

Homework 5: Use Cases, Sequence Diagram, and State Diagram

Name : _____

Student Number : _____

Laboratory Time : _____

Objectives

- Create a Use Case Diagram in Rational Software Development Platform
- Create a Sequence Diagram in Rational Software Development Platform
- Create a State Diagram in Rational Software Development Platform

Preamble

In a previous lab we learned how to create Class Diagrams from Java Code and vice versa. In this lab, we will learn how to create Use Case Diagrams, Sequence Diagrams, Activity Diagrams, and State Diagrams using Rational Software Development Platform.

Modifications

Since the lab computers were so slow and Rational Software Development Platform was so difficult to use in Labs 1 and 2, the lab portion of this assignment is now optional and can be completed for extra credit.

One question has been added to the take-home portion of the assignment. You will select, download, and install a UML tool to create the diagrams. You can choose any tool you wish as long as it is a UML or software engineering tool, NOT a drawing or paint program.

Take Home

1. (30 points) Select, download, and install a UML tool on your computer. Write a paragraph on your experiences completing do this. Some questions you can answer are: How did you choose the tool? Was the installation difficult? If it is a tool other than Rational, what was it like changing tools? Did your knowledge of UML help or hinder?
2. (40 points) Draw a sequence diagram for a when one PackagedProduct is purchased.
 - You can assume that SelfCheckOut is already running, but the transaction has not started.
 - Make sure your diagram clearly shows which objects are created when a PackagedProduct is added to the CheckOutCart and when paying.
 - The order in which objects are created or invoked is important.
 - Objects that you must include are
 - SelfCheckOut,
 - PaymentCollector,
 - ProductDB,
 - GroceryItem,
 - CheckOutCart,
 - UPC, and
 - PackagedProduct.
 - You can include other objects if you need them to show the return transactions.
 - You can use the addUPC method in edu.uci.ics.inf111.selfcheckout.gui.Actions as starting point for your diagram.
3. (10 points) Create a state chart diagram to show the states and transitions in the SelfCheckOut class (checkOutState enumeration).
 - For each state, make sure to include the state name.
 - For each transition, make sure to include the event name and guard condition, when applicable.
4. (40 points) Create a use case for buying groceries from the SelfCheckOut.
 - You must use the Use Case template provided.
 - Fill in every field in the template with relevant details.
 - You must include at least two questions in the Miscellaneous field.

Submit both diagrams and the use case in a single document (Word or pdf).

Grading Checklist (30 points)

By the end of the laboratory session, you need to demonstrate to the TA that you can do the following tasks.

- Use Case Diagram
- Initial Sequence Diagram (Steps 1-6)
- Completed Sequence Diagram
- State Diagram

TA Initials: _____

Instructions for the Laboratory

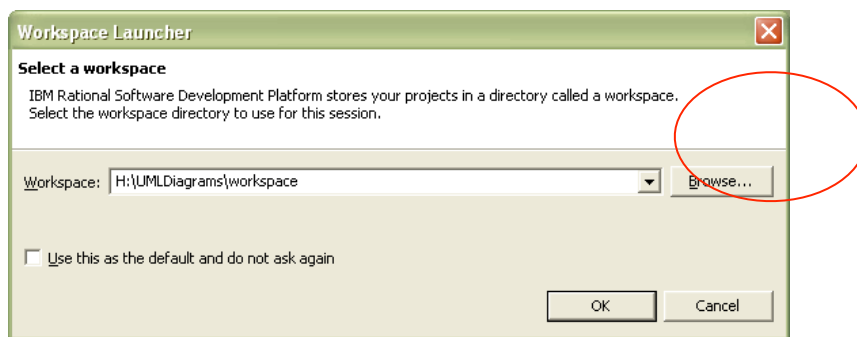
Task 1: Create a UML Use Case Diagram

For this task, you will create a UML Use Case Diagram using Rational Software Development Platform. It will be based on the requirements provided to you.

