LOGISTICKS ✔

UML CLASS DIAGRAM

CS301P-2023-06 Keshav(003), Harsh(016) and Subhajeet(022)

SentBy
- senderPhoneNumber : String
- orderId : Integer
- orderTime : DateTime

ToBeReceivedBy
- receiverPhoneNumber : String
- orderId : Integer
- timeOfReceipt : DateTime
- receptionOTP : Integer

+ markAsReceived()
+ generateOTP()

OrderStatus
- orderId : Integer
- currentLocationId : Integer
- status : StatusDomain

+ updateStatus( )
+ updateLocationId()

Order
- id : Integer
- deliveryRate : Float
- weight : Float
- isFragile : Boolean
- isExpressDelivery : Boolean

+ placeOrder()

User
- phoneNumber : String
- name : String
- addressId : Integer
- isAdmin : Boolean

+ signUp()
+ signIn()

Address
- id : Integer
- houseNumber : String
- locality : String
- locationId : Integer

+ updateAddress()

Agent
- locationId : Integer
- vehicleNumber : String
- salary : Integer

+ updateSalary()
+ updateVehicleNumber()

ToBeDeliveredBy
- agentPhoneNumber : String
- orderId : Integer

+ markedAsDelivered()

Location
- id : Integer
- district : String
- city : String
- state : String

+ updateLocation()

Rate
- fromLocationId : Integer
- toLocationId : Integer
- baseRate : Float

+ calculateRate() : Float

[Lucid Chart Link](#)

NOTE : We have extensively used association classes in our design, for eg. in the SentBy, ToBeReceivedBy and the ToBeDeliveredBy relations. The primary key of all these tables is orderId itself leading to associations in both directions: Order.id references SentBy.orderId in some joins and SentBy.orderId references Order.id in other joins.

Order

Primary key: id

Foreign key: id references
SentBy.orderId,
ToBeReceivedBy.orderId,
ToBeDeliveredBy.orderId,
OrderStatus.orderId

id, deliveryRate, weight, isFragile, isExpressDelivery NOT NULL

OrderStatus

Primary key: orderId

Foreign key: orderId references
Order.id

currentLocationId references
Location.id

orderId NOT NULL

SentBy

Primary key: orderId

Foreign key: orderId references
Order.id

senderPhoneNumber references
User.phoneNumber

orderId, senderPhoneNumber NOT NULL

ToBeReceivedBy

Primary key: orderId

Foreign key: orderId references
Order.id

receiverPhoneNumber references
User.phoneNumber

orderId, receiverPhoneNumber NOT NULL

ToBeDeliveredBy

Primary key: orderId

Foreign key: orderId references
Order.id

agentPhoneNumber references
User.phoneNumber

orderId, agentPhoneNumber NOT NULL

User

Primary key: phoneNumber

Foreign key: addressId references
Address.id

name, phoneNumber, addressId, isAdmin NOT NULL

Agent (Inherited from User)

Primary key: phoneNumber

Foreign key: addressId
references Address.id

locationId references
Location.id

name, phoneNumber, addressId,
isAdmin, locationId NOT NULL

Address

Primary key: id

Foreign key: locationId
references Location.id

id, locationId NOT NULL

Location

Primary key: id

id, city, state, district NOT NULL

Rate

Primary key: { fromLocationId,
toLocationId }

Foreign key: fromLocationId
references Location.id

toLocationId references
Location.id

fromLocationId, toLocationId
NOT NULL

Relational Schema:

Order relation:

id	DeliveryRate	weight	isFragile	isExpressDelivery

OrderStatus relation:

orderId	currentLocationId	status

SentBy relation:

orderId	senderphoneNumber	orderTime

ToBeReceivedBy relation:

orderId	receiverPhoneNumber	timeofReceipt	ReceptionOTP

ToBeDeliveredBy relation:

orderId	agentPhoneNumber

User relation:

phoneNumber	name	addressId	isAdmin

Agent relation:

locationId	addressId	name	vehicleNumber	phoneNumber	isAdmin	salary

Address relation:

id	houseNumber	Locality	locationId

Location relation:

id	district	city	state

Rate relation:

fromLocationId	toLocationId	baseRate