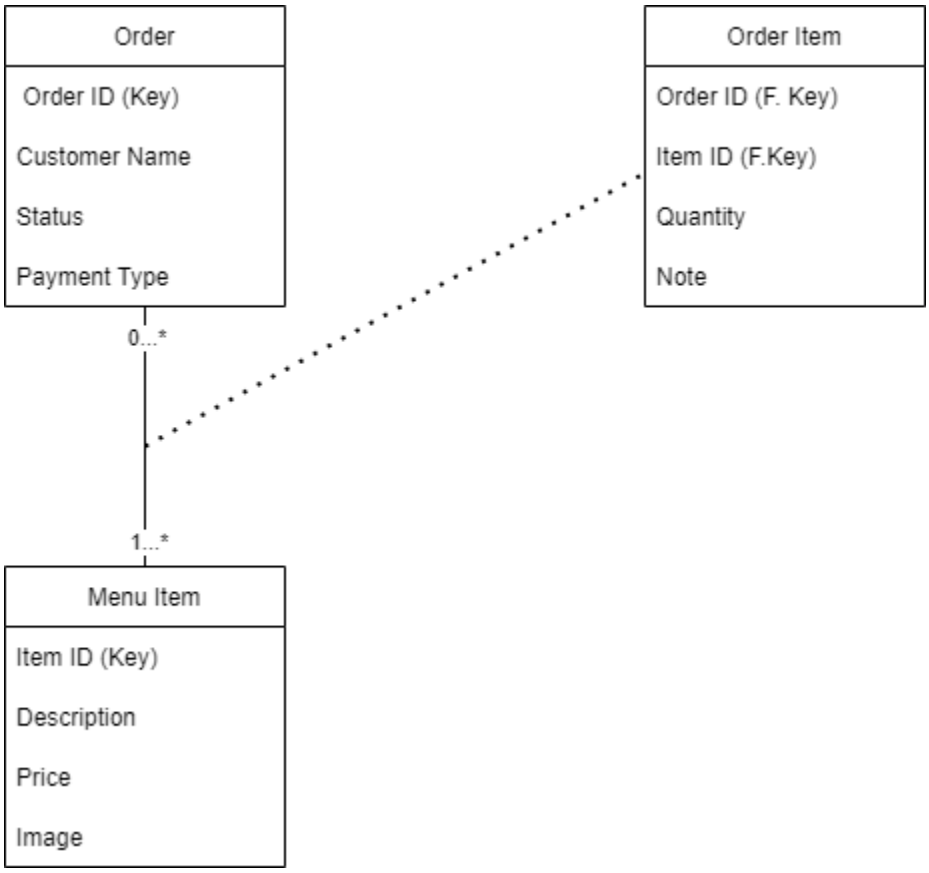
Kyle Just, Joshua Jenkins, Abdul Kalam Syed, Taft Ring, Russell Rickards
CS 3560.02
Prof. Damavandi
October 23, 2022

Group 5 Assignment #2

Part 1- Domain Modeling & Use Case Modeling:

A. Use methods covered in the class to identify "thing" (class). Only include courses that are related to your system. Think carefully about each specified class, all its related attributes, and how classes are related (multiplicities). Identify super and subclasses if any. Use UML tools to develop your system domain model class diagram. (10 Pts.)



B. In group assignment #1, you identified your system use cases. Revise your list! If any change is required, do it! you might need to add, remove or change some use cases. Once you are confident with your use case list, choose one of them and create its fully developed description table. (10 Pts.)