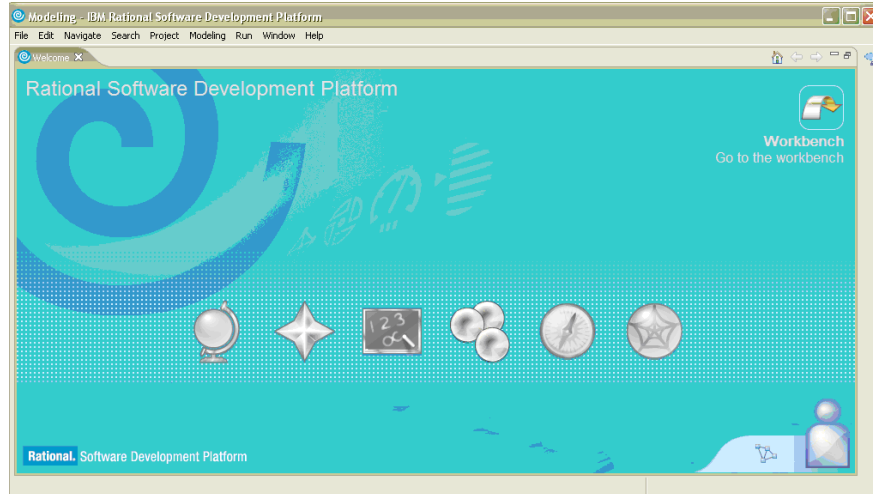
- a) Start Rational Software Development Platform.



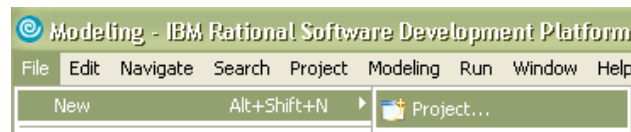
- b) Change the Workspace location to H:\homework5



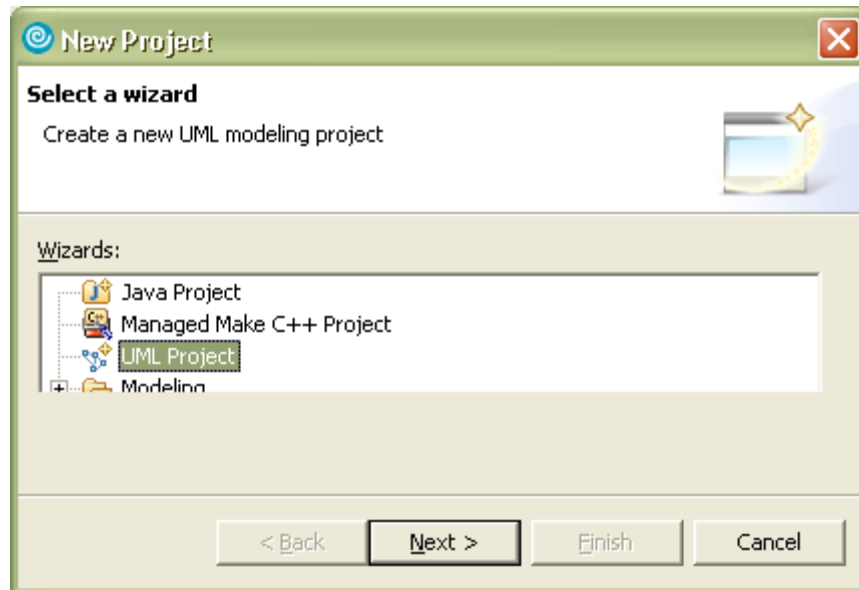
- c) Click on Workbench.



