



Test Documentation

Student Number	Name
2015318385	김현우 Hyun Woo Kim
2013313737	김무성 Mu Sung Kim
2015310250	김동환 Dong Hwan Kim
2016315146	이상우 Sang Woo Lee

0. Table of Contents

1. Preface	5
1.1 Objective	5
1.2 Introduction	5
1.3 Unit Test	5
1.4 Component/sub-System Test	5
1.5 System Test	5
1.6 Release Test	6
1.7 Index	6
1.8 Reference	6
2. Introduction	7
2.1 Objective	7
2.2 System Architecture	7
2.3 Testing Plan	8
3. Unit Test	9
3.1 Objective	9
3.2 User Management System	9
3.2.1 회원가입	9
3.2.2 회원탈퇴	10
3.2.3 회원정보수정	11
3.2.4 로그인	12
3.2.5 로그아웃	12

3.2.6	샵 등록	13
3.2.7	검증자 등록	13
3.3	Product Management System.....	14
3.3.1	샵 조회.....	14
3.3.2	상품 등록.....	15
3.3.3	상품 삭제.....	17
3.3.4	상품 정보 수정	18
3.3.5	상품 검색.....	19
3.3.6	상품 조회.....	19
3.4	Transaction Management System	20
3.4.1	구매자 거래 요청.....	21
3.4.2	판매자 거래 승낙.....	22
3.4.3	거래 등록.....	22
3.4.4	거래 조회.....	22
3.5	Validation Management System.....	23
3.5.1	거래 검증.....	23
3.5.2	거래 기록 검색	24
3.5.3	거래 최종 결정	25
4.	Component/Sub-System Test.....	26
4.1	Objective	26
4.2	System Description	26
4.2.1	User Management System	26
4.2.2	Product Management System.....	26
4.2.3	Transaction Management System.....	26
4.2.4	Validation Management System.....	26
4.3	System Integration Test	27

4.3.1 Blockchain 과 User Management System 간의 interface.....	27
4.3.2 Blockchain 과 Transaction Management System 간의 interface	27
4.3.3 User Management System 과 Product Management System 간의 interface	28
4.3.4 User Management System 과 Transaction Management System 간의 interface	28
4.3.5 Product Management System 과 Transaction Management System 간의 interface	28
5. System Test.....	29
5.1 Objective	29
5.2 System Test Scenario	29
5.2 System Test Data	30
6. Release Test.....	36
6.1 Objective	36
6.2 시나리오와 요구사항 명세서 비교 - 기능	36
6.3 시나리오와 요구사항 명세서 비교 -비기능.....	37
7. Reference	38

1. Preface

1.1. Objective

본 테스트 문서의 전체 구조와 간략한 내용에 대한 서술을 서문에서 다룬다.

1.2. Introduction

Used2Block 시스템의 전체 구조와 테스트 계획에 대해서 서술한다.

1.3. Unit Test

Used2Block 시스템의 최소단위인 함수와 기능에 대한 테스트이다.

각 기능마다 테스트 케이스를 바탕으로 정상 동작과 결함을 설계한다.

1.4. Component and Sub-System Test

유닛 테스트 통과 이후 통합된 Component로서의 기능을 테스트한다.

기능을 통합한 서브시스템의 Interface 가 정상 작동하는지 파악한다.

1.5. System and Integration Test

컴포넌트 테스트 통과 이후 통합된 하나의 Used2Block 시스템이 정상 작동하기 위해서, 시스템 간의 연결, Interaction 을 테스트한다.

전체적인 유저 시나리오를 기반으로, 시스템 간의 연결을 확인한다.

1.6. Release Test

최종적으로 배포 바로 직전에 요구사항 명세서, 전체 유저 시나리오를 기반으로
종합 테스트한다.

안정적인 구매자, 판매자, 검증자의 상호작용이 필요하다.

1.7. Index

문서에 사용된 그림, 도표 등을 정리하여 명시한다.

1.8. Reference

문서에서 참조한 글 또는 그림을 정리하여 명시한다.

2. Introduction

2.1 Objective

테스트 문서 도입부에서는, Used2Block 시스템의 전체적인 구조 및 테스트 계획에 대한 전체적인 관점을 제공한다. 시스템 구조도를 통해서 어떤 모듈이 있으며 어떤 시스템 위계를 구성하는지 파악할 수 있다. 테스트 계획 도식을 통해서 전체적인 테스트 진행 방향을 알 수 있다.

2.2 System Architecture

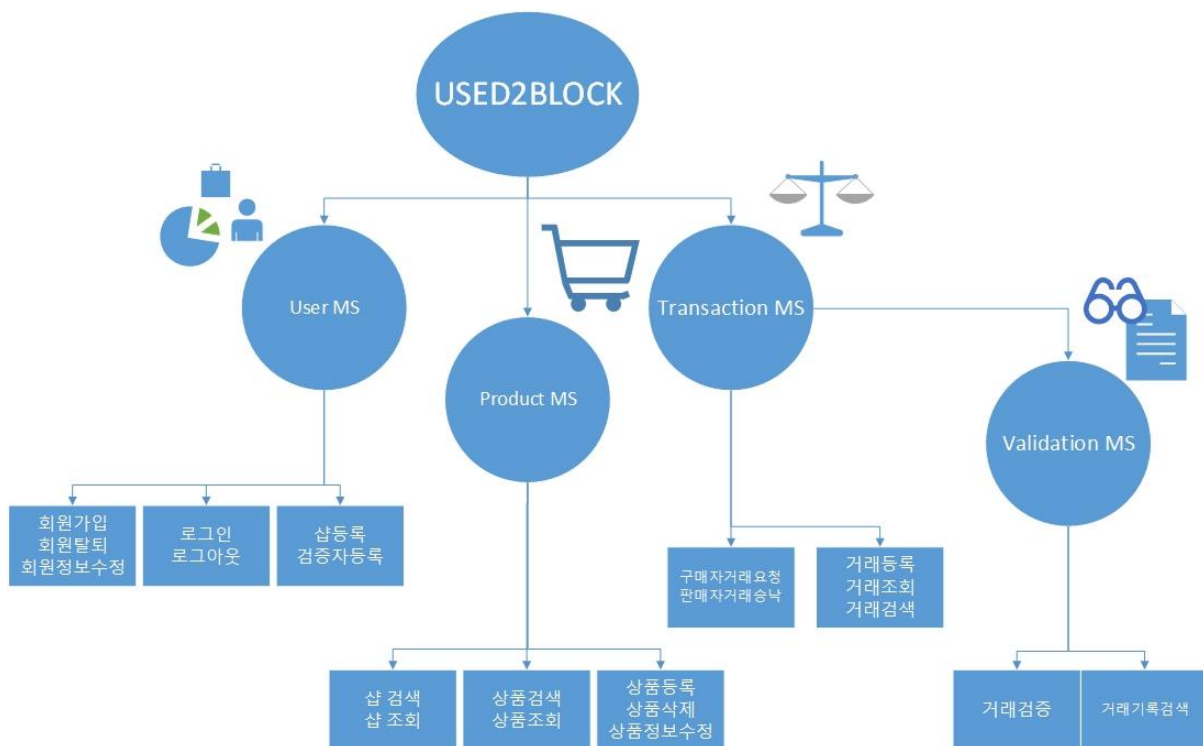


Figure 1. Used2Block System Architecture.

Object System인 Used2Block은 총 4개의 관리 system과 그것을 이루는 기능들로 이루어져 있다. 우선, User Management System은 회원의 정보를 저장하고 관리하는 역할을 한다.

Product Management System은 회원이 올린 상품에 대한 정보를 저장하고 관리하는 역할을 한다.

Transaction Management System은 거래 정보에 대한 전반적인 처리를 담당한다.

끝으로 Validation Management System은 Used2Block 시스템의 핵심 기능인 블록체인을 기반으로 한 판매자 정보 검증을 통한 고-신뢰 중고거래 성립을 가능하게 한다.

