Q1 Identify the actors. Write a description for each actor.

|  |
| --- |
| Actor specification |
| Actor name: potential client |
| Description: un-registered customer |

|  |
| --- |
| Actor specification |
| Actor name: the app client |
| Description: registered customer who interact with system |

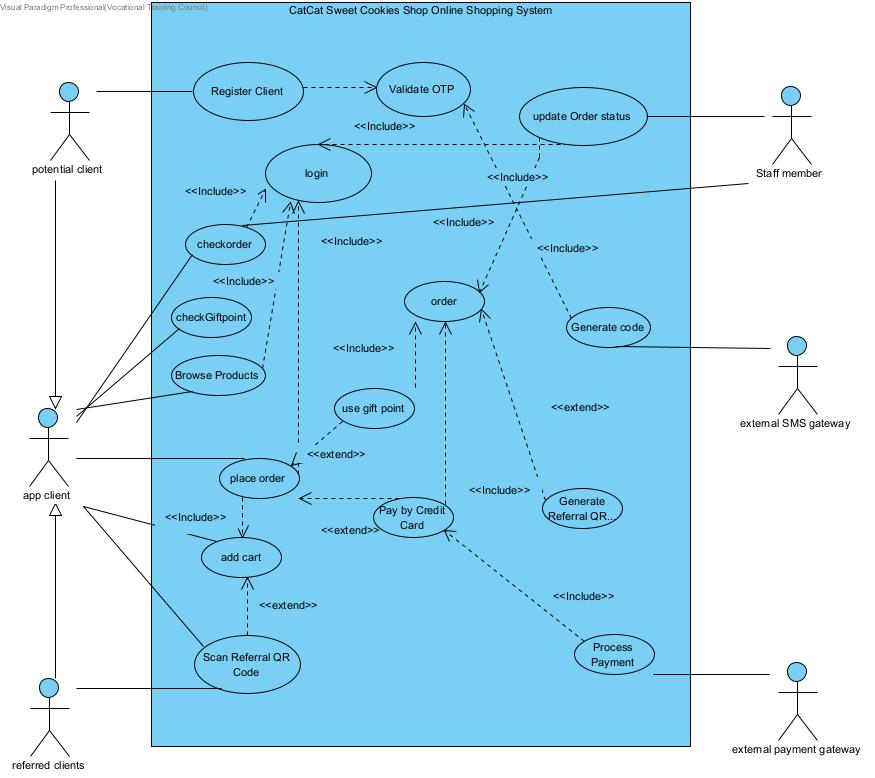
|  |
| --- |
| Actor specification |
| Actor name: referred clients |
| Description: customer that referred by another app client |

|  |
| --- |
| Actor specification |
| Actor name: Staff member |
| Description: view clients’ orders and updating the order’s status. |

|  |
| --- |
| Actor specification |
| Actor name:external SMS gateway |
| Description: generates and send the one-time passcode via an SMS. After successful registration, it send confirmation SMS message. |

|  |
| --- |
| Actor specification |
| Actor name: external payment gateway |
| Description: processes credit card payments during checkout. It validates transactions and returns approval/reject codes to the system. |

Q2 Draw the finalized Use Case Diagram for the online shopping system. Use <> and <> relationships if appropriate.



Q3 Write an initial use case description for each use case given in your answer to Task 2.

