Delivery Service Application

Phase II:
System Architecture
and Sequence Diagram

Presented to Dr. Joumana Dargham

Sarah Amri 40210908 - Leader
Ayesha Mahmood 40189093 - Member
Adib Akkari 40216815- Member
Safaa Kentaoui 40228872 - Member
Amine Arrachid 40263585 - Member
Younes Bouhaba 40205816 - Member

Monday, October 21st, 2024

Table of Contents

I - Summary of the project

Our delivery service system aims to simplify the delivery request process, offering various features. The core functionality includes secure payment options and a 24/7 ChatBot for assistance with service inquiries or package status updates. Another key feature is the real-time tracking system, which provides transparency throughout the delivery process. Our platform also supports interactive features, like pinpointing pick-up and drop-off locations on a map for accuracy.

The system offers a quotation service that calculates delivery costs based on the package details and distance, providing customers with different delivery options. After a delivery is completed, customers can rate the service and the delivery person, contributing to an overall rating. The chatbot not only provides assistance but can also escalate unresolved issues to a human agent.

While designing our system, we kept in mind that it should handle increased demand without compromising performance. The system's architecture is organized in the following layers to promote modularity and code reusability: the UI layer (user interaction and app presentation), the application layer (core app services), the domain layer (core business domain logic), the business infrastructure layer (interaction with external systems), technical services (technical support utilities) and the foundation layer (the database).

II - System Architecture

II.1 UI Layer

This layer primarily focuses on user interaction, containing packages related to the user interface..
Classes:

- **UserInterface:** Manages the user's interaction with the delivery system (React).
- **DeliveryRequestPage:** Allows users to enter details for delivery requests (React).
- **OrderTrackingPage:** Enables users to track their delivery status (React).
- **PaymentPage:** Facilitates secure payment processing (React).
- **ConfirmationPage:** Provides delivery order confirmations and tracking information (React).
- **ChatbotInterface:** Allows users to interact with the chatbot for assistance (React).

II.2 Application Layer

This layer contains the core application logic and services.
Classes:

- **DeliveryService:** Manages the delivery request process, including validating details and checking availability (Node.js, Express.js).
- **OrderTrackingService:** Provides information about the status of deliveries using tracking IDs (Node.js, Express.js).
- **PaymentService:** Handles payment transactions securely (Node.js, Express.js).
- **NotificationService:** Sends automated notifications to users regarding order status and updates (Node.js, Express.js).
- **ChatbotService:** Manages interactions with the chatbot and provides responses to user inquiries (Node.js, Express.js).

II.3 Domain Layer

The core business domain logic resides in this layer.
Classes:

- **Delivery:** Represents delivery orders, including sender and recipient details (Mongoose models).
- **Order:** Stores information about the delivery, including weight, dimensions, and special instructions (Mongoose models).
- **PaymentTransaction:** Represents payment details and transaction records (Mongoose models).

II.4 Business Infrastructure Layer

This layer contains classes that interact with external systems and services.

Classes:

- **PaymentGateway:** Interacts with payment processing services for secure transactions (e.g., Stripe, PayPal).
- **NotificationGateway:** Communicates with external notification services (e.g., email via SendGrid, SMS via Twilio).

II.5 Technical Services

This layer provides technical support functions and utilities.
Everything in this layer is optional and subject to addition or deletion upon implementation
Classes:

