

1. Title Page

Project #2. Software Design Description (SDD)

Team 9:

Rakhman Asmatullayev	20180830
Hongxiao Yao	20196360
Umar Taufiqulhakim	20170896
Hyunseung Lim	20170548

Date: 13.11.19

Table of Contents:

1. Title Page.....	1
2. System Overview.....	1
3. System Design.....	6
4. Open Source ALM.....	19
5. Acknowledgment.....	19

2. System Overview

A. Requirements

Functional Requirements		
R. ID	Requirement Description	Dependencies/ Assumptions
R.F.1	The System must contain Account Create and Login Function.	-
R.F.1.1	There is a function that enables user to create an account.	-
R.F.1.1.1	There is a function that enables users to authenticate themselves when creating an account.	-
R.F.1.2	There is a function that enables user to modify their account information.	R.F.1.1 is completed
R.F.1.3	There is a function that enables user to login and logout of their accounts.	R.F.1.1 is completed
R.F.1.4	There is a function that enables manager to sign in administrator account.	R.F.1.1 is completed
R.F.2	The System must contain Upload and Search Functions.	R.F.1 is completed
R.F.2.1	There is a function that enables the seller to upload the	R.F.1 is completed

	good they want to sell.	
R.F.2.2	There is a function that enables buyer to search for the item they need by name.	R.F.1 is completed
R.F.2.3	There is a function that enables buyer to see the detailed information about the specific item.	R.F.2.1 is completed
R.F.3	The System must contain Auction Function.	R.F.2 is completed
R.F.3.1	There is a function that enables sellers to set the price for the good and appropriate deadline.	R.F.2 is completed
R.F.3.2	There is a function that enables buyers to bid the price for the good.	R.F.2 is completed
R.F.3.3	There is a function that enables buyers and sellers to check the current price.	R.F.2 is completed
R.F.4	The System must contain the Payment Function.	R.F.3 is completed
R.F.4.1	There is a function that creates a virtual account that can be deposited by buyer.	R.F.3 is completed
R.F.4.2	There is a function that enables buyers to set the preferable delivery methods.	R.F.3 is completed
R.F.4.3	There is a function that enables buyer and seller to confirm the transactions.	R.F.3 is completed
R.F.4.4	There is a function that enables sellers to make a refund if the quality of item bought doesn't match the description.	R.F.4.1, R.F.4.3 are completed
R.F.4.5	There is a function that automatically sends money to the seller if delivery and satisfaction are confirmed by buyer.	R.F.4.1-R.F.4.3 are completed

Nonfunctional Requirements	
R. ID	Requirement Description
R.N.1	There should be Requirements, Code and End-User Documentations for the application
R.N.2	System is proposed to be used on mobile devices at Android platform with API level 19 or higher (KitKat and later versions)
R.N.3	Device used is proposed to have at least 50 Mb of free space
R.N.4	System should respond within 2 seconds
R.N.5	System is proposed to use internet connection at speed about 250 Kb/s

R.N.6	System may have a delay when working at low internet connection, but it still should work properly
R.N.7	System should check the input for a correct(expected) type and output an error message for improper input
R.N.8	When user uploads the image for the product to sell, it should be of type *.jpg or *.jpeg, and its' size should be no more than 2 Mb.
R.N.9	App is expected to have a 100% reliability, working 24/7
R.N.10	The downtime of a system should be 0
R.N.11	In case of failure the app should restart within 5-10 seconds.
R.N.12	As long as the new environment fulfill the given minimum specification, the system should work normally
R.N.13	The UI or payment security are the most likely to be modified in the future
R.N.14	The application is expected to be hosted at Google Play servers, so there will not be any extra equipment
R.N.15	All input and output of the system will be available only inside the application at mobile device
R.N.16	System administrator should have access to any data so that he/she can control and modify any unacceptable information
R.N.17	The system is expected to be backed up once a week
R.N.18	Back up will be done automatically under the control of system administrator
R.N.19	Physical security isn't an issue as there is almost no private information used in the system
R.N.20	The application will be designed through the Android Studio Application by 4 members of our team. Application is supposed to be done in about 1 month of developing. There are no financial resources needed
R.N.21	Developers need to know the Object-Oriented Programming in Java and developing in Android Studio
R.N.21	The deadline is projected to be at the beginning of December
R.N.22	There is no budget for the application, all required hardware already is available and developers will not earn anything
R.N.23	All 4 developers are responsible for system installation and system maintenance

