Assignment #2

UML – CSP 586

Ritika Kumari
CWID : A20414073

Project Overview Statement:

Develop an application for the GameSpeed retailer that will allow its customers to buy/trade-in products from the retailer either in-store or online.

Deliverables:

1. Any Three System Sequence Diagrams for the use-case diagram

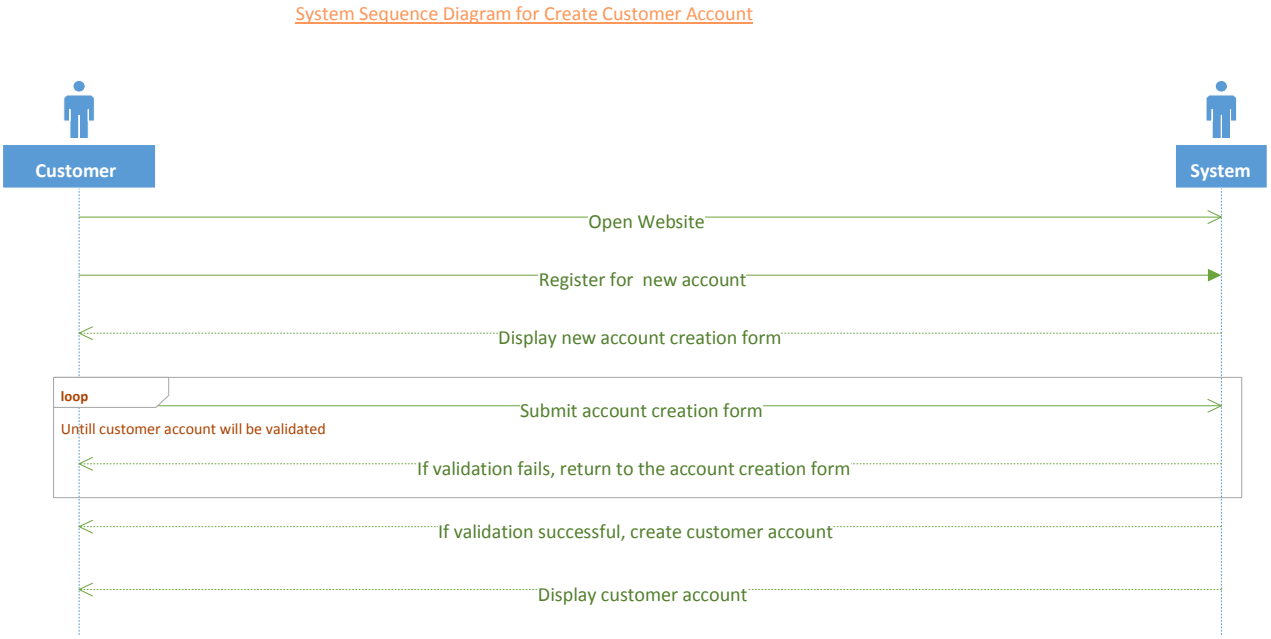
A system sequence diagram (SSD) illustrates events sequentially input from an external source to the system

- A sequence diagram is a good way to visualize and validate various runtime scenarios.

- An SSD shows – for one scenario of a use case –
- the events that external actors generate,
- their order, and
- inter-system events
- The system is treated as a black-box
- System sequence diagrams are a timeline drawing of an expanded use case.

A) System Sequence Diagram for Use Case: "CreateCustomerAccount"

