**Use Case Detailed Use Case Descriptions :**

### **Use Case 1: Sending Money to Friends/Family**

**Goal**: To allow users to send money to their friends or family quickly and securely.

**Actors**:

* **User (Customer)**: The person sending the money.
* **Payment Gateway (Stripe/PayPal)**: The third-party service responsible for processing the payment.
* **Recipient (Friend/Family)**: The person receiving the money.

#### **Preconditions:**

* The user is logged into their account on the app.
* The user has at least one verified payment method (e.g., credit card, bank account) linked to their account.
* The recipient is either already registered on the app, or they can be invited to join the platform by the user.

#### **Basic Flow:**

1. **User Opens the App and Navigates to the "Send Money" Page**:
   1. The user launches the app and selects the "Send Money" option from the main menu or home page.
2. **User Enters Recipient's Details**:
   1. The user can either:
      1. Select a recipient from their contacts (the app syncs with the phone’s contact list), or
      2. Manually input the recipient’s email address or phone number.
3. **User Specifies the Payment Amount**:
   1. The user enters the amount of money they want to send.
4. **User Selects the Payment Method**:
   1. The app presents the available payment methods linked to the user’s account (e.g., credit card, bank account).
   2. The user selects their preferred payment method for the transaction.
5. **User Reviews Transaction Details**:
   1. The app displays a summary of the transaction, including:
      1. Recipient’s name or contact details.
      2. Amount to be sent.
      3. The selected payment method.
      4. Any associated transaction fees (if applicable).
6. **User Confirms the Transaction**:
   1. After reviewing the details, the user confirms the payment by clicking the "Confirm" button.
7. **Payment is Processed**:
   1. The app securely communicates with the payment gateway (e.g., Stripe/PayPal) to process the transaction.
   2. The funds are transferred from the sender’s account to the recipient’s account.
8. **User Receives Confirmation**:
   1. The user is notified that the payment was successful via a confirmation message in the app.
   2. A notification is also sent to the recipient about the incoming funds.

#### **Alternative Flow:**

1. **Recipient Not Found**:
   1. If the recipient is not registered with the app, the system prompts the user to send an invitation to the recipient, encouraging them to join the platform.
   2. Once the recipient joins, the process can continue from the beginning, or the user can choose to send money to another recipient.

#### **Postconditions:**

* The money is successfully transferred to the recipient’s account.
* Both the sender and the recipient receive notifications confirming the transaction.
* The transaction is recorded in the sender’s transaction history.

### **Use Case 2: Booking Go-Bus Tickets**

**Goal**: Enable users to book Go-Bus tickets for intercity or local travel.

**Actors**:

* **User**: The person booking the ticket.
* **App Backend**: The system responsible for managing the booking process.
* **Go-Bus Operator Systems/APIs**: External systems that provide bus schedule, availability, and payment processing services.

#### **Preconditions:**

* The user must have a valid account on the app.
* The app must be integrated with external bus operator systems or APIs for fetching available buses, schedules, and processing payments.

#### **Basic Flow:**

1. **User Navigates to the "Go-Bus Tickets" Section**:
   1. The user opens the app and navigates to the "Go-Bus Tickets" section from the main menu.
2. **User Enters Travel Details**:
   1. The user is prompted to enter the following travel details:
      1. **Source**: The departure location.
      2. **Destination**: The arrival location.
      3. **Travel Date**: The date of travel.
      4. **Number of Passengers**: The number of people traveling.
3. **System Fetches Available Buses**:
   1. The app communicates with the bus operator systems to fetch available buses based on the user’s travel details.
   2. A list of available buses, their fares, and schedules is displayed.
4. **User Selects a Bus**:
   1. The user selects a bus from the available options based on preferences such as fare, schedule, and seating availability.
   2. The app then displays the seating options for the selected bus.
5. **User Chooses a Seat**:
   1. The user selects a preferred seat from the available options for the chosen bus.
6. **User Confirms the Booking**:
   1. The user reviews the bus details, seat selection, and total fare.
   2. After confirming the details, the user proceeds to the payment screen.
7. **Payment Processing**:
   1. The user selects a payment method and enters the payment details.
   2. The app processes the payment via integrated payment gateways (e.g., Stripe, PayPal).
   3. Once the payment is successful, the booking is confirmed.
8. **E-Ticket Generation**:
   1. The system generates a digital ticket for the user and sends it to them via in-app notification and email.
9. **User Receives the Ticket**:
   1. The user receives the e-ticket along with travel details like the bus number, departure time, and seat number.

#### **Alternative Flow:**

1. **No Buses Available**:
   1. If no buses are available for the selected route on the chosen date, the system notifies the user and suggests alternative dates or other bus operators.
2. **Payment Failure**:
   1. If the payment fails for any reason (e.g., insufficient funds, invalid payment details), the system notifies the user and offers the option to retry the payment or use another payment method.

#### **Postconditions:**

* The user successfully books the Go-Bus ticket, and the booking details are saved in the user’s booking history.
* The user receives a digital copy of the ticket for their reference.