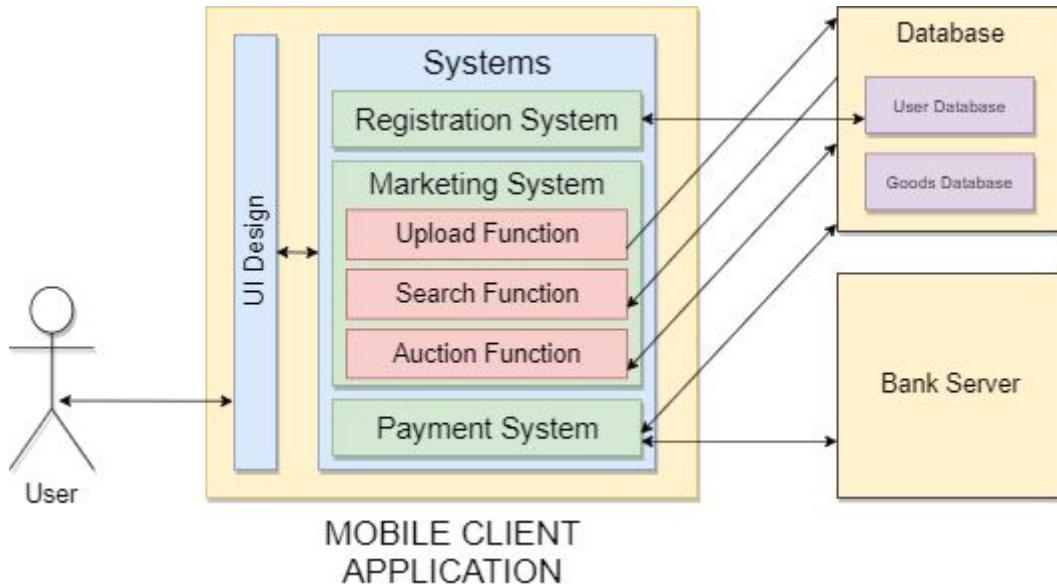
B. Tasks

Task Model		
Task ID	Task description	Related Req(s).
T.1	Build new development environment	-
T.2	Create application UI	-
T.2.1	Create home page UI	R.F.1
T.2.2	Create registration and login page UI	R.F.1.1
T.2.2.1	Create 'Sign up' page UI	R.F.1.3
T.2.2.2	Create 'Information Modification' page UI	R.F.1.2
T.2.2.3	Create 'Administrator Login' page UI	R.F.1.4
T.2.2.4	Create user's home page UI	R.F.1.2
T.2.3	Create goods list UI	R.F.2.2
T.2.3.1	Create goods categories UI	R.F.2.2
T.2.3.2	Create item's home page UI	R.F.2.3
T.2.4	Create Auction System UI	R.F.3
T.2.4.1	Create 'Set Price and Deadline' page UI	R.F.3.1
T.2.4.2	Create 'Submit Bid Price and Real-time Information' page UI	R.F.3.2;R.F.3.3
T.2.5	Create Payment System UI	R.F.4
T.2.5.1	Create 'Create account' page UI	R.F.4.1
T.2.5.2	Create 'The result of payment' page UI	R.F.4.3
T2.6	Implement the jump between different pairs of pages	R.N.4
T.3	Create database to store the information of users, goods and so on	-
T.4	Implement Registration and Login System	R.F.1;R.N.1
T.4.1	Implement 'Create Account' functionality	R.F.1.1;R.F.1.1.1;R.N.7
T.4.2	Implement 'Modify Information' functionality	R.F.1.2;R.N.7
T.4.3	Implement 'User Landing' functionality	R.F.1.3;R.N.7