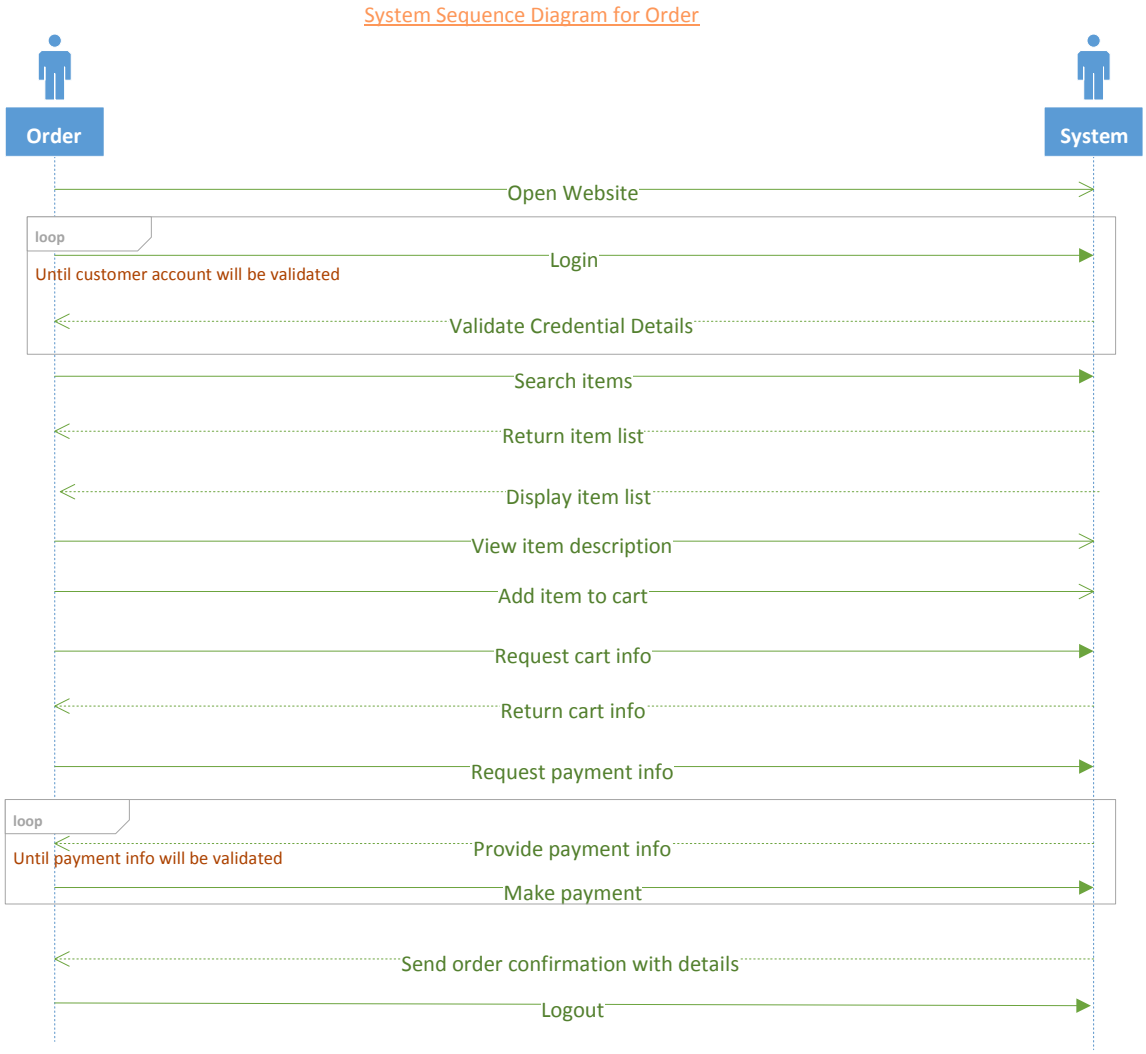
- Customer will go to the website and request for register a new account
- System displays a registration form to the customer
- Customer fills the registration form and submit the form
- System will validate the form using loop (loop: if validation fails)
- Customer account gets created after validation and acknowledgement is sent by system.



B) System Sequence Diagram for Use Case: "Order"

- Customer opens the website and login to his account.
- System validates the credential details. If it is successful, it will allow customer to login. If the attempt is unsuccessful, it will go back to the login page.
- Customer requests for search items.
- System returns the requested items with descriptions.
- Customer add items to the shopping cart and requests for the cart info.
- System displays the cart info to the Customer.

- Customer enters the payment details. System validates the payment details. If it is successful, order will be placed, and system will send a notification to customer. If it is unsuccessful, System asks the Customer to re- enter the payment details.
- Finally, Customer logs out from his account.



C) System Sequence Diagram for Use Case: "ViewProducts"

- Store Manager logs in to his account.
- System validates the credential details. If it is successful, it will allow Store Manager to login. If the attempt is unsuccessful, it will go back to the login page.
- Store manager requests to view the products' page and system displays the requested products' page.
- Store manager will add new products to the products page and the page is updated by the system.
- Store manager will update discounts for the products to the products page and the page is updated by the system.

- Store manager deletes products from the page and it is updated by the system.
- Store manager updates the product info and it is updated by the system.
- Store manager logs out from his account after doing all the operations.

2. The Class diagram for your analysis model

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the systematic of the application, and for detailed modelling translating the models into programming code.

In the diagram, classes are represented with boxes that contain three compartments:

- The top compartment contains the name of the class. It is printed in bold and cantered, and the first letter is capitalized.
- The middle compartment contains the attributes of the class. They are left-aligned, and the first letter is lowercase.