2.3 Testing Plan

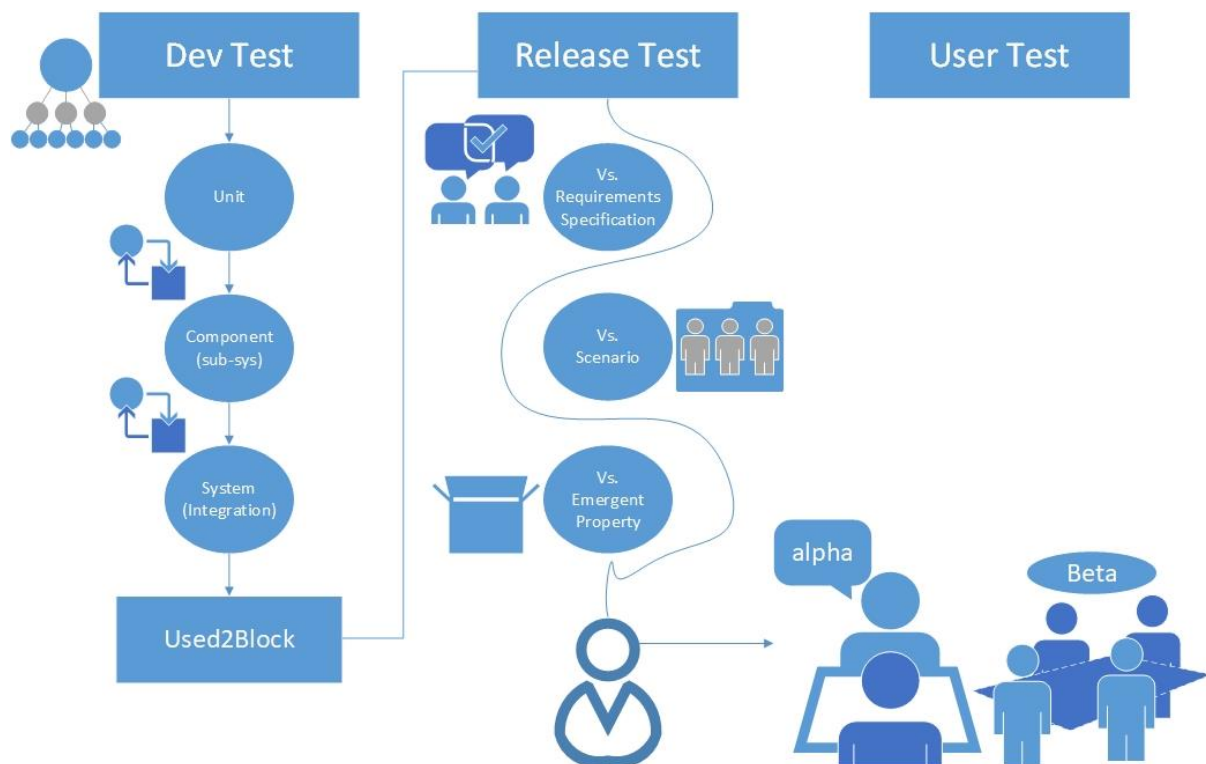


Figure 2. Used2Block overall Testing Plan.

테스트는 크게 Development Test, Release Test, User Test 총 3 개의 단계로 나뉜다. 먼저 개발 단계의 테스트는, Bottom-Up 방식을 통해서, 가장 작은 기능부터 전체 통합된 시스템에 이르기까지 순차적으로 진행한다. 각 세부 테스트에 관해서는 다음 장부터 다룬다. 이후 Release Test 단계에서는, 요구사항 명세서에 포함된 충족할 요구사항과 제약과의 비교, 디자인 문서에서 명시된 유저 시나리오와의 비교를 통해서 완성된 시스템이 구현 계획에 부합하는지 테스트한다.

이후, Used2Block 이 Generic Product 인 특징으로부터, Alpha, Beta test 에 대한 계획의 필요가 대두되는데, 이 User 단계 테스트에서는 Release Test 이후 책정된 예산과 시간제약에 기반해서 테스트를 진행한다. 개발팀의 손에서 테스트 전문가에게 테스트가 이관되어, 유저의 환경에서 Used2Block 시범 사용이 이루어져야 한다.

3. Unit Test

3.1 Objective

Unit Test에서는, Used2Block 시스템의 세부 기능과 모듈에 대한 테스트 계획을 서술한다. 앞서 시스템 구조도를 통해서 어떤 모듈이 있는지 파악할 수 있었다. 이를 바탕으로 각 Component를 구성하는 Unit들을 서버 시스템 별로 나누어 테스트한다. 각 기능은 테스트 케이스를 통과해야 하고, 발견가능한 잠재적인 defect들도 파악되고 수정되어야 한다.

3.2 User Management System

User Management System에서는 유저의 정보 저장 및 관리를 통제한다. 회원가입으로부터 로그인, 유저 상태 변경에 이르기까지 유저 관리에 관한 모든 기능이 있다.

3.2.1 회원가입

회원가입을 위해서 입력해야하는 데이터에는 이름, 이메일, 비밀번호, 전화번호가 있다.

이 중 이메일은 <xxx@xxx.xxx> 의 형태여야 하며, 비밀번호는 6자리 이상이어야 하고, 전화번호는 <xxx-xxxx-xxxx> 의 형태여야 한다.

또한 이메일은 중복되어서는 안된다.

위 사항들을 고려해서 정상적인 경우와 비정상적인 경우로 테스트 케이스를 나누어서 진행한다.

Input		Test type	Result
Email	test@test.com	Normal	새로운 계정이 만들어진다.
PW	123456		
phone	010-1234-5678		
Email		Abnormal	이메일을 입력해야 한다는 안내문구가 나온다.
PW		Abnormal	유효한 이메일을 입력하라는 안내문구가 나온다.
phone			

Email	test@test	Abnormal	
PW	123456		
phone			
Email	test@test.com	Abnormal	이미 존재하는 이메일이라는 안내문구가 나온다.
PW	123456		
phone	010-1234-5678		
Email	test@test.com	Abnormal	비밀번호를 입력해야 한다는 안내문구가 나온다.
PW			
phone			
Email	test2@test.com	Abnormal	비밀번호가 최소 6자리 이상 이어야 한다는 안내문구가 나온다.
PW	123		
phone	010-1234-5678		
Email	test2@test.com	Abnormal	전화번호를 입력해야 한다는 안내문구가 나온다.
PW	12345		
phone	010-1234-5678		
Email	test2@test.com	Abnormal	유효한 전화번호를 입력하라는 안내문구가 나온다.
PW	123456		
phone	010		
Email	test2@test.com	Abnormal	새로운 계정이 만들어진다.
PW	123456		
phone	010-1234		
Email	Test2@test.com	Normal	
PW	123456		
phone	010-1234-5678		

3.2.2 회원탈퇴

회원탈퇴는 프로필 페이지에서 회원탈퇴 버튼을 클릭해서 진행된다.

회원탈퇴 버튼을 클릭했을 때, 팝업 메시지 창이 나오면서 다시 한 번 계정을 삭제할 것인지 묻는다. 여기서 삭제하겠다고 할 시, 계정이 삭제되며 취소를 누르면 계정 삭제가 취소된다.

Input	Test type	Result
CONFIRM 클릭	Normal	계정이 삭제된다.
CANCEL 클릭	Normal	계정 삭제가 취소된다.

3.2.3 회원정보수정

회원정보수정은 프로필 페이지에서 수정 버튼을 클릭하여서 진행된다.

입력하는 창은 회원가입 할 때와 거의 똑같으며, 대신 여기서 입력값을 입력하지 않았을 경우에는 해당 입력값에 대해서는 수정이 이루어지지 않는다. 또한 여기서 market 에서 상품을 등록할 수 있는 판매자와 검증인으로서의 권한을 받을 수 있는데, 이 부분에 대해서는 각각 <샵 등록>과 <검증자 등록>에서 다룬다.

테스트 진행은 test2@test.com 계정으로 진행한다.

