use case-Tina

---

need item post
Scope: Bear Trade system
Level: user goal
Primary Actor: buyer
Stakeholders and Interests:

- buyer: can make the request of the specific items;
- seller: can repond the request in order to sell the unused items and make the benefit;
- manager: manage the post, and filter out the unqualified item post

Preconditions: the buyer is able to make the post, and the buyer is already logged in
Postconditions: the details about the request is filled and the users in the same community are able to see it and make the comments. The buyer can manage the request post later like delete or make the deal.
Main Success Scenario :

1. the buyer make the request post
2. the buyer fill the information: items name,pri ce, items description and prefered payment method
3. system record the information and return the information buyer entered
4. buyer consent the posting
5. system post the request into the community

Extension:
a. duplicated post

1. system returns the previous post, and ask whether make a change or cancel the new post
2. if the buyer choose to make a change, system will return the editing post page, and buyer make the change, then save and make visible to the other user. if choose to cancel, then close the request window

b.invalid price range(negative number)

1. the system will show the warning message and ask the user the type the price again untill the price is in valid range

c.no entered for prefer payment method

1. the system will set the google pay in default

Bidding
Scope: Bear Trade system

Level: user goal
Primary Actor: buyer
Stakeholders and Interests:

- buyer: can bid the higher price
- seller: can get the higher price deal than the initial price
- manager: manage the bid, and filter out the unqualified bid

Preconditions: the buyer is visible to the bid post, and buyer credit is able to make a bid
Postconditions: the price of the bid is elavated, and the buyer who post the highest bid so far temporarily take the bid
Main Success Scenario :

1. the buyer saw the bid information about the price and time last
2. the buyer input the price
3. the system record the price and the bid owner
4. the system update the price of the bid post

Extension:
a.invalid price input

1. the system will show the error message
2. the system will ask the input the price again or give up bidding untill the valid price input or give up bidding

b.invalid credit

1. the system will show the error message and cancel the bidding request
2. the system return to the home page

evaluation
Scope: Bear Trade system
Level: user goal
Primary Actor: buyer, seller
Stakeholders and Interests:

- buyer: can reflect the experience of this trade, and report the bad behavior throught the trade
- seller: can reflect the experience of this trade, and report the bad behavior throught the trade
- manager: manage thecommunity user, and restrict the user which has low credit

Preconditions: the trade is finished successfully
Postconditions: the buyer and seller credit might be changed becuase the evaluation.
Main Success Scenario :

1. The participants(buyer or seller) will receive a questionaire from system about grade the other participant, comment the trade and report the user

2. once the both participants finished the questionaire, the system will end the evaluation and recalculate and update the credit for each users

Extension:

a.no evaluation input

1. if the user is not evaluated the other in certain time, the evaluation will not influence toward the other credit

b. report the user

1. the system will send the message to the manager
2. manager will verify and evaluate the report, and reflect the deduction on the other's credit
3. the system will record the deduction and recalculate and update the credit

use case - Ting-chen

■ Use Case: Leave messages
■ Primary Actor: Users
■ Scope: Bear Trade system
■ Level: User goal
■ Stakeholders and interests
Buyers
-people that want to make a purchase on the platform
Sellers:
-people who want to sell their items on the platform
System maintainer
-person who makes sure the platform is running properly and is responsible for updating and maintaining the system.
■ Preconditions: Bear trade platform is running and the customers and the sellers are identified and authorized to access the platform.
■ Success Guarantees(Post conditions): A user, either a buyer or a seller, can leave messages to other users.
■ Main Success Scenario (Basic Flow):
1. A user opens the profile of another user.
2. The profile shows basic information such as account id, name, evaluation about the user, contact information, and credit history.
3. On the profile page, a user can leave messages to the user.
■ Extensions:
■ a. Chatting box is not working.
– The system maintainer should look into it and fix the bug.

■ a. System stops functioning.
– System maintainer should look into the issues and restart the system.
■ b. Users find system fails to update page.
– System maintainer should fix the problem.
■ c. Any actions that alerts the security system of the trading platform.
- The system should send an urgent emergency note to the system maintainer.

■ Use Case: Making a payment on Bear trade platform
■ Primary Actor: Users
■ Scope: Bear Trade system
■ Level: User goal
■ Stakeholders and interests
-Buyers
-people that want to make a payment on the platform
Sellers:
-people who want to sell their items on the platform
System maintainer
-person who makes sure the platform is running properly and is responsible for updating and maintaining the system.
■ Preconditions: Bear trade platform is running and the customers and the sellers are identified and authorized to access the platform.
■ Success Guarantees(Post conditions): The correct amount of charge is calculated. Card validation and other online payment methods are checked when the payment is paid by a credit card or the online payments methods. A transaction is processed. Payment authorization approval is recorded. The receipt is sent through text message or email. Items are being shipped to the address.
■ Main Success Scenario (Basic Flow):
1. Buyers enters an item id on the platform.
2. The platform checks if an item id is valid.
3. The cost of the item is calculated.
4. The amount is showed on the web page.
5. The order page asks for the payment option, either cash or online payment methods, such as google pay or apple pay.
6. The transacting system verifies the payment option.
7. A transaction is processed with customer's payment option.
8. The item purchased is recorded.
9. The platform asks for mailing address.
10.The platform asks if the user want to receive the receipt, either through text message or email address.
11. The transaction process record is saved.

■ Extensions:
■ a. Invalid credit card information
– Transaction process stops. System returns a message requiring the customer to try again with other credit cards or other payment options.
■ b. Invalid item ID
– System signals an error message and rejects entry
- System prompts the user to enter a valid item ID again
■ c.: System does not receive a response from the user in a timely manner after prompting for a payment method.
- System stops the process.
- System returns to the starting status and gets ready for processing a new transaction.
■ d any actions that alerts the security system of the trading platform.
- The system should send an urgent emergency note to the system maintainer.

■ Use Case: Sell items on Bear trade platform
■ Primary Actor: Sellers
■ Scope: Bear Trade system
■ Level: User goal
■ Stakeholders and interests
Sellers:
-people who sell items on the platform and want to know whether an item has been sold.
System maintainer
-person who makes sure the platform is running properly and is responsible for updating and maintaining the system.
■ Preconditions: Bear trade platform is running and the customers and the sellers are identified and authorized to access the platform.
■ Success Guarantees(Post conditions): Items claimed by a user will be marked as sold and will no longer be visible to the rest of the community.
■ Main Success Scenario (Basic Flow):
1. The item's id being claimed is marked sold.
2. The item id is removed from the platform visible to the buyers.
■ Extensions:
■ a. System stops functioning.
– System maintainer should look into the issues and restart the system.
■ b. Customers find system fails to update page.
– System maintainer should fix the problem.
■ c. Customers find system fails to remove sold items.
– System maintainer should fix the problem and update the web page.

use case - Eric

Use Case 1: Post a Comment

Scope: Bear Trade System

Level: User Goal

Primary Actor: Commentator

Stakeholders and Interests:

- Commentator: A user who wishes to post a comment on the Bear Trade Platform
- User: Wish to use system and receive comments

Preconditions: The commentator is logged in and can comment.

Postconditions: A comment is posted by the commentator to the chosen page and is made visible to all who view the page.

Main Success Scenario:

1. The commentator navigates to another user or product that they wish to comment on.
2. The commentator navigates to the comments for the other user or product.
3. The commentator types a comment.
4. The commentator submits their desired comment.
5. The comment appears with any other comments on the user or product's page.

Extensions:

a. At any time, the system stops functioning.
    1. The system restarts.
b. At any time, the commentator no longer wishes to make a comment.
    1. The commentator cancels the act of making a comment.

5a. The commentator no longer wishes for their comment to be visible.