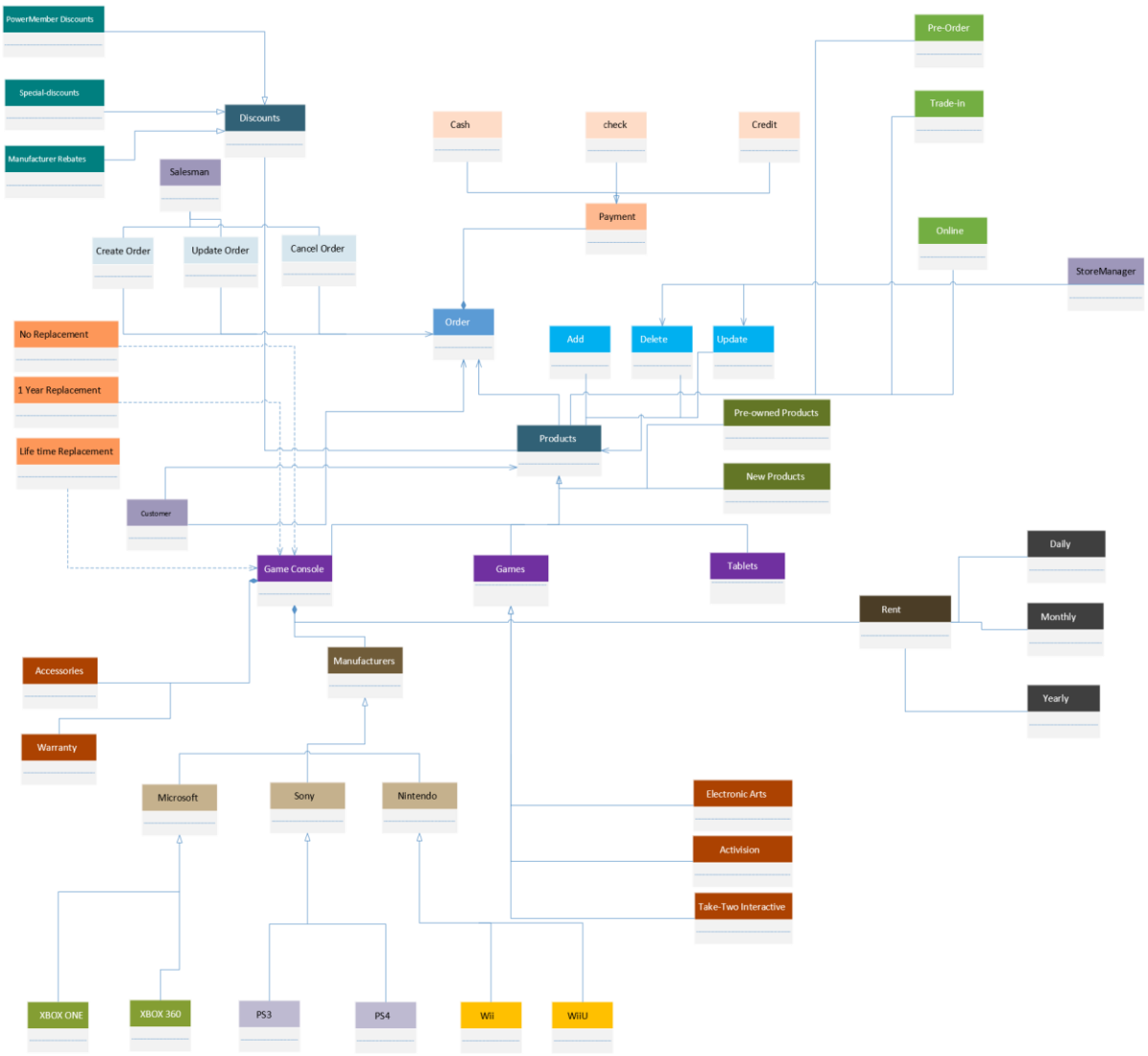
- The bottom compartment contains the operations the class can execute. They are also leftaligned, and the first letter is lowercase.

Sl No.	Class name
1	Customer
2	Products a) Pre-Owned Products b) New Products
3	Order
4	Payment a) Cash b) Check c) Credit
5	Discounts a) PowerMember discounts b) Special-discounts c) Manufacturer rebates
6	Salesman a) Create Order b) Update Order c) Delete Order
7	StoreManager d) Add Products e) Update Products f) Delete Products
8	Games a) Electronic Arts b) Activision c) Take-two Interactive
9	Game consoles
10	Tablets
11	Accessories
12	Warranty
13	Manufacturers a) Microsoft 1. XBOXONE 2. XBOX360 b) Sony 1. PS3 2. PS4
	c) Nintendo 1. Wii 2. WiiU
14	Rent a) Daily b) Monthly c) Yearly