Input		Test type	Result
Email PW phone		Normal	회원 정보가 바뀌지 않는다.
Email PW phone	test@test.com	Abnormal	중복된 이메일이라는 안내문구가 나온다.
Email PW phone	test3	Abnormal	유효한 이메일을 입력하라는 안내문구가 나온다.
Email PW phone	test3@test	Abnormal	
Email PW phone	test3@test.com	Normal	이메일이 변경된다.
Email PW phone	123	Abnormal	비밀번호가 최소 6 자리 이상이어야 한다는 안내문구가 나온다
Email PW phone	12345	Abnormal	
Email PW phone	1234567	Normal	비밀번호가 변경된다.
Email PW phone	010	Abnormal	유효한 전화번호를 입력하라는 안내문구가 나온다
Email		Abnormal	

PW phone	010-5678		
Email PW phone	010-5678-1234	Normal	전화번호가 변경된다.

3.2.4 로그인

Used2Block 서비스를 이용하기 위해서는 회원가입 한 계정으로 로그인을 필요로 한다. 로그인 할 때는 이메일과 비밀번호를 입력해야 하며, 데이터베이스에 저장되어 있지 않은 사용자이거나 이메일과 비밀번호가 매치되지 않는 경우에는 로그인이 되지 않는다.

Input		Test type	Result
Email PW	test@test.com 123456	Normal	로그인에 성공한다.
Email PW	test10@test.com 123456	Abnormal	존재하지 않는 사용자라는 안내문구가 나온다
Email PW	test@test.com 12345	Abnormal	이메일과 비밀번호가 맞지 않는다는 안내문구가 나온다.

3.2.5 로그아웃

Used2Block의 사용자는 자신의 계정을 보호하기 위해 로그아웃 할 수 있다. 로그아웃 하기 위해서는 로그인이 되어 있는 상태여야 하며, 로그아웃 할 때에는 별도의 질문을 하지 않고 바로 로그아웃 된다.

Input		Test type	Result
로그아웃 버튼 클릭		Normal	로그아웃에 성공한다.

3.2.6 샵 등록

Used2Block의 사용자는 상품을 판매하기 위해서는 seller로서의 권한을 받고 샵을 등록해야 한다. Seller의 권한을 받기 위해서는 프로필 수정 페이지에서 seller account를 유효하게 바꿔야 한다. Seller account를 유효하게 바꾸었을 시, 상단 MY PROFILE 옆에 MY SHOPS 메뉴가 생기며, 샵 등록이 가능하게 된다. 샵 등록은 My SHOPS 메뉴로 들어가서 NEW SHOP 버튼을 클릭하여 진행한다. 입력값으로는 샵의 프로필 사진이 되는 이미지 파일, 샵 이름, 설명이 들어간다. 여기서 이미지 파일을 추가하지 않은 경우에는 default 이미지가 프로필 사진으로 사용된다. 설명에 대한 입력을 하지 않을 경우에는 샵이 만들어지지만, 샵에 대한 설명이 없는 채로 생성된다. 또한 샵의 이름은 중복될 수 있다.

Input		Test type	Result
이미지 이름 설명	Test	Normal	샵이 만들어진다
이미지 이름 설명	Test.jpg Test Test	Normal	샵이 만들어진다.
이미지 이름 설명	Test.jpg Test	Abnormal	이름을 입력하라는 안내문구가 나온다.

3.2.7 검증자 등록

Used2Block에서 거래가 이루어지기 위해서는 구매자와 판매자를 제외한 제 3자의 검증 과정이 필요하다. 검증 과정은 사용자에게 안전한 거래를 보증하기 위해 진행되며, 검증자가 거래인들의 과거 거래 내역을 확인하여 거래인이 믿을만한 사람이라는 것을 보이는 과정이다. 검증자 등록은 모든 사용자가 할 수는 있지만, 거래 검증을 위해서는 일정량 이상의 화폐를 가지고 있어야 한다. 이 부분에 대해서는 <거래 검증>부분에서 다룬다.

Input	Test type	Result
검증자 권한 On으로 변경	Normal	검증자로서의 권한이 생긴다.
검증자 권한 Off로 변경	Normal	검증자로서의 권한이 사라진다.

3.3 Product Management System

Unit Test에서는, Used2Block 시스템의 세부 기능과 모듈에 대한 테스트 계획을 서술한다. 앞서 시스템 구조도를 통해서 어떤 모듈이 있는지 파악할 수 있었다. 이를 바탕으로 각 Component를 구성하는 Unit들을 서브 시스템 별로 나누어 테스트한다. 각 기능은 테스트 케이스를 통과해야 하고, 발견가능한 잠재적인 defect들도 파악되고 수정되어야 한다.

3.3.1 샵 조회

모든 샵의 목록이나 <샵 검색>에서 찾은 샵의 목록에서 원하는 샵을 선택하여 해당 샵의 정보를 볼 수 있다. 샵의 정보는 이름, 프로필 사진, 설명, 판매하고 있는 상품들이 나타난다.

Input	Test type	Result
Test 이름의 샵 선택	Normal	Test 샵의 정보가 나타난다.

3.3.2 상품 등록

상품을 판매하고자 하는 사용자(이하 판매자)는 자신이 판매하고자 하는 상품을 등록하는 과정을 거쳐야 한다. 상품을 등록하기 위해서는 MY SHOPS 메뉴에서 자신이 상품을 등록하려는 샵의 정보 수정 페이지로 진입 후 NEW PRODUCT 를 눌러서 진행할 수 있다.

상품을 등록하기 위해서 입력해야 하는 정보로는 상품의 사진, 이름, 설명, 카테고리, 수량, 가격이 있다. 여기서 이름과 수량, 가격은 필수적으로 입력해야 하며 수량과 가격은 숫자로만 입력할 수 있다. 사진을 입력하지 않았을 시에는 default 이미지로 나타나게 되며, 설명과 카테고리를 입력하지 않은 경우에는 추후 상품 정보를 확인할 때 해당 정보가 나타나지 않게 된다.

하지만 카테고리 같은 경우에는 입력하는 것이 구매자로 하여금 접근성을 높이는 방법이기 때문에, 판매자가 상품을 판매하고 싶다면 적절한 카테고리를 입력하는 것이 바람직하다.

Input		Test type	Result
이미지 이름 설명 카테고리 수량 가격	Test 1 1	Normal	해당 상품이 등록된다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test 1 1	Normal	
이미지 이름 설명 카테고리 수량 가격	Test Test 1 1	Normal	
이미지 이름 설명 카테고리 수량 가격	Test Test 1 1	Normal	
이미지 이름 설명 카테고리 수량 가격	Test Test 1 1	Normal	

이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test 1 1	Abnormal	이름을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test 1	Abnormal	수량을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test Test 1	Abnormal	수량에는 숫자만 들어갈 수 있기 때문에 주어진 입력값을 입력할 수 없다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test 1	Abnormal	가격을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test 1 Test	Abnormal	가격에는 숫자만 들어갈 수 있기 때문에 주어진 입력값을 입력할 수 없다.

3.3.3 상품 삭제

판매자는 본인의 변심 등의 사유로 등록한 상품을 삭제할 수 있다. 상품 삭제는 MY SHOPS 메뉴에서 삭제하고자 하는 상품이 등록된 샵 정보 수정 페이지를 진입한 후 삭제할 수 있다. 상품 삭제를 선택할 시, 다시 한 번 상품을 삭제할 것이냐 묻는 팝업 메시지 창이 나타나고, 판매자는 여기서 삭제하겠다 라는 의사를 표현해야 상품 삭제에 성공한다.

Input	Test type	Result
CONFIRM 선택	Normal	상품 삭제에 성공한다.
CANCEL 선택	Abnormal	상품 삭제가 취소된다.

3.3.4 상품 정보 수정

판매자는 이미 등록한 상품에 대해 정보를 수정할 수 있다. 수정할 수 있는 정보는 상품 등록 시와 똑같이 상품에 대한 사진, 이름, 설명, 카테고리, 수량, 가격이 있다. Default 에 대한 값도 마찬가지로 사진, 설명, 카테고리에 대해서는 입력하지 않아도 된다. 사진을 등록하지 않을 시에는 default 이미지로 나타나게 된다.