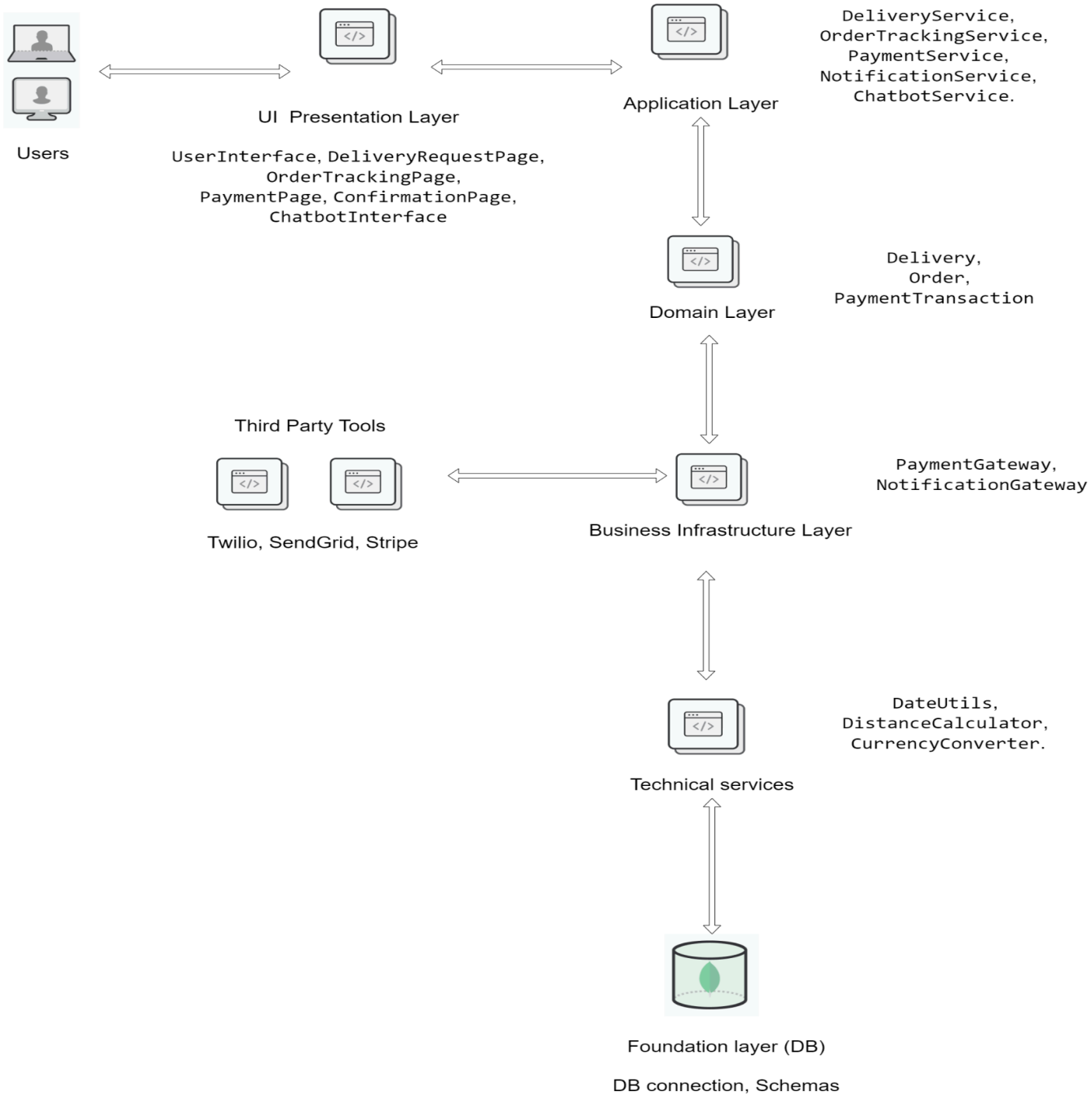
- **DateUtils:** Provides date-related utilities for scheduling and tracking (JavaScript).
- **DistanceCalculator:** Offers functions for calculating distances between pick-up and drop-off locations (JavaScript).
- **CurrencyConverter:** Provides currency conversion functions for international delivery services (JavaScript).

II.6 Foundation Layer

This layer includes foundational elements and domain objects that support the application.
Classes:

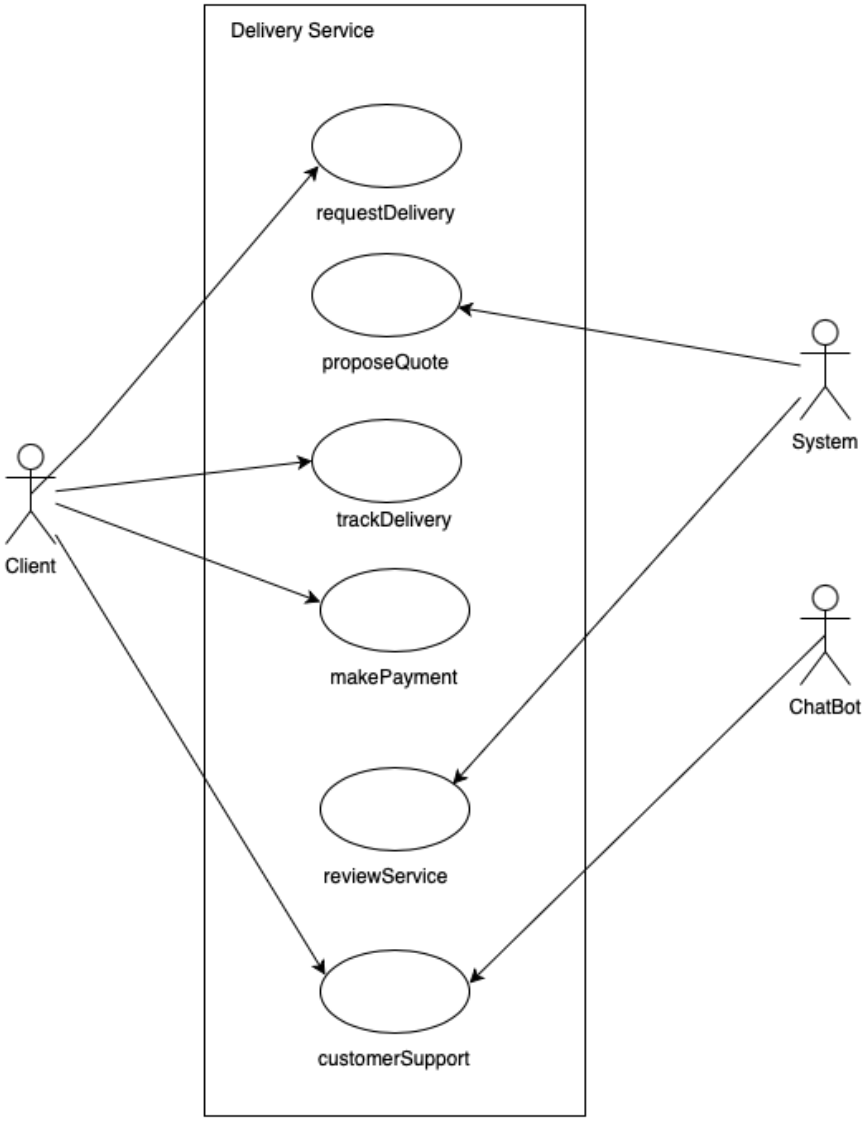
- **MongooseConnection:** Handles the connection to the MongoDB database (Mongoose).
- **SchemaDefinitions:** Contains schema definitions for MongoDB collections (Mongoose models).