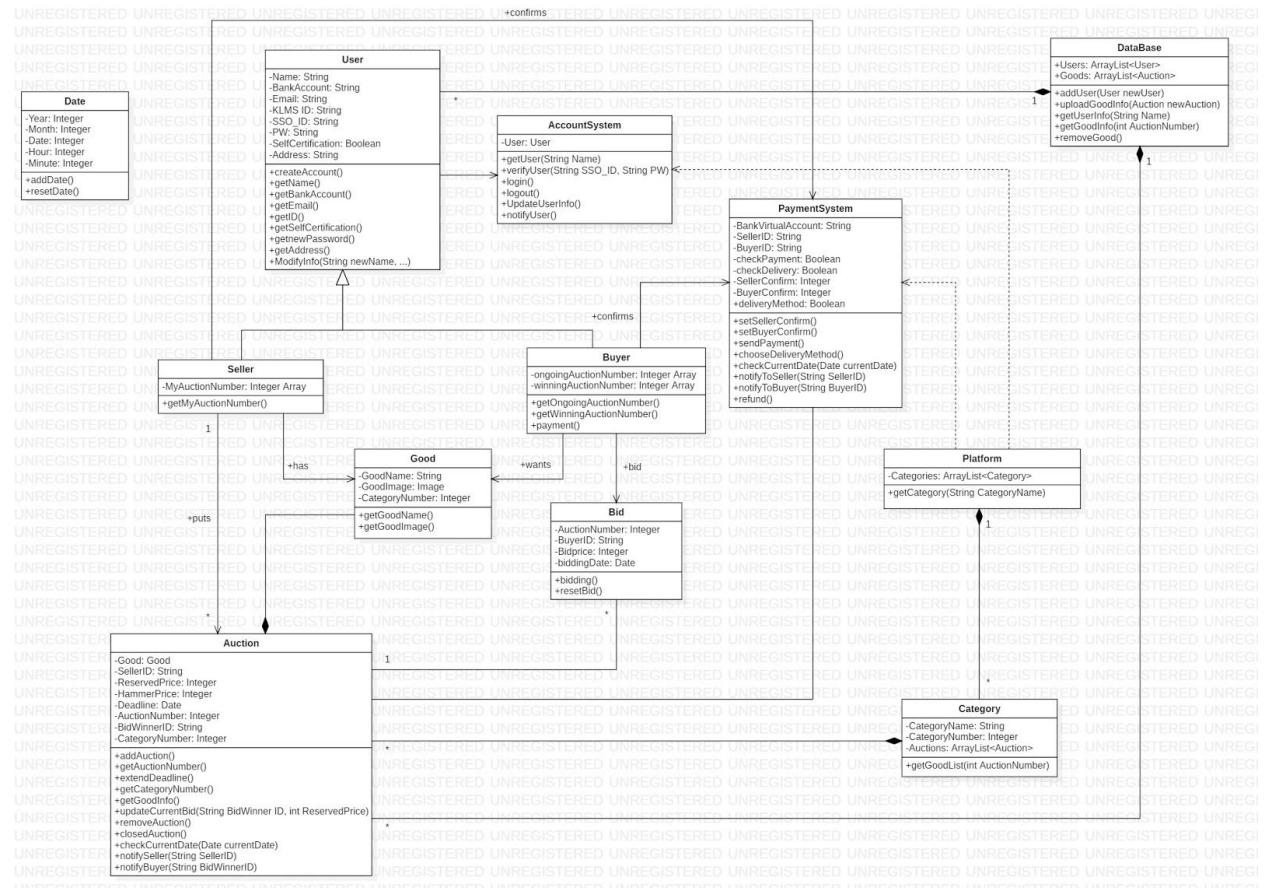
T.4.4	Implement 'Administrator Landing' functionality	R.F.1.3;R.N.7;R.N.16
T4.5	Connect 'Registration and Login System' with registration and login page UI	-
T.5	Implement 'Upload Goods' functionality	R.F.2;R.F.2.1
T.5.1	Implement upload the goods information in database	R.F.2;R.F.2.1;R.N.8
T.5.2	Implement add the item in goods list and right category	R.F.2;R.F.2.1
T.6	Implement 'Search Goods' functionality	R.F.2;R.F.2.2
T.6.1	Implement 'Retrieve the available good list'	R.F.2.2;R.F.2.3
T.7	Connect 'Upload Goods' and 'Search Goods' functions with good list UI	-
T.8	Implement 'Auction System' functionality	R.F.3
T.8.1	Implement 'Set price and deadline' functionality	R.F.3.1
T.8.2	Implement 'Submit bid price' functionality	R.F.3.2
T.8.3	Implement 'Display and update real-time information' functionality	R.F.3.3
T.8.4	Connect 'Auction System' to auction system UI	-
T.9	Implement 'Payment System' functionality	R.F.4
T.9.1	Create bank account for our application	-
T.9.2	Acquire pre-authorization qualification	-
T.9.3	Implement 'Automatic Payment Recognition' functionality	R.F.4.1
T.9.4	Implement 'Set delivery method' functionality	R.F.4.2
T.9.5	Implement 'Confirm transaction is success' functionality	R.F.4.3
T.9.6	Implement 'Forward payment automatically' functionality	R.F.4.5
T.9.7	Implement 'Refund' functionality	R.F.4.4
T.9.8	Connect 'Payment System' to Payment System UI	-

3. System Design

A. System Architecture



B. Class Diagram



C. User Interface Design (Mockup Design)

Category



Book

Book Contemporary Linear Algebra by...
Current price 30000won
Deadline 11/13 23:59

Book Calculus Early Transcendentals
Current price 48000won
Deadline 11/16 23:59

Book Biology with MasteringBiology 8th
Current price 55000won
Deadline 11/17 23:59

Browse

Categories

See More








Recent Posts

See More

Book Contemporary Linear Algebra by...
Current price 30000won
Deadline 11/13 23:59

 MY AUCTION  HOME  PROFILE



(1 / 3)

Contemporary linear algebra by howard anton
Current price 30000won
Deadline 11/13 23:59

 **Jinu Choi**
Korean

30000won   **Bid Now**

My auction

 **Hyunseung Lim**
Korean

Ongoing

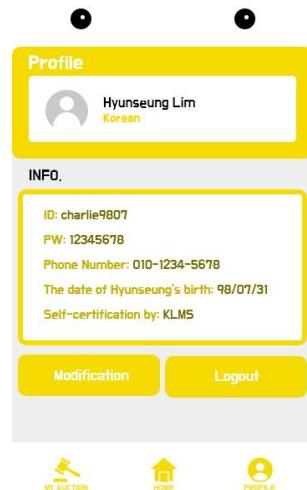
Book Contemporary Linear Algebra by...
Current price 30000won
Deadline 11/13 23:59

You are the biggest bidder now!