이 테스트를 위해 선행조건으로 등록된 상품에 대한 정보는 다음과 같다.

이미지	Test.jpg
이름	Test
설명	Test
카테고리	Test
수량	1
가격	1

Input		Test type	Result
이미지	Test.jpg	Normal	상품의 이름이 Test2 로 변경된다.
이름	Test2		
설명	Test		
카테고리	Test		
수량	1		
가격	1		
이미지	Test2.jpg	Normal	상품의 사진이 Test2.jpg 에 해당하는 이미지로 변경된다.
이름	Test		
설명	Test		
카테고리	Test		
수량	1		
가격	1		

이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test2 Test 1 1	Normal	상품에 대한 설명이 Test2 로 변경된다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test2 1 1	Normal	상품의 카테고리가 Test2 로 변경된다
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test 1 1	Abnormal	상품의 이름을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test Test 1	Abnormal	상품의 수량을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test Test 1	Abnormal	상품의 가격을 입력하라는 안내문구가 나타난다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test T 1	Abnormal	수량에는 숫자만 들어갈 수 있기 때문에 주어진 입력값을 입력할 수 없다.
이미지 이름 설명 카테고리 수량 가격	Test.jpg Test Test Test 1 T	Abnormal	가격에는 숫자만 들어갈 수 있기 때문에 주어진 입력값을 입력할 수 없다.

3.3.5 상품 검색

어떠한 상품을 구매하고자 하는 사용자(이하 구매자)는 메인 페이지에서 자신이 구매하고자 하는 상품을 검색할 수 있다. 상품 검색에 대한 조건은 카테고리나 상품의 이름이 있다. 카테고리에 대한 입력은 모든 카테고리와, 각 상품들의 카테고리를 묶은 것들 중에서 선택 할 수 있다. 상품의 이름은 등록된 상품의 이름 중 일부만 입력해도 그 일부가 이름에 들어간 상품이 모두 결과로 출력된다. 하지만 상품 이름 입력란에 아무것도 기입하지 않은 경우에는 검색 기능이 실행되지 않는다.

이 테스트에서의 선행 조건으로 등록된 상품들은 다음과 같다.

Category	Name
Test1	T1
Test2	T1
Test2	T2

Input		Test type	Result
Category	All	Normal	Test1 카테고리의 T1 과 Test2 카테고리의 T1 이 나타난다.
Name	T1		
Category	All	Abnormal	어떤 상품도 나타나지 않는다
Category	Test1	Normal	Test1 카테고리의 T1 이 나타난다.
Name	T1		
Category	Test1	Abnormal	해당 상품이 없다는 안내문구가 나타난다.
name	T2		
Category	Test2	Normal	Test2 카테고리의 T2 가 나타난다.
Name	T2		

3.3.6 상품 조회

구매자는 자신이 구매하고자 하는 상품을 선택하여 상품에 대한 정보를 확인할 수 있다. 상품에 대한 정보로 판매자가 상품을 등록할 때 입력한 이름, 설명, 카테고리, 가격, 상품의 사진을 확인할 수 있다.

Input	Test type	Result
상품 선택	Normal	상품에 대한 정보가 나타난다.

3.4 Transaction Management System

User Management System에서는 유저의 정보 저장 및 관리를 통제한다. 회원가입으로부터 로그인, 유저 상태 변경에 이르기까지 유저 관리에 관한 모든 기능이 있다.

3.4.1 구매자 거래 요청

구매자는 구매하고자 하는 상품을 선택한 후 판매자에게 거래 요청을 해야 한다.

거래 요청을 할 때는 이름, 이메일, 주소(도로명 주소, 시, 도, ZIP code, 국가)를 입력해야 한다.

Input		Test type	result
Name Email Street City State ZIP code Country		Abnormal	거래에 실패했다는 안내 메시지가 나타난다.
Name Email Street City State ZIP code Country	test test@test.com	Abnormal	
Name Email Street City State ZIP code Country	test test@test.com test test test test test	Normal	주문이 완료되었다는 것을 알리는 화면이 나타나며, 검증 내역에 따라 거래를 완료/취소/신고 할 수 있는 기능이 나타난다.

3.4.2 판매자 거래 승낙

판매자는 구매자가 자신이 등록한 상품에 대한 거래 요청을 했을 때, 해당 요청을 받아들여 거래를 진행할 수 있다. 거래 승낙 시 해당 거래에 대한 정보가 자동으로 등록되고, 검증자가 거래 참가자들에 대한 검증을 진행할 수 있게 된다.

거래 요청은 MY SHOPS 메뉴에서 해당 거래가 일어난 샵의 VIEW ORDERS 메뉴를 선택하여 확인할 수 있다.

Order 의 상태는 (Not processed / Processing / Shipped / Delivered / Cancelled) 가 있으며, 판매자는 각각의 상태를 업데이트하여 현재 거래가 어떤 상태인지 알릴 수 있다.

Input	Test type	result
Not processed	Normal	거래의 상태가 Not processed 임을 알림
Processing	Normal	거래의 상태가 Processing 임을 알림
Shipped	Normal	거래의 상태가 Shipped 임을 알림
Delivered	Normal	거래의 상태가 Delivered 임을 알림
Cancelled	normal	거래의 상태가 Cancelled 임을 알림

3.4.3 거래 등록

거래 등록은 판매자가 거래 요청을 승낙한 경우에 자동으로 진행되며, 검증자들은 등록된 거래들을 보고 검증 과정을 진행할 수 있다. 판매자는 등록된 거래들의 현재 상황을 확인할 수 있다.

Input	Test type	Result
구매자가 거래 요청	Normal	Validation 목록과 거래에 참여한 사용자들의 ORDERS 에 해당 거래가 게시된다.

구매자와 판매자는 거래 대상인 상품에 대한 정보, 총 가격, 거래가 시작된 시간, 구매자의 이름과 이메일, 구매자가 상품을 수령할 주소와 검증 진행 상태를 확인할 수 있다.

Validation 리스트에서는 거래가 시작된 시간과 판매자의 이메일, 판매자의 가상 화폐 계좌, 전화번호를 확인할 수 있다.

3.4.4 거래 조회

판매자는 등록된 거래들 중 확인하고자 하는 거래에 대한 정보를 확인할 수 있다.

Input	Test type	Result
거래 선택	Normal	거래에 대한 정보가 나타난다.

3.5 Validation Management System

Validation Management System에서는 거래 검증의 전반적인 기능을 통제, 관리한다. 거래 관리 시스템에서 거래가 시작된 상품이 Order로 등록되고, 검증 리스트에 출력된다. 출력된 거래를 검증자로 등록한 유저들이 검증하고 보고하는 기능까지 구현된다.

3.5.1 거래 검증

거래는 검증자의 검증이 이뤄진 후에 거래 참가자들이 거래를 진행할지 말지를 선택한 후 이뤄진다. 여기서 검증은 거래 참가자들로 하여금 상대의 신뢰성에 대한 확인을 거치는 단계로 거래 참가자들이 검증 결과를 보고 거래를 진행할지 파기할지를 선택할 수 있다.

거래에 대한 검증은 검증자가, 판매자의 이메일/가상화폐계좌/전화번호를 확인하여 과거의 사기 내역이 있었는가를 검증하는 과정이다.

확인된 사기 내역에 대해서 사기를 쳤던 사이트의 이름과 주소를 입력하여 블록체인에 업데이트 시킬 수 있다.

Input		Test type	Result
Name of page	Test	Normal	Test 라는 사이트의 Test 라는 페이지에 확인할 수 있다는 내역이 업데이트되고, google.com 으로 링크된다.
Name of site	Test		
Access time	191111		
URL	Google.com		

3.5.2 거래 기록 검색

검증자들은 거래에 대한 검증을 위해 거래 참가자들의 거래 기록을 검색하여 그들의 신뢰도를 판단할 수 있다. 본 시스템에서는 사용자가 편하게 접근할 수 있도록 번개장터의 조회 시스템을 가져왔으며, 입력할 수 있는 값에는 계좌, 휴대폰, 카카오톡이 있다.

