우 없다는 메시지를 출력하고 빈 목록을 보여준다.

<표 7.6 Search>

7. Chatting

Use case	Chatting
Actor	Seller, Buyer
Description	판매자와 구매자가 채팅창을 통해서 상품에 대한 대화를 나눈다.
Trigger	구매자가 Online Chatting 버튼을 누른다.
Success Response	구매자가 Online Chatting 버튼을 누르면 Messenger system에 구매자와 판매자가 연결된다. 두 사용자는 채팅창을 통해서 상품에 대한 대화를 나눈다.
Failure Response	판매자가 Chatting이 불가능한 상황이면 오류 메시지를 출력한다.

<표 7.7 Chatting>

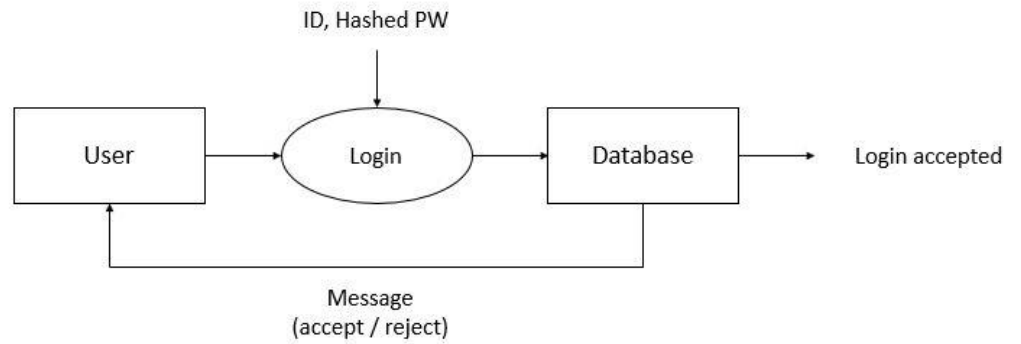
8. Purchase items

Use case	Purchase item
Actor	Seller, Buyer, Database(Item)
Description	사용자가 상품을 구매하기 위해 결제 페이지로 이동한다.
Trigger	구매자가 가능한 결제 수단 리스트 중 하나를 선택한다.
Success Response	해당하는 결제수단에 대한 페이지로 이동하고 결제를 수행할 수 있도록 한다. 구매자는 결제를 수행한다.
Failure Response	해당 결제를 사용할 수 없는 경우 오류 메시지를 출력하고 다른 결제수단을 사용하도록 한다.