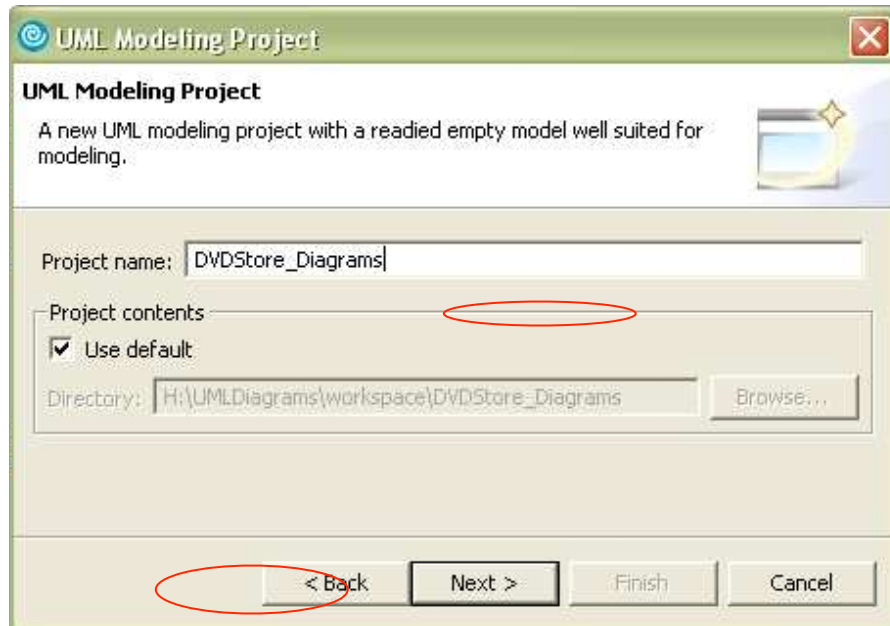
d) Create a new Project. Go to File -> New -> Project



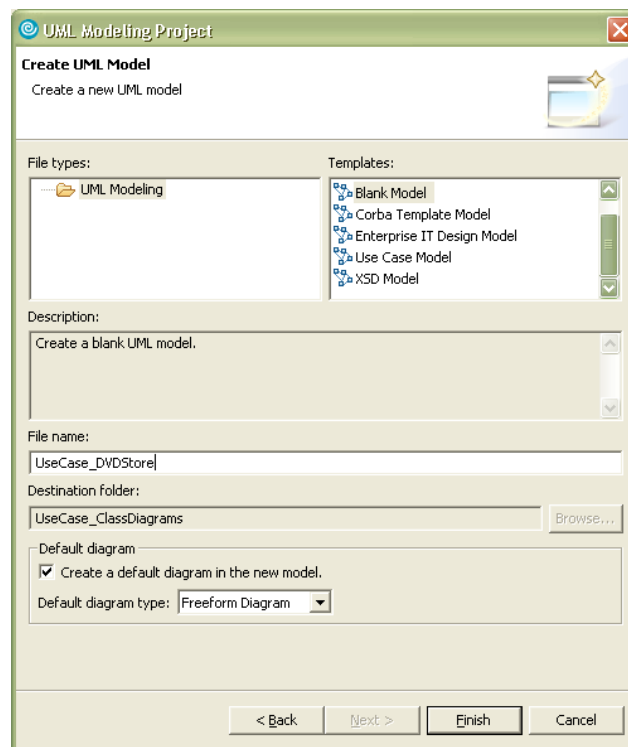
e) Select the UML Project wizard. Click Next.



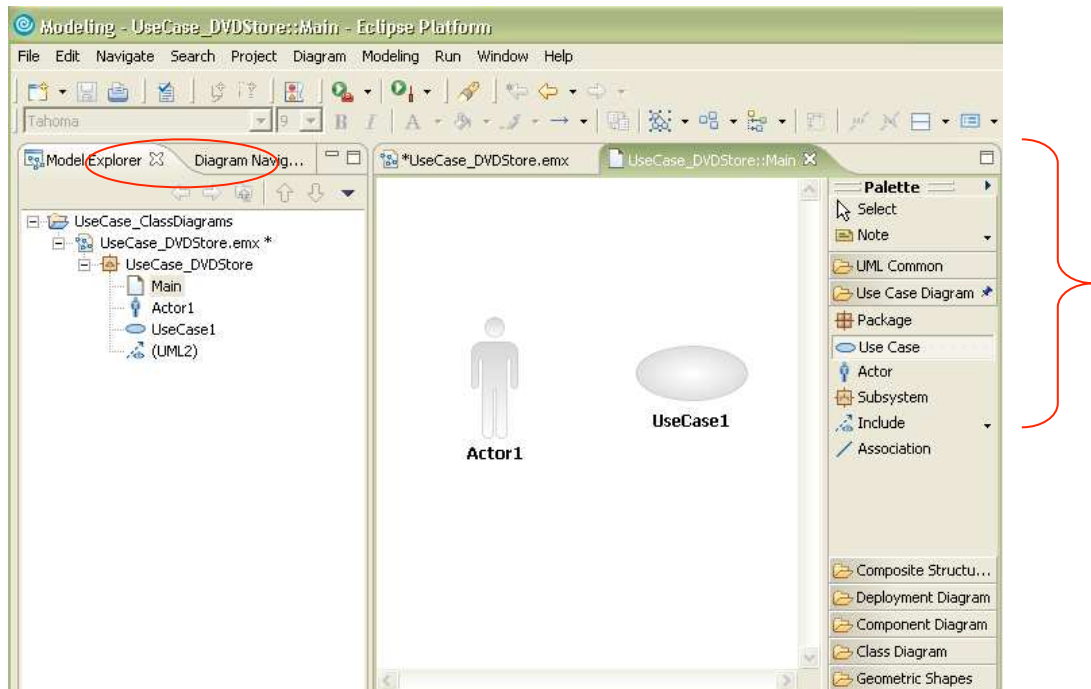
f) In the Project name field, enter "DVDStore_Diagrams". Click Next.