1. The commentator requests to delete the comment.
2. The system deletes their comment.

Use Case 2: Create a Selling Post

Scope: Bear Trade System

Level: User Goal

Primary Actor: Seller

Stakeholders and Interests:

- Seller: Wishes to post an item to sell on the Bear Trade Platform
- Buyer: Wishes to buy items from sellers

Preconditions: The user is logged in and wishes to post an item to sell.

Postconditions: A new item is posted by the user to the Bear Trade Platform.

Main Success Scenario:

1. The seller navigates to their page.
2. The seller starts a new selling post.
3. The seller enters the name of the item they wish to sell.
4. The seller enters a description for the item.
5. The seller provides a price for the item.
6. The seller submits the post of the item.
7. The system registers the post, and it becomes visible.

Extensions:

a. At any time, the system stops functioning.

1. The system restarts.

1-5a. The seller no longer wishes to create a selling post.

1. The user cancels the act of making a selling post.

7a. The user no longer wishes for their selling post to be visible.

1. The user requests to delete the selling post.
2. The system deletes their post.

Use Case 3: Creating a Bidding Post

Scope: Bear Trade System

Level: User Goal

Primary Actor: Seller

Stakeholders and Interests:

- Seller: Wishes to post an item to bid for on the Bear Trade Platform
- Buyer: Wishes to buy items from sellers

Preconditions: The user is logged in and wishes to post an item to bid.

Postconditions: A new item is posted by the user to the Bear Trade Platform.

Main Success Scenario:

1. The seller navigates to their page.
2. The seller starts a new bidding post.
3. The seller enters the name of the item they wish to auction off.
4. The seller enters a description for the item.
5. The seller provides an initial bidding price for the item.
6. The seller provides a timeframe for the bidding.
7. The seller submits the post of the item.
8. The system registers the post, and it becomes visible.

Extensions:

a. At any time, the system stops functioning.

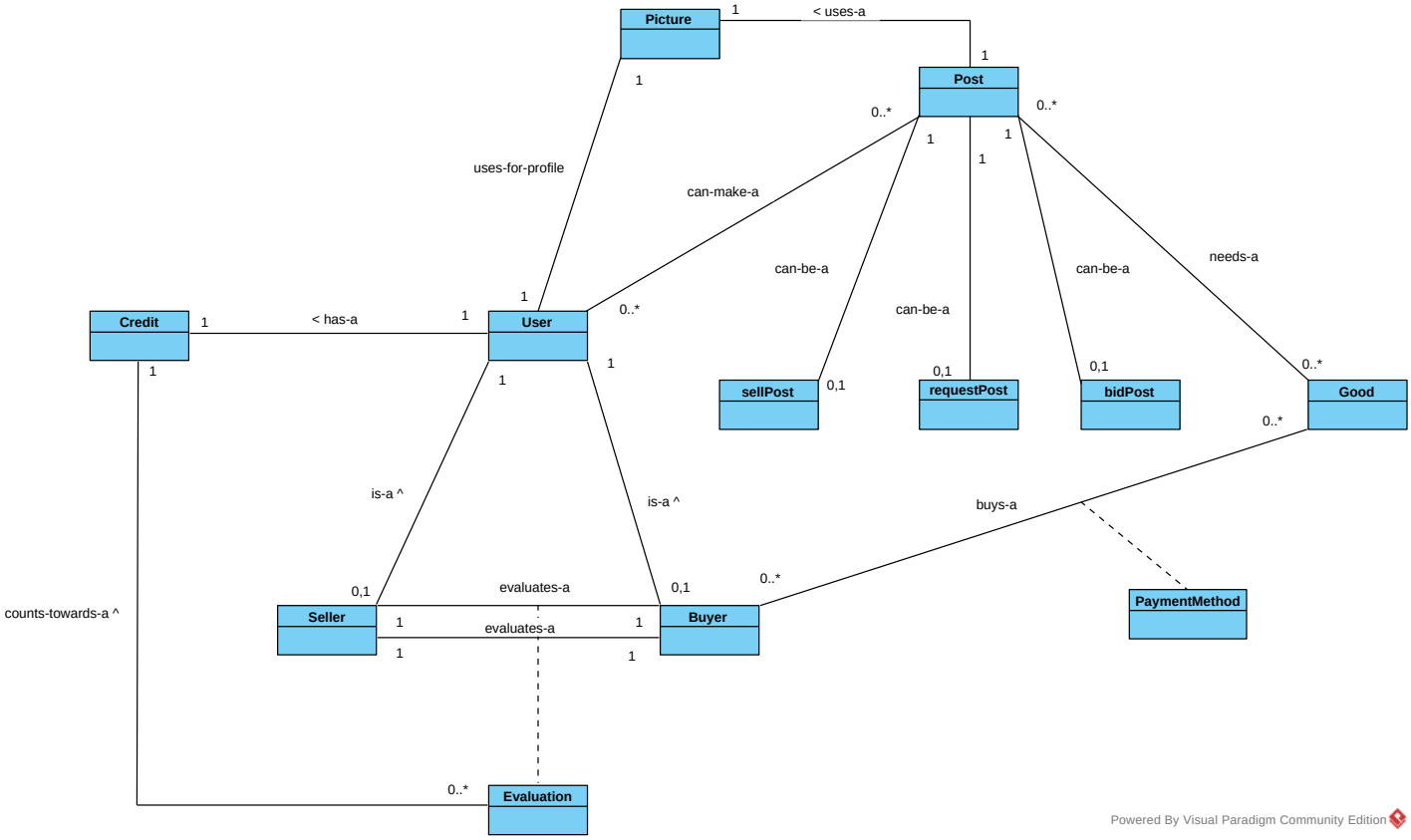
2. The system restarts.

1-6a. The user no longer wishes to create a bidding post.

2. The user cancels the act of making a bidding post.

8a. The user no longer wishes for their bidding post to be visible.

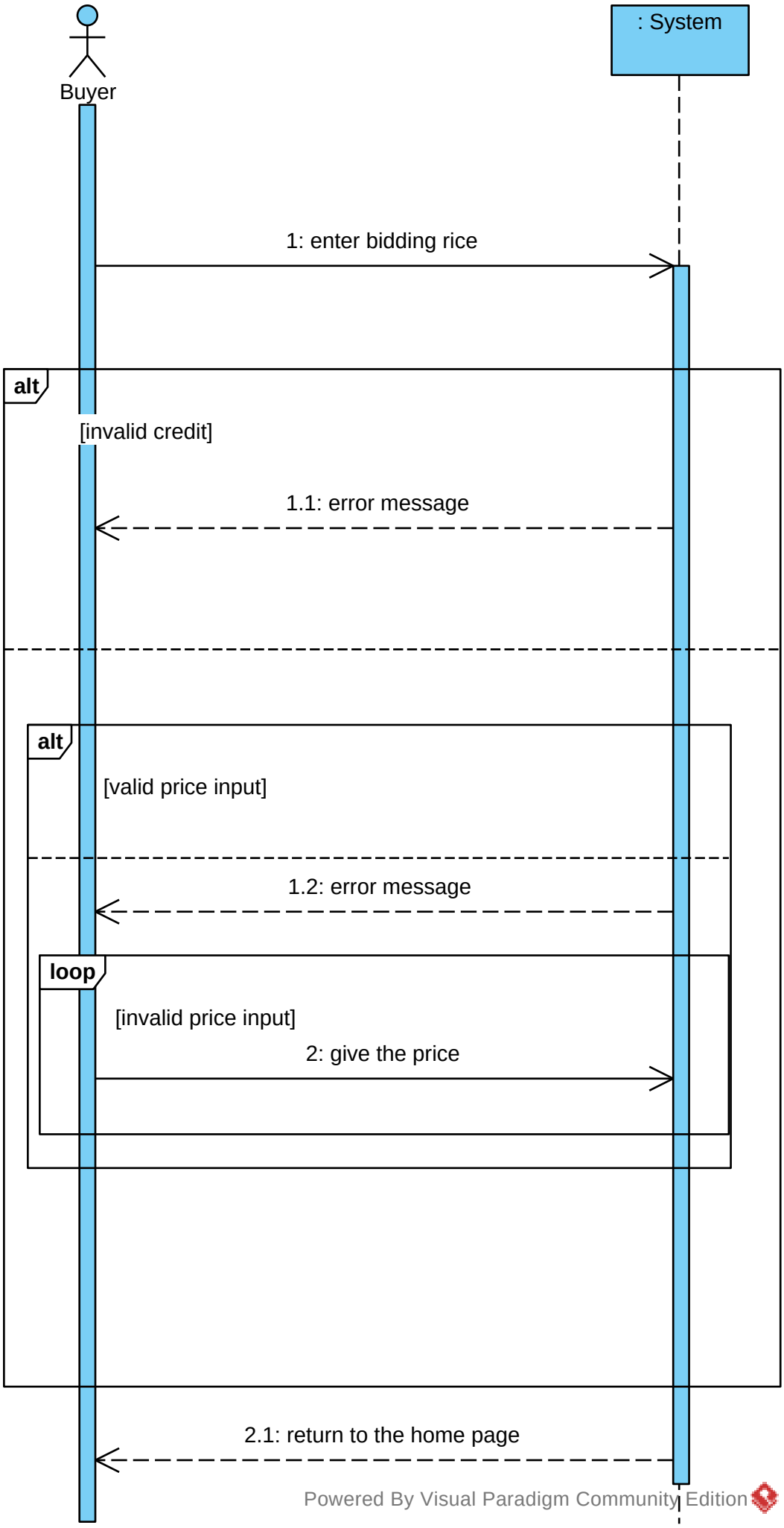
3. The user requests to delete the bidding post.
4. The system deletes their post.