<표 7.8 Purchase items>

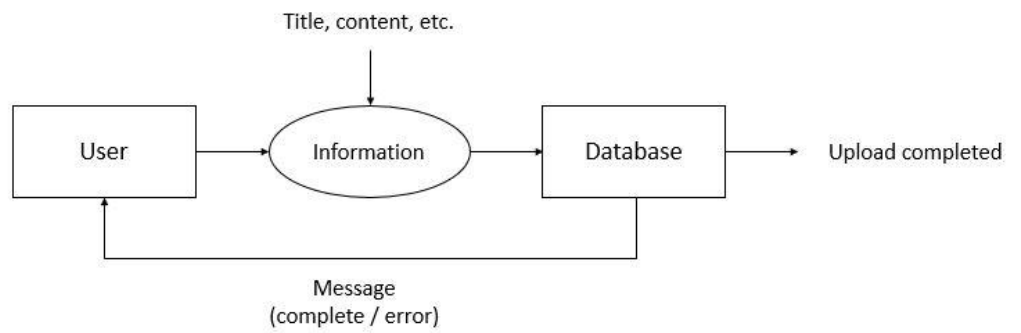
iii. Behavior Models

1. Login



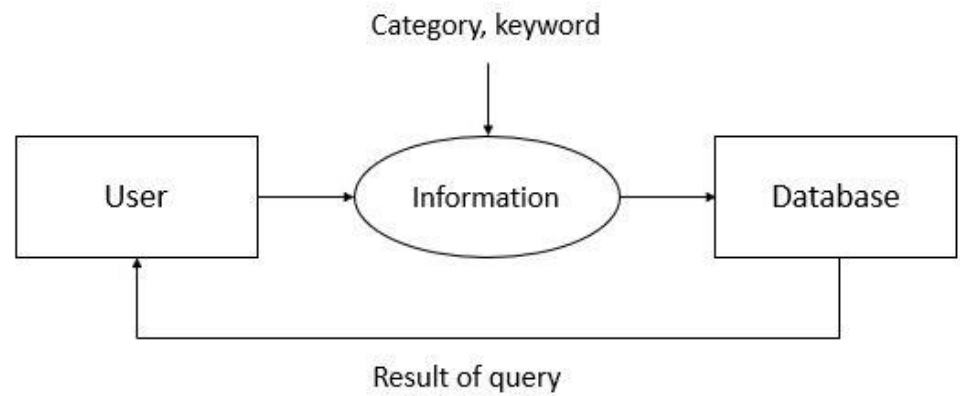
<그림 7.4 Login>

2. Upload product and posts



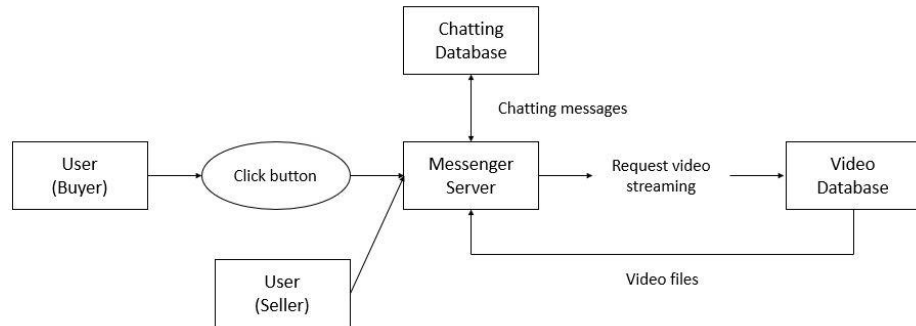
<그림 7.5 Upload product and posts>

3. Search



<그림 7.6 Search>

4. Chatting and video streaming



<그림 7.7 Chatting and video streaming>

8. System Evolution

A. Objective

System Evolution에서는 시스템이 기반하고 있는 주요 가정들과 제약사항들에 관해서 서술한다. 또한 개발 이후 일어날 것으로 보이는 Hardware, User requirement, Environment의 변화에 관해서도 간단히 서술한다. 이후에 생길 변화를 미리 서술해 두어 변화에 잘 적응할 수 있는 시스템을 만들 기 위해 서술되는 부분이기도 하다.

B. Limitation and Assumption

사람들의 중고거래에 관한 인식이 긍정적으로 변화하면서 중고거래를 시도하는 사람도 많이 늘어나고 있다. 단순히 가격이 저렴해서라는 이유보다는 특정 제품을 '소유'하는 것에 큰 가치를 두는 것이 아니라 '사용'한다는 것에 더 큰 가치를 두는 것도 중고거래의 규모가 커진 이유이기도 하다. 실제로 트렌드모니터에서 조사한 자료에 따르면 약 75.8%가 소유보다 사용 가치를 중시한다고 답했다. 그리고 앞으로 중고거래 규모는 더욱 커질 것으로 보인다. 이제는 단순히 중고거래가 '돈을 절약하기 위함'에만 그 목적이 있는 것이 아니라는 의미도 된다.

하지만 규모가 커지면 항상 그 이면에는 부작용이 따른다. 중고거래의 부작용은 앞서 계속 언급한 것처럼 사기 피해 또한 늘어났다는 것이다. 사람들이 점점 다양한 물품을 중고거래 시장에 내놓으면서, 사기 발생 건수, 사기피해 금액 모두가 이전에 비해 큰 폭으로 증가했다. 중고거래 규모가 커짐에 따라 중고거래와 관련한 스타트업 기업들이 생기고 있고 각 기업에서 사기 방지를 위한 다양한 서비스를 제공하려고 노력 중이다. 그럼에도 불구하고 사기를 완전히 방지하기 어렵다는 점이

중고거래 서비스의 한계점이다. 어떤 방식의 사기방지 서비스를 제공하더라도 완벽히 안전한 서비스를 제공할 수는 없고 이에 따른 사용자의 신뢰도 하락 또한 이 서비스가 갖는 한계점이다.