II.7 System Architecture Diagram


Users
UI Presentation Layer
UserInterface, DeliveryRequestPage,
OrderTrackingPage,
PaymentPage, ConfirmationPage,
ChatbotInterface
Application Layer
DeliveryService,
OrderTrackingService,
PaymentService,
NotificationService,
ChatbotService.
Domain Layer
Delivery,
Order,
PaymentTransaction
Third Party Tools
Twilio, SendGrid, Stripe
Business Infrastructure Layer
PaymentGateway,
NotificationGateway
Technical services
DateUtils,
DistanceCalculator,
CurrencyConverter.
Foundation layer (DB)
DB connection, Schemas

III - Use Cases

III.1 Use Case Diagram



III.2.1 Use Case 1: Requesting the delivery (entering delivery details and receiving delivery identification).

ID:	UC-01
Title:	Requesting a Delivery
Description:	This use case allows the customer to place a delivery order by providing the relevant delivery details. Upon submission, the system generates a unique tracking ID to the order for tracking and future reference.
Primary Actor:	Customer
Preconditions:	☐ The system is operational and connected to the database
Postconditions	☐ The delivery request is recorded in the system ☐ A unique tracking ID is assigned to the delivery and shared to the customer
Inputs:	➔ Name of sender ➔ Email of sender ➔ Name of recipient ➔ Pick-up location ➔ Drop-off location ➔ Weight of the package ➔ Dimensions of the package ➔ Special instructions (optional) ➔ Payment information
Outputs:	➔ Order confirmation ➔ Unique tracking ID
Main Success Scenario:	1. Customer clicks the “Request Delivery” button. 2. Customers enter relevant details such as pick-up and drop-off locations, package weight, package dimensions and optionally special instructions. 3. System validates the input data to ensure all fields are correctly filled. 4. Customer is directed to the payment process. 5. Customer is redirected to this process upon successful completion of payment process 6. System processes the order request and assigns a unique tracking ID to it. 7. System confirms the order and sends the tracking ID to the sender’s email.

III.2.2 Use Case 2: Proposal of a quotation for the delivery service (details of the request)

ID:	UC-02
Title:	Proposal of a Quotation for the Delivery Service
Description:	This use case allows the user to request a quotation for the delivery service by providing the relevant delivery details. The system calculates the estimated cost and returns a quotation based on the delivery distance, package weight and other relevant details.
Primary Actor:	Customer
Preconditions:	☐ Customer has filled in all the necessary delivery details ☐ System is connected to the database of rates and services
Postconditions	☐ System generates a quotation ☐ Customer can pick delivery speed and proceed to placing a delivery order
Inputs:	➔ Pick-up location ➔ Drop-off location ➔ Weight of the package ➔ Dimensions of the package
Outputs:	➔ Estimated delivery cost ➔ Estimated delivery time
Main Success Scenario:	8. Customer clicks on the “Request a Quote” button. 9. Customers enter relevant details such as pick-up and drop-off locations, package weight and package dimensions. 10. System validates the input data to ensure all fields are correctly filled. 11. System calculates the estimated delivery cost options considering the Customer input and the available delivery options. 12. System generates quotations including delivery cost and time. 13. System displays a few quotations varying in cost and delivery time to the Customer. 14. Customers can pick one of the options and proceed to place a delivery order.

III.2.3 Use Case 3: Tracking the delivery (initiate tracking by Id and getting the information about the delivery: time to arrival, contact person info).

ID:	UC-03
Title:	Tracking a Delivery
Description:	This use case allows the customer to track the status of their delivery by entering the unique tracking ID assigned to the order and provided to the customer through email.
Primary Actor:	Customer
Preconditions:	☐ Customer has a valid tracking ID ☐ Database is up-to-date with delivery status information
Postconditions	☐ .The customer receives live location of their delivery
Inputs:	➔ Unique tracking ID
Outputs:	➔ Live location of the package
Main Success Scenario:	1. Customer clicks on the “Track Package” button. 2. Customer enters tracking ID. 3. System validates tracking ID and retrieves current status of the delivery. 4. System displays the current location to the Customer.