Picture
1
< uses-a
1
Post
1
0..*
0..*
1
1
1
uses-for-profile
can-make-a
can-be-a
can-be-a
can-be-a
needs-a
1
Credit
1
< has-a
1
User
0..*
1
1
1
1
sellPost
0,1
requestPost
0,1
bidPost
0,1
Good
0,1
0..*
0..*
is-a ^
is-a ^
buys-a
0,1
evaluates-a
0,1
0..*
counts-towards-a ^
Seller
1
evaluates-a
1
Buyer
1
1
PaymentMethod
0..*
Evaluation
Powered By Visual Paradigm Community Edition

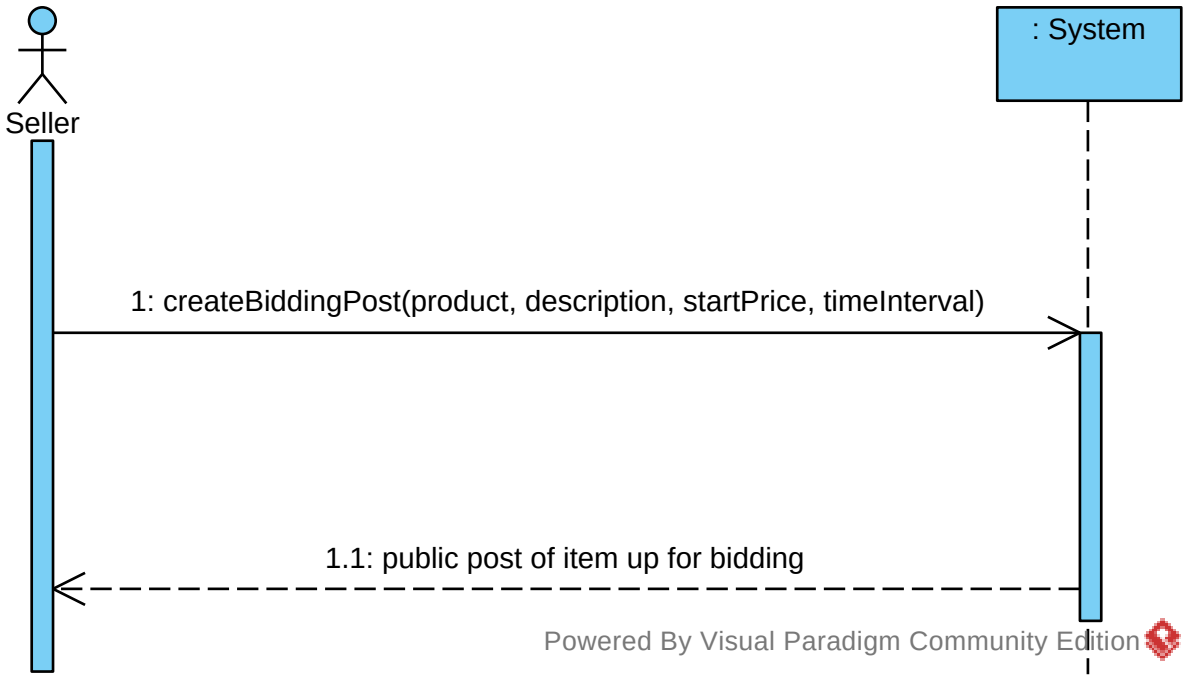
System

+goTo(item)
+createComment(item, message)
+createSellingPost(product, pictures description, price)
+createBiddingPost(product, pictures, description, startPrice, timeInterval)
+createRequestPost()
+giveInfo(product, pictures, description, preferPayment)
+changePost(description, preferPayment)
+reInputPrice(price)
+consentPost(choice)
+GradeTrade(grade, comment)
+doesReport(choice)
+enterBiddingPrice(price)
+makePayment(itemID)
+givePaymentOption(option)
+provideAfterPaymentInfo(address, receipt choice)
+sellItem(itemID)
+updateAvailbleItems()
+viewProfile(profileId)
+leaveMessage(text)

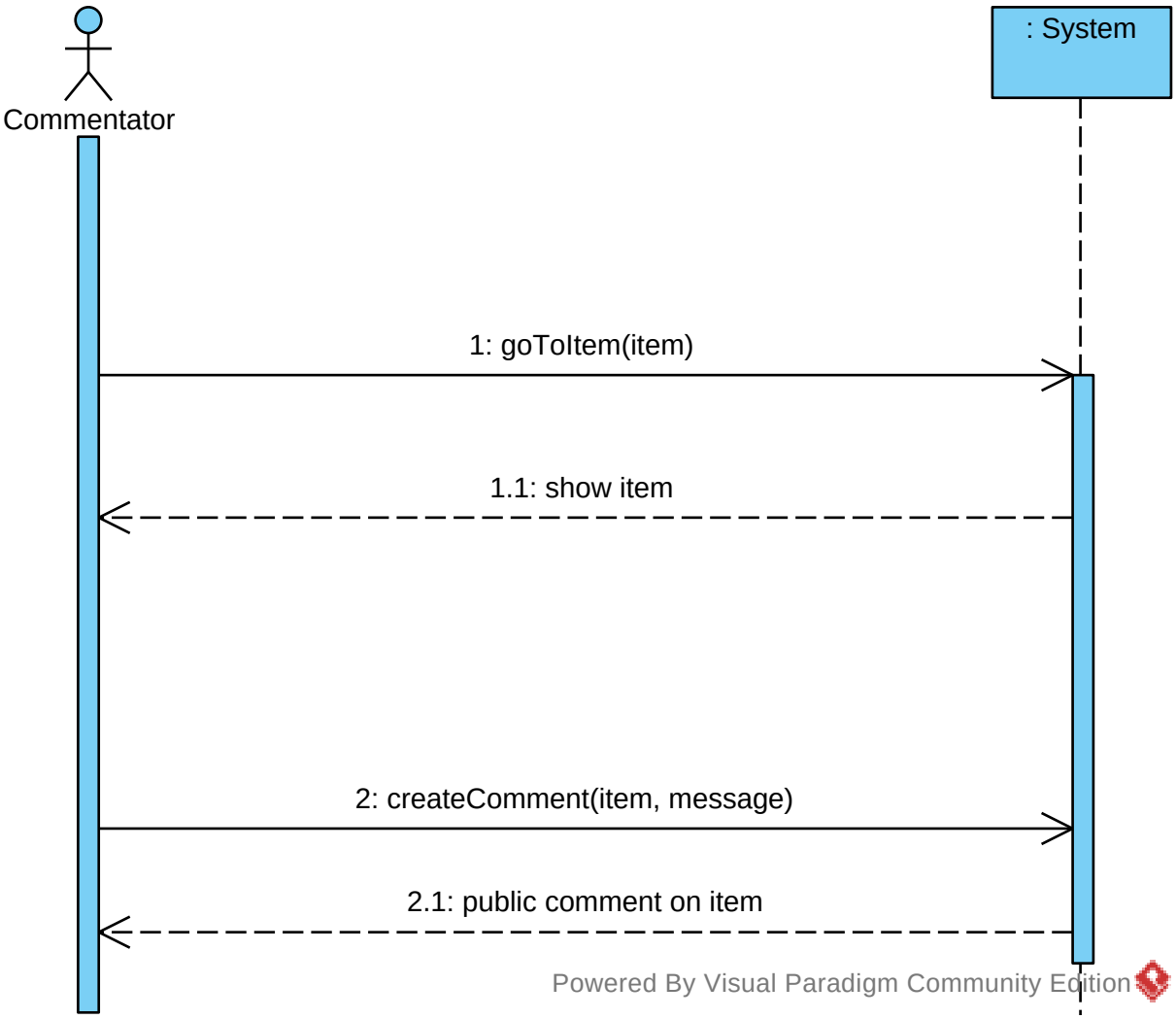
Powered By Visual Paradigm Community Edition

