BearGo
(Requirements, Use Case, Use Case Diagram, System Sequence Diagram, System Operations)

AGM Islam
Grad Student, Computer Science
agm_islam1@baylor.edu
Baylor University

Maisha Binte Rashid
Grad Student, Computer Science
maisha_rashid1@baylor.edu
Baylor University

Razwan Ahmed Tanvir
Grad Student, Computer Science
razwan_tanvir1@baylor.edu
Baylor University

Swapnil Saha
Grad Student, Computer Science
swapnil_saha1@baylor.edu
Baylor University

Tonni Das
Grad Student, Computer Science
tonni_jui1@baylor.edu
Baylor University

Sep 07, 2022

Project Vision The vision of the project is to create a platform for people to send items to their known acquaintances at a lower cost in a convenient way.

Functional Requirements

1. Product tracking: Traveller will update the status of different stages of the product delivery. The sender will finally confirm if the product is delivered.
2. Report of user: Admin will be able to see any user's activity on a monthly or half yearly or yearly basis.
3. Report collection: Collect information against a post if more than four users have reported it and send a notification to the admin for review of the post in case it needs to be removed.
4. User feed post: Product info posting (routing info, pick up location & travel time) in application feed to find a shipper.
5. Feed post interaction: Comment section under sender's post.
6. Travel blog feed: Registered users can upload their travel stories as blogs in Travel Blogging option.
7. Post search: Users can search other senders' posts based on different filters such as - date, location.
8. Live tweet display: Our system will display tweets from people posting anything about our system or the service.
9. Sender's post update: Those who post about sending products, can also update the post. Senders can change the date, route, and pickup location. But once an invoice has been made or the delivery has passed then sender can not update the post anymore.
10. Invoice: Once Sender and Traveler agree on the shipping, the system will generate an invoice.
11. Post moderation: Admin will decide to remove/keep the post if the post is reported by the users.
12. Notification: real-time web push notification and email notification. Notification will be triggered on different events like: agreement signed, products delivered.
13. Review and rating: When a product will be delivered, the traveler will change the product's tracking info to 'Delivered'. Then a sender will see an option there to post a review and rating.
14. Peer-to-peer chat: Any traveler/sender can contact to any particular sender/traveler to discuss the product details or any other travel information.
15. Social media share: A sender can share his/her post on social media.

Use Case: Product Tracking

ID: UC01

Actors:

1. Traveler
2. Sender

Preconditions:

1. A contract (traveler has accepted sender's request to ship the product) has been made between traveler and sender.

Flow of Events:

1. The use case starts when the traveler receives the product.
2. The traveler selects tracking of the product.
3. The system shows the list of the available delivery status.
4. If the traveler selects the status as "Product Received",
   4.1. The system updates the status of the product as "Received".
5. If the traveler selects the status as "Product In Transit",
   5.1. The system updates the status of the product as "In Transit".
6. If the traveler selects the status as "Product Delivered",
   6.1. The system updates the status of the product as "Delivered".
7. If the sender selects the status as "Delivery Confirmed",
   7.1. The system records the contract as "Successful"

Postconditions: The customer receives the product and the status of the contract has been updated as "Successful".

Alternative flow:

