**VNPAY Document subsystem elements**

**I. VNPaySubsystem package**

a. Class VNPay Subsystem

1. Class name: VNPaySubsystemController

2. Class description: The VNPaySubsystemController class plays the role of controlling the payment process via VNPay in the system. It processes payment requests, communicates with the VNPay system and returns payment transaction results.

3. Operation definition:

- payOrder(order: Order, content: String): PaymentTransaction : This method initiates the payment process for a given order. It will send a payment request to the VNPay system and return a PaymentTransaction object containing information about the payment transaction.

@param order (order): The Order object contains information about the order that needs payment.

@param content (String): String describing payment content  
@return: The PaymentTransaction object contains information about the payment transaction.

b. Class Respone

1. Class name: Respone

2. Class description: The class processes the form the VNPay system and extracts payment transaction information from it.

3. Operation definition:

- getPaymentTransaction(String): PaymentTransaction: This method extracts payment transaction information from VNPay's  string and returns a PaymentTransaction object.

@param respone (String):  string from the VNPay system.  
@return: The PaymentTransaction object contains information about the payment transaction.

- handleErrorCode(errorCode: String): void: This method handles error codes from VNPay's .

@param errorCode (String): Error code string.

@return: Does not return a value (void).

-buildPaymentTransaction(: String): void: This method constructs a PaymentTransaction object from VNPay's  string.

@param (String): string from the VNPay system

@return: None

c. Class VNPayBoundary

1. Class name: VNPayBoundary

2. Class description: The VNPayBoundary class provides an interactive interface with the VNPay system. It processes payment transaction requests and returns results from the VNPay system.

3. Operation definition:

- doPaymentTransaction(query: String): String: This method sends a payment transaction request to the VNPay system and returns the results from VNPay as a string.

@param query (String): String containing the payment transaction request.

@return: String containing payment transaction results from the VNPay system.

d. Class Request

1. Class name: Request

2. Class description: Request handles building the payment transaction request URL for the VNPay system based on input information.

3. Operation definition:

- buildPayOrderQueryURL(amount: int, contents: String, currCode: String, Locale: String, OrderType: String): String        : This method constructs the payment request URL based on inputs such as amount, content, currency code, language, and order type.

@param amount (int): Integer representing the amount to be paid.

@param contents (String): String describing the payment content

@param currCode (String): Currency code string.

@param locale (String): Language string

@param OrderType (String): Order type string

@return: String containing the URL to request a payment transaction.

e. Class Config

1. Class name: Config

2. Class description: The Config class contains the configuration and information necessary to communicate with the VNPay system. It also provides support methods for payment transaction processing.

3. Attributes:

vnp\_PayUrl: String containing the URL of VNPay used to send payment requests. Data type: String.

vnp\_Version: String containing the version of VNPay. Data type: String.

vnp\_TmnCode: Payment service provider code string. Data type: String.

vnp\_ReturnUrl: URL string to return payment results. Data type: String.

vnp\_SecureHash: String of security codes to verify the request. Data type: String.

4. Operation definition:  
- getIpAddress(): String: This method returns the client's IP address

@return: String containing the client's IP address.

- getRandomNumber(): String: This method generates a random number.

@returns: String containing the generated random number.

f. Class ResultScreen

1. Class name: ResultScreen

2. Class description: The ResultScreen class handles displaying payment transaction results to the user.

3. Operation definition:

- displayResult(response: String):void : This method displays payment transaction results on the screen based on the response string from VNPay.

@param response (String): String containing the response from the VNPay system about the payment transaction results.

@return: None (void)

**II. Exception package**

a. Class PaymentException

1. Class name: PaymentException

2. Class description: The PaymentException class is the base exception class for exceptions related to the payment process. It contains information about the error in the form of a message.

3. Attribute:

- Message (String): String containing the error message.

4. Operation definition:

- getMessage(): String: This method returns an error message related to the exception.

@return: String containing the error message.

b. Class InvalidPaymentException:

1. Class name: InvalidPaymentException

2. Class description: The InvalidPaymentException class is a subclass that inherits from PaymentException. It represents exceptions related to invalid payment requests.

3. Operation definition: None

b. Class TimeoutException

1. Class name: TimeoutException

2. Class description: The TimeoutException class is a subclass that inherits from PaymentException. It represents exceptions related to the payment process timing out.

3. Operation definition: None

c. Class ErrorTransactionException

1. Class name: ErrorTransactionException

2. Class description: The ErrorTransactionException class is a subclass that inherits from PaymentException. It represents exceptions related to errors during payment transactions.

3. Operation definition: None

d. Class PaymentTransaction

1. Class name: PaymentTransaction

2. Class description: The PaymentTransaction class represents a payment transaction in the system. It contains information related to the transaction such as transaction code, customer code, amount, transaction time, transaction content and error message (if any).

3. Attributes:

transactionId (String):  String containing the unique ID of the payment transaction.

customerId (String): String containing the customer ID making the transaction.

Amount (int): Integer representing the amount of the payment transaction.

transactionTime (String): String containing the time the payment transaction took place.

Contents (String): String describing the content of the payment transaction.

errorMessage (String): String containing the error message (if any) during the transaction.

4. Operation definition:

- savePaymentTransaction(): void: This method stores payment transaction information into the system (database or file).

@return: Does not return a value (void).