15	PreOrder
16	Trade in
17	Online

CLASS DIAGRAM-ANALYSIS MODEL

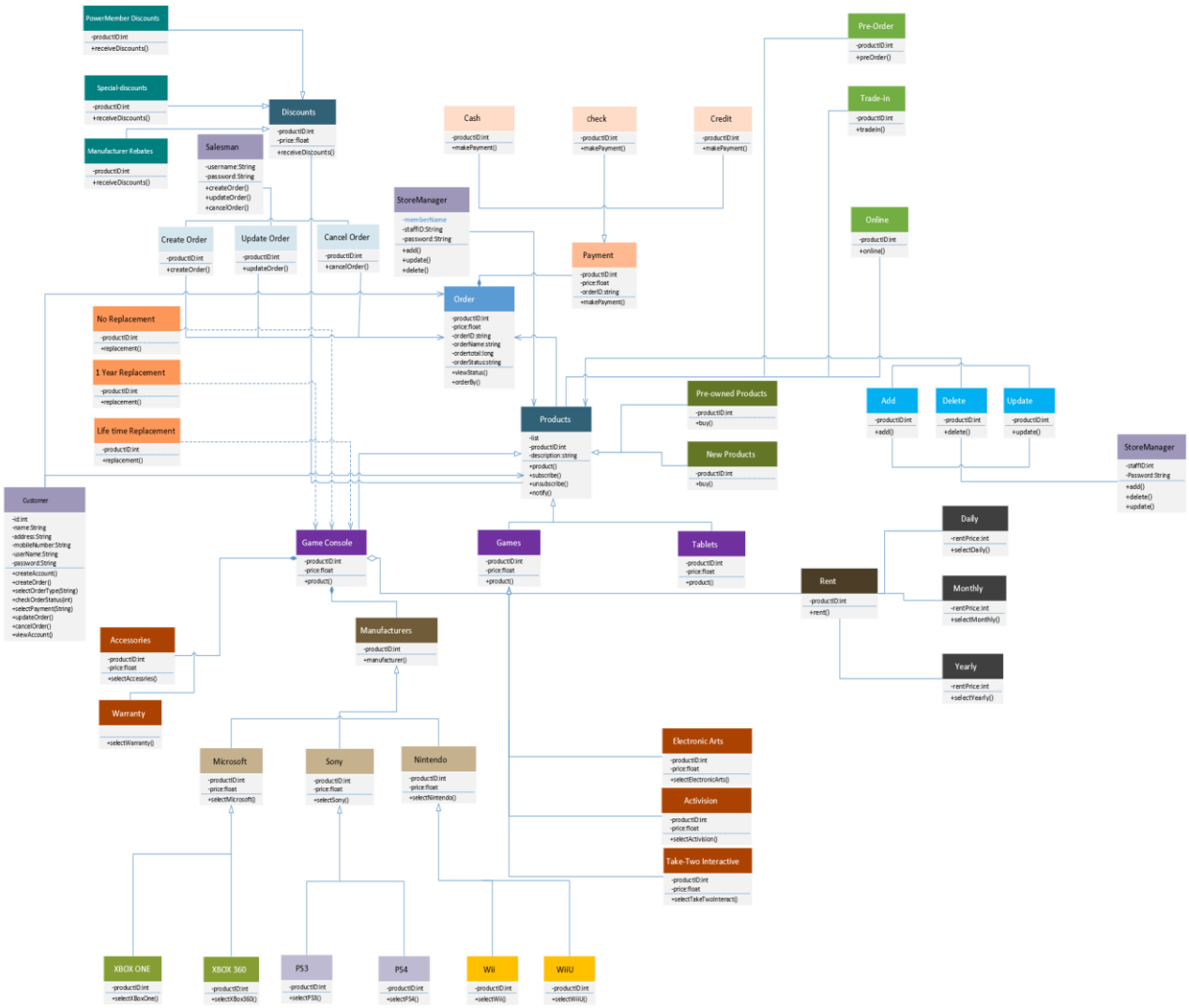
Analysis Class Diagram



3. The Class diagram and package diagram for your design model

A. CLASS DIAGRAM -DESIGN MODEL

Design Class Diagram



A. PACKAGE DIAGRAM-DESIGN MODEL

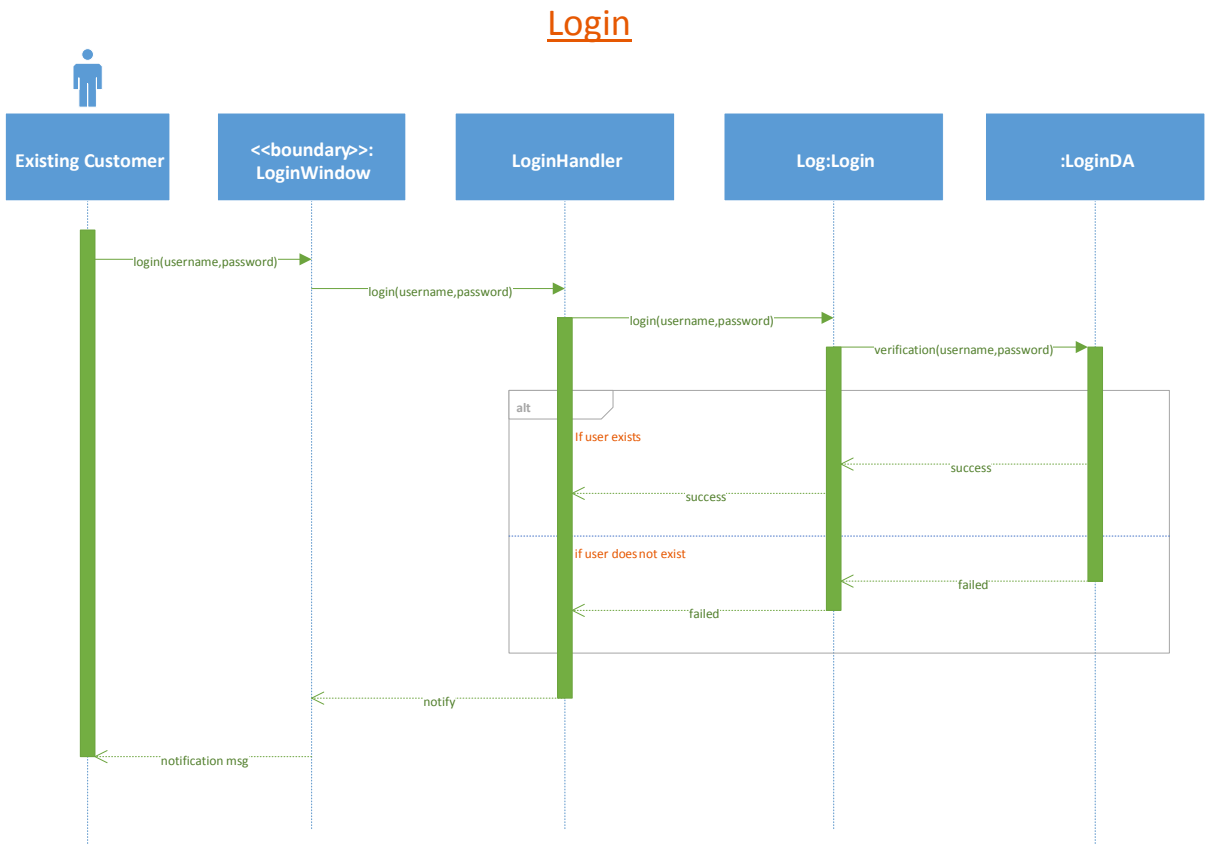
Package diagram is UML structure diagram which shows packages and dependencies between the packages.