7.2 If the traveler/sender selects "Product Not Delivered", the system will record the contract as "Unsuccessful".

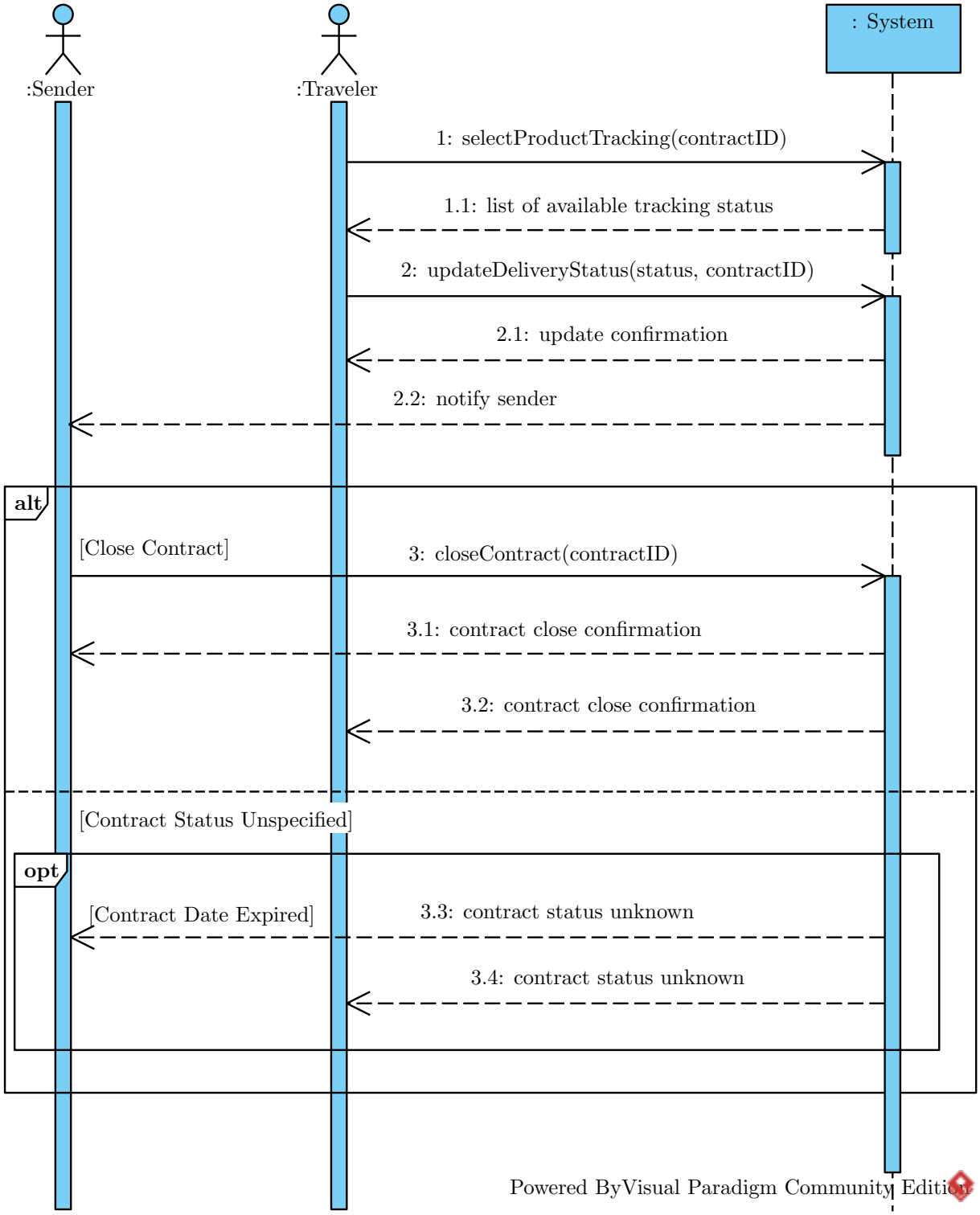
Postconditions: The status of the contract has been updated as "Unsuccessful".

Alternative flow:

7.3 If the sender does not select or forgets to select the status for the product and the contract expires (exceeds product delivery end date), the system will record the contract as "Unknown".

Postconditions: The status of the contract has been updated as "Unknown".

System Sequence Diagram for **Product Tracking**



Use Case: Create Post

ID: UC2

Actors:

1. Sender

Preconditions:

1. The sender is authenticated.

Flow of events:

1. The use case starts when a sender selects "Create Post".
   1.1. The system will return a form for post creation along with "Upload Image", "Upload Post", and "Cancel" options.
2. The sender will fill up the form with product details (dimension, weight, type, value) and itinerary details (source and destination addresses).
3. System will check the character limit.
4. If the sender selects "Upload Image",
   4.1. The system will return a dialog to select image(s).
   4.2. The sender will select image files from his local disk.
   4.3. System will validate the image size.
   4.4. The system will show the names of selected images.
5. If the sender selects "Upload Post",
   5.1. The system will verify the post details (missing fields, incorrect addresses) and image files (file type, file size).
   5.2. If the post details are verified,
      5.2.1. The system will record the post.

Postconditions: The post will be visible on the homepage.

Alternative flow:

3.1 If the character limit exceeds,
   3.1.1 System will show a warning message.

Postconditions: The "Upload Post" button will be disabled.