g) Select Blank Model template. In the File name, enter "UseCase_DVDStore." Click Finish.



- h) Select Main under UseCase_DVDStore in the Model Explorer. Expand the "Use Case Diagram" section in the Palette. Select the objects you need to create for the next step using the Use Case Diagram Palette.



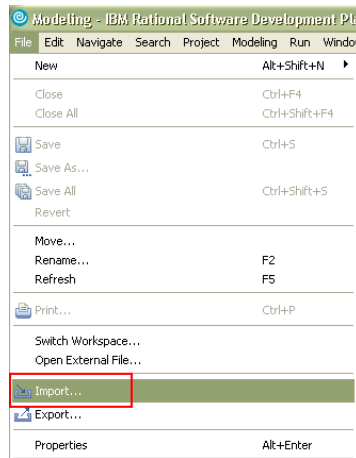
- i) Create the UML Use Case Diagram for a system to buy DVDs online. Customers, Delivery Agent, and Administrators of the System will use the system. They can perform the following actions in the system:
- Customer:
- Add a DVD to shopping cart (requires the user to log in)
 - Remove a DVD from shopping cart (requires the user to log in)
 - Check out shopping cart (requires the user to log in)
 - Review contents of shopping cart (requires the user to log in)
 - Search DVDs by Title and Category (Some examples of categories are: Series, Movies, Music)
- Delivery Agent:
- Check online orders
 - Change the status of the order to delivered
- Administrator:
- Add DVD
 - Update inventory of DVDs

Note: Use the "include" and "extend" relationships as needed. The multiplicity in all cases can be *

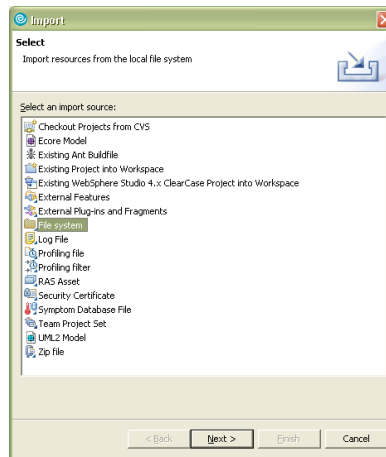
Preparation for Task 2: Import a Class Diagram of the DVD Online Store into Rational Software Development Platform

For this task, you will add a UML Class Diagram for the DVD Online Store provided to you into the current DVDStore_Diagrams project. You need to do this, so that you can create sequence diagrams in the next task in this lab.