Finished

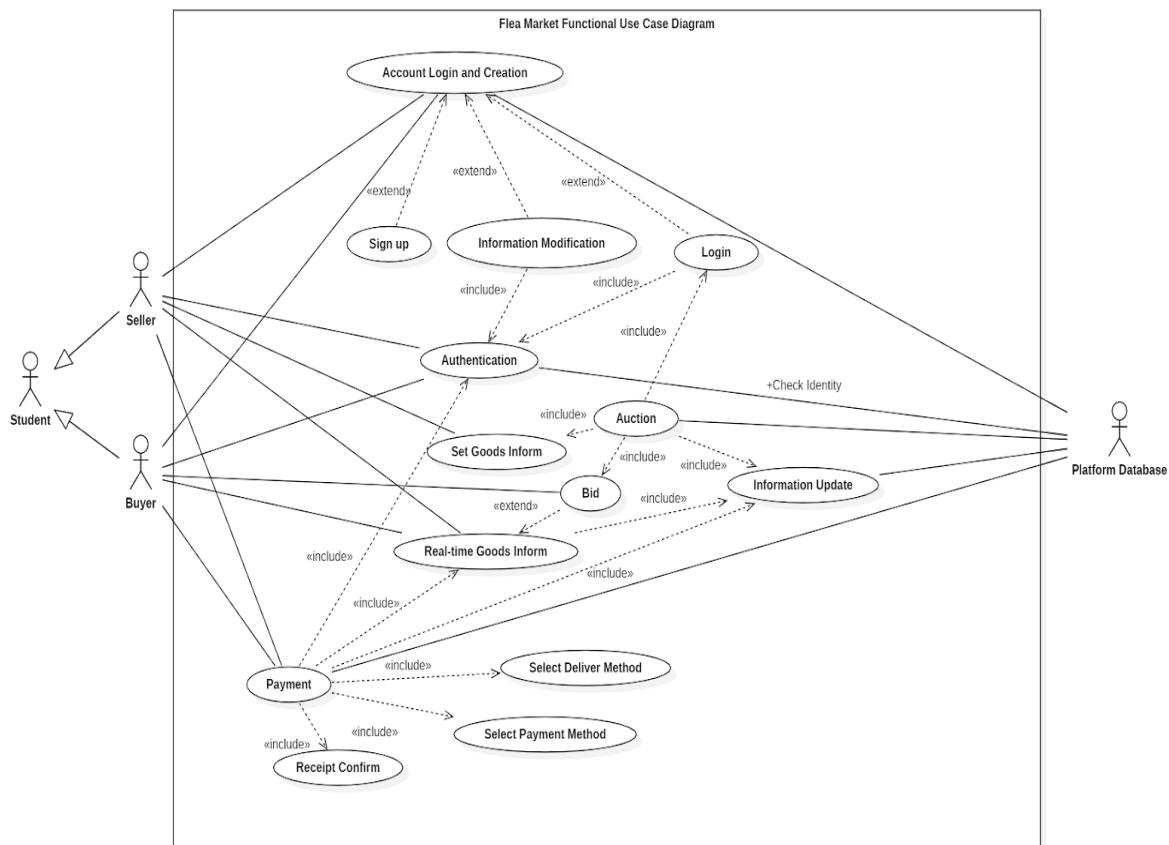
 MY AUCTION  HOME 

 MY AUCTION  HOME 



MY AUCTION HOME PROFILE

D. (Refined) Use Case Diagram and Description



Use Case Name	Account Login and Creation System
Related Requirements	

Goal in Context	The student who expects to sell something which they will never use again or get some tools or goods at a preferential price requests an account and uses this account to trade or search for useful information.
Preconditions	The student who wants to use this APP needs to own KLMS ID, effective email and appropriate proof of identity at first.
Successful End Condition	An account is created and stored in the database of our platform. Or successfully login somebody's account with their unique KLMS ID or email and correct password. In the latter case, someone modifies the password through the email from our platform.
Failed End Condition	The application for an account is rejected because the lack of KLMS ID. Somebody can not log in their account for the wrong password, KLMS ID or email. Somebody can not modify their password because of fallacious email.
Primary Actors	Users
Secondary Actors	Platform Database
Trigger	The user needs to apply for an account, login or modify his or her password.