4. Any Five sequence interaction diagrams that you can pick form the list of potential sequence interaction diagrams for your design model.

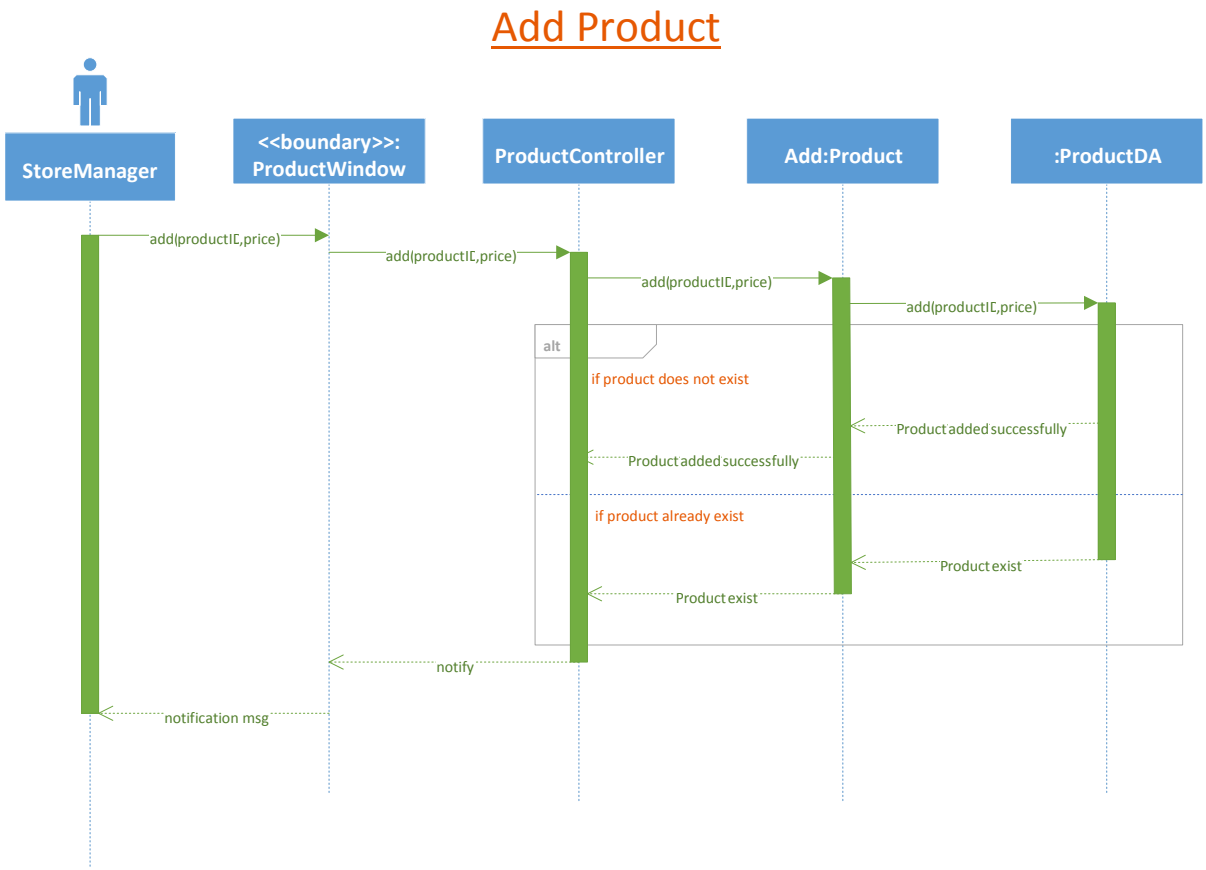
Sequence diagrams describe interactions among classes in terms of an exchange of messages over time. They're also called event diagrams. A sequence diagram is a good way to visualize and

validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.