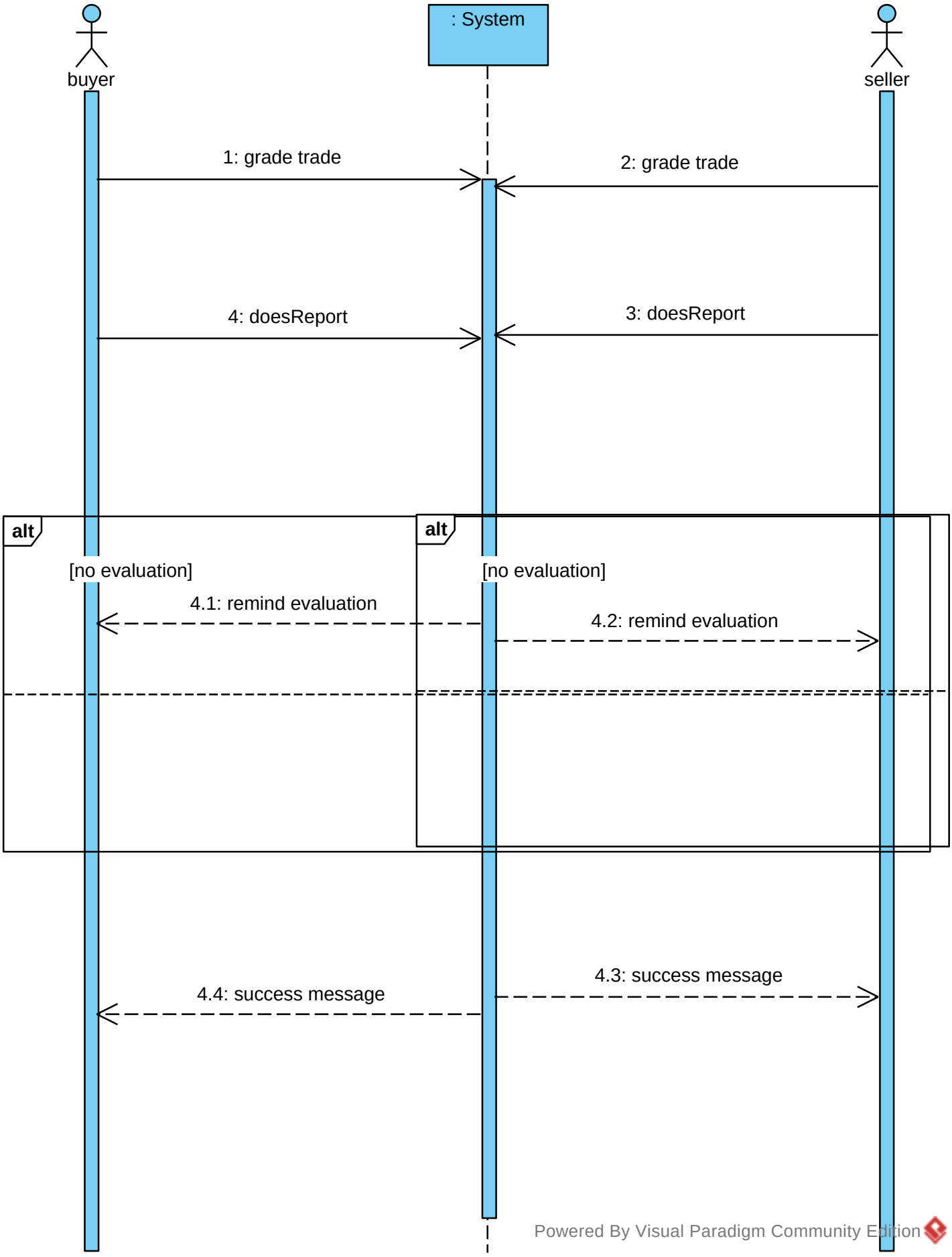


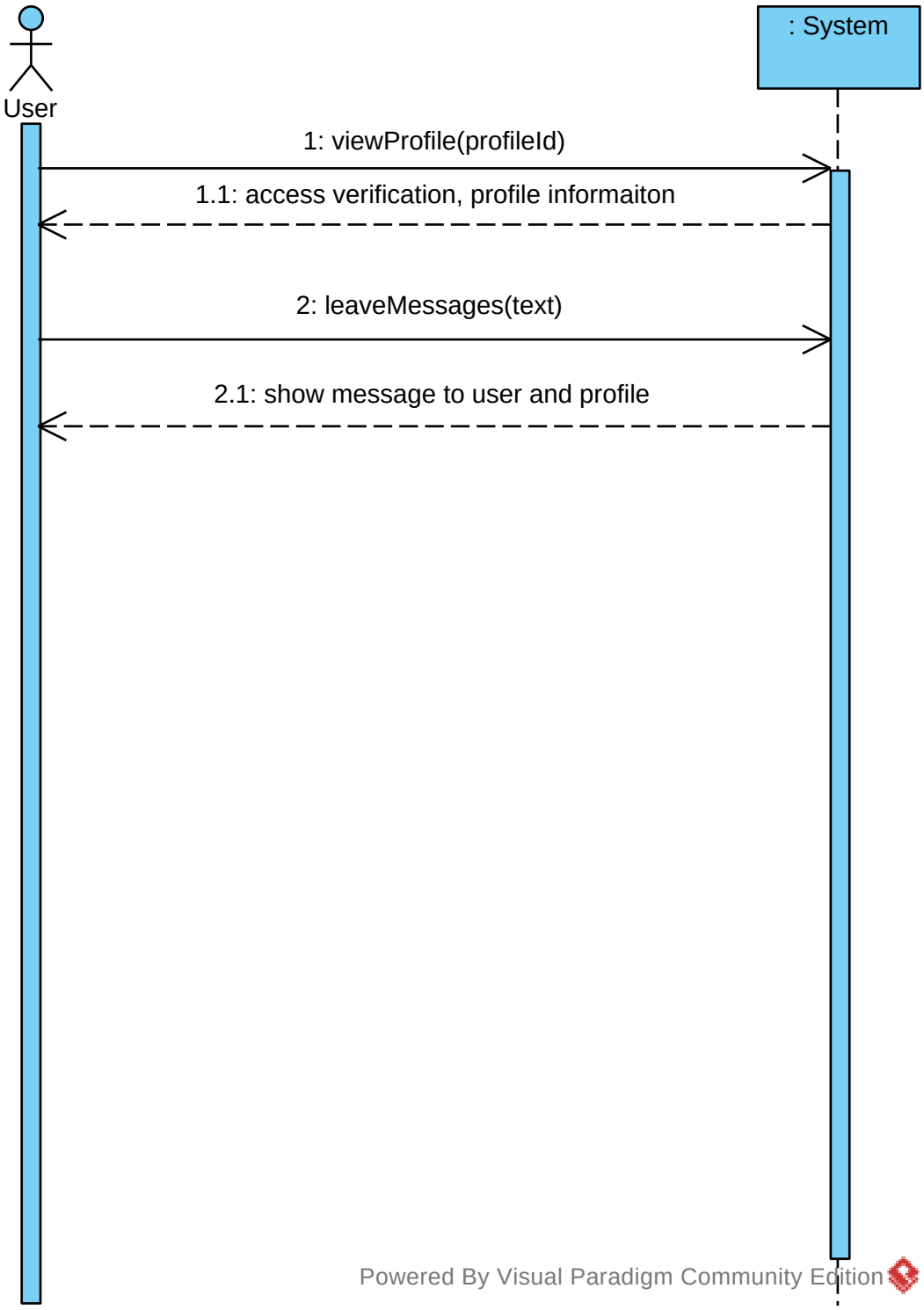
Powered By Visual Paradigm Community Edition