Input	Test type	Result
계좌	Normal	해당 계좌에 대한 사기 내역이 확인되지 않은 경우 정상 사용자라고 알리고, 사기 신고가 된 경우에는 '차단 내역 있음'이라는 메시지를 나타낸다.
휴대폰 번호	Normal	해당 휴대폰 번호에 대한 사기 내역이 확인되지 않은 경우 정상 사용자라고 알리고, 사기 신고가 된 경우에는 '차단 내역 있음'이라는 메시지를 나타낸다.
카카오톡 아이디	Normal	해당 카카오톡 아이디에 대한 사기 내역이 확인되지 않은 경우 정상 사용자라고 알리고, 사기 신고가 된 경우에는 '차단 내역 있음'이라는 메시지를 나타낸다.

3.5.3 거래 최종 결정

검증자들의 검증이 끝나고, 거래 참가자들은 상대에 대한 검증 결과를 확인 후 거래를 진행할지 파기할 지에 대한 최종 결정 단계를 가진다.

Input	Test type	Result
COMPLETE TRANSACTION	Normal	거래가 완료되고, 거래에 필요한 금액만큼의 가상화폐가 판매자에게 송금된다.
REPORT TRANSACTION	Normal	판매자에 대한 신고 기록이 블록체인에 저장된다.
REVERT VALIDATION	Normal	거래가 취소된다.

4. Component/sub-System Test

4.1 Objective

Component Test에서는, Used2Block 시스템의 세부 기능과 모듈을 통합하고 있는 Component 들을 테스트한다. 앞서 Module/Unit Test를 통해서 어떤 모듈이 있는지 파악할 수 있었다. 또한 시스템 구조도를 통해서 어떤 sub-system단위가 있는지 파악할 수 있었다. 이를 바탕으로 각 Component의 Interface를 테스트한다. Component test 이전, 각 기능은 unit test 케이스를 통과해야 하고, 이후에 component interface를 test plan에 의거해서 검증한다.

4.2 System Description

4.2.1 User Management System

User Management System에서는 유저의 정보 저장 및 관리를 통제한다. 회원가입으로부터 로그인, 유저 상태 변경에 이르기까지 유저 관리에 관한 모든 기능이 있다.

4.2.2 Product Management System

Product Management System에서는 상품의 정보 저장 및 관리를 통제한다. Shop등록, 상품 추가, 상품 수정, 상품 삭제 검색 등 유저가 관리하는 유저의 판매대(shop)과 상품(product)에 관한 관리기능이 테스트의 대상이다.

4.2.3 Transaction Management System

Transaction Management System에서는 거래 정보 저장 및 관리를 통제한다. 등록된 상품이 거래 버튼을 통해서 거래의 대상이 되고, 거래가 시작되면서 생기는 기능과 정보 처리를 테스트한다.

4.2.4 Validation Management System

Validation Management System에서는 제3자의 거래 검증 기능을 관리한다. 거래를 찾고, 검증하고, 그 결과가 반영되는 절차를 테스트한다.

4.3 System Integration Test

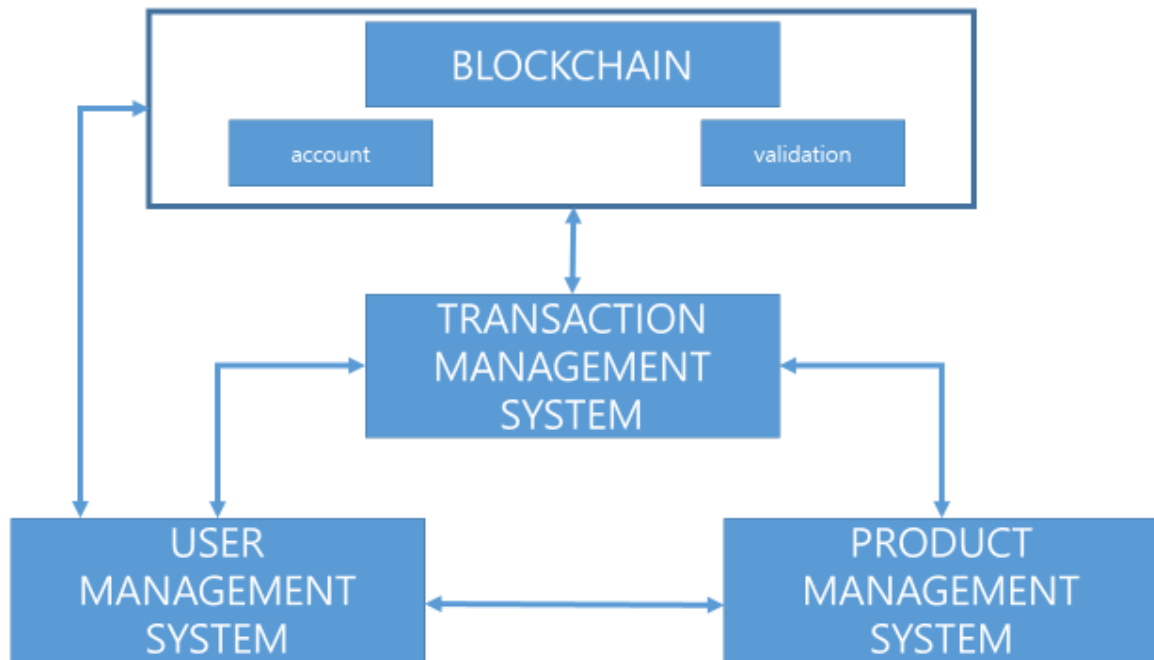


Figure 3. Used2Blcok overall Structure.

위 그림에서 User Management System에는 사용자에 관한 정보가, Product Management System에는 등록된 상품에 관한 정보, Transaction Management System에는 진행중인 거래에 관한 정보가 저장된다.

Blockchain 상에는 Used2Block에서 사용되는 가상 화폐가 유통되고, 거래를 진행하면서 작성한 Smart Contract들이 저장되어 후에 사용자에게 대한 검증을 하는데 쓰인다.

4.3.1 Blockchain과 User Management System간의 interaction

Used2Block에서의 거래는 블록체인 기반의 가상 화폐를 통해 이루어진다. 그렇기 때문에 각 사용자마다 가상 화폐 계좌가 있다.

각 사용자 별로 가상 화폐 계좌가 생성되는지 확인한다.

4.3.2 Blockchain과 Transaction Management System간의 interaction

사용자(이하 A)의 거래 기록을 블록체인 상에 저장한다. 이 거래 기록은 나중에 A가 거래를 시도할 때, 검증인으로 참여하는 사용자가 A의 거래 기록을 확인하여 A가 신뢰할 수 있는 판매자인지 확인하는데 쓰인다.

또한, 거래가 완료되면 Blockchain 에 거래된 물품의 값에 해당하는 금액만큼의 이동을 요청하게 된다.

거래를 실시하여, 검증자로 거래에 참가했을 때 거래 기록을 확인할 수 있는지 확인한다.

거래가 종료되었을 때, 거래에 사용될 만큼의 가상 화폐가 이동되었는지 확인한다.

4.3.3 User Management System과 Product Management System간의 interaction

사용자가 Used2Block에 상품을 등록할 때, 어떤 사용자가 상품을 등록했는지 명시되어야 하기 때문에 User Management System에서 사용자 이름에 해당하는 값을 가져오게 된다.

상품을 등록했을 때, 등록자의 이름이 등록한 사용자의 이름으로 명시되는지 확인한다.

4.3.4 User Management System과 Transaction Management System간의 interaction

거래가 이루어질 때, 어떤 사용자가 거래에 참여하는지 확인되어야 한다. 이는 구매자와 검증자로 하여금 판매자가 어떤 사람인지 확인할 수 있도록 도와주며, 판매자가 후에 물건을 구매자에게 보낼 때, 어디로 보내는 지도 확인할 수 있게 해준다.