이를 해결하기 위해 여러 단계의 사기방지 시스템을 두어 완전히 사기를 방지할 수는 없더라도 일정량의 사기는 여러 단계의 시스템을 통과하면서 걸러질 수 있게끔 최대한 여러 겹의 방지 시스템을 구축하고자 하였다. 비디오 스트리밍, Indirect dealing system, Chatting system, Alerting a high risk of fraud product 모두가 각 단계에 해당한다. 더 신뢰도 높은 서비스를 구축하기 위해서 번거롭더라도 여러 단계를 더 추가해 사기 방지에 힘쓰는 것이 필요하다.

하지만 이 3단계 만으로도 일정량의 사기는 방지할 수 있을 것으로 기대되고 그에 따른 사람들의 중고거래 인식도 긍정적으로 변화할 것으로 기대된다. 서비스를 완성한 후에도 사기방지를 위해 여러 방안을 고안해 나갈 것이며 서비스 개선을 위해 꾸준히 노력할 것이다.

C. Evolution of User Requirement

i. 앱 어플리케이션 출시

현재 본 'SSH SHOP'서비스는 웹 기반 서비스를 제공한다. 하지만 요즘 모바일 앱 쇼핑 시장이 늘어남에 따라 많은 사람들이 앱 기반의 서비스 또한 필요로 할 것이다. 따라서 SSH SHOP 서비스를 앱과 연동하여 제공한다면 편리할 것으로 보인다. 또한 현재 제공하는 비디오 스트리밍 서비스 또한 앱으로 서비스를 확장 시킨다면 더 많은 사람들이 쉽게 접근할 수 있을 것이라고 예상한다.

ii. 통계를 기반으로 한 경고시스템

이 또한 사기 방지 시스템 중 하나로 어떤 품목에서 어느 정도의 금액대가 넘어가면 사기 위험성이 높다는 점을 통계자료로 이용해 사기 방지 시스템을 추가할 수 있다. 현재 단순히 사기 위험성이 높은 품목만 경고해주는 시스템에서 더 나아가 금액대, 판매자 등 여러 정보를 활용해서 사기 위험성을 판단 할 수 있는 시스템으로 확장할 수 있다.

iii. 판매자 평가 시스템

구매자가 물품을 주문하고 난 뒤 해당 상품이나 해당 판매자에 관해 평점을 부여하는 기능을 추가할 수 있다. 상품에 대한 후기, 판매자에 대한 후기가 제공된다면 구매자가 구입결정을 하기 전에 많은 도움을 받을 수 있고 이를 통해 판매자가 사기를 칠 가능성이 더 줄어든다. 또한 일정 점수 이하로 떨어진

판매자는 사기위험성이 높은 품목에 판매를 제한한다는 등 여러 사기방지 시스템도 마련할 수 있는 기반 정보로 쓸 수도 있다.

D. Evolution of Environment

i. 중고거래 품목의 다양화

중고거래 시장 규모가 커짐에 따라 기존에 거래하던 품목에서 다양한 상품을 추가해 서비스의 규모를 늘릴 수 있다. 단순히 쓰던 물품, 샀는데 환불 시기를 놓쳐서 다시 팔려고 하는 물품 이외에도, 쓰던 물품을 리폼해서 파는 경우, 혹은 본인이 직접 만든 물품들 까지도 중고거래의 범위에 포함시킬 수 있다.

9. Appendices

A. Objective

Appendices에서는 개발되는 시스템에 대한 더 자세하고 구체적인 정보들을 제공하고자 한다. 하드웨어, 데이터베이스 등을 설명하고 시스템 사용에 적합한, 혹은 최고 설정에 관해 설명한다.