|  |
| --- |
| Use Case: Register Account |
| Use Case ID: UC-100 |
| Actor: potential client |
| Description: A public user provides email address, home address, password to register a customer account |

|  |
| --- |
| Use Case: update Order status |
| Use Case ID: UC-110 |
| Actor: Staff member |
| Description: Staff member updates the order’s status from "OUTSTANDING" to "READY FOR SHIPPING." After all products of an order have been successfully delivered to the shipping address, the staff member updates the status of the order from " |

|  |
| --- |
| Use Case: checkGiftpoint |
| Use Case ID: UC-120 |
| Actor: app client |
| Description: he client can check his/her gift points balance in the app. |

|  |
| --- |
| Use Case: checkorder |
| Use Case ID: UC-130 |
| Actor: app client, Staff member |
| Description: view clients’ orders |

|  |
| --- |
| Use Case: Browse Products |
| Use Case ID: UC-140 |
| Actor: app client |
| Description: A app client can search Products by entering keywords. |

|  |
| --- |
| Use Case: Generate code |
| Use Case ID: UC-150 |
| Actor: external SMS gateway |
| Description: Generate code for validate otp |

|  |
| --- |
| Use Case: place order |
| Use Case ID: UC-160 |
| Actor: app client |
| Description: provides shipping address to place order |

|  |
| --- |
| Use Case: add cart |
| Use Case ID: UC-170 |
| Actor: app client |
| Description: The client add a product to the shopping cart |

|  |
| --- |
| Use Case: Scan Referral QR Code |
| Use Case ID: UC-180 |
| Actor: app client, referred clients |
| Description: app client scan a referral QR code to select and add products into the shopping cart. referred clients receive a 5% discount on their order. |

|  |
| --- |
| Use Case: Process Payment |
| Use Case ID: UC-190 |
| Actor: external payment gateway |
| Description: external payment gateway processes the payment. If successful, the gateway returns an approval code to the system and the system saves the order and the payment record and provides the client with an order number. If the payment fails, the gateway returns a reject code and the above payment process will be repeated again. |

1. Write the base use case descriptions for all concrete/abstract use cases related to the process of "Place Order".

|  |  |
| --- | --- |
| Use case name: | **Place Order** |
| Use case ID: | UC-170 |
| Primary actor: | App Client |
| Secondary actor(s): | external SMS gateway |  |
| Brief description: | The client places an order by selecting products, checking out, providing shipping details, and completing payment. |
| Preconditions: | Logged in the system |
| Flow of events: | 1. Client adds products to the cart 2. Client clicks "CHECK OUT," enters shipping address. 3. Client enters payment details   System processes payment and generates a referral QR code |
| Postconditions: | Placed order and qr code was generated |
| Alternative flows and exceptions: | If the payment fails, the gateway returns a reject code and the above payment process will be repeated again. |
| Non-behavior requirements: |  |

|  |  |
| --- | --- |
| Use case name: | add cart |
| Use case ID: | UC-170 |
| Primary actor: | App Client |
| Secondary actor(s): |  |
| Brief description: | Client adds products to the cart by browsing categories or scanning a QR code. |
| Preconditions: | Logged in the system |
| Flow of events: | 1. Client selects a category and adds products **or** scans a QR code. 2. Products are added to the cart. |
| Postconditions: | Cart contains selected products. |
| Alternative flows and exceptions: | If there’s no product in the cart, ask the client to add product |
| Non-behavior requirements: |  |

|  |  |
| --- | --- |
| Use case name: | **Scan Referral QR Code** |
| Use case ID: | UC-180 |
| Primary actor: | App Client |
| Secondary actor(s): |  |
| Brief description: | * Client scans a referral QR code to auto-add products to the cart. |
| Preconditions: | Logged in the system |
| Flow of events: | 1. Client scans QR code. 2. System validates and adds linked products to the cart |
| Postconditions: |  |
| Alternative flows and exceptions: |  |
| Non-behavior requirements: |  |

