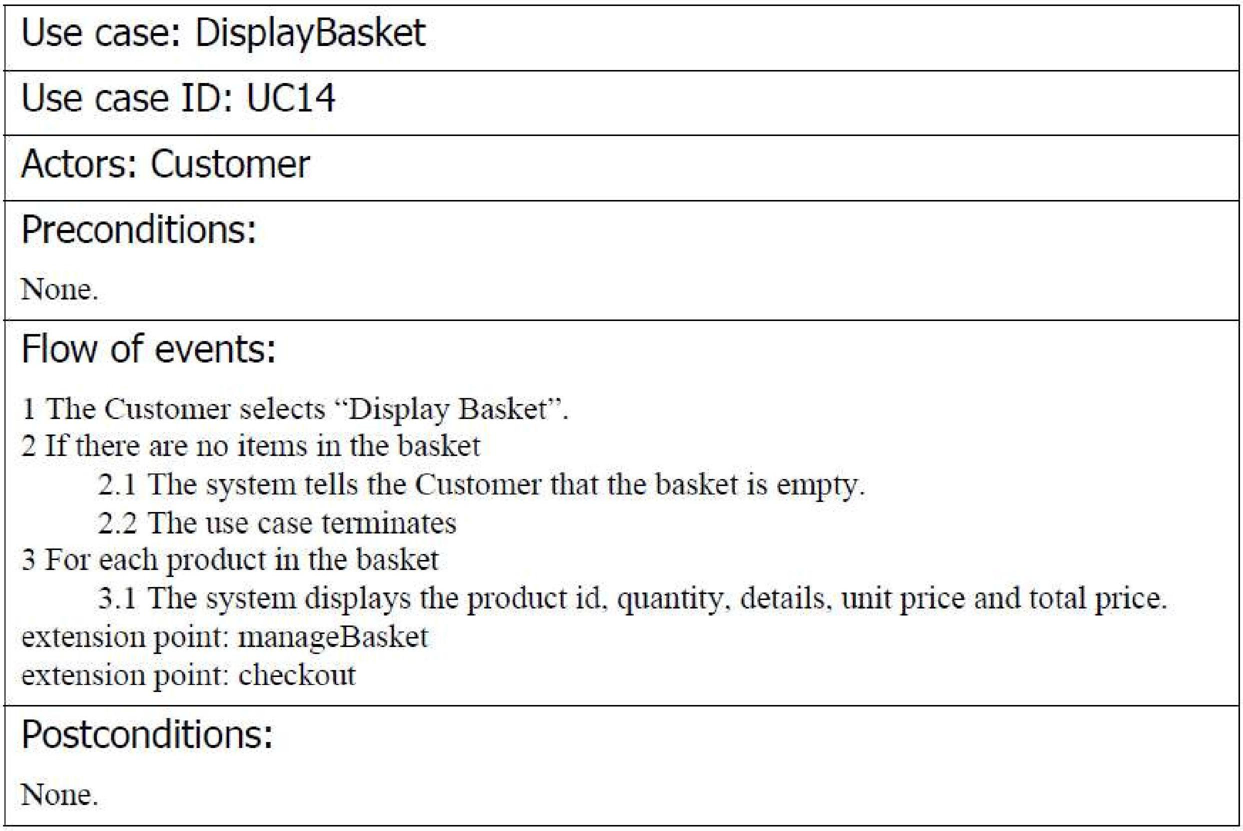
강의 2일차 Solution – part2



Use case: DisplayBasket
Use case ID: UC14
Actors: Customer
Preconditions: None.
Flow of events: 1 The Customer selects “Display Basket”. 2 If there are no items in the basket 2.1 The system tells the Customer that the basket is empty. 2.2 The use case terminates 3 For each product in the basket 3.1 The system displays the product id, quantity, details, unit price and total price. extension point: manageBasket extension point: checkout
Postconditions: None.