B. Database Requirements

ii. User(판매자, 구매자)

사용자 정보를 저장하기 위한 DB entity

iii. Item

업로드 된 제품을 기억하기 위한 DB entity

iv. Board

신고 게시판 등 다양한 게시판에 올라온 정보를 저장할 DB entity

v. Chatting

구매자와 판매자 사이에 메신저 서비스를 통해 대화한 내용을 저장할 DB entity

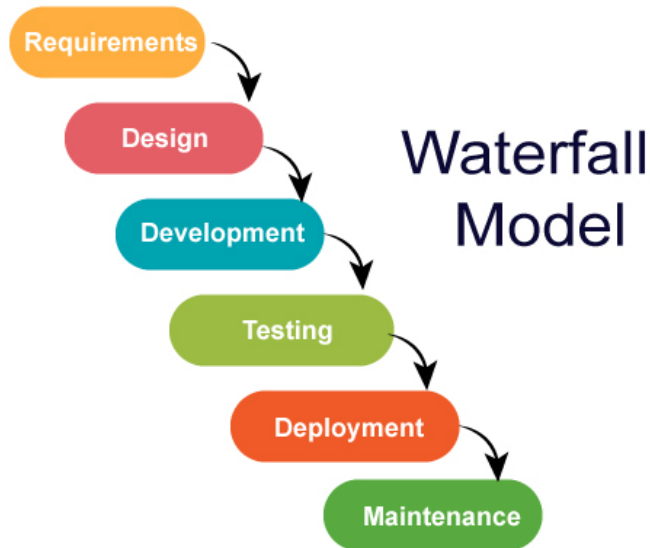
C. User-system Requirements

본 시스템은 웹 기반 서비스로 HTML5를 지원하는 웹 브라우저를 갖추어야 하며 항상 네트워크에 연결되어 있어야 한다.

데스크탑 브라우저	
최소 사양	HTML5를 지원하는 웹 브라우저
추천	구글 크롬을 이용하기를 추천한다.

<표 9.1 User-system Requirements>

D. Development Process



<그림 9.1 waterfall model>

본 조는 waterfall 모델을 사용할 개발을 진행할 예정이다. Waterfall model은 미리 계획이 짜여져 있기 때문에 parallel development 방법을 이용할 수 있다는 장점이 있다. 물론 모든 계획이 완벽하게 짜여져 있어야 한다는 점이 문제가 될 수도 있지만 변경사항이 발생한다면 최대한 빨리 변경사항을 분석하고 반영해 그에 따른 비용을 줄일예정이다. 따라서 11/10일까지 design specification을 완성하고 12/8일까지 개발 및 테스트를 완료할 계획이다.

10. Index

Index를 표시할 때는 <목차번호. 목차내에 사용된 자료 순서> 순서로 표시하였다.

A. 표 index

- i. 표 1.1 Version of the Document
- ii. 표 3.1 사용자 관련용어
- iii. 표 3.2 서비스 관련 용어
- iv. 표 3.3 개발 관련 용어
- v. 표 6.1 Login function
- vi. 표 6.2 My page display function
- vii. 표 6.3 Upload product function
- viii. 표 6.4 Product list function
- ix. 표 6.5 Board function
- x. 표 6.6 Search function
- xi. 표 6.7 Messenger function
- xii. 표 6.8 Streaming function
- xiii. 표 7.1 login
- xiv. 표 7.2 Register new items
- xv. 표 7.3 Write posts on board
- xvi. 표 7.4 View recommendation
- xvii. 표 7.5 View bookmark
- xviii. 표 7.6 Search
- xix. 표 7.7 Chatting
- xx. 표 7.8 Purchase items
- xxi. 표 9.1 User-system Requirements

B. 그림, Diagram index

- i. 그림 2.1 증가하고 있는 중고거래
- ii. 그림 2.2 증가하고 있는 사기 발생 건수
- iii. 그림 2.3 Alert a high risk of fraud product
- iv. 그림 4.1 Product Upload 창
- v. 그림 4.2 Messenger 창
- vi. 그림 5.1 System structure as a whole

- vii. 그림 5.2 Frontend architecture
- viii. 그림 5.3 Backend architecture
- ix. 그림 5.4 Messenger System
- x. 그림 5.5 User product dealing system
- xi. 그림 7.1 Context Diagram
- xii. 그림 7.2 Process Diagram
- xiii. 그림 7.3 Use case diagram
- xiv. 그림 7.4 Login
- xv. 그림 7.5 Upload product and posts
- xvi. 그림 7.6 Search
- xvii. 그림 7.7 Chatting and video streaming
- xviii. 그림 9.1 waterfall model