Main Flow 1	Step	Action
	1	The user asks the system to create an account.
	2	The user enters his or her detailed information.(KLMS ID, Email and Password)
	3	The user's information is checked using database.
	4	The user's information is stored into the database of our platform.
	5	The new account is created.
	6	The account's details are emailed to the user.
Extensions	Step	Branching Action
	1	The user's KLMS ID can not be verified.
	2	The user's account application is rejected.
Main Flow 2	Step	Action

	1	The user asks the system to log in his or her unique account.
	2	The user enters his or her KLMS ID or email and password.
	3	The user's information is checked by our platform's database.
	4	Successfully login.
Extensions	Step	Branching Action
	1	The database of our platform does not verify the user's information.
	2	The user's application for login is rejected.
Main Flow 3	Step	Action
	1	The user asks the system to modify his or her information of account.
	2	The user does not forget the account's password and successfully login.
	3	The user changes the account's information, and the database updates its information.
Extensions	Step	Branching Action
	1	The user forgets his or her password of account.
	2	The user chooses to enter his or her email.
	3	The database checked this email.
	4	Our platform sends the user an email to help modify the password.
	5	The user enters the new password and double check password.
	6	The database updates the information of user's account.
	7	Our platform sends the user an email to notify him or her and the password has been modified.

Use Case Name	Auction System
Related	This system requires every seller and buyer have been registered

Requirements	through our Account Login and Creation System.	
Goal in Context	The student who wants to sell something with a nice price can puts the item on our APP, and our APP will sell this item to someone based on the seller's request(Auction or One to one)	
Preconditions	The users have been registered and verified the identities. What's more, they do not have bad credit or fraud history.	
Successful End Condition	The seller successfully put his or her item on our platform and the buyers can get the item through the auction or one-to-one.	
Failed End Condition	Because seller or buyer has bad credit or fraud history, they could not sell goods or buy something through our platform. The seller has the item off the middle of the auction or the buyer abandons to pay for the item.	
Primary Actors	Seller and Buyer	
Secondary Actors	Platform database	
Trigger	The user wants to sell or buy something at a preferential price.	
Main Flow 1	Step	Action
	1	The seller selects the item to put on our APP.
	2	The seller needs to fill some necessary information and price of the item, and chooses the way of sale.(Auction or one-to-one)
	3	The information will be updated in our database and put the current information on our APP.
	4	In the auction system, the buyers firstly set their expecting price or add price before deadline.
	5	The system will select the highest price automatically and updated the price in database, and our platform will show the current price to both sellers and buyers.
	6	After the deadline, the item will be the buyer who set the highest price.
	7	The system will notice the buyer to pay and jump to payment system.
Extension 1	Step	Branching Action
	1	If no buyer chooses the item after the deadline, the information of the item will be erased from the platform.

	2	The transaction will be closed.
Extension 2	Step	Branching Action
	1	If the buyer abandons the payment, the system will give this item to the buyer who provides the second highest price.
	2	The platform will send an email to the buyer to remind him or her.
	3	Back to the main page of Auction system.
Extension 3	Step	Action
	1	The seller selects the one-to-one method for sale.
	2	The buyer chooses the item.
	3	The buyer fill the payment information.
	4	The system will remind the seller to prepare to deliver the item to the buyer.
	5	Jump to the payment system.
Extension 4	Step	Branching Action
	1	The buyer abandons the payment.
	2	The system will put the item on the platform again.

Use Case Name	Payment System
Related Requirements	This system is based on the former two systems, so it should be connected to the Account Login and Creation system and Auction a Market System.
Goal in Context	The buyers who have selected their biased items need to pay for the goods and wait for them delivered from sellers. And our platform needs to ensure transaction processes are safe and reliable, so we introduced deposit mechanism and double check mechanism to protect our users interest from threatening and losing.
Preconditions	The buyer has selected the item or supported the highest price after deadline.

Successful End Condition	Both buyers and sellers confirm they have received the goods or delivered the goods, and buyers make sure the goods are in good condition. Then money will be paid to the seller by our platform. If buyers have received the goods and make sure they are in good condition but forget to click the confirm button, then the seller can give feedback to our platform. If the situation is true, the money will be paid to sellers after 5 days.	
Failed End Condition	Buyers abandon to pay for the goods. Buyers do not receive the goods in 5 days. Buyers think the items are in poor quality. Sellers do not confirm that they have delivered the goods in 5 days. Sellers abandon to sell the items.	
Primary Actors	Seller and Buyer	
Secondary Actors	Platform database	
Trigger	The buyers have selected their interested goods.	
Main Flow	Step	Action
	1	The buyer selects the item on our APP.
	2	The buyer needs to fill some necessary information and choose the postal methods.(may include extra fee), and pay for the item(credit card or mobile payment)
	3	The information will be updated in our database and the money will be kept safely by our platform.
	4	The platform will send the address and the postal method to sellers.
	5	After the sellers have send the goods successfully, they need to click the confirm button and the platform will inform the buyers that their goods are on the way.
	6	After buyers receive the goods, they should check the condition of goods and click the confirm receipt button.
	7	Our platform will give the money to sellers.
	8	The transaction will be done.
Extension 1	Step	Branching Action
	1	If buyers fill the wrong address or other information, and also notice us.