Extension Use Case: ManageBasket

Use case ID: UC21

Actors: Customer

Preconditions:

1 The System is displaying the shopping basket.

Flow of events:

1 While the Customer is updating the basket
1.1 The Customer selects an item in the basket.
1.2 If the Customer selects "Remove Item"
1.2.1 The system displays the message "Are you sure you want to remove the selected item from your basket?".
1.2.2 The Customer confirms the removal.
1.2.3 The system removes the selected item from the basket.
1.3 If the Customer enters a new quantity for the selected item
1.3.1 The system updates the quantity for the selected item.

Postconditions:

None.

Use case: AddItemToBasket

Use case ID: 2

Actors:

Customer

Preconditions:

1 The Customer is browsing products.

Flow of events:

1 The Customer selects a product
2 The Customer selects "Add Item".
3 The system adds the item to the Customer's shopping basket.
4 include(DisplayBasket)

Postconditions:

1 A product has been added to the Customer's basket.
2 The contents of the basket are displayed.

Extension Use Case: Checkout

Use case ID:8

Actors:

Customer
Inventory

Preconditions:

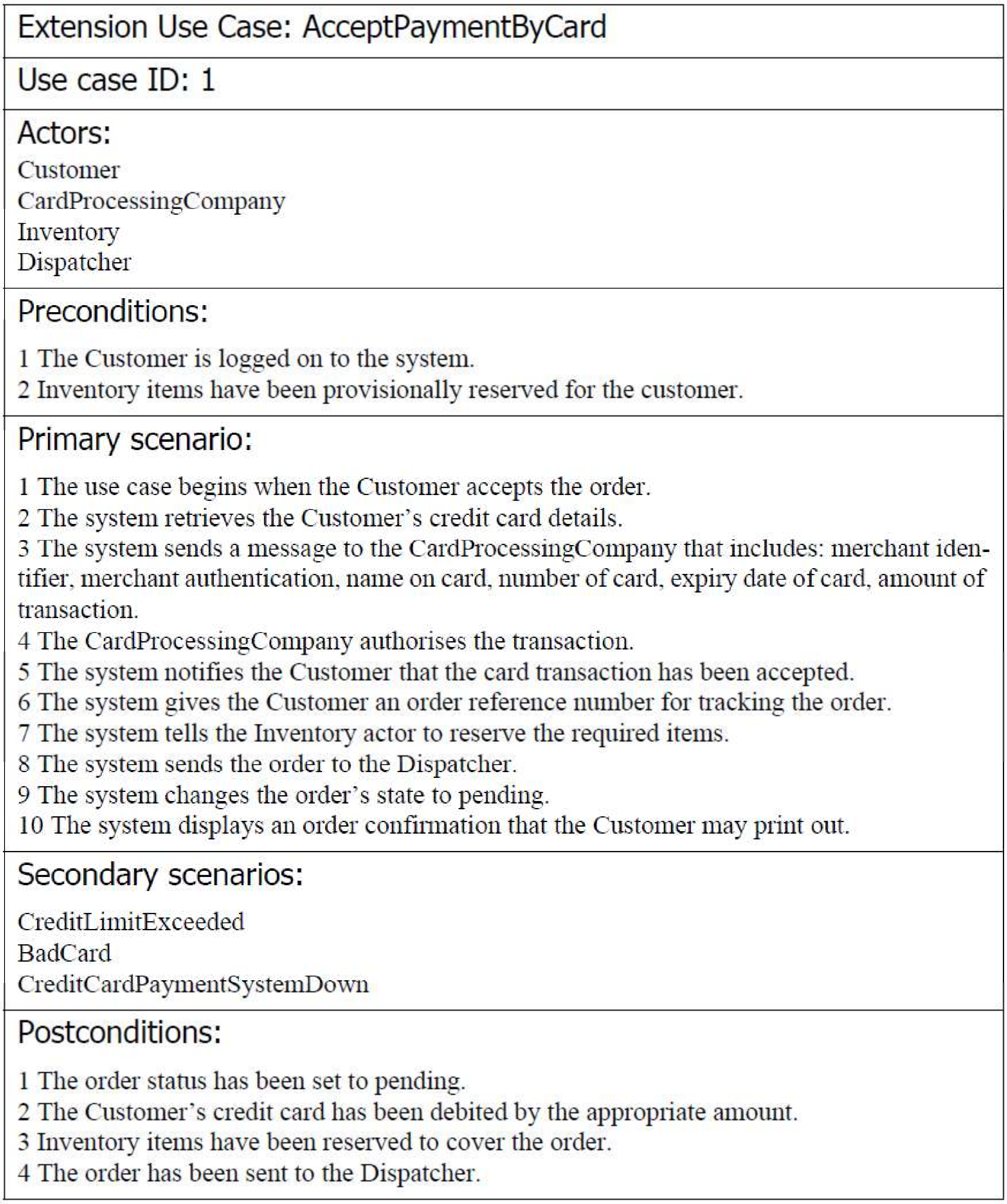
1 The Customer is logged on to the system.