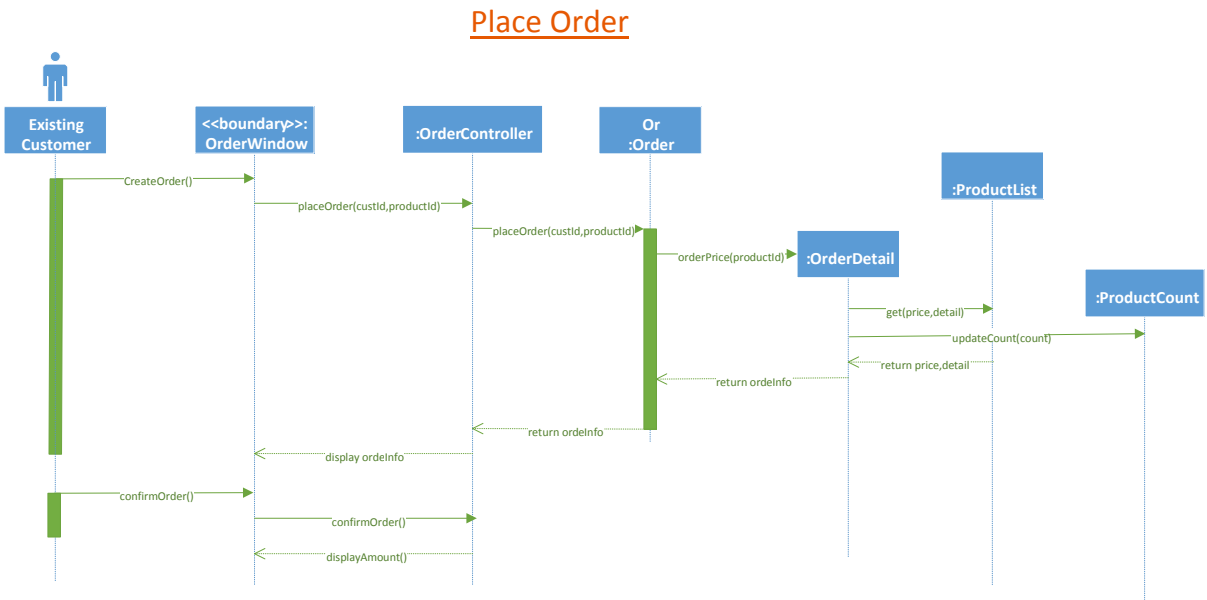
1. Login



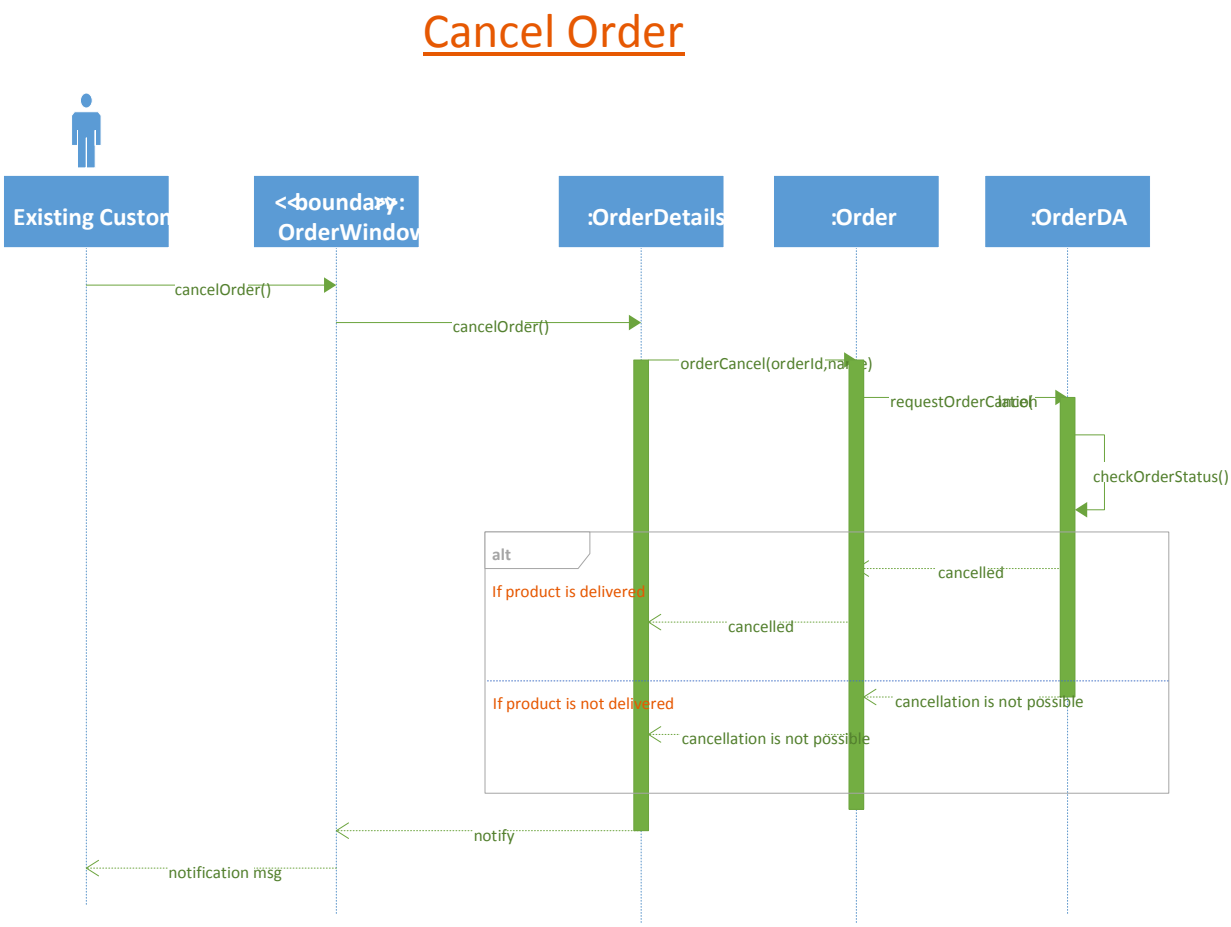