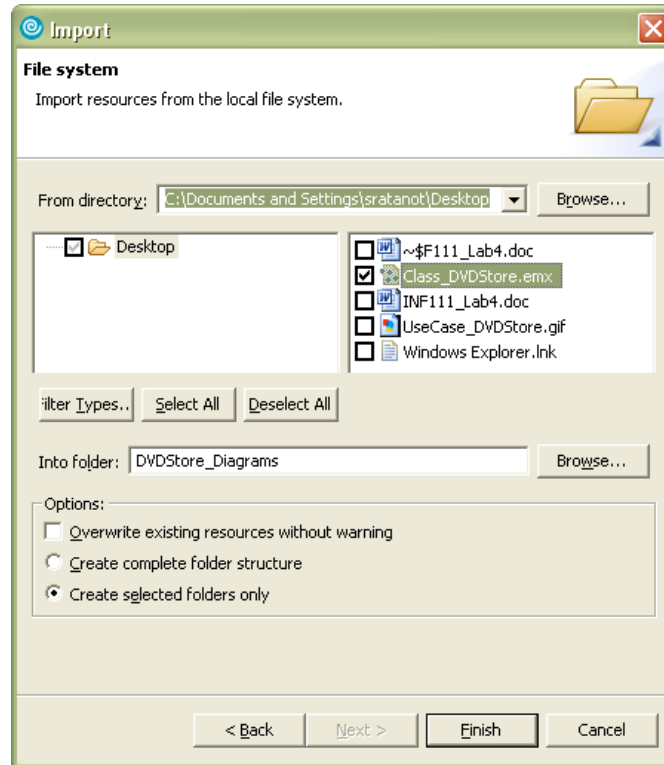
- a) Download the Class_DVDStore.emx from the class Website and save it on your computer (in this example, it is saved on the Desktop.)
- b) In order to add the model Class_DVDStore.emx that you downloaded, select the project DVDStore_Diagrams, and then go to File -> Import.



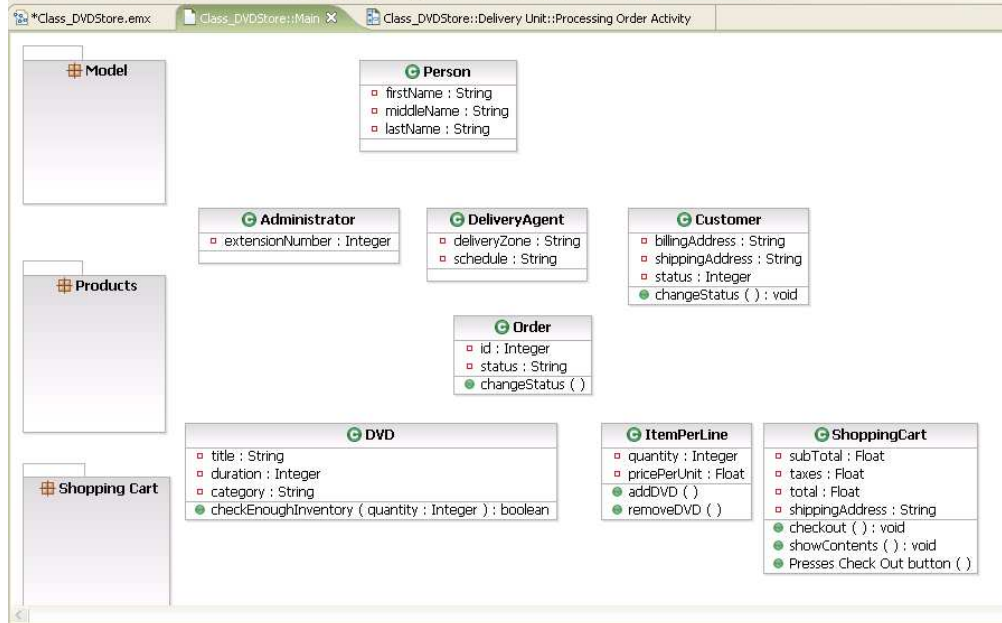
- c) In the Import window, choose File system and click Next.



- d) To import a file, select the folder (Desktop) where you saved Class_DVDStore.emx and check Class_DVDStore.emx. Click Finish.



- e) In the Model Explorer, you can see the imported Class_DVDStore.emx. Double click on the model Class_DVDStore.emx to see its contents.



