Place Order

4.1.2: calculateDeliveryFee(location, deliveryType)
4.1.2: addTiptoTotal(tipAmount)
4.1.3: calculateFinalTotal()
5.1.1: validatePaymentMethod()

[if applyVoucher] 3: enterVoucherCode()
4: selectDeliveryOptions(addressId, deliveryType, tipAmount)
5: selectPaymentMethod()
[if additional instruction] 6: addAdditionalInstruction()
7: confirmOrder()

3.1: showCheckoutAfterVoucher()
4.1: showDeliveryDetails()
5.1: showPaymentMethodAndDetails()
6.1:showAdditionalInstruction(instruction)
7.1: showConfirmedOrderDetails()

1: getCart(customerId)
4.1.1: validateDeliveryAddress(addressId)

Customer

:CheckoutUI

2: displayCheckoutSummary()

:OrderController

:Customer

4.1.1.1: getLocation()

7.2: checkAvailableRider(resLocation, deliveryLocation)

:RiderMatchingService

:Location

6.1.1:addAdditionalInstruction()
7.1.1: createOrder(customerId, restaurantId, totalAmount)
7.1.2: *[for each cartItem] createOrderItem(item)
7.1.3: associateDiscount(order, discount)
7.1.4: finalizeOrder(order)

1.1: validateCart(cart)
1.2: calculateTotal()
7.4: clearCart(customerId)

3.1.1: retriveDiscount(voucherCode)
[if valid] 3.1.2: applyDiscount()

1.1.1: checkEmpty()
1.2.1: *[forEachItem] getSubtotal()
7.4.1: *[for each item] removeItem()

1.2.1.1: getMenuItem()

:MenuItem

:CartItem

:Cart

:Order

3.1.1.1: checkValidity()

:ActiveDiscounts

:Discount

7.1.4.1: recordOrderHistory(order)

[if rider found]7.3: sendOrderToRestaurant(order)

:OrderHistory

7.3.1: notifyRestaurant(orderId, orderDetails)

:Notification

:OrderManagementSubsystem