2.Add Product



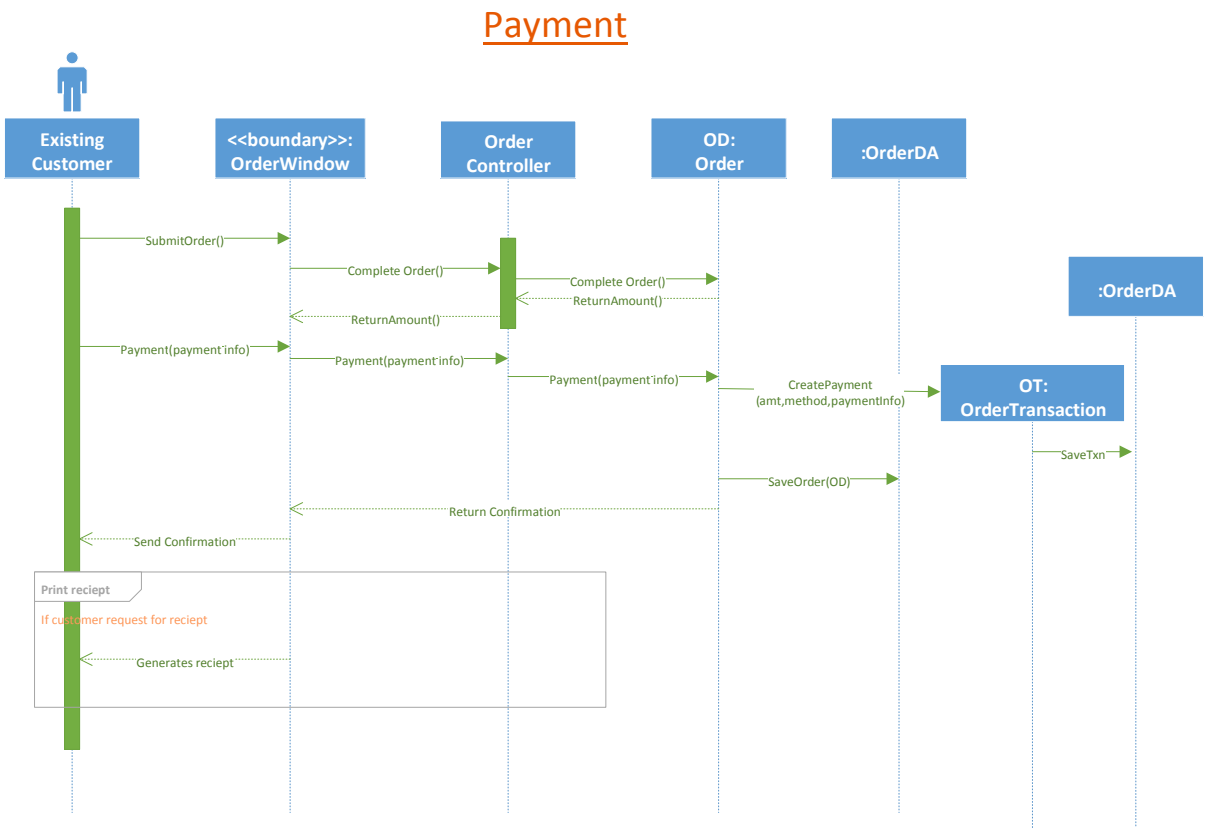
3. Place Order



4.Cancel Order