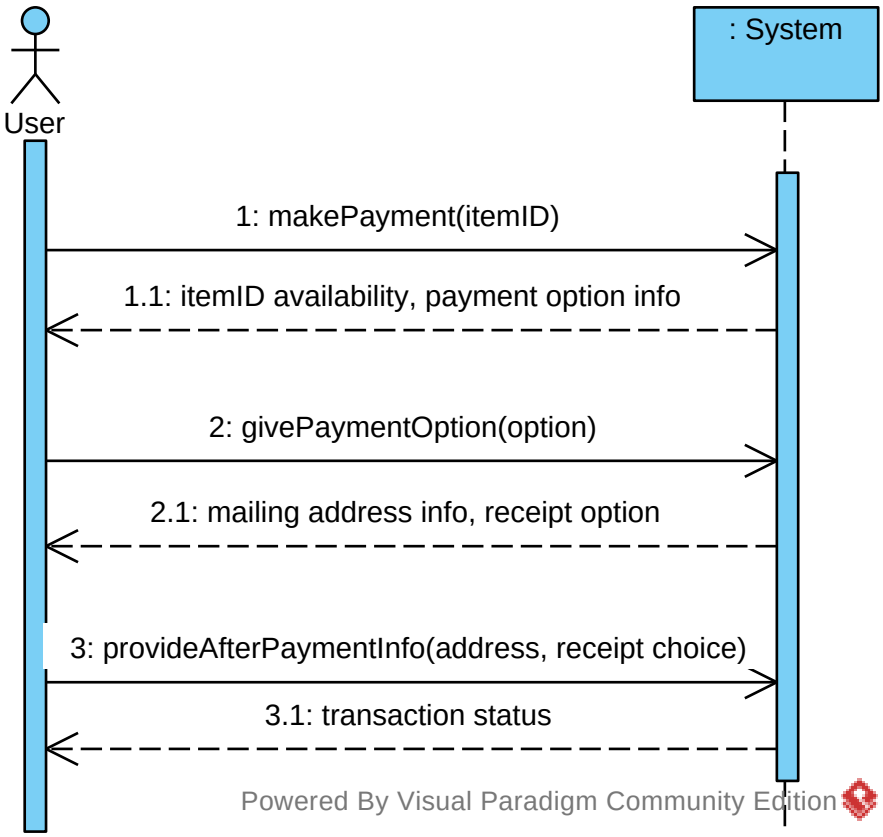

Seller
: System
1: createBiddingPost(product, description, startPrice, timeInterval)
1.1: public post of item up for bidding
Powered By Visual Paradigm Community Edition

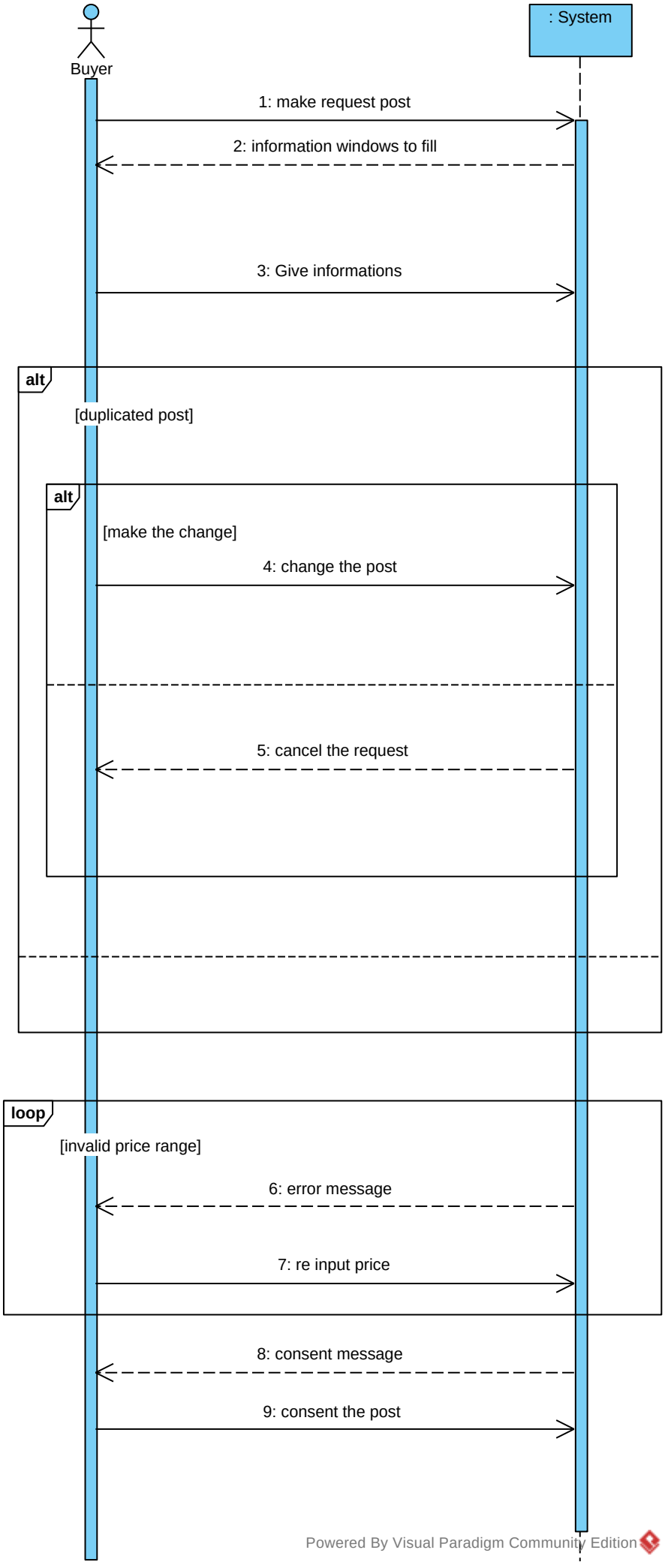
Seller
: System
1: createSellingPost(product, description, price)
1.1: public post of item to sell
Powered By Visual Paradigm Community Edition


Commentator
: System
1: goToItem(item)
1.1: show item
2: createComment(item, message)
2.1: public comment on item
Powered By Visual Paradigm Community Edition