진행중인 거래를 확인하여, 구매자와 판매자가 정상적으로 나타나는지 확인한다.

4.3.5 Product Management System과 Transaction Management System간의 interaction

Used2Block이 거래를 위한 플랫폼이기 때문에, 플랫폼에 등록되는 물품들을 관리하는 Product Management System과 Transaction Management System 간의 interface 또한 존재한다. 거래가 이루어질 때는 거래를 하고자 하는 상품이 정상적으로 등록되어야 한다.

어떠한 상품에 대한 거래를 할 때, 해당 상품이 정상적으로 Transaction Management System에서 사용되는지 확인한다.

5. System Test

5.1 Objective

System Test에서는, Used2Block 시스템의 Component들의 통합과 그 Component들 간의 상호작용을 테스트한다. 앞서 Unit test, Component test를 모두 성공적으로 마친 상태를 전제로 하며, 그래야 완성된 Component, sub-System들의 상호작용이 테스트 가능하다. 특정 기능과 Interface에 대해서 상호작용해야 하는 시스템들 간의 충돌은 없는지, 통합된 시스템 전체가 정상적으로 작동하는 지를 주안점으로 시나리오, 테스트 케이스와 계획을 구성한다.

5.2 System Test Scenario

핵심 테스트 시나리오는 간단하다. 회원가입을 거친 구매자 Yes와 판매자 Poo 그리고 검증에 참여하는 제3자 No가 존재하고, 그 3 Actor들이 중고상품 또는 P2P 재화의 안전 거래를 구성한다. 그 거래 시나리오를 순서대로 서술하면 다음과 같다.

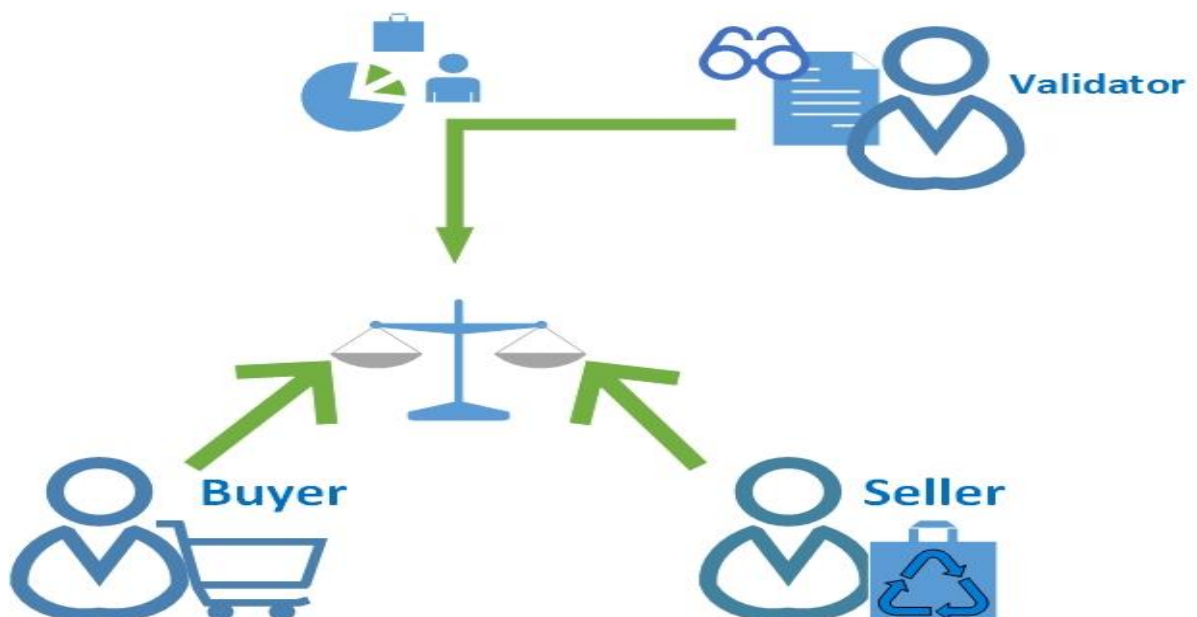
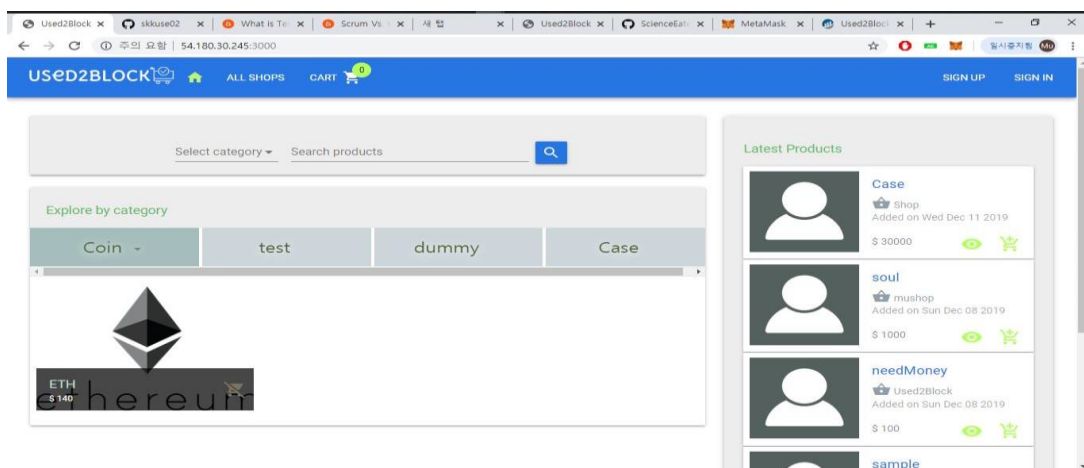


Figure 4. Used2Block Scenario.

1) 판매자 Poo는 상품을 올린다 -> 2) 구매자 Yes는 검색을 통해서 원하는 상품을 확인한다 -> 3) 구매자는 카트에 담고, 물건을 구매한다 -> 4) 판매자는 거래를 확인하고, 거래는 검증을 위한 리스트로 간다 -> 5) '검증자' No는 검증을 기다리고 있는 리스트를 확인한다 -> 6) 검증을 위해서 거래의 거래 정보를 블록체인에 검색 또는 다른 사이트의 사기 내역을 조사, 오점이 발견되면 보고한다 -> 7) 부적절한 거래 내역을 보고 받으면, 구매자는 검증자가 확인해준 거래 기록을 통해서 해당 거래를 취소함으로써, 가능한 사기 거래를 피한다.

물론,中间的 빠른 거래를 성사시키기 위해서 검증 단계 '(6)'을 거치지 않고 구매자는 바로 거래를 승낙할 수 있다. 그리고 구매자와 판매자가 생각하는 적절한 시간 안에 '(7)'의 사기 검증이 생기지 않으면, 거래를 승낙하게 된다.

5.3 Scenario Test Data



<Main Page>

0) 회원가입(생략) 후 로그인.

poo@gmail.com / poopoo

& yes@gmail.com / yesyes

& no@gmail.com / nonono

Sign In

Email


Password


SUBMIT

<Log-In Page>

1) 판매자 Poo는 프로필 페이지에서 판매자(seller)로 신분을 추가, 본인의 매장을 개설(Shop)한 후 상품을 올린다. 여기서는 hope라는 이름의 물건을, 아래와 같이 입력한다.