Alternative flow:

4.3.1 If the image size exceeds,
   4.3.1.1 System will show a warning message.

Postconditions: The image will be removed.

Alternative flow:

5.2.2 If the post details are not verified,
   5.2.2.1 the system will highlight the errors in the same product information form.

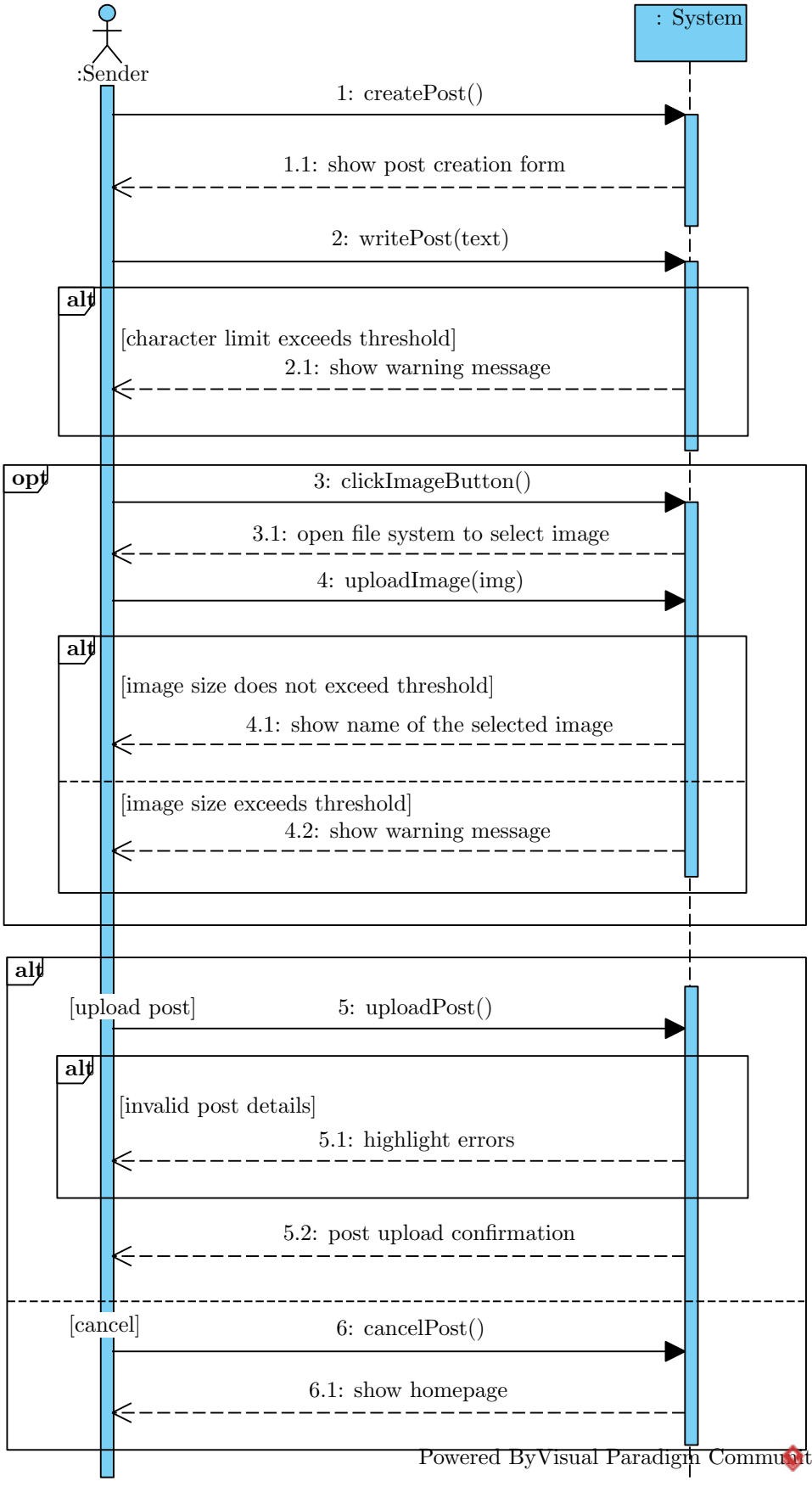
Postconditions: The post will not be recorded.