Event	Use Case	Actors
Cashier creates the order	Create Order	Cashier
Cashier wants to edit order	Edit Order	Cashier
Cashier marks the order as deleted after sent to database	Delete Order	Cashier
Cook prepares the order and marks it as done	Prepared order	Cook
When a customer receives the order, cashier marks the order as complete in the system which processes the payment	Complete order	Cashier
The restaurant closes for the day	Generate sales report	Manager

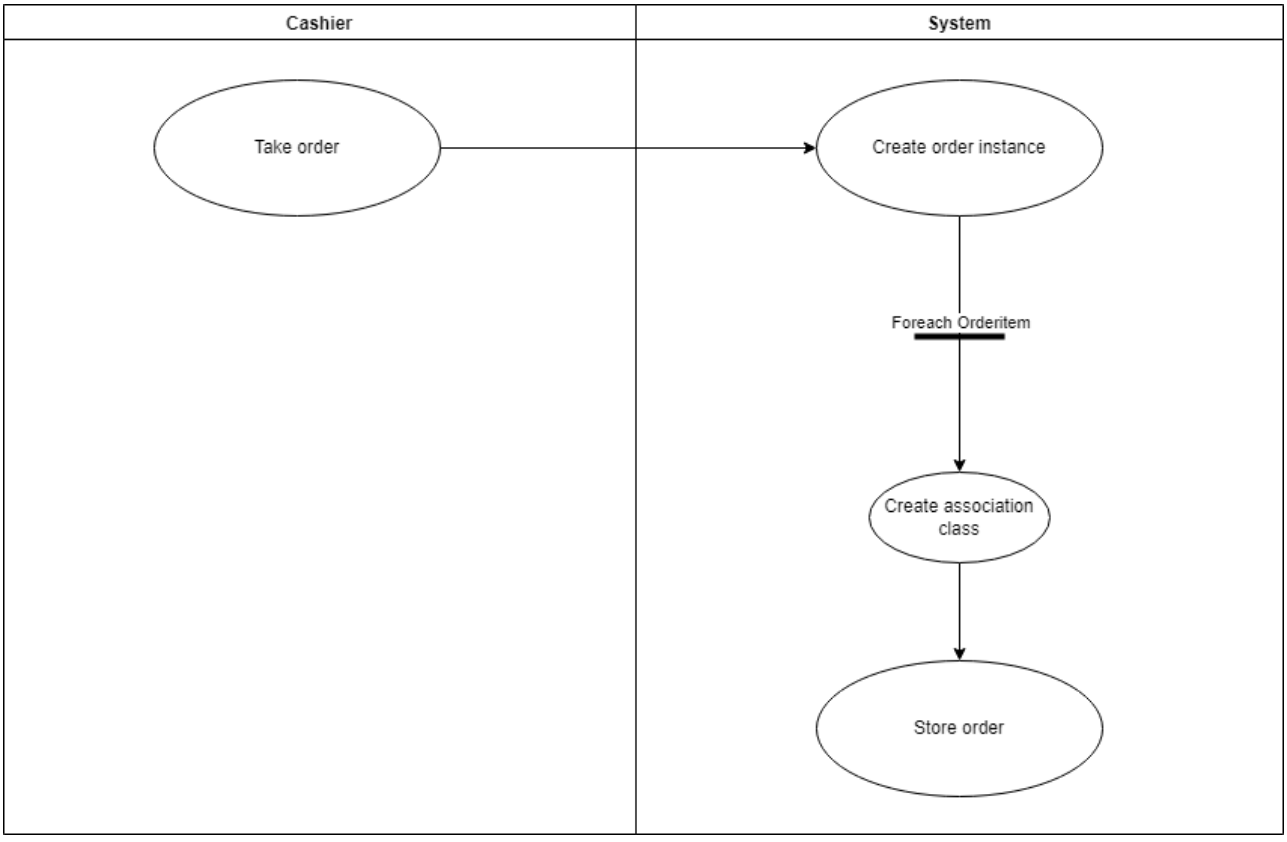
Use case name:	Create Order
Scenarios:	Customer pulls up to the drive-thru and tell the cashier their order, the Cashier then creates order and enter the customer's request.
Triggering event:	The cashier creates a new instance of an order
Brief description:	A cashier creates an order when the customer comes to the drive-thru, this order then goes to the database for the order to be completed.
Actors:	Cashier
Related use cases:	Edit order, Delete order, Prepared order, Complete order, Generate sales report

