### Process of retrieving order details

The initial GET request will be sent to the buyer service through ESB. The order id is sent as a parameter and the order is retrieved using the order id. The retrieved order will be sent back to the client through the ESB. If an error is occurred then a response is sent back to the client with 422 status code.

### Process of payment through credit card

The initial POST request will be sent to the credit card payment gateway service through ESB if the request has a parameter “payment\_type” with its value “card”. The request body will be validated by the “validatePaymentRequest” method to check if it has order id, order hash code, transfer amount, credit card number, CVC number and a card holder name.

If the validation fail an error response with the respective message will be sent back to the client via ESB. Else, the order hash will be validated. To validate the hash sent from the request, a new hash will be created by using the order secret string constant along with the order id and transfer amount.

If the newly created hash string does not match with the hash code sent by the request that means the payment details does not match with the order details therefore an error response will be sent back to the client.

If hash matches credit card object will be retrieved from the database based on the credit card number. If there is no credit card object then an error response will be sent back. Else then the credit card details will be validated.

To validate credit card details “matchCardDetails” function will be used. Credit card number, CVC number and the card holder name must be matched with the request data and also the remaining balance of the account must be greater than the transfer amount.

If the validation fail an error response with the respective message will be sent back to the client via ESB. Else the transaction will be made updating the credit card balance. If the transaction failed then an error response will be sent back. Else a success message will be sent back via ESB and the buyer service will be notified that the order was completed.

Notify server method will send a POST request to the buyer service with data such as order id, transfer amount and payment hash code. Payment hash code is generated using payment secret (string constant in the environment configuration), order id and the transfer amount. This hash code is useful to validate that the payment was completed through the payment gateway.

### Process of payment through mobile payment gateway

The initial POST request will be sent to the mobile payment gateway service through ESB if the request has a parameter “payment\_type” with its value “mobile”. The request body will be validated by the “validatePaymentRequest” method to check if it has order id, order hash code, transfer amount, mobile number and mobile pin number.

If the validation fail an error response with the respective message will be sent back to the client via ESB. Else, the order hash will be validated. To validate the hash sent from the request, a new hash will be created by using the order secret string constant along with the order id and transfer amount. If the newly created hash string does not match with the hash code sent by the request that means the payment details does not match with the order details therefore an error response will be sent back to the client. If hash matches then the mobile payment details will be validated.

To validate the mobile payment, mobile object will be retrieved from the database based on the mobile number. If the mobile object does not exist then an error response will be sent. Else, mobile pin will be validated. If the validation fails then an error response will be sent back. Else, the mobile balance will be updated (transfer amount will be added) and saved. If the payment fail then an error response will be sent back. Else, a success message will be sent back via ESB and the buyer service will be notified that the order was completed.

Notify server method will send a POST request to the buyer service with data such as order id, transfer amount and payment hash code. Payment hash code is generated using payment secret (string constant in the environment configuration), order id and the transfer amount. This hash code is useful to validate that the payment was completed through the payment gateway.

### Process of requesting mobile pin number

The initial POST request will be sent to the mobile payment service through ESB. The mobile number is sent as a parameter. Mobile number will be validated from the below function and if any errors were found an error response will be sent back.

After the validation was completed successfully then a four digit pin number will be created.

The mobile object will be retrieved based on the mobile number from the database and the pin number will be updated and saved. If there is no object in the database then a new mobile object will be created and the pin number will be saved.

If an error is caught while saving an error response will be sent back to the client through the ESB. Else, a success message will be sent back and a SMS will be sent to the number with the pin number. SMS will be sent using Twilio.

### Process of listening to payment notifications

After completing the payment by payment gateways (credit card / mobile ) it will notify the buyer service. A POST request will be sent through the ESB to the buyer service with order id, transfer amount and the payment hash. Order will be retrieved from the order id and if there is an order then the payment hash needs to be validated.

To validate the payment hash, a new md5 hash will be created from the payment secret (Environment variable), order id and order payment value. Then if the payment hash from the request and the newly created hash from the order object matches, the payment is validated.

If the payment was not validated, “notifyPaymentFailed” method will be called and an unsuccessful email and SMS will be sent based on the order buyer information. Else, “notifyPaymentSuccessful” method will be called and a successful email and SMS will be sent based on the order buyer information.

### Process of placing cash on delivery order.

The initial POST request will be sent to the buyer service through ESB if the request has a parameter “payment\_type” with its value “COD”. The request is required to send the order ID. Then the order object will be retrieved. Order details including the buyer and delivery information must be validated before placing the order. As in above functions “ValidateOrderDetails” function is used to validate the request. If any errors are found then an error response with 422 status code will be sent.

Then update the order details setting payment status to “pending”, delivery status to “packing”, order status to “validating” and payment type to “COD”. Setting the order status to validating will avoid users from updating its details. Order details can only be updated if the order status is “pending”. This order object is saved and if it fails to save an error message is sent back to the client through ESB and error SMS and an error email will be sent to the user. Else, the user will be notified as the payment completed and a success SMS will be sent to the order’s buyer phone number and an email will be sent to the order’s buyer email.

(please refer “Process of listening to payment notifications” for the screenshots of email and SMS and the code)