	2	The transaction will be closed, and give the chance for buyers to change the information.
	3	Then the transaction will be continued.
Extension 2	Step	Branching Action
	1	If the buyer abandons the payment, the system will close the current transaction.
	2	The platform will send email to the seller to remind him or her.
	3	Back to the Auction System
Extension 3	Step	Action
	1	If buyers do not receive the goods in 5 days and the situation is true after the platform confirmation.
	2	The platform will remind the sellers.
	3	The sellers do not respond in a day, the transaction will be stopped.
	4	The money will be back to buyers by our platform.
Extension 4	Step	Branching Action
	1	If buyers think the goods are not in good condition, they can feedback to our platform.
	2	The platform will remind the sellers.
	3	The goods will be back to the sellers.
	4	The transaction will be stopped.
	5	The money will be back to buyers.
Extension 5	Step	Branching Action
	1	If buyers forget to confirm and do not give any feedback to our platform in 5 days after they have received the goods.
	2	The money will be sent to the sellers automatically.
Extension 6	Step	Branching Action
	1	If sellers do not confirm that they have sent the goods out in 5 days.

	2	The platform will remind them.
	3	If sellers do not response in 2 days.
	4	The transaction will be closed.
	5	The money will be back to buyers.

E. (Refined) Sequence Diagram