III.2.4 Use Case 4: Payment (identification of the service with due payment and payment).

ID:	UC-04
Title:	Payment
Description:	This use case allows the customer to complete payment for an order they have placed after requesting a delivery service. The system processes the payment and provides confirmation to the customer, updating the order status accordingly.
Primary Actor:	Customer
Preconditions:	☐ The customer has requested a delivery service, entering all the relevant details ☐ The customer has been directed to the payment portal ☐ The customer has sufficient funds and an accepted payment method available (ex: credit card, PayPal).
Postconditions	☐ The payment is successfully processed and the customer receives a confirmation. ☐ The system updates the delivery status to "Payment Complete" and stores the transaction details. ☐ A receipt is generated and sent to the customer.
Inputs:	➔ "Make a Payment Request" process ➔ Payment details (ex: card details, payment method).
Outputs:	➔ Payment confirmation ➔ Receipt for transaction
Main Success Scenario:	1. System directs Customer to the payment screen after delivery details are confirmed. 2. Customer selects a payment method and enters required payment information. 3. System validates the payment details (ex: checking card validity and funds availability). 4. System processes payment transactions. 5. System confirms the successful payment and generates a receipt. 6. System updates the payment status to "Payment Complete" and stores the transaction details. 7. Customer is redirected to the Request Delivery process.

III.2.5 Use Case 5: Review of service (details of the usage of the service, comments on the delivery person and other details related to the offered service).

ID:	UC-05
Title:	Reviewing the delivery service
Description:	This use case allows the user to review the service after the completion of a delivery. Customers can provide feedback by rating the service and the delivery person on a scale of 1 to 5 stars and include any additional comments on specific aspects such as timeliness, condition of the delivered goods, the delivery person and the overall experience. The system will then store the reviews, and calculate an updated average rating.
Primary Actor:	Customer
Preconditions:	☐ The delivery has been successfully completed ☐ The system is operational, and the review feature is available.
Postconditions	☐ The customer's review is stored in the system. ☐ The system updates and calculates the average rating alongside the total number of reviews of the service and the delivery person.
Inputs:	➔ Tracking ID ➔ Service rating (1-5 stars) ➔ Optional: written comments about the service ➔ Delivery person rating (1-5 stars) ➔ Optional: written comments about the delivery person
Outputs:	➔ Confirmation of the submitted review ➔ Updated average star rating (overall service and the delivery person) ➔ Updated number of reviews (overall service and the delivery person)
Main Success Scenario:	1. The customer enters the tracking id to confirm the item has been delivered. 2. The system announces that the item has been delivered and prompts the customer to leave a star rating and an optional comment. 3. The customer enters their rating and comments (optional) and submits the review. 4. The system stores the review in the database. 5. The system calculates the updated average ratings for both the overall service and delivery person. 6. The customer receives confirmation that the review has been successfully submitted.