Alternative flow:

5.3 If the sender clicks the "Cancel" button,
   5.3.1 the system will discard the post and take the user to the homepage.

Postconditions: The homepage will remain the same.

System Sequence Diagram for **Create Post**



Use Case: Review and Rating

ID: UC03

Actors:

1. User

Preconditions: Current date exceeds the contract end date.

Flow of Events:

1. The use case starts when the user selects "Review".
   1.1. The system will return a form for writing the review along with the rating (1 to 5) option, the "Upload Image", "Cancel", and the "Submit" options.
2. The user will fill up the rating option with any rating between 1 to 5.
3. The user will fill up the form with comments.
4. System will check the character limit.
5. If the user selects "Upload Image",
   5.1. The system will return a dialog to select image(s).
   5.2. The reviewer will select image files from his local disk.
   5.3. System will validate the image size
   5.4. The system will show the names of selected images.
6. If the user selects the "Submit" button,
   6.1. The system will verify the rating details (at least a rating between 1 to 5 has to be given).
   6.2. If the review and rating details are verified,
      6.2.1. The system will record that review (may be empty) and rating (has to be given).

Postconditions: The overall rating of the user will be updated.

Alternative flow:

4.1 If the character limit exceeds
   4.1.1 System will show a warning message.

Postconditions: The "Submit" option will be disabled.

Alternative flow:

5.3.1 If the image size exceeds
   5.3.1.1 System will show a warning message.

Postconditions: System will remove the image.

Alternative flow:

6.2.2 If the reviewer does not give a rating,
   6.2.2.1 the system will highlight the rating option.

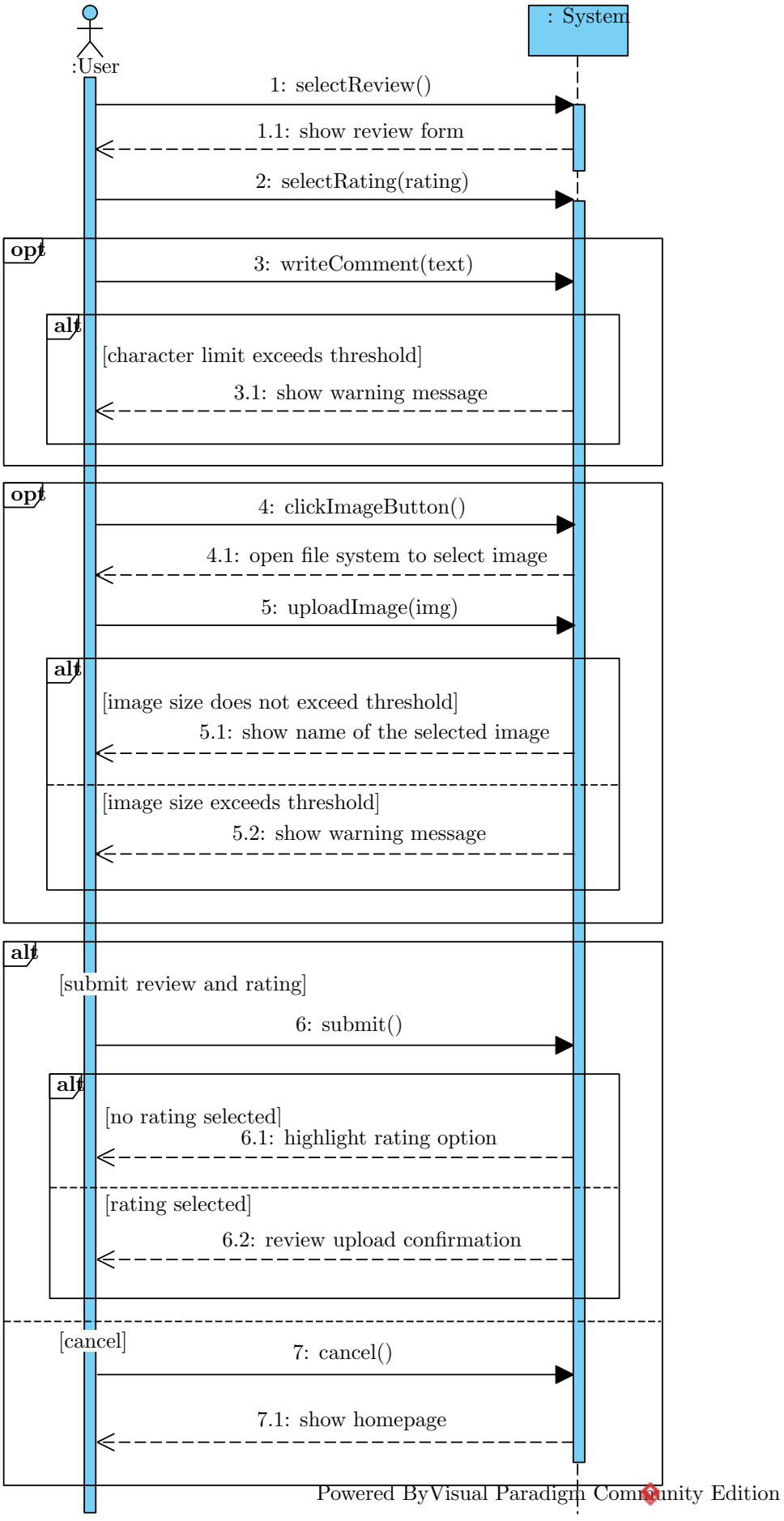
Postconditions: The user will see that the review is uploaded.