Postconditions:

1 The use case begins when the Customer selects “Checkout”.
2 The system asks the Inventory actor to provisionally reserve the items in the shopping basket.
3 For each item that is out of stock
   3.1 The system informs the Customer that the item is currently unavailable and it is removed from the order.
4 The system presents the final order to the Customer. The order includes an order line for each product that shows the product identifier, the product name, the quantity, the unit price, the total price for that quantity. The order also includes the shipping address and credit card details of the Customer and the total cost of the order including tax and postage and packing.
5 The system asks the Customer to accept or decline the order
6 The Customer accepts the order.
7 include(AcceptPaymentByCard)

Postconditions:

1 The Customer has accepted the order.
2 The ordered items have been reserved by the Inventory actor.



Extension Use Case: AcceptPaymentByCard

Use case ID: 1

Actors:

Customer
CardProcessingCompany
Inventory
Dispatcher

Preconditions:

1 The Customer is logged on to the system.
2 Inventory items have been provisionally reserved for the customer.

Primary scenario:

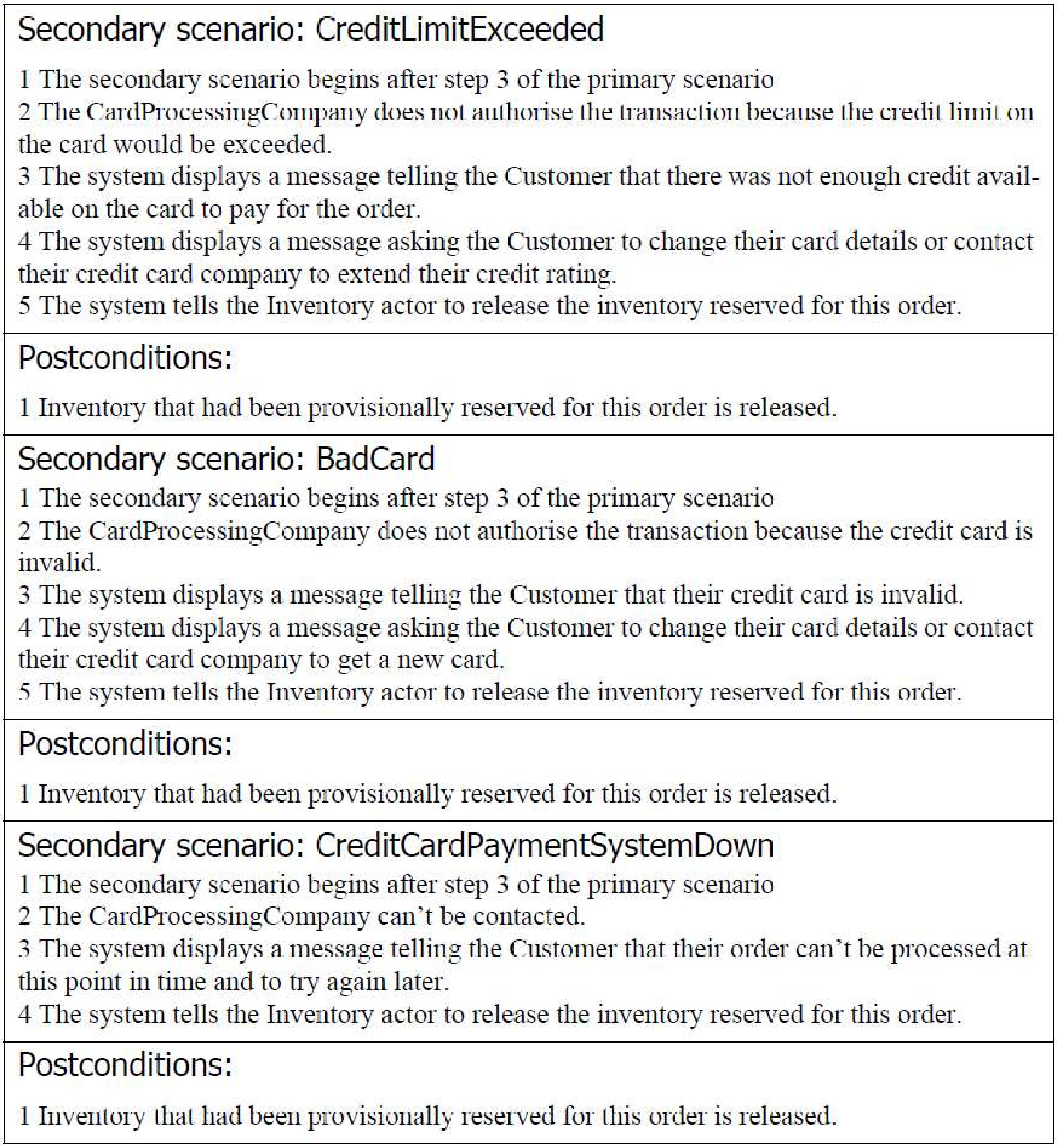
1 The use case begins when the Customer accepts the order.
2 The system retrieves the Customer's credit card details.
3 The system sends a message to the CardProcessingCompany that includes: merchant identifier, merchant authentication, name on card, number of card, expiry date of card, amount of transaction.
4 The CardProcessingCompany authorises the transaction.
5 The system notifies the Customer that the card transaction has been accepted.
6 The system gives the Customer an order reference number for tracking the order.
7 The system tells the Inventory actor to reserve the required items.
8 The system sends the order to the Dispatcher.
9 The system changes the order's state to pending.
10 The system displays an order confirmation that the Customer may print out.