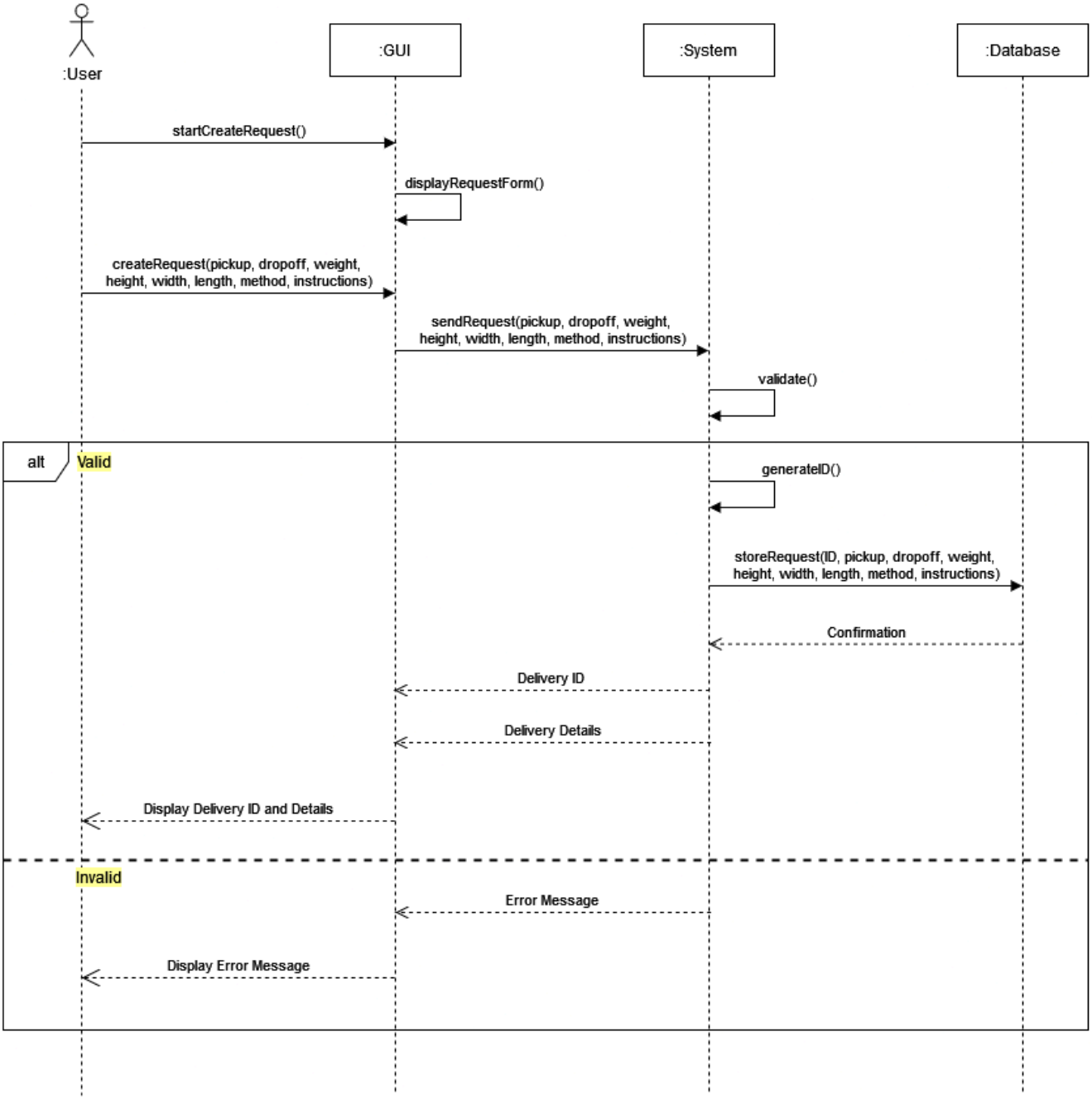
III.2.6 Use Case 6: Customer support (for general or specific requests)

ID:	UC-06
Title:	Chatbot Assistance for Delivery Help
Description:	This use case enables customers to receive assistance regarding their delivery inquiries through a chatbot. The chatbot can answer frequently asked questions, provide status updates, and guide users through the delivery process.
Primary Actor:	Customer
Preconditions:	☐ The chatbot system is operational and integrated with the delivery service database.
Postconditions	☐ The customer receives accurate and timely assistance regarding their delivery inquiries.
Inputs:	➔ User query (text input) ➔ previous interactions
Outputs:	➔ Chatbot response (information) ➔ Confirmation of information provided (if applicable)
Main Success Scenario:	1. Customer initiates a chat session with the chatbot on the delivery service platform. 7. Customer types in a query related to their delivery (e.g., "What is the status of my delivery?"). 8. The system processes the input and identifies keywords and intents. 9. If the query relates to a tracking ID, the customer provides the ID. 10. The chatbot retrieves relevant information from the database (e.g., order status, estimated delivery time). 11. The chatbot responds to the customer with the requested information or guidance. 12. If the inquiry cannot be resolved by the chatbot, it escalates the request to a human agent. 13. Customers receive helpful responses or further instructions as needed.