Alternative flow:

6.3 If the user selects "Cancel",
   6.3.1 the system will close the form.

Postconditions: The user will be redirected to the homepage.

System Sequence Diagram for **Review and Rating**


:User
: System
1: selectReview()
1.1: show review form
2: selectRating(rating)
opt
3: writeComment(text)
alt
[character limit exceeds threshold]
3.1: show warning message
opt
4: clickImageButton()
4.1: open file system to select image
5: uploadImage(img)
alt
[image size does not exceed threshold]
5.1: show name of the selected image
[image size exceeds threshold]
5.2: show warning message
alt
[submit review and rating]
6: submit()
alt
[no rating selected]
6.1: highlight rating option
[rating selected]
6.2: review upload confirmation
[cancel]
7: cancel()
7.1: show homepage
Powered ByVisual Paradigm Community Edition

Use Case: Peer-to-peer Chat

ID: UC04

Actors:

1. User

Preconditions: The user is authenticated.

Flow of Events:

1. The use case starts after the user selects the "Chat" option.
2. The system asks the user for a name (name of the person that the user is looking for).
3. The user enters the user's name.
4. The system searches for users that match the name.
5. If the system finds some matching users then,
   5.1. The system shows a list of matched users with a "Message" option beside each of these users' names.
6. If the user selects the "Message" option,
   6.1. If there exists previous chat history between them,
      6.1.1. The conversation will be displayed with a text dialog and a "Send" option.
7. If the user writes something in the text dialog and selects "Send",
   7.1. System will send the text to the other user.

Postconditions: The user will see that the conversation between them is updated with the latest text of the user.

Alternative flow:

