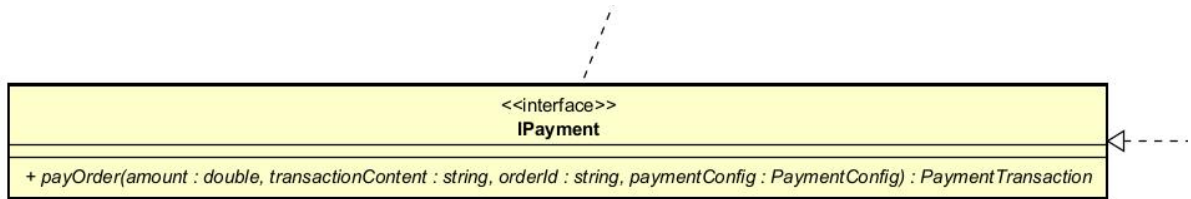


## Class Design for Interface "IPayment"



### Operation Design example

#	Name	Return type	Description
1	<code>getPaymentTransaction</code>	<code>PaymentTransaction</code>	Customer sends payment transaction request, if successful, subsystem will return <code>PaymentTransaction</code>

`getPaymentTransaction` method:

- Parameter:
  - o `amount`: The total amount to be paid for the order.
  - o `transactionContent`: A description or note for the transaction.
  - o `orderId`: The unique identifier of the order.
  - o `paymentConfig`: Contains merchant ID, secret key and other payment setup details.
- Exception:
  - o `UnregisteredInternetBankingException` if internet banking account is not registered.
  - o `NotEnoughBalanceException` if balance in account is insufficient.

- *SuspiciousException* if suspicious or potentially fraudulent activity is detected.
- *Return Parameter: PaymentTransaction* includes the following information:
  - *transactionNo: String*
  - *amount: double*
  - *bankCode: String*
  - *bankTransactionNo: String*
  - *cardType: String*
  - *payDate: DateTime*
  - *transactionContent: String*
- *How to use parameters/attributes: Assign function parameters to corresponding object attributes*
- *Flowchart / activity diagram / sequence diagram: None*
- *State: None*