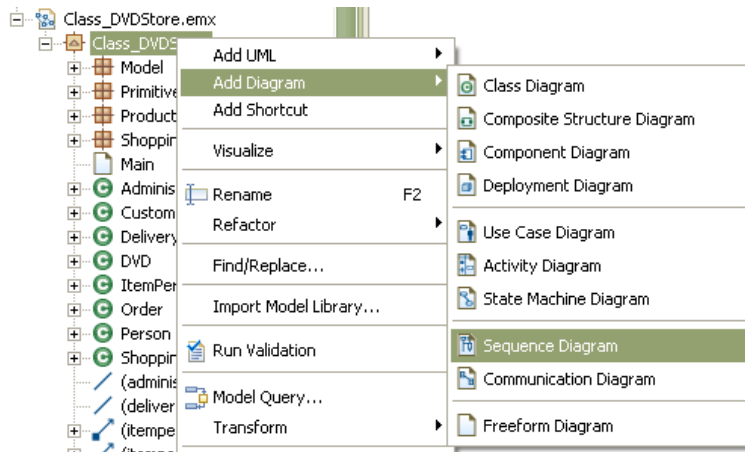
Task 2: Create a Sequence Diagram in Rational Software Development Platform

For this task, you will create a UML Sequence Diagram for the Use Case “Check out shopping cart” using Rational Software Development Platform. It will be based on the Use Case description provided below:

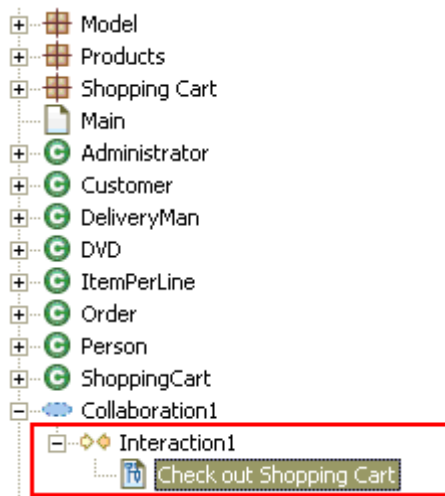
Use Case Name	Check out shopping cart
Scope	DVD Online Store
Level	subfunction
Primary Actor	Customer Bank System
Stakeholders and Interests	Customer wants to buy DVDs The DVD store wants to sell DVDs and charge money for the DVDs Bank System wants to process credit card information to charge money
Preconditions	The customer is registered in the system and the user is logged in
Success Guarantee	The order is placed
Main Success Scenario	<ol style="list-style-type: none"> 1. The Customer presses the “Check out” button 2. The system shows the contents of the shopping cart and asks for the Shipping Address 3. The Customer enters the Shipping Address 4. The system asks for credit card information 5. The Customer enters credit card information 6. The system sends the payment information to the Bank system 7. The Bank system returns transaction OK 8. The System creates the Order 9. The System sends a notification to the Deliver man 10. The System shows the Confirmation of Place Order page

Extensions	7. If the Bank system returns transaction NO OK, the system should show an Error Page explaining the reason of the failure.
Special Requirements	
Technology and Data Variations List	
Frequency	The store expects 1 orders placed every 10 minutes
Miscellaneous	

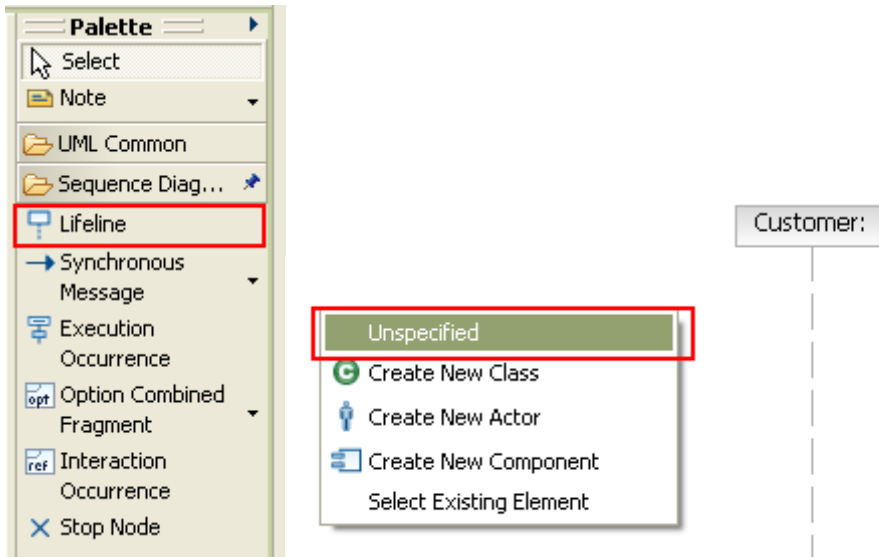
- a) Create the Sequence Diagram in Rational Software Development Platform. Select the Class_DVDStore model under the Class_DVDStore.emx. Right click on it and select Add Diagram -> Sequence Diagram.