5.2 If the system cannot find any matching user,
   5.2.1 the system will tell that no matching user could be found.

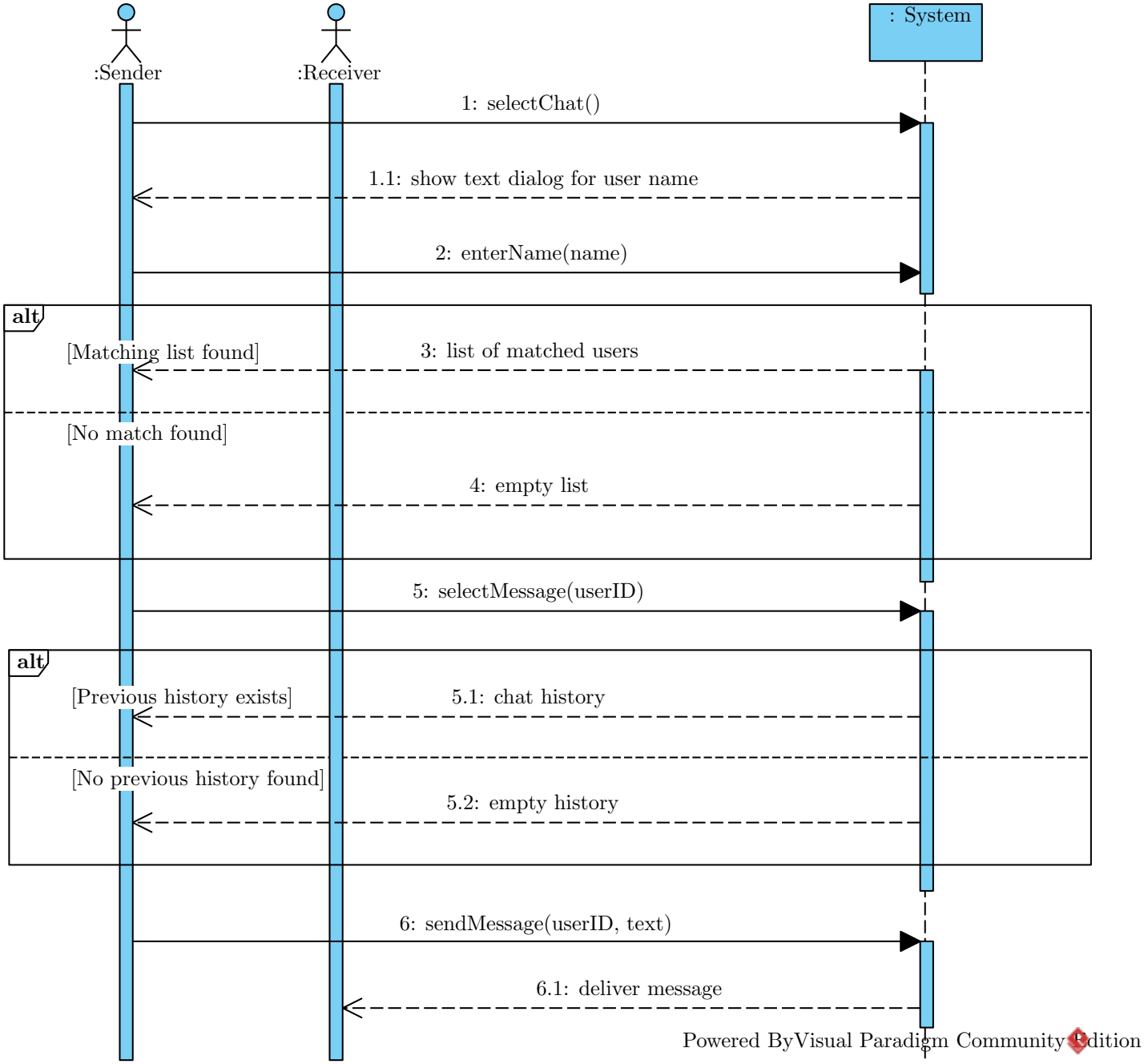
Postconditions: The user will see an empty list.

Alternative flow:

6.1.2 If there is no previous chat history between the users,
   6.1.2.1 the system will only show a text dialog and a "Send" option.