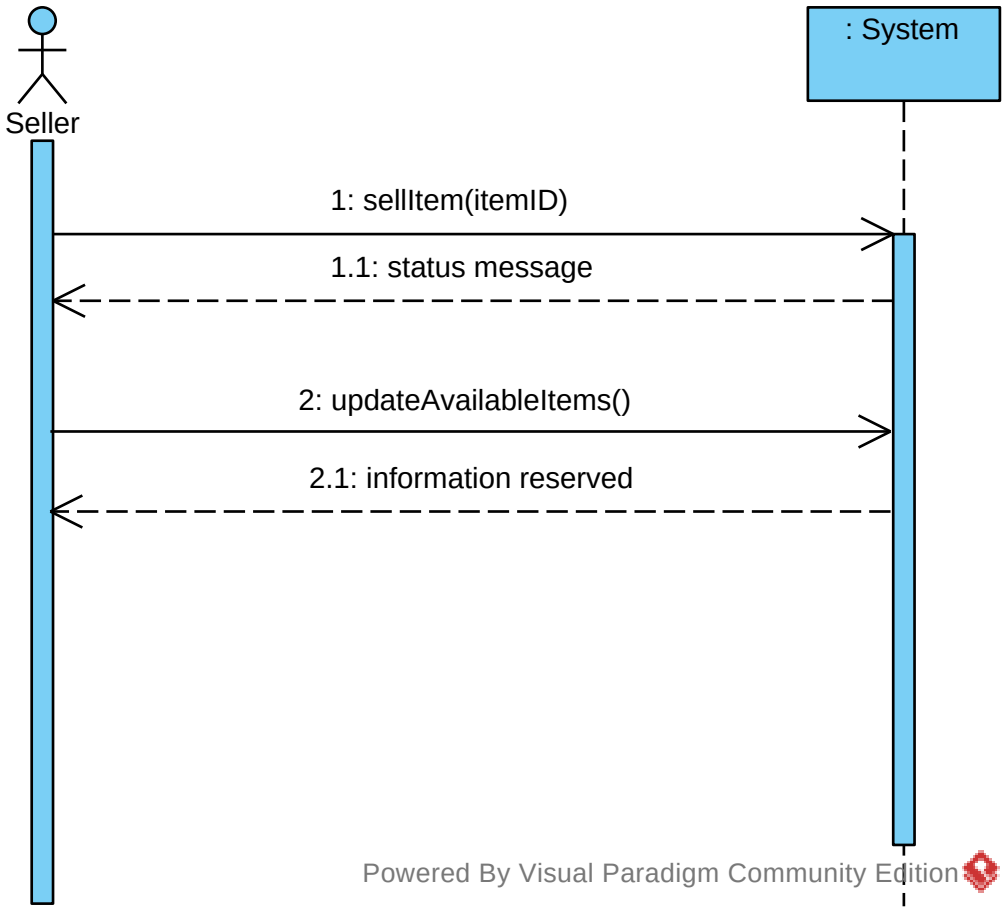

buyer
: System
seller
1: grade trade
2: grade trade
4: doesReport
3: doesReport
alt
[no evaluation]
4.1: remind evaluation
alt
[no evaluation]
4.2: remind evaluation
4.3: success message
4.4: success message
Powered By Visual Paradigm Community Edition

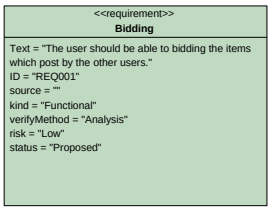

User
: System
1: viewProfile(profileId)
1.1: access verification, profile informaiton
2: leaveMessages(text)
2.1: show message to user and profile
Powered By Visual Paradigm Community Edition

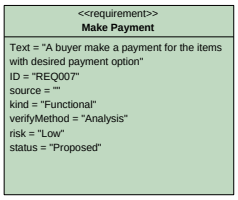

User
: System
1: makePayment(itemID)
1.1: itemID availability, payment option info
2: givePaymentOption(option)
2.1: mailing address info, receipt option
3: provideAfterPaymentInfo(address, receipt choice)
3.1: transaction status
Powered By Visual Paradigm Community Edition

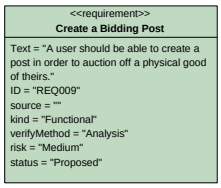


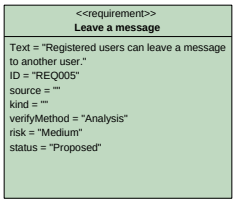
Powered By Visual Paradigm Community Edition

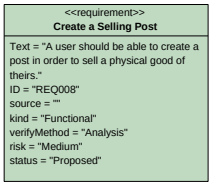

Seller
: System
1: sellItem(itemID)
1.1: status message
2: updateAvailableItems()
2.1: information reserved
Powered By Visual Paradigm Community Edition

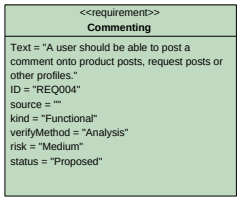


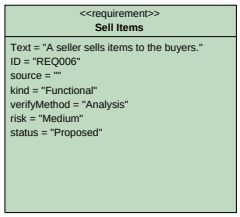


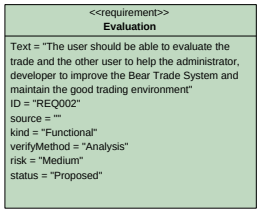


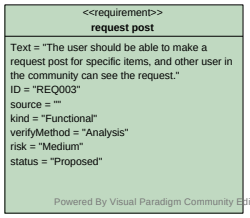












Powered By Visual Paradigm Community Edition

Tina

Contract 1: create request post
Operation: createRequestPost()
Cross References: use Cases: need Item post
Preconditions: The buyer is able to make the post, and the buyer is logged in
Postconditions: it creates a request Post windows and returns to the user

contract 2: give information
Operation: giveInfo(product,description, price, preferPayment)
Cross References: need item post
Preconditions: The request Post window is created
Postconditions: product name, description, price and preferPayment are recorded

contract 3: change post
operation: changePost(description, price, condition, preferPayment)
Cross References: need item post
Preconditions: the request post already existed in the users posting list
Postconditions: some of fields in request post is changed

contract 4: reinput price
operation: reInputPrice(price)
Cross Reference: reques
Preconditions: the price entered previously is not in valid range
Postconditions: the changed price is in valid range

contract 5: consent post
operation: consentPost(choice)
Cross reference: request post
Preconditions: the valid request post is recorded
Postconditions: the request post is released to the community

contract 6: grade trade
operation: gradeTrade(grade, comment)
Cross reference: evaluation
Preconditions: the trade is complished
Postconditions: grade has been reflect to the users' credit

contract 7: does report
operation: doesReport(choice)
Cross reference: evaluation
Preconditions: the trade is complished
Postconditions: the user whether skip the reporting or report the other user

contract 8: enter bidding price
operation: enterBiddingPrice(price)
Cross reference: bidding
Preconditions: the bid post is visible to user, and user is able to bid(above threshold credit)
Postconditions: bidding price is valid, and new price is reflected on the bid post

Eric

Contract 1:
Operation: goToItem(item)
Preconditions:

- The user is logged in.

Postconditions:

- The user arrives at the location of the item online.
- No change to the data in the server

Contract 2:
Operation: createComment(item, message)
Preconditions:

- The user is logged in.
- The item exists online.

Postconditions:

- A comment is posted on "item"'s page.
- Comment contains message "message".

Contract 3:
Operation: createSellingPost(product, description, price)
Preconditions:

- The user is logged in.
- The user has a valid product to sell.

Postconditions:

- A selling post is created for "product".
- The post contains the description "description".
- The post's product is sold for price "price".

Contract 4:
Operation: createBiddingPost(product, description, startPrice, timeInterval)
Preconditions:

- The user is logged in.
- The user has a valid product to sell.