<table>
<tr><td>Stakeholders:</td><td>Restaurant Manager / Owner, Customer, Front of House Employees (Cashiers), Back of House Employees (Cooks)</td></tr>
<tr><td>Preconditions:</td><td>Cashier must be tendering order, customer must be in the drive-thru, customer must order something</td></tr>
<tr><td>Postconditions:</td><td>The completed order is entered into the database successfully</td></tr>
<tr><td>Flow of activities:</td><td>
<table>
<tr><th>Actor</th><th>System</th></tr>
<tr><td>1. The cashier creates an order with all the items desired by the customer.</td><td>1.1 System reads through order and associates with the Menu Item.

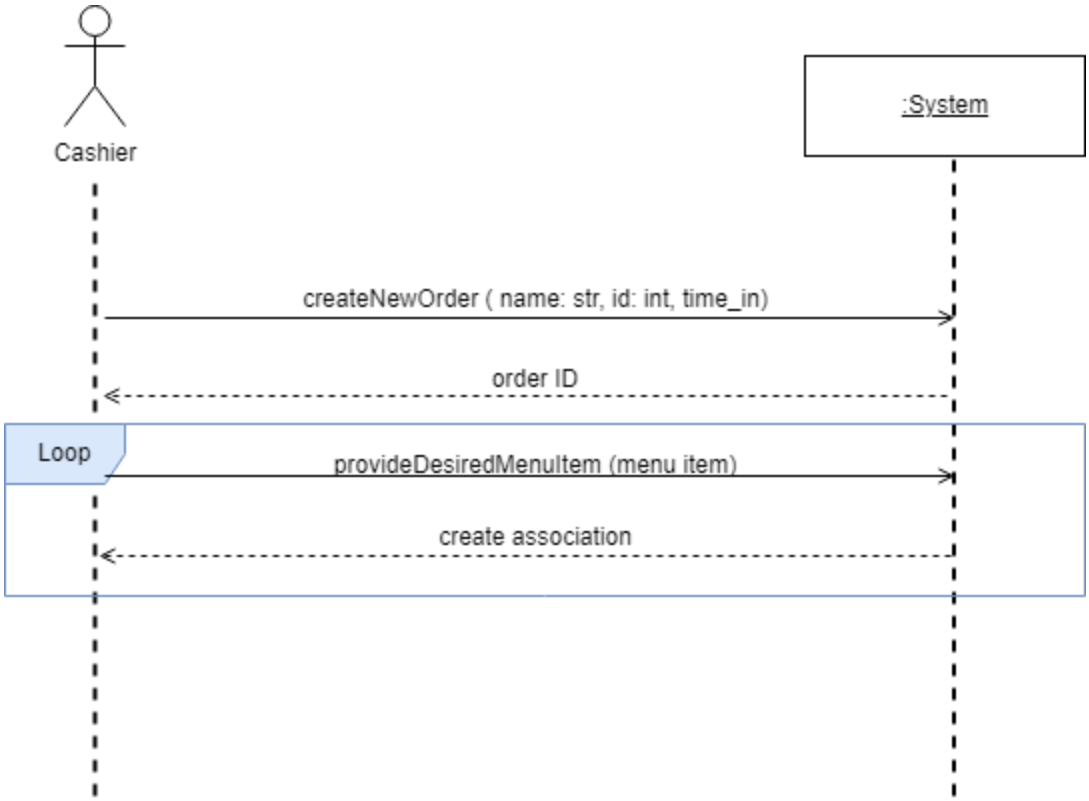
1.2 System Stores New instance of Order Item and stores it in the database.</td></tr>
</table>
</td></tr>
<tr><td>Exception conditions:</td><td>1.1 Cashier creates new order but does not complete it, the order is not sent to the system.</td></tr>
</table>

C. Use UML tools to develop an activity diagram and SSD for the use case described in part B. (10 Pts.)

Activity Diagram:



SSD Diagram:


Cashier
:System
createNewOrder (name: str, id: int, time_in)
order ID
Loop
provideDesiredMenuItem (menu item)
create association