Edit Product



CHANGE IMAGE 

Name

hope

Description

hope

Category

test

Quantity

2

Price

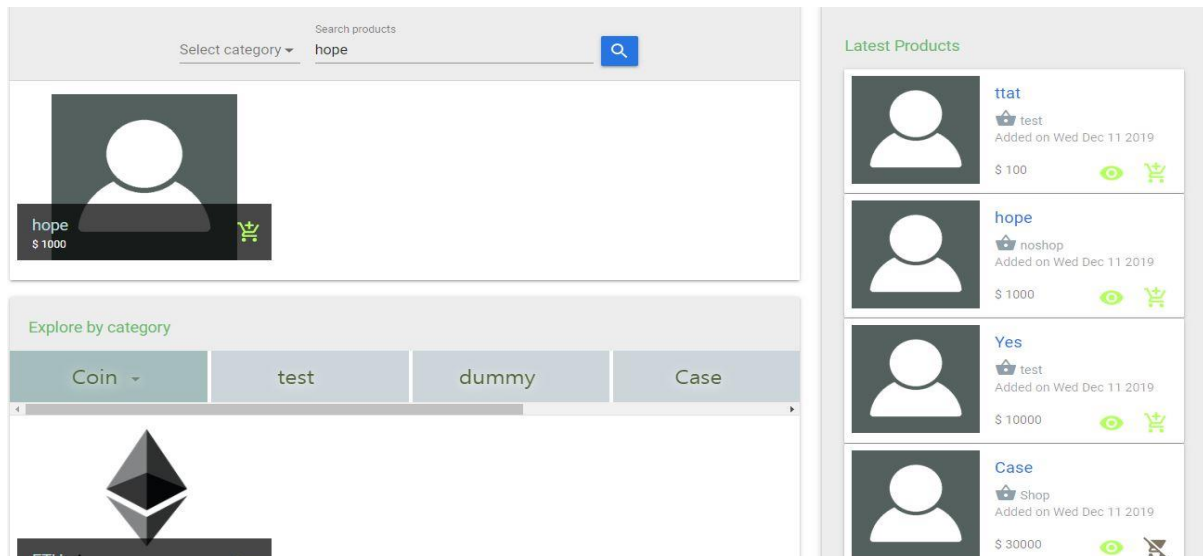
1000

UPDATE

CANCEL

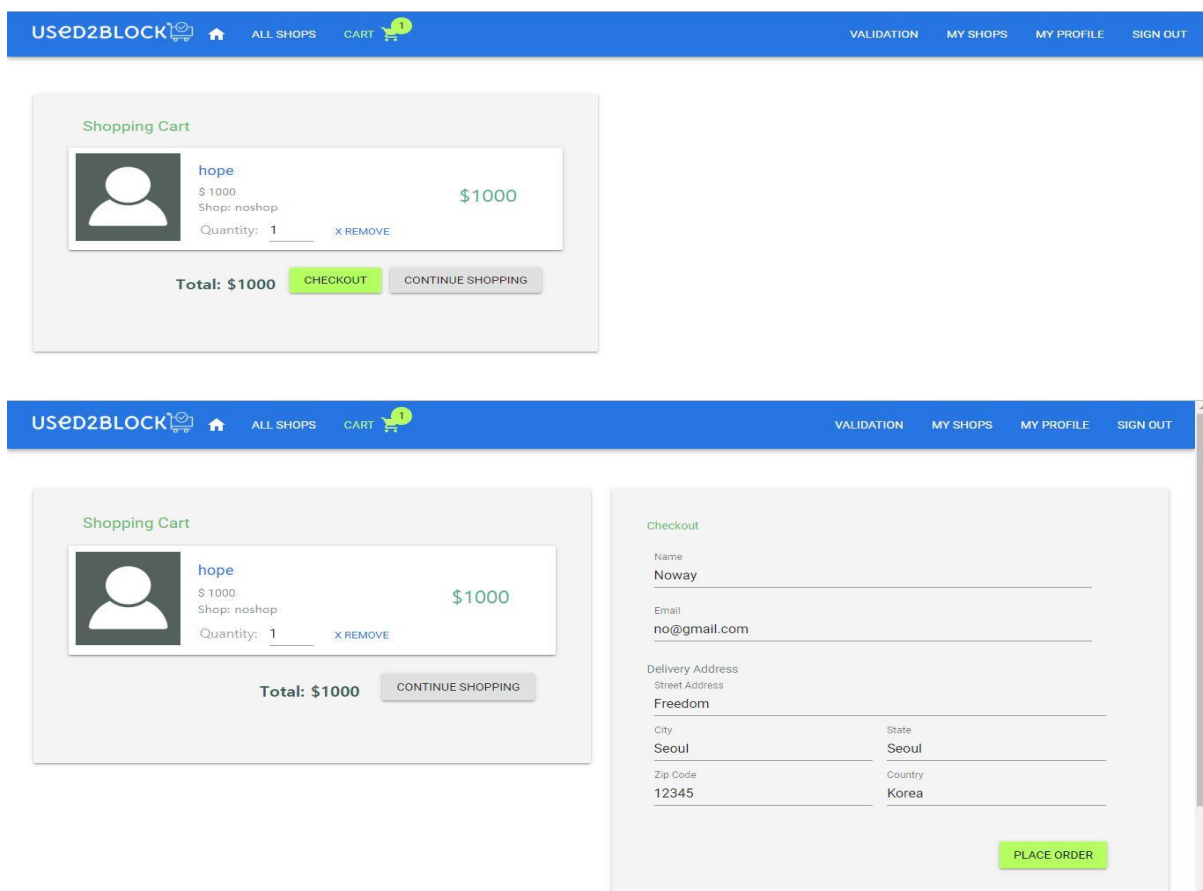
<Product Enrollment>

2) 구매자 Yes는 검색을 통해서 원하는 상품을 확인한다



<Search Product>

3) 구매자는 카트에 담고, 물건을 구매한다. 순서와 입력 데이터는 아래와 같다.



<장바구니에 담긴 물품과 거래 신청>

4) 판매자는 거래를 확인하고, 거래는 검증을 위한 리스트로 간다

The image displays two screenshots related to a transaction on the Used2Block platform.

Top Screenshot: Used2Block Order Confirmation

The header shows the site name "USED2BLOCK" with navigation links: ALL SHOPS, CART (0 items), VALIDATION, MY SHOPS, MY PROFILE, and SIGN OUT.

Order Details:

- Order Code: 5df0f18d4b68aa1163f85a4d
- Placed on: Wed Dec 11 2019
- Status: Pending
- Link To Blockchain Network

Item Details:

- Item: hope
- Price: \$ 1000 x 1
- Shop: noshop
- Status: Not processed
- Total: \$1000

Delivery Information:

- Deliver to: Ted
- Email: yes@gmail.com
- Address: Freedom, Seoul, Seoul 12345, Korea
- Message: Thank you for shopping with us! You can track the status of your purchased items on this page.

Bottom Screenshot: Etherscan Transaction Details

The browser address bar shows the URL: ropsten.etherscan.io/tx/0xcc1a2dfa390cdb48ba0f5fe1e9e72a3ce392e8affee3fd74ebdce7f1169098ba.

Etherscan Transaction Details:

- Transaction Hash: 0xcc1a2dfa390cdb48ba0f5fe1e9e72a3ce392e8affee3fd74ebdce7f1169098ba
- Status: Success
- Block: 6949239 (1 Block Confirmation)
- Timestamp: 21 secs ago (Dec-11-2019 01:39:36 PM +UTC)
- From: 0x37f6bbe96749fea346f97ceba6c64c71b012e938
- To: Contract 0xd7d5167f8c356a1217f1019942f18d9c75572a02
- Value: 0 Ether (\$0.00)
- Transaction Fee: 0.0202080000 Ether (\$0.000000)

A footer notice states: "This website uses cookies to improve your experience and has an updated Privacy Policy. Got It"

<Used2Block 사이트와, 이더리움에 기록된 거래>

5) '검증자' No는 검증을 기다리고 있는 리스트를 확인한다.