IV - Sequence Diagrams

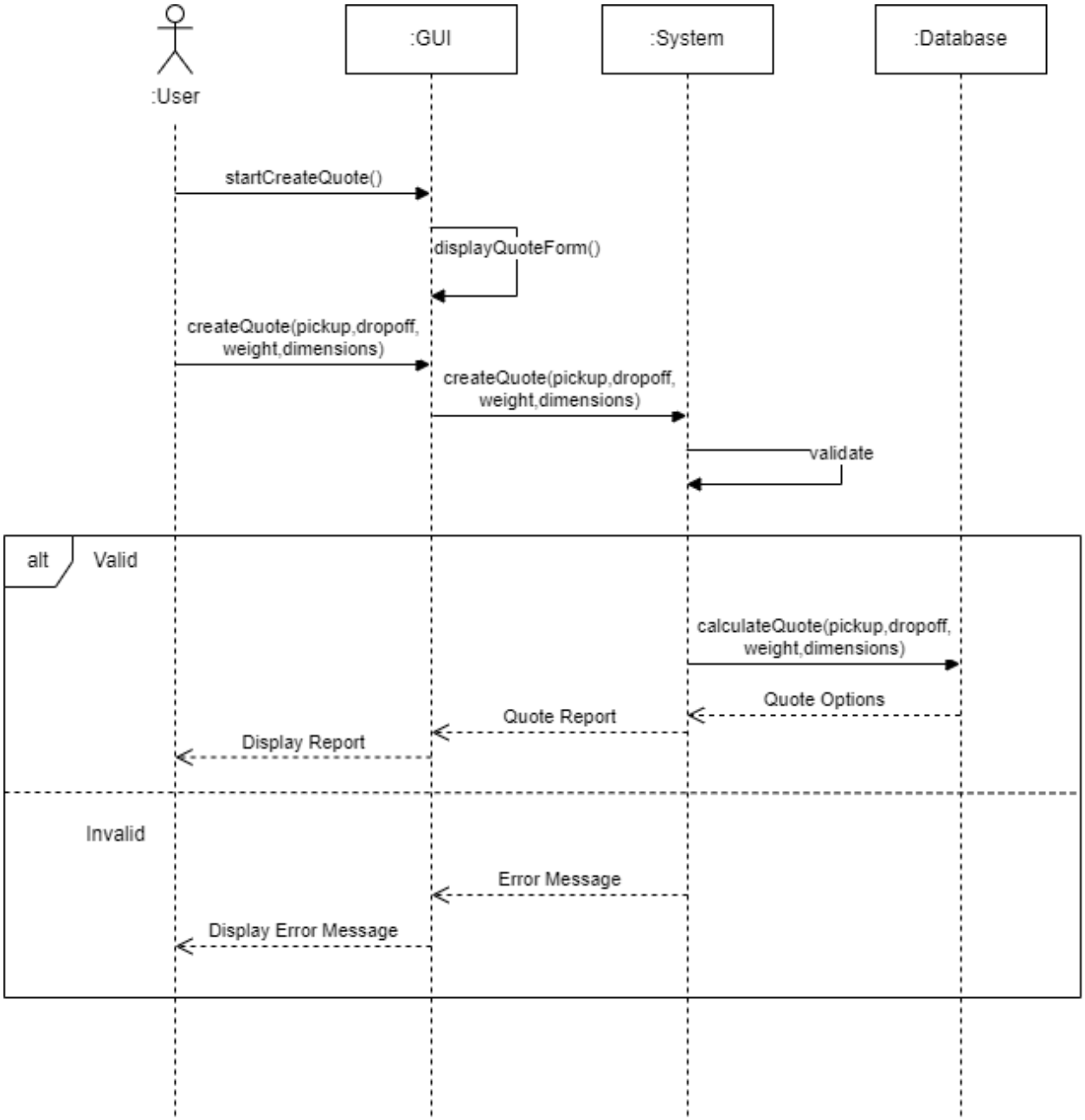
IV.1 Sequence Diagram 1

Requesting the delivery (entering delivery details and receiving delivery identification)