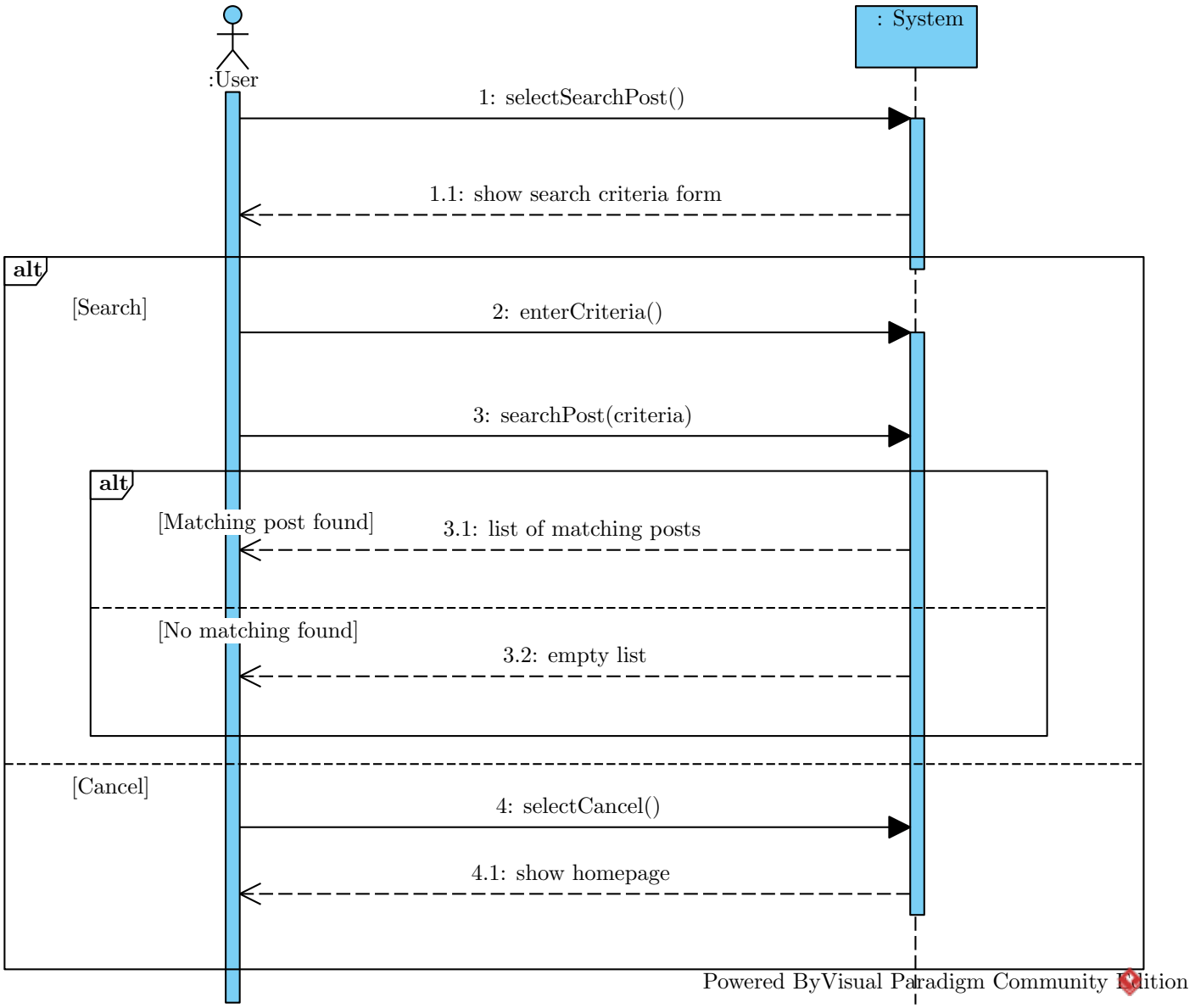
Postconditions: The user will see a conversation with only the recently sent text message.