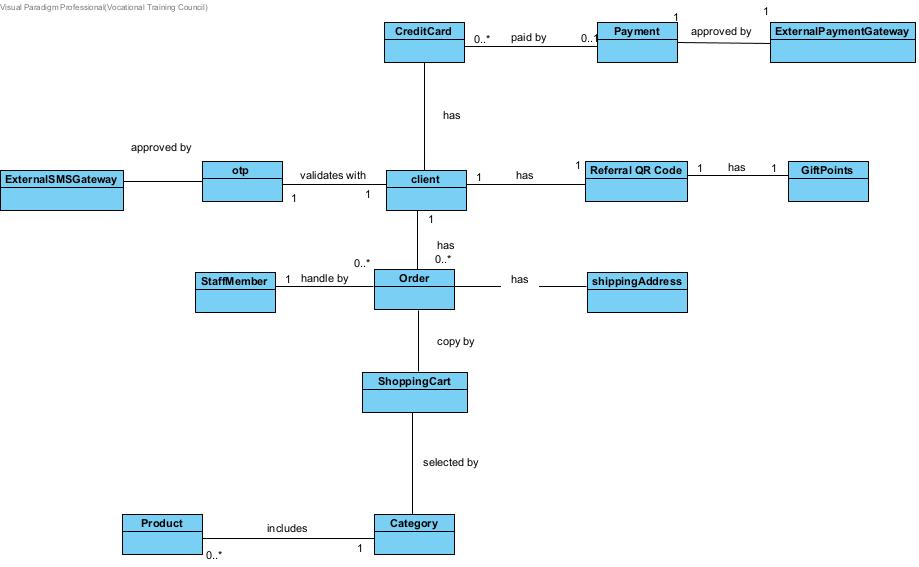
|  |  |
| --- | --- |
| Use case name: | **Process Payment** |
| Use case ID: | UC-190 |
| Primary actor: | App Client |
| Secondary actor(s): | external payment gateway |
| Brief description: | System processes payment via external gateway. |
| Preconditions: | Payment details provided |
| Flow of events: | 1. external payment gateway processes the payment with credit card information and payment amount 2. returns an approval code to the system |
| Postconditions: | Payment status updated |
| Alternative flows and exceptions: | If the payment fails, the gateway returns a reject code and the above payment process will be repeated again. |
| Non-behavior requirements: |  |
| Use case name: | **Generate Referral QR Code** |
| Use case ID: | UC-190 |
| Primary actor: | System |
| Secondary actor(s): | external payment gateway |
| Brief description: | System generates a referral QR code post-payment. |
| Preconditions: | Payment successful |
| Flow of events: | 1. System generates QR code linked to ordered products. |
| Postconditions: | QR code available for sharing. |
| Alternative flows and exceptions: |  |
| Non-behavior requirements: |  |
| Assumptions: |  |

|  |  |
| --- | --- |
| Use case name: | use gift point |
| Use case ID: | UC-190 |
| Primary actor: | App Client |
| Secondary actor(s): |  |
| Brief description: | Client have gift points |
| Preconditions: | * 1. System checks gift points balance.   2. Client opts to use points.   3. System deducts points and updates order total.   4. client chooses gift points |
| Flow of events: | System generates QR code linked to ordered products. |
| Postconditions: | Gift points balance reduced |
| Alternative flows and exceptions: | Client choose not to use points |
| Non-behavior requirements: |  |
| Assumptions: |  |

5. Identify candidate classes by using textual analysis. Prepare a data dictionary for the candidate classes.

|  |  |
| --- | --- |
| Candidate classes | Reason |
| Client | Role play |
| external SMS gateway | Role play/ external system |
| **OTP** | Conceptual thing |
| **Product** | Conceptual thing |
| **Category** | Conceptual thing |
| **Shopping Cart** | Conceptual thing |
| **Order** | event |
| **Shipping Address** | Conceptual thing |
| **Gift Points** | Conceptual thing |
| **Credit Card** | Conceptual thing |
| **Payment** | event |
| **Payment Gateway** | Role play/ external system |
| **Referral QR Code** | event |
| **staff Member** | Role play/ external system |
| **System** | boundary class |

1. Draw a domain class diagram to show the relationships among the candidate classes found in Task 5. Show attributes of classes, the inheritances and multiplicities of the associations between classes. Give appropriate names to associations.



7. Draw a three-tier sequence diagram 