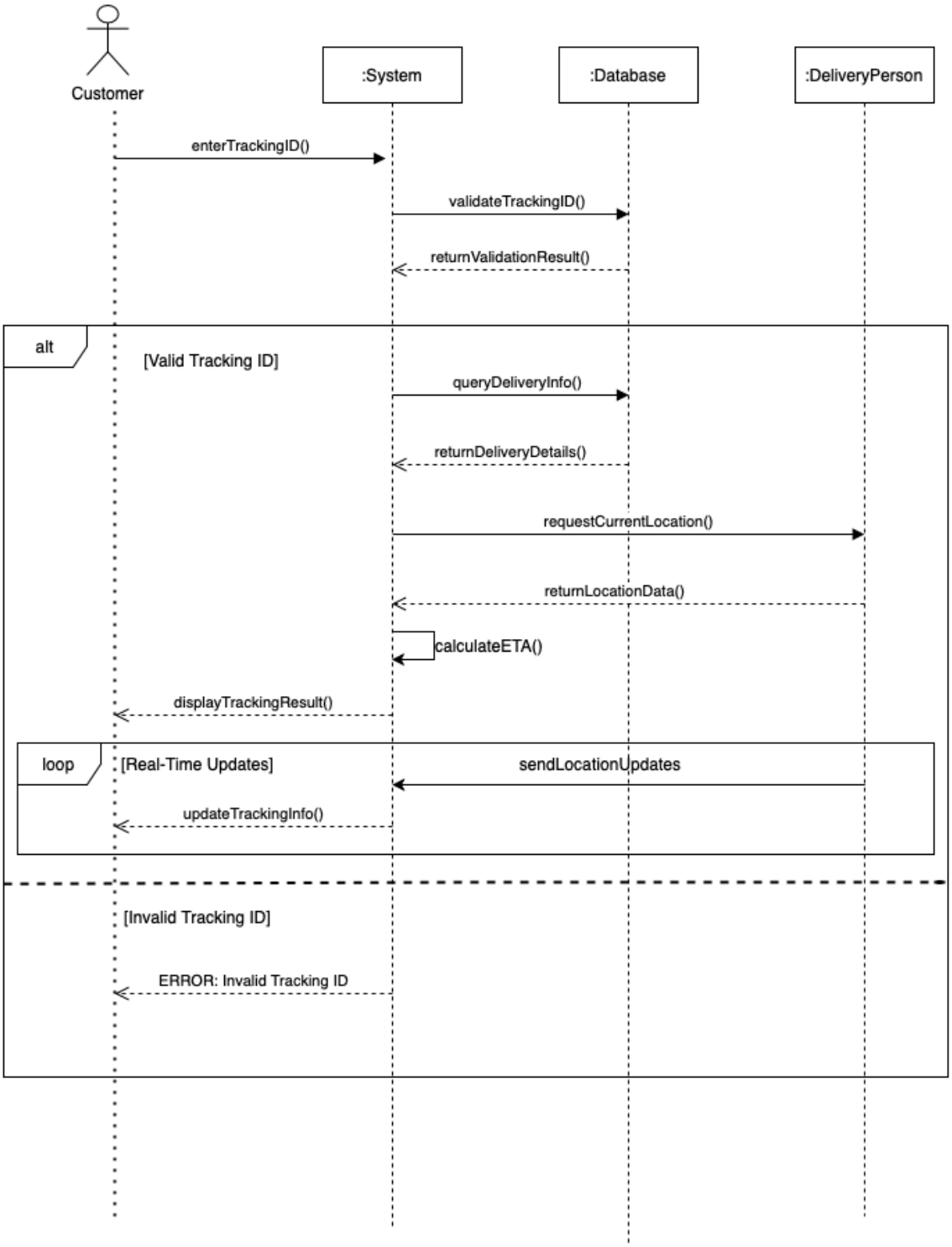
IV.2 Sequence Diagram 2

Proposal of a quotation for the delivery service (details of the request)



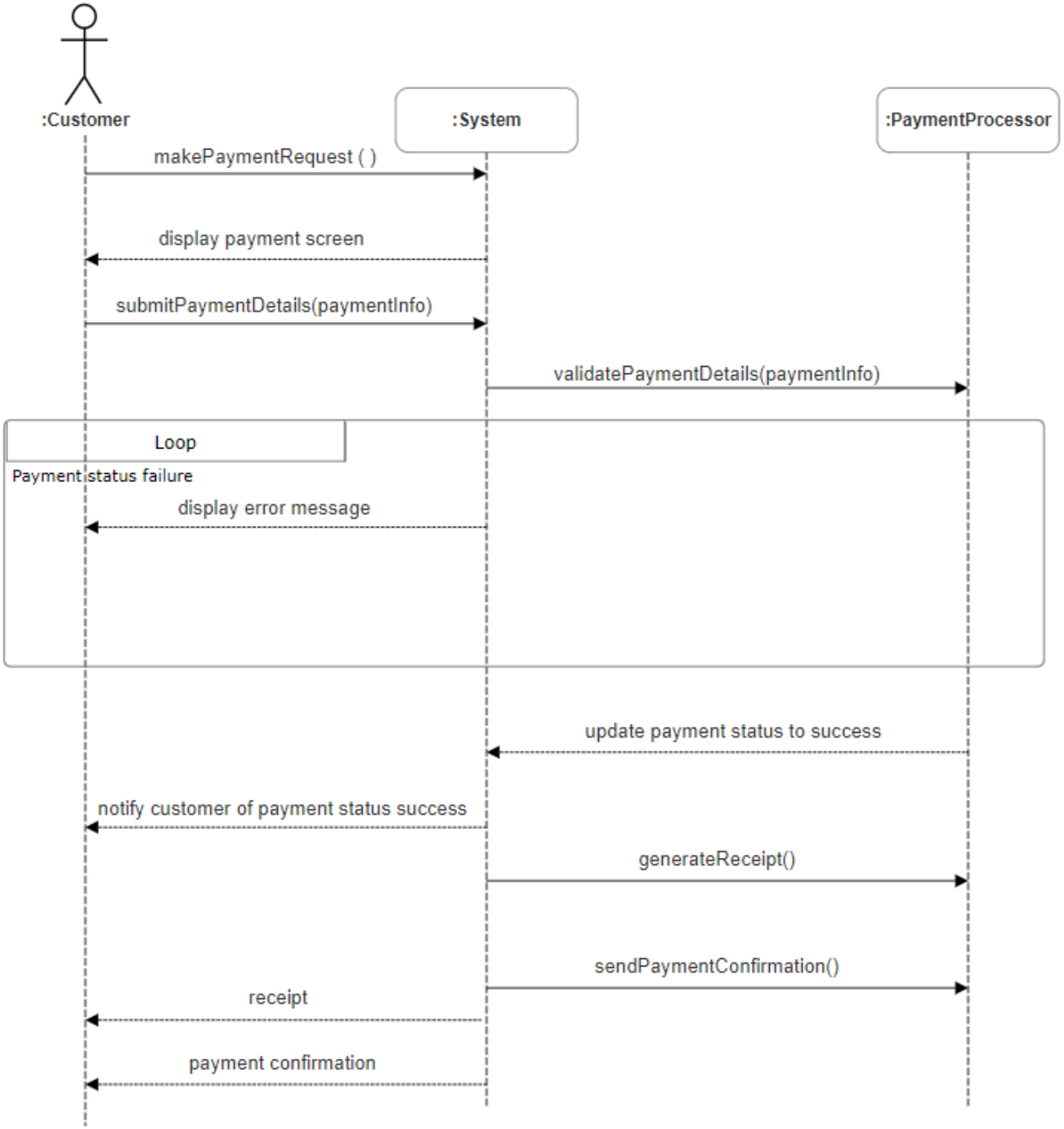
IV.3 Sequence Diagram 3

Tracking the delivery (initiate tracking by Id and getting the information about the delivery: time to arrival, contact person info). It is initiated when a customer clicks on “Track Package”.