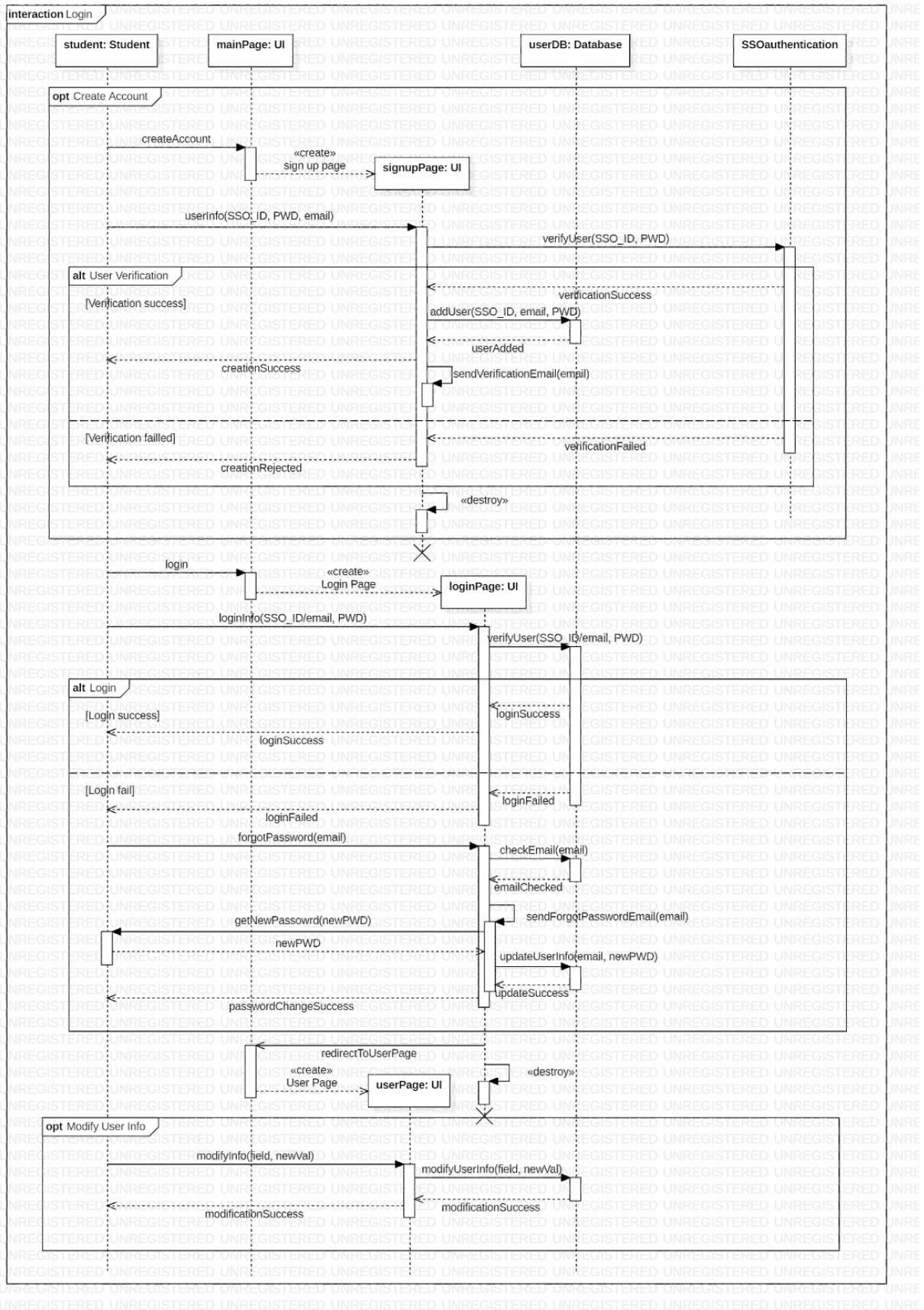


Diagram E.1 Refined Login Sequence Diagram

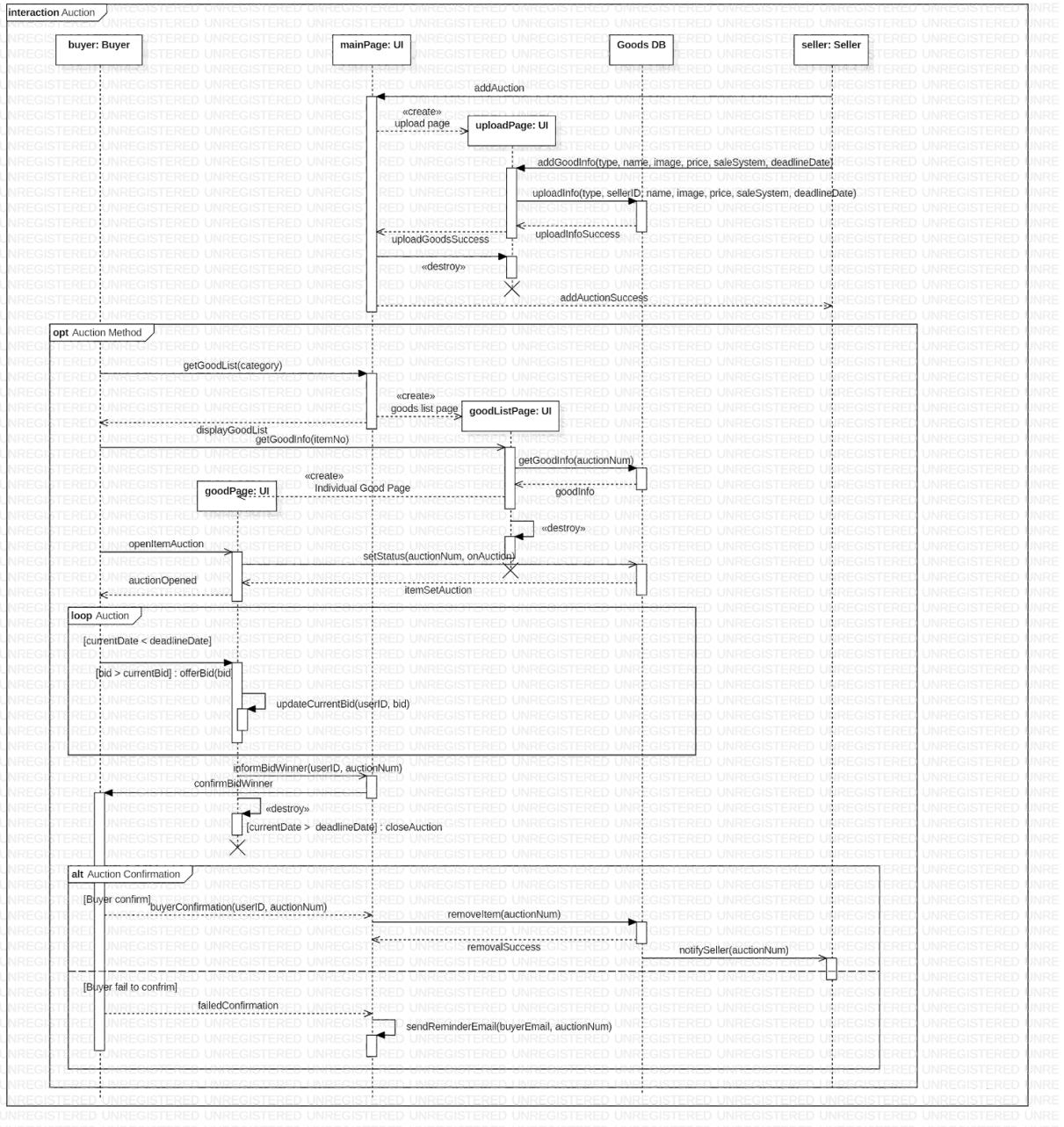
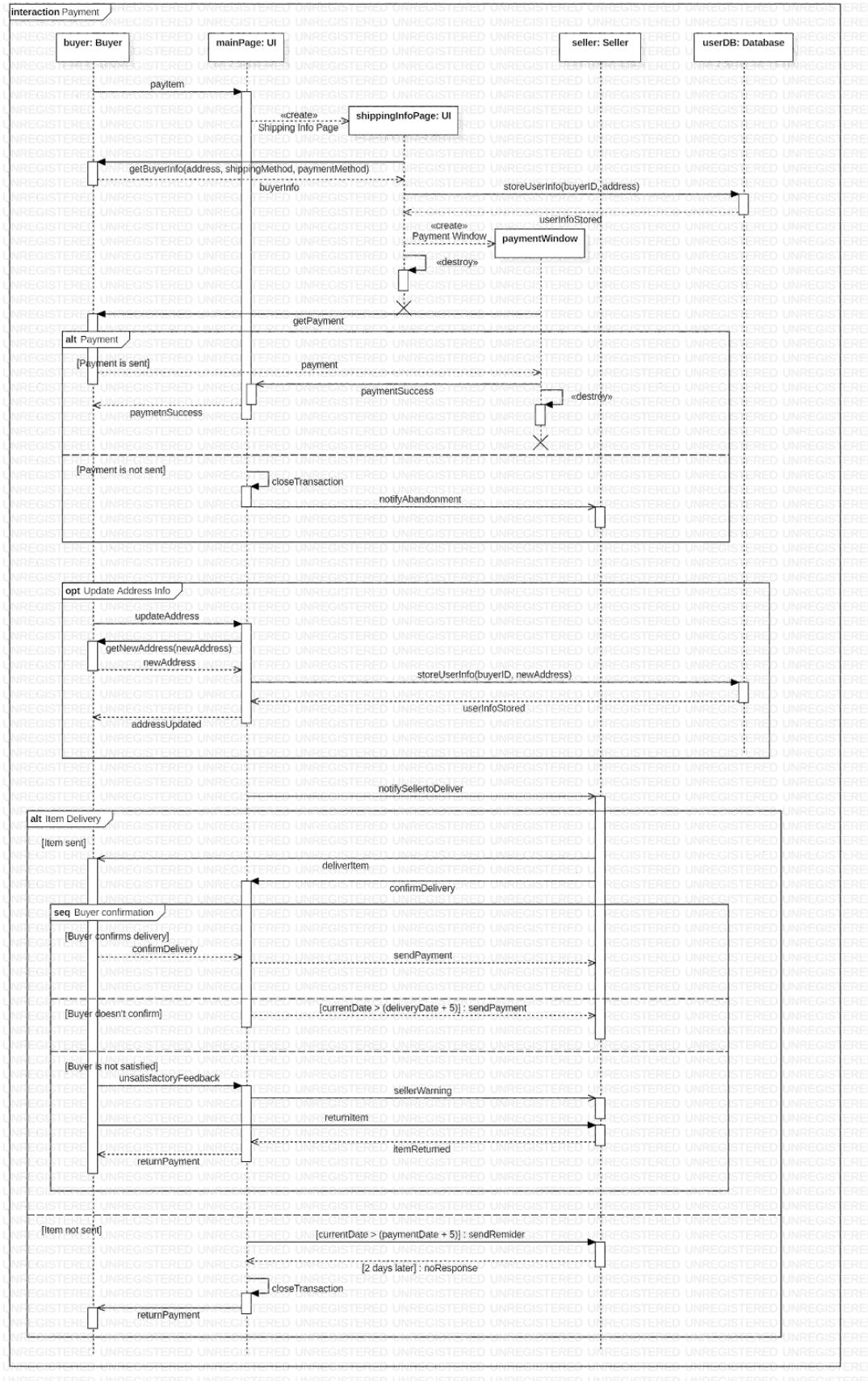


Diagram E.2 Refined Auction Sequence Diagram



4. Open-source ALM (Application Lifecycle Management)

Tools	Purpose
IntelliJ	Integrated Development Environment (IDE)
JUnit	Testing
Maven	Build
Git	Version Control System
Jenkins	Continuous Integration
Sonarqube, PMD, Cobertura	Static Analysis
Github	Issue Management

5. Acknowledgment

This document was done by our team in this partition:

Rakhman Asmatullayev: Section 2.A, Section 3.A

Hongxiao Yao: Section 2.B, Section 3.D

Hyunseung Lim: Section 3.B, Section 3.C

Umar Taufiqulhakim: Section 3.E, Section 4