Postconditions:

- A bidding post is created for "product".

- The post contains the description "description".
- The post's product is initially priced at "startingPrice".
- The post will only be active during the time interval "timeInterval".

Ting-chen

Contract1: makePayment
Operation: makePayment(itemID:itemID)
Cross References: Use Cases: make payment
Pre-conditions: There is a transaction being processed.
Post-conditions:
A corresponding itemID will be verified for availibity to purchase.

Contract2: givePaymentOption
Operation: givePaymentOption(option)
Cross References: Use Cases:make payment
Pre-conditions: A payment is being processed.
Post-conditions:
A payment option is generated and return.

Contract3: provideAfterPaymentInfo
Operation: provideAfterPaymentInfo(address, receipt choice)
Cross References: Use Cases: ask for post payment information
Pre-conditions: A transcation is being processed.
Post-conditions:
The item ID that a user desired to buy is verified.
The corresponding item is marked sold.
A mailing address is created
A receipt choice is created.

Contract4: sellItems
Operation: sellItems(itemID)
Cross References: Use Cases: sell item
Pre-conditions: Items claimed by users.
Post-conditions:
ItemID marked as sold.
The itemID will no longer be available and visible to users.

Contract5: updateAvailableItems
Operation: updateAvailableItems()
Cross References: Use Cases: update list of avilable items
Pre-conditions: items being sold from the platform.
Post-conditions:
ItemId of sold items will be removed from the viewable list to users.

The quantity of items will be updated.

Contract6: viewProfile
Operation: viewProfile(profileId)
Cross References: Use Cases: view profile
Pre-conditions: user profiles already exist.
Post-conditions:
Individual profile page can be accessed by other users based on profileId.

Contract7: leaveMessage
Operation: leaveMessage(text)
Cross References: Use Cases: leave messages
Pre-conditions: message boxes are set up.
Post-conditions:
Instances of messages(text) will be created.
Messages can be put into the message box.

Untitled section

main page

click here

Name

credit : 4.5

trade complished = 9999

1,234 likes

chat

need item post

sell item post

bid item post

need item :

Java book

need item

Java book

prefer payment = Cash

Description:

sell item

UML book

price: $999

Description :

request window

bid item

place a bid

Keyboard price: $999 time last : 1 min

Description

need a item

name

Descriptions

post

sell a item

name

price $

Descriptions

Post

create a bid

name

Price $

timeInterval

Descriptions

post

pop windows when you clicks the request

place a bid

Name

Price $

update

place a bid window

Eric

Hi, Eric! Is UML book still available?

Tina

ENTER

chat windows

Time Tracking:

Tina

---

7 hours

Ting-chen

---

7 hours

Eric

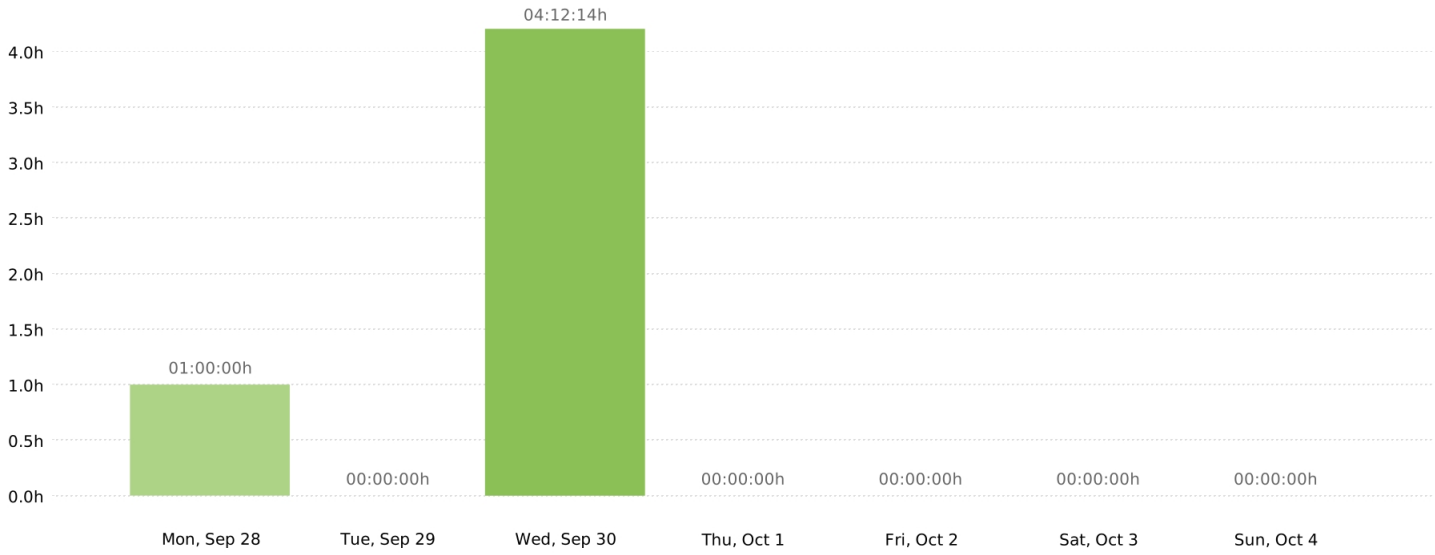
---

7 hours

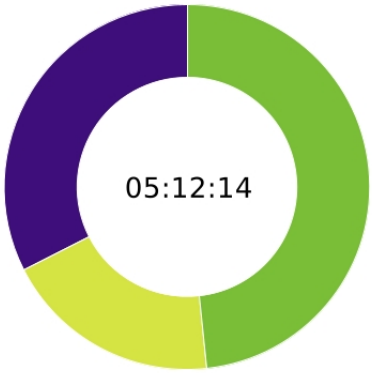
Summary report

09/28/2020 - 10/04/2020

Total: 05:12:14 Billable: 04:12:14 Amount: 0.00 USD

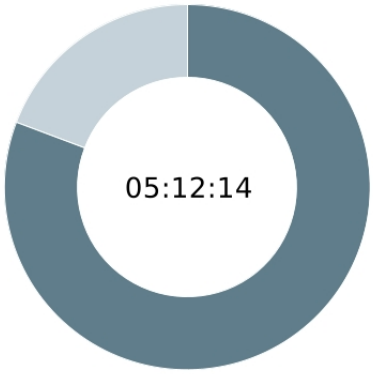


User



Eric Jaroszewski	01:41:38	32.48%
Tina	01:00:00	19.22%
Tingchenwang2019	02:30:36	48.23%

Project



(Without Project)	01:00:00	19.22%
Bear Trade Platform	04:12:14	80.71%

User / Project	Duration	Amount
Eric Jaroszewski	01:41:38	0.00 USD
Bear Trade Platform	01:41:38	0.00 USD
Tina	01:00:00	0.00 USD
(Without Project)	01:00:00	0.00 USD
Tingchenwang2019	02:30:36	0.00 USD
Bear Trade Platform	02:30:36	0.00 USD

Project manager	
Project dates	Sep 21, 2020 - Oct 20, 2020
Completion	0%
Tasks	18
Resources	3