System Sequence Diagram for **Peer-to-peer Chat**

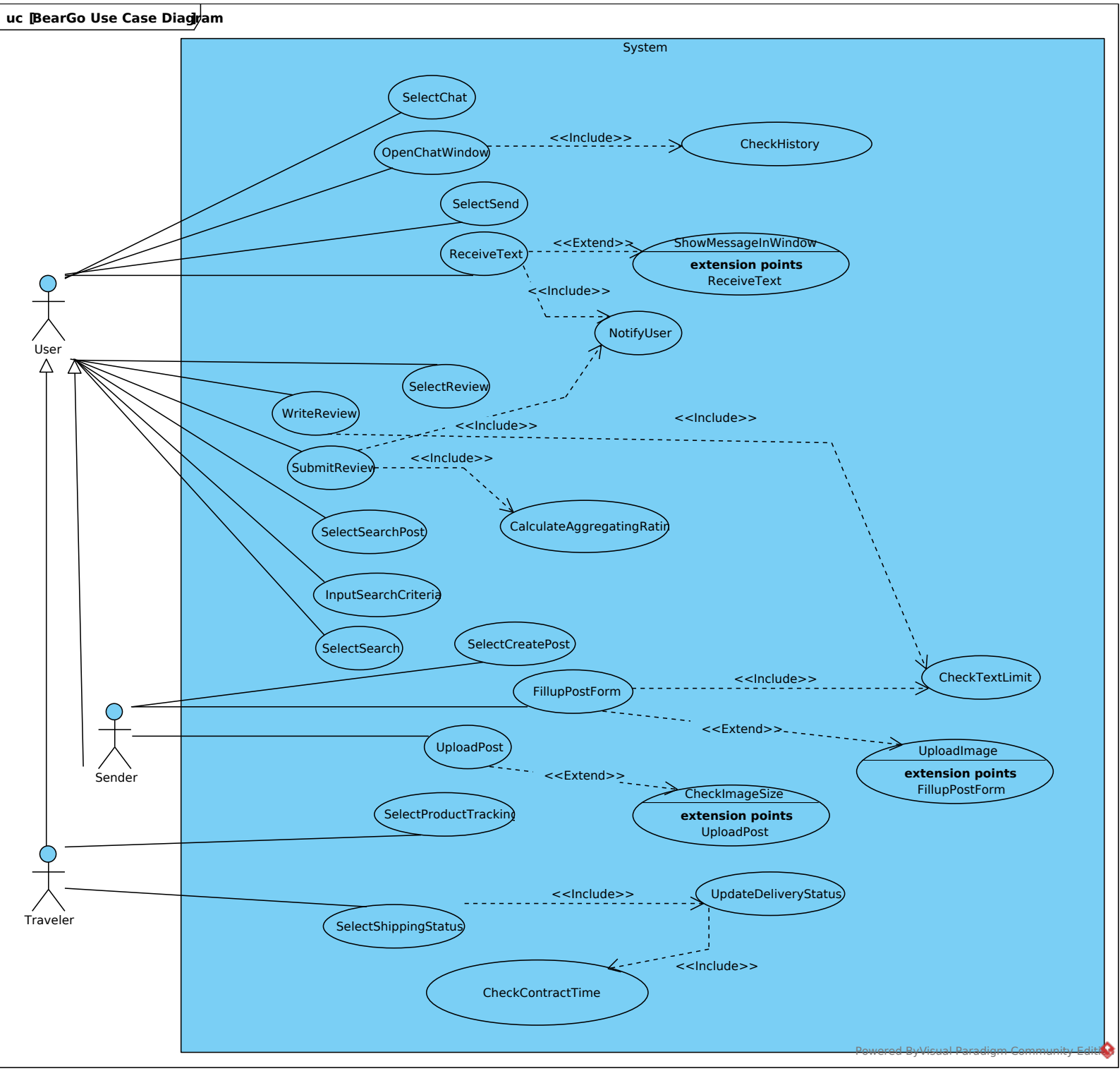


Use Case: Search Post
ID: UC05
Actors: 1. User
Preconditions: The user is authenticated.
Flow of Events: 1. The use case starts when a user selects “Search Post”. 2. The system will ask the user for search criteria. 3. The user enters the requested criteria. 4. The system searches for posts that match the users’ criteria. 5. If the system finds some matching posts then, 5.1 The system displays a list of matching posts.
Postconditions: The user will be redirected to the post page.
Alternative flow: 2.1 If the user selects “cancel” option, 2.1.1 The system will redirect the user to the homepage.
Postconditions: The user will see the homepage.
Alternative flow: 5.2 If the system cannot find any matching post 5.2.1 The system will tell that no post was found.
Postconditions: The user will see an empty list.

System Sequence Diagram for **Search Post**



Use Case Diagram


uc [BearGo Use Case Diagram]
System
SelectChat
OpenChatWindow
<<Include>>
CheckHistory
SelectSend
ReceiveText
<<Extend>>
ShowMessageInWindow
extension points
ReceiveText
<<Include>>
NotifyUser
User
SelectReview
WriteReview
<<Include>>
<<Include>>
SubmitReview
<<Include>>
CalculateAggregatingRatin
SelectSearchPost
InputSearchCriteria
SelectSearch
SelectCreatePost
CheckTextLimit
FillupPostForm
<<Include>>
<<Extend>>
Sender
UploadPost
UploadImage
extension points
FillupPostForm
<<Extend>>
CheckImageSize
extension points
UploadPost
SelectProductTracking
Traveler
<<Include>>
UpdateDeliveryStatus
SelectShippingStatus
<<Include>>
CheckContractTime
Powered By Visual Paradigm Community Edition

System Operations

System

+selectProductTracking(contractID)

+updateDeliveryStatus(status, contractID)

+closeContract(contractID)

+createPost()

+writePost(text)

+clickImageButton()

+uploadImage(img)

+uploadPost()

+cancelPost()

+selectReview()

+selectRating(rating)

+writeComment(text)

+submit()

+cancel()

+selectChat()

+enterName(name)

+selectMessage(userID, text)

+selectSearchPost()

+enterCriteria()

+searchPost(criteria)