- b) Change the name "Diagram 1" to "Check out Shopping Cart".



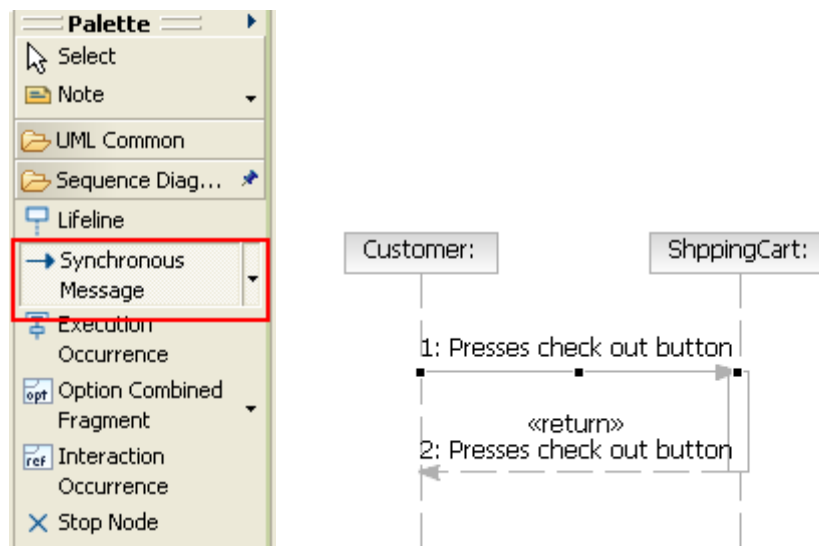
- c) Using the Sequence Diagram Palette, create a Lifeline for Customer. Select the option Unspecified and rename the Lifeline as Customer.



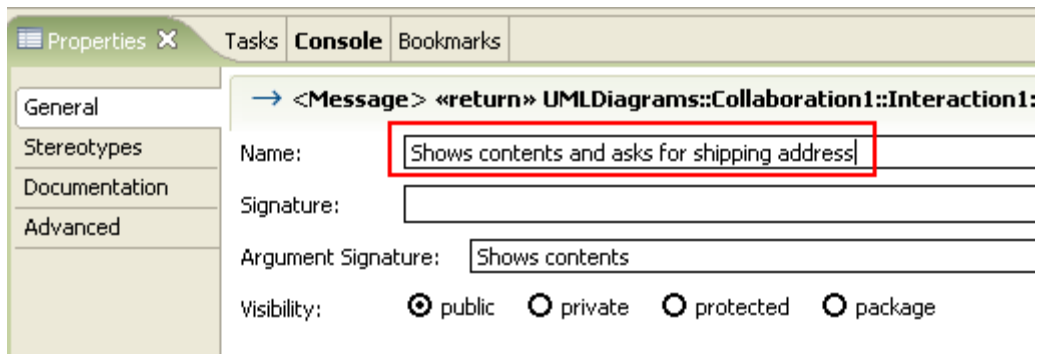
- d) Create Lifelines for "ShoppingCart," "Order," "Bank System," and "Delivery Agent."