Secondary scenarios:

CreditLimitExceeded
BadCard
CreditCardPaymentSystemDown

Postconditions:

1 The order status has been set to pending.
2 The Customer's credit card has been debited by the appropriate amount.
3 Inventory items have been reserved to cover the order.
4 The order has been sent to the Dispatcher.



Secondary scenario: CreditLimitExceeded

1 The secondary scenario begins after step 3 of the primary scenario
2 The CardProcessingCompany does not authorise the transaction because the credit limit on the card would be exceeded.
3 The system displays a message telling the Customer that there was not enough credit available on the card to pay for the order.
4 The system displays a message asking the Customer to change their card details or contact their credit card company to extend their credit rating.
5 The system tells the Inventory actor to release the inventory reserved for this order.

Postconditions:

1 Inventory that had been provisionally reserved for this order is released.

Secondary scenario: BadCard

1 The secondary scenario begins after step 3 of the primary scenario
2 The CardProcessingCompany does not authorise the transaction because the credit card is invalid.
3 The system displays a message telling the Customer that their credit card is invalid.
4 The system displays a message asking the Customer to change their card details or contact their credit card company to get a new card.
5 The system tells the Inventory actor to release the inventory reserved for this order.

Postconditions:

1 Inventory that had been provisionally reserved for this order is released.

Secondary scenario: CreditCardPaymentSystemDown

1 The secondary scenario begins after step 3 of the primary scenario
2 The CardProcessingCompany can't be contacted.
3 The system displays a message telling the Customer that their order can't be processed at this point in time and to try again later.
4 The system tells the Inventory actor to release the inventory reserved for this order.

Postconditions:

1 Inventory that had been provisionally reserved for this order is released.

출처: http://www.umlandtheunifiedprocess.com/books/supporting-materials-for-our-books/worked-example-1-ecp