BearGo
(Domain Model)

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 08, 2022

1 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.

2 High level overview of our procedures:

We went through the textbook and class lecture to find out the strategies. We followed conceptual category list to identify the concepts. We also prepared some sketch of class responsibility collaborator (CRC) which have been added in the pdf before the domain model. After that, we found out some candidate concepts and classified good and bad concepts for our project. Then we figured out multiplicities and associations between the concepts, and prepared the domain model using UML class diagram.

3 Approach to identify conceptual class:

According to the book, we know there exists two strategies which are -

- Using conceptual class category list.

- Identifying noun phrases.

Also, there is another method known as - class responsibility collaborator.

We used a conceptual class category list to identify candidate concepts. Our initial sketch for the conceptual class category list is given below:

Conceptual class category	Examples
physical or tangible objects	Product
roles of people	Traveler
	Sender
	Admin
calculation	OverallRating
rules and policies	Contract
records of agreements	Invoice
stream of activities	TweetFeed
	NewsFeed
interaction between users	Message
	Comment
	ReviewRating
records of communication	MessageHistory
events	Post
	PostObjection
	ProductTracking
documents	Report
personalization	Preferences

4 Candidate Concepts:

Product	Sender	Admin	Traveler
Contract	OverallRating	Invoice	TweetFeed
Message	Comment	ReviewAndRating	MessageHistory
Post	PostObjection	ProductTracking	Report
Preferences			

4.1 Then, we filtered the good concepts and bad concepts -

Good Concepts	Bad Concepts
Product	OvearallRating
Sender	TweetFeed
Admin	MessageHistory
Traveler	Report
Contract	Preferences
Invoice	
Message	
Comment	
ReviewRating	
Post	
PostObjection	
ProductTracking	

Class

Responsibility

Collaborator

ReviewRating	
Review ID Reviewer ID Reviewee ID Review text Rating Date Contract ID Add Rating	Contract User

Chat	
Chat ID Sender ID Receive ID Time Send Text	

ProductTracking	
Tracking ID Contract ID Updated By Updated Time Update Status	User Contract

| Post	
Post ID User ID Text Date Source Location Destination Location Start Date End Date Add Image	User

ReportUser	
Report ID Date Post ID User ID Report Post Send Notification	User Notification Post

Invoice	
Invoice ID Date Contract ID Traveler ID Sender ID Generate Invoice	User Contract

Contract	
Contract ID Post ID Traveler ID Sender ID Start Date End Date Tracking log Add tracking Info Generate Invoice	User Post ProductTracking

User	
User ID First Name Last Name Email Phone Dob Address Create Post Create Contract Chat with user	Post Contract Chat

Domain Model
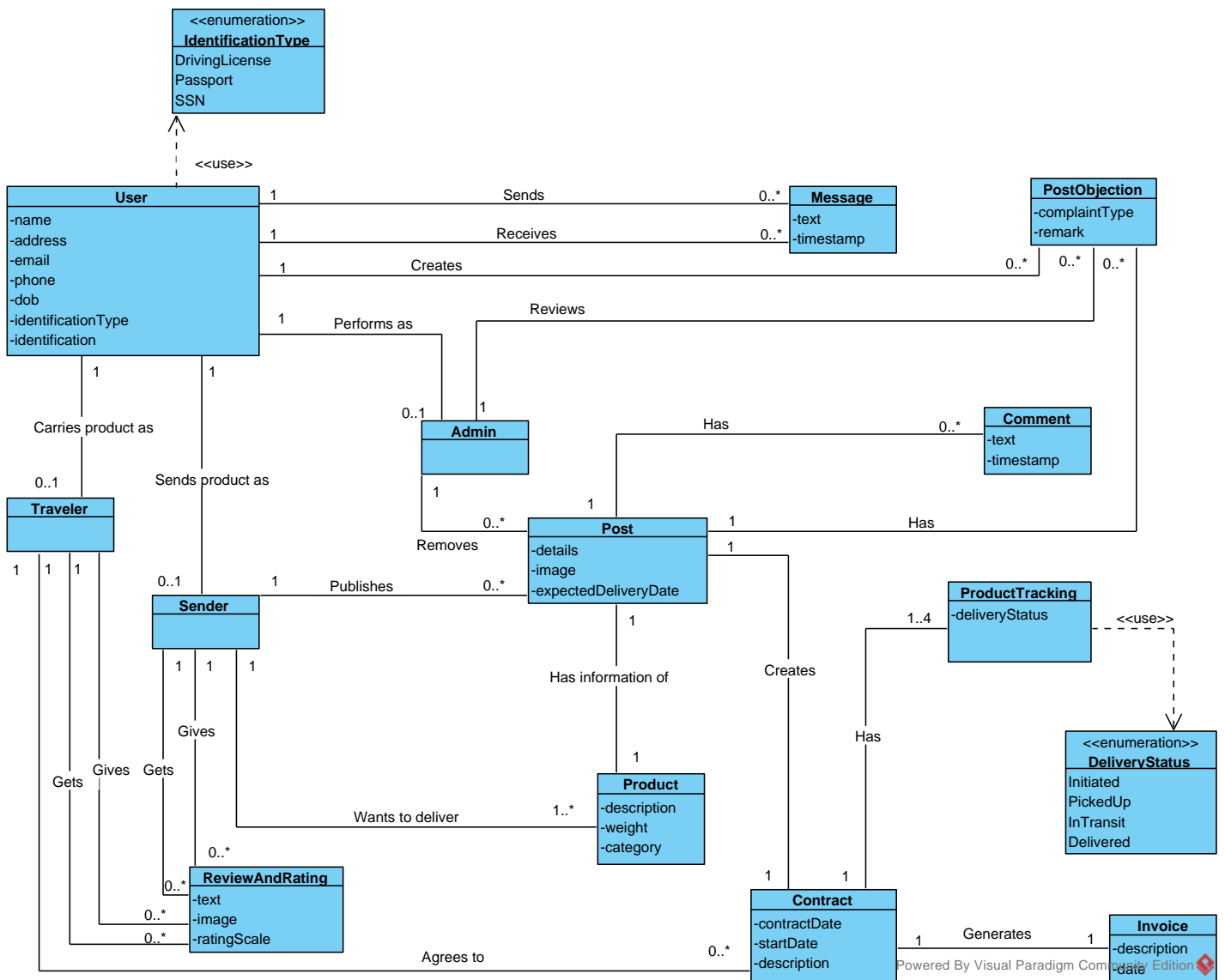
5 Model explanation:

1. **User -**

   - User is composed of sender, traveler and admin.
   - User uses identification type to prove identity.
   - User carries product as traveler.
   - User sends product as sender.
   - User can perform as admin.
   - User sends message.
   - User receives message.
   - User creates post objection.

2. **Sender -**

   - Sender publishes post.
   - Sender gets review and rating.
   - Sender gives review and rating.
   - Sender wants to deliver product.

3. **Traveler -**

   - Traveler agrees to contract.
   - Traveler gets review and rating.
   - Traveler gives review and rating.

4. **Admin -**

   - Admin removes post.
   - Admin reviews post objection.

5. **Post -**

   - Post has post objection.
   - Post has comment.
   - Post creates contract.
   - Post has information of product.

6. **Contract -**

   - Contract has product tracking info.
   - Contract generates invoice.

7. **Product tracking -**

   - Product tracking uses delivery status.