Transaction ID # 0x501316B3F90D3af6F9c0362B4b36c4595183393 11 Dec 2019 15:15:26	▼
Transaction ID # 0x36b59184F2204d90a0F94675b37402A744cF1bd5 11 Dec 2019 19:28:35	▼
Transaction ID # 0x8659e2b4131E77F87A9f1ca6dfAaC2229335B258 11 Dec 2019 15:34:41	▼
Transaction ID # 0x7B85DdEe1fc22c05264f4B46E0338d69E2178D37 11 Dec 2019 19:44:6	▼
Transaction ID # 0x841ebD8B99004C751717e18791Cb663eDA0373C2 8 Dec 2019 22:53:28	▼

Open Transactions 1

Transaction ID # 0xcBD246AD882E4912d46b0C028EC4CD31b728f0D8
11 Dec 2019 22:39:36

Seller ID: no@gmail.com
Account: 0xF57BD1b3F7814e228Ad3EAE5F73E3A125A7274A1
Phone: 010-0200-0121

Name Of Page

Name Of Site

Access Time

URL

REPORT

계좌 조회

의심스러운 계좌를 조회해보세요.

반개장터의 등록된 사기계좌정보와 디지털, 경찰청의 계좌정보를 함께 조회할 수 있습니다.

휴대폰 조회

의심스러운 번호를 조회해보세요.

반개장터의 등록된 사기계좌정보와 디지털, 경찰청의 계좌정보를 함께 조회할 수 있습니다.

카카오톡 조회

의심스러운 카카오톡 ID를 조회해보세요.

반개장터의 등록된 사기정보를 이용하여 조회할 수 있습니다.

<Validation List and Report Form>

6) 검증을 위해서 거래의 거래 정보를 블록체인에 검색 또는 다른 사이트의 사기 내역을 조사, 오점이 발견되면 보고한다

Open Transactions 1

Transaction ID # 0xcBD246AD882E4912d46b0C028EC4CD31b728f0D8
11 Dec 2019 22:39:36

Seller ID: no@gmail.com
Account: 0xF578D1b3F7814e228Ad3EAE5F73E3A125A7274A1
Phone: 010-0200-0121

Name Of Page
skku main

Name Of Site
skku

Access Time
20191208

URL
www.skku.edu

REPORT

계좌 조회

의심스러운 계좌를 조회해보세요.

변개장터의 누적된 사기계좌정보와 디지털, 경찰청의 계좌정보를 함께 조회할 수 있습니다.

조회

휴대폰 조회

의심스러운 번호를 조회해보세요.

변개장터의 누적된 사기계좌정보와 디지털, 경찰청의 계좌정보를 함께 조회할 수 있습니다.

조회

카카오톡 조회

의심스러운 카카오톡 ID를 조회해보세요.

변개장터의 누적된 사기정보를 이용하여 조회할 수 있습니다.

조회

<가정: skku.edu에서 보고된 사기를 찾은 경우>

7) 부적절한 거래 내역을 보고 받으면, 구매자는 검증자가 확인해준 거래 기록을 확인하고, 이를 통해서 해당 거래를 취소함으로써, 가능한 사기 거래를 피한다.

Validation Report

Status: Waiting
Reported Time:
Name of Page:
Name of Site:
Accessed Time:
[Link To Resource](#)

COMPLETE TRANSACTION

REPORT TRANSACTION

REVERT VALIDATION

<구매자의 선택 기능>

최종적으로 링크를 확인해서 거래 기록을 확인하거나, 검증이 원하는 시간만큼 충분히 되었다고 생각하면, 거래를 종료한다.

6. Release Test

6.1 Objective

Release Test에서는, 가장 먼저 Requirement Specification을 바탕으로 해당 충족사항을 만족하는지 파악한다. 동시에 Emergent Property를 파악하여 그것을 충분히 이행하고 있는지 파악한다. 동시에, 디자인 문서를 참조하여 유저 시나리오에 따라서 정상적으로 혹은 비정상 적으로 시스템이 제 기능을 수행하는지 확인한다. 테스트 데이터는 시나리오를 그대로 쓰며, 유저의 환경에서 유저가 원하는 거래를 원하는 만큼 거래하면 된다.

6.2 시나리오와 요구사항 명세서와의 비교 - 기능

먼저, 요구사항 명세서의 기능적 요구사항이 충족되었는지 확인한다.

문서에 따르면, 기능적 요구사항은 크게 User, Product, Transaction 관리 기능으로 나뉘는데,

Register, Log-in/out, Profile Management, Wallet으로 유저 관리 기능을 요구한다.

Searching Product, Selling Item, Manage Item, Item Page 등으로 상품 관리 기능을 요구한다.

Validation, Buying Item, Verifying Transaction 등으로 거래/검증 기능을 요구한다.

위 모든 기능들은 Unit, Component test에서 각각 통과되었으며, System Scenario test에서 Normal Case를 통해서 작동함을 검증했다. 각 기능을 추가적으로 테스트하기 위해서, 앞서 만들어 둔 계정을 통해서 원하는 어떤 거래의 경우라도 안정적으로 시도할 수 있다.

6.3 시나리오와 요구사항 명세서와의 비교 - 비기능적 특성

다음으로, 요구사항 명세서의 비기능적 요구들이 충족되었는지 확인한다.

비 기능적 요구사항은 첫째로 Product Requirement, Organization Requirement, External

Requirement가 있었는데, 각각 우리가 제작하는 서비스/제품의 요구, 조직적인 요구사항, 외부적인 요구사항 등이 있다.

먼저 Used2Block 제품 요구사항은, 보안, 안정성, 속도, 이용용이성 등이 있다.

먼저 보안에 대한 요구사항은 가상화폐의 private key가 시스템 내부에서 hash되어 저장되는 기본적인 보안성 만이 아니라 거래에 대한 정보 역시 블록체인에는 꼭 필요한 데이터만 hash되어 기록되게 구현되었다.

서비스 안정성 역시, smart contract 상에서 계정 생성시 미리 주입된 초기 화폐 값만 충족하면, 거래를 진행하는 데에 있어서 어려움이 없다. 또한 블록체인 시스템과 다른 데이터 베이스는 안정적으로 연동되어 이용자는 끊임 없이 사용 가능하다.

속도 측면에 있어서는 블록체인의 검색을 이용하기에 다른 데이터베이스 거래보다 조금 느리다고 할 수 있지만, pending에서 success라고 결과를 처리하기까지 걸리는 시간 역시 여러 테스트 결과 5~10초 내외였기에, 기본적으로 충족됐다고 생각된다. 검증 과정이 길어지는 문제는 차후 많은 검증자들을 수용하고 그들이 활동함으로써 채워질 수 있다.

사용하기에는 어느정도 기호에 대한 해석이 가능하면 모두 각자의 역할에 맞게 사용 가능하다. 사용 브라우저가 크롬, 언어가 영어라는 두 제약 빼고는 직관적으로 그림버튼을 통한 해당 기능의 작동이 가능하다. UX에 대한 경우는, 추후 평가가 더 필요하다.

Organizational Requirement에 있어서는, 개발이 정해진 시간에 맞춰 Working System을 만들어 낼 수 있었고, 개발과 관리자의 스택을 MERN으로 변경해서 능력과 배경에 맞는 개발이 이루어졌다.

External Requirement에 있어서는, 서버와의 호환성이 요구사항으로 있었는데, 이는 앞서 언급한 기능 충족과 더불어 안정적으로 이루어 졌다. 윤리적인 요구사항에 있어서는, 개인정보 유출을 hash를 통해 최소화함으로써 이루어졌고, 유저들의 Misbehave에 대해서는 Evolution 단계에서 시스템의 악용에 대한 대응, 변화가 필요하다 또한 법적인 요구사항에 있어서는, 가입시 정보 공개에 대한 동의와 가상화폐 생성과 유통에 대한 금융감독원의 허가가 필요한데, 가입시 정보 공개 및 동의는 추후 구현하고, 가상화폐에 대한 거래는 거래소에 이관함과 동시에, 보완이 필요하다.

7. Reference

<https://www.guru99.com/levels-of-testing.html> 단계적 테스트에 대해서 이해 및 재정리.

<https://www.guru99.com/test-case.html> 테스트 케이스 작성에 대한 참조.

<https://www.guru99.com/test-scenario.html> 테스트 시나리오 작성에 대한 참조.