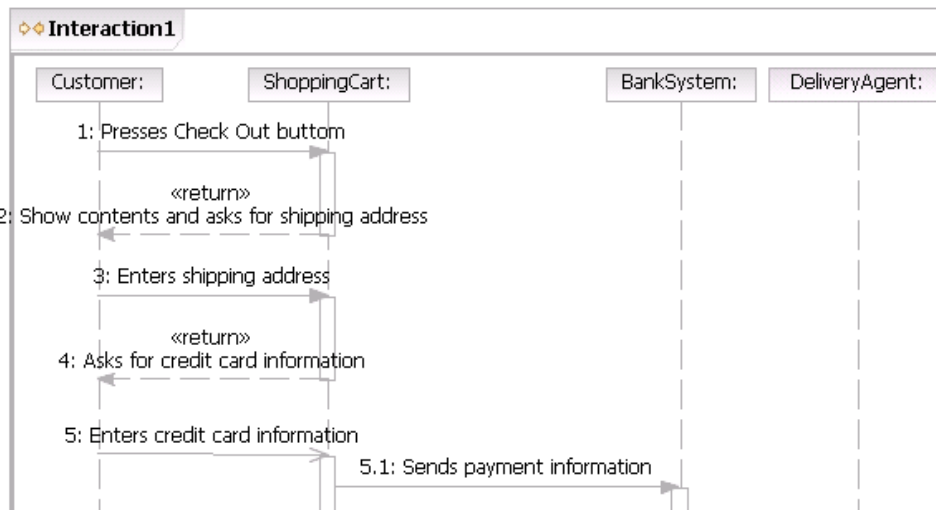
- e) Create a Synchronous message using the Palette. Drag the mouse from Customer to Shopping Cart. Rename the message "Presses Check Out button."



- f) Select the return message. In the properties view, change the Name of the message to "Shows contents and asks for shipping address."

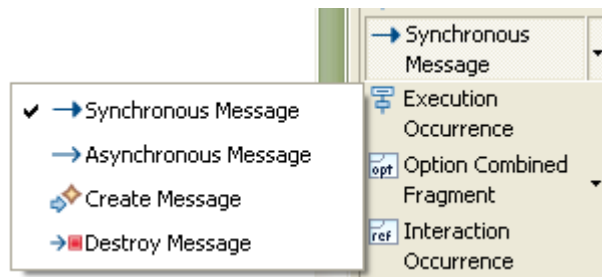


- g) The Sequence Diagram for the first 6 steps in the Main Scenario of the Extended Use Case Format will look similar to the following diagram. Notice that step 5 has been created using an "Asynchronous Message."



Task 3: Complete the remaining steps in the Sequence Diagram

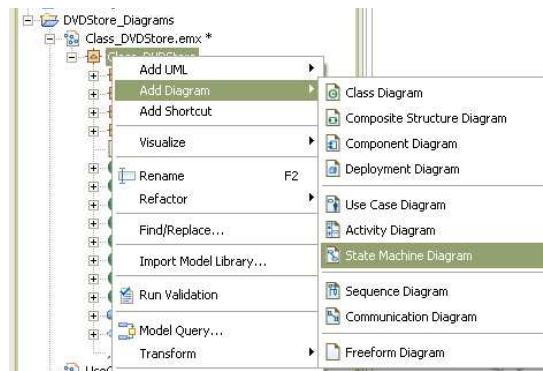
- a) Add events to the sequence diagram corresponding to steps 7-10 of the Main Success Scenario. You should use a "Create Message" for step 8, and synchronous or asynchronous messages when necessary.



Task 4: Create a State Diagram in Rational Software Development Platform

For this task, you will create a UML State Diagram of the ShoppingCart object of the Online DVD Store using Rational Software Development Platform. The ShoppingCart object can be in three states: "Empty", "Ready for Check Out", and "Checking Out". The ShoppingCart starts in the Empty state. When a user adds an item, the ShoppingCart will transit to the "Ready for Check Out" state. The Shopping cart will continue to be in this state while more items are added. The ShoppingCart will transit to "Checking Out" state when the user is checking out the items in the cart. Once the user finishes checking out, the ShoppingCart will transit back to the "Empty" state.

- a) Create the State Diagram in Rational Software Development Platform. Select the Class_DVDStore model under the Class_DVDStore.emx. Right click on it and select Add Diagram -> State Machine Diagram.



- b) Rename the diagram to "ShoppingCart State Machine"



- c) Complete the State Machine using the State Machine Palette. The finished diagram should look like the diagram figure below. To add a guard condition, Click on a Transition and add the guard condition in the Body field in the Guard area.