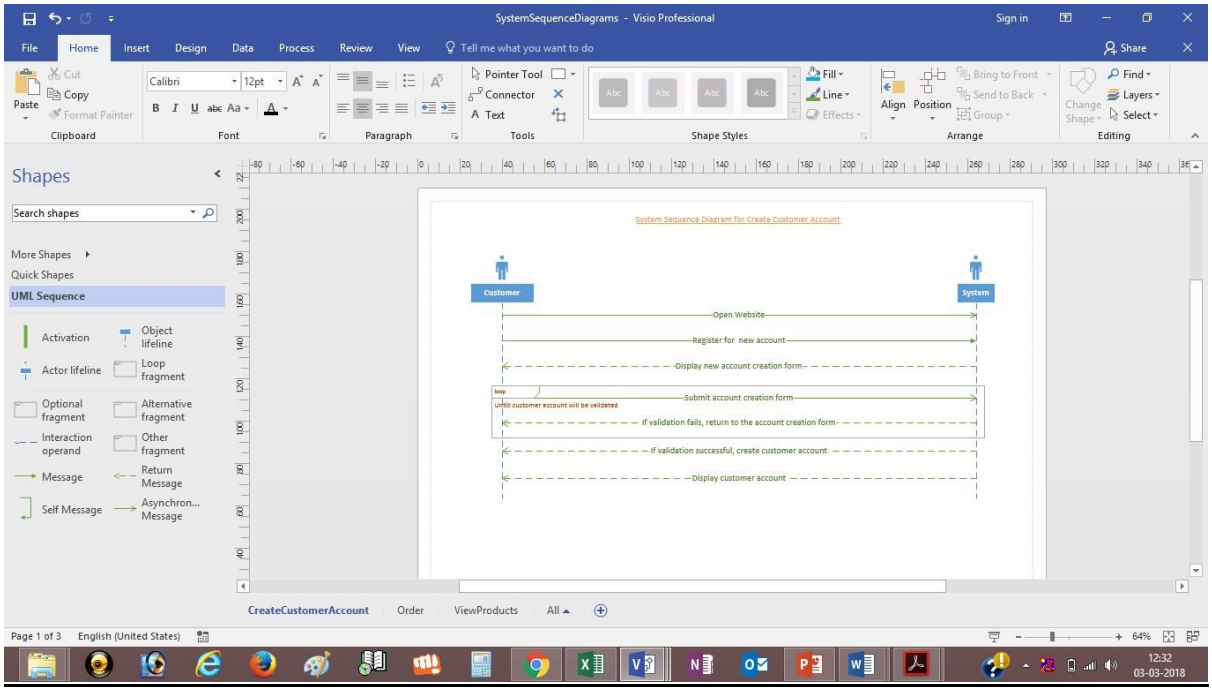


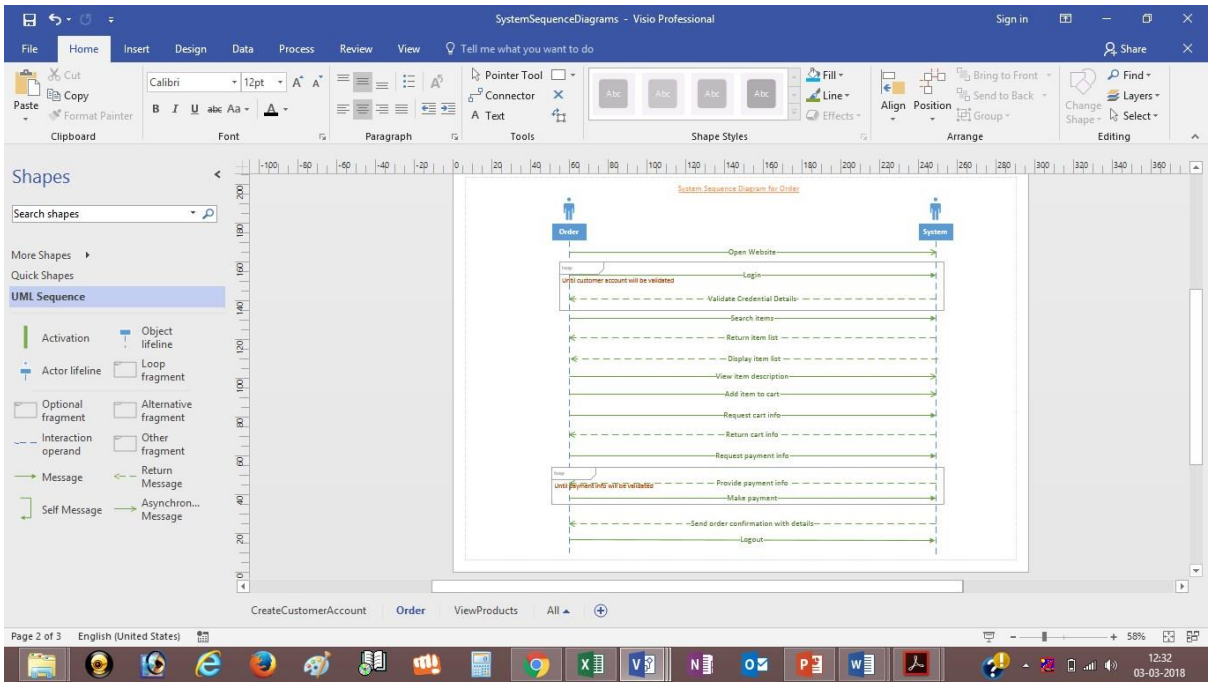
5. Payment