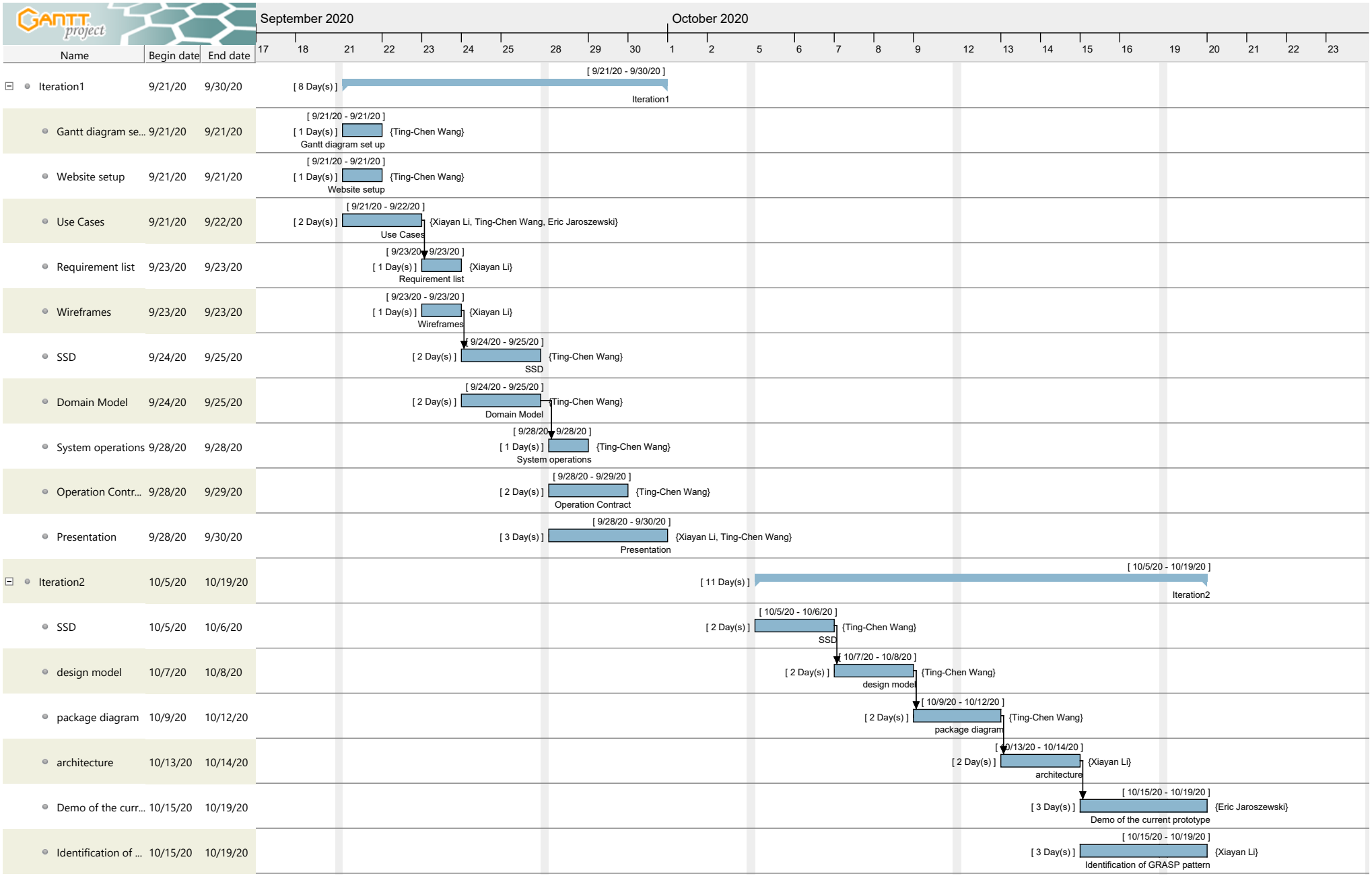
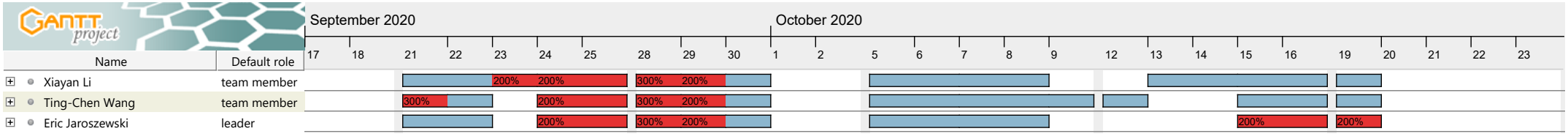
Tasks

Name	Begin date	End date
Iteration1	9/21/20	9/30/20
Gantt diagram set up	9/21/20	9/21/20
Website setup	9/21/20	9/21/20
Use Cases	9/21/20	9/22/20
Requirement list	9/23/20	9/23/20
Wireframes	9/23/20	9/23/20
SSD	9/24/20	9/25/20
Domain Model	9/24/20	9/25/20
System operations	9/28/20	9/28/20
Operation Contract	9/28/20	9/29/20
Presentation	9/28/20	9/30/20
Iteration2	10/5/20	10/19/20
SSD	10/5/20	10/6/20
design model	10/7/20	10/8/20
package diagram	10/9/20	10/12/20
architecture	10/13/20	10/14/20
Demo of the current prototype	10/15/20	10/19/20
Identification of GRASP pattern	10/15/20	10/19/20

Resources

Name	Default role
Xiayan Li	team member
Ting-Chen Wang	team member
Eric Jaroszewski	leader

Gantt Chart


GANTT project
September 2020
October 2020
17 18 21 22 23 24 25 28 29 30 1 2 5 6 7 8 9 12 13 14 15 16 19 20 21 22 23
Name
Begin date
End date
Iteration1 9/21/20 9/30/20
[8 Day(s)]
[9/21/20 - 9/30/20]
Iteration1
Gantt diagram se... 9/21/20 9/21/20
[9/21/20 - 9/21/20]
[1 Day(s)]
{Ting-Chen Wang}
Gantt diagram set up
Website setup 9/21/20 9/21/20
[9/21/20 - 9/21/20]
[1 Day(s)]
{Ting-Chen Wang}
Website setup
Use Cases 9/21/20 9/22/20
[2 Day(s)]
[9/21/20 - 9/22/20]
{Xiayan Li, Ting-Chen Wang, Eric Jaroszewski}
Use Cases
Requirement list 9/23/20 9/23/20
[9/23/20 - 9/23/20]
[1 Day(s)]
{Xiayan Li}
Requirement list
Wireframes 9/23/20 9/23/20
[9/23/20 - 9/23/20]
[1 Day(s)]
{Xiayan Li}
Wireframes
SSD 9/24/20 9/25/20
[9/24/20 - 9/25/20]
[2 Day(s)]
{Ting-Chen Wang}
SSD
Domain Model 9/24/20 9/25/20
[9/24/20 - 9/25/20]
[2 Day(s)]
{Ting-Chen Wang}
Domain Model
System operations 9/28/20 9/28/20
[9/28/20 - 9/28/20]
[1 Day(s)]
{Ting-Chen Wang}
System operations
Operation Contr... 9/28/20 9/29/20
[9/28/20 - 9/29/20]
[2 Day(s)]
{Ting-Chen Wang}
Operation Contract
Presentation 9/28/20 9/30/20
[9/28/20 - 9/30/20]
[3 Day(s)]
{Xiayan Li, Ting-Chen Wang}
Presentation
Iteration2 10/5/20 10/19/20
[10/5/20 - 10/19/20]
[11 Day(s)]
Iteration2
SSD 10/5/20 10/6/20
[10/5/20 - 10/6/20]
[2 Day(s)]
{Ting-Chen Wang}
SSD
design model 10/7/20 10/8/20
[10/7/20 - 10/8/20]
[2 Day(s)]
{Ting-Chen Wang}
design model
package diagram 10/9/20 10/12/20
[10/9/20 - 10/12/20]
[2 Day(s)]
{Ting-Chen Wang}
package diagram
architecture 10/13/20 10/14/20
[10/13/20 - 10/14/20]
[2 Day(s)]
{Xiayan Li}
architecture
Demo of the curr... 10/15/20 10/19/20
[10/15/20 - 10/19/20]
[3 Day(s)]
{Eric Jaroszewski}
Demo of the current prototype
Identification of ... 10/15/20 10/19/20
[10/15/20 - 10/19/20]
[3 Day(s)]
{Xiayan Li}
Identification of GRASP pattern

Resources Chart