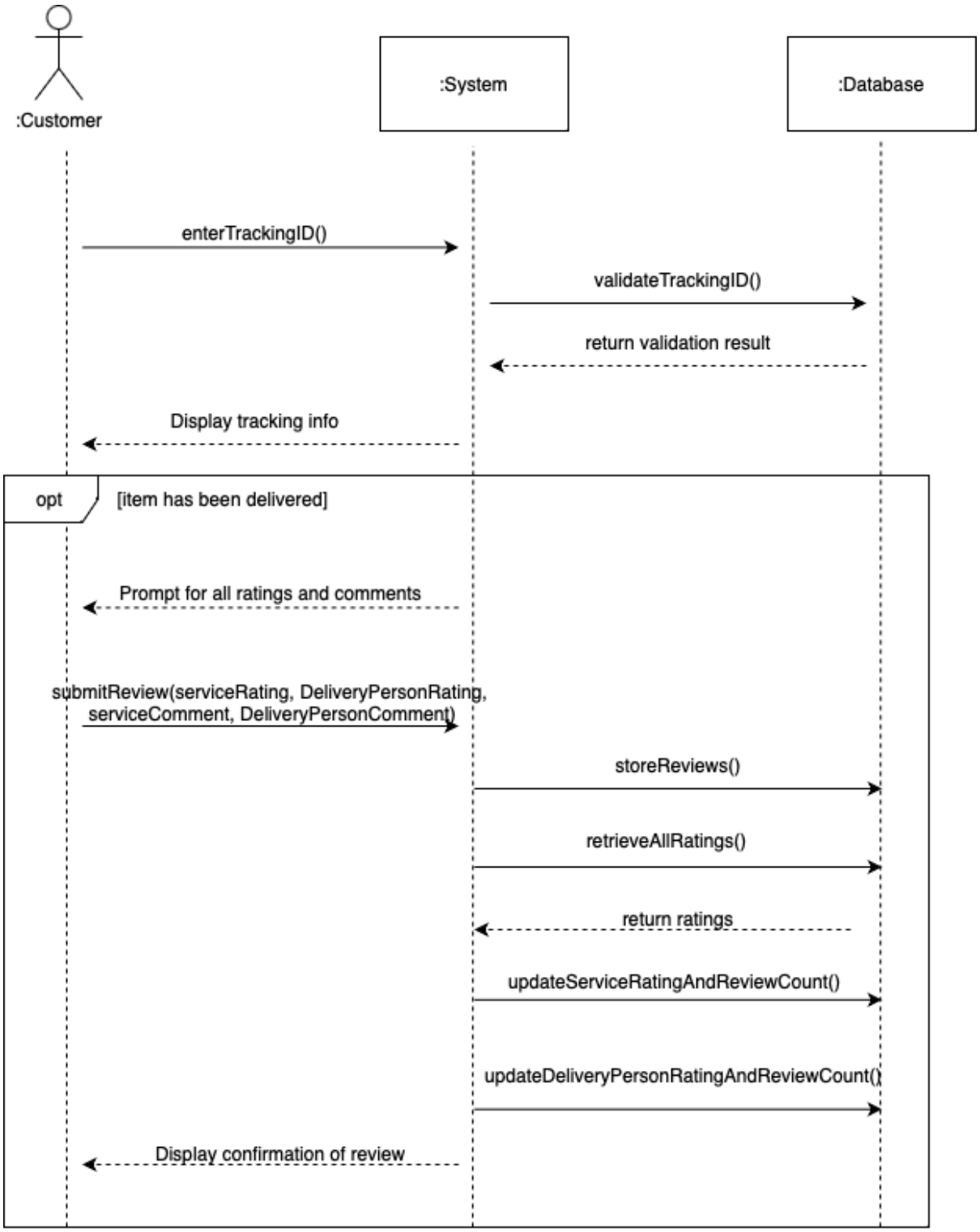
IV.4 Sequence Diagram 4

Initiation of payment functionality in real life: The payment functionality is initiated when the customer completes their delivery request and clicks the "Pay" button to enter their payment information, leading to the processing of the payment.



IV.5 Sequence Diagram 5

Review of service:The review functionality is initiated when the user enters the tracking id and the item has been delivered.



IV.6 Sequence Diagram 6

Customer support (for general or specific requests): It is initiated when the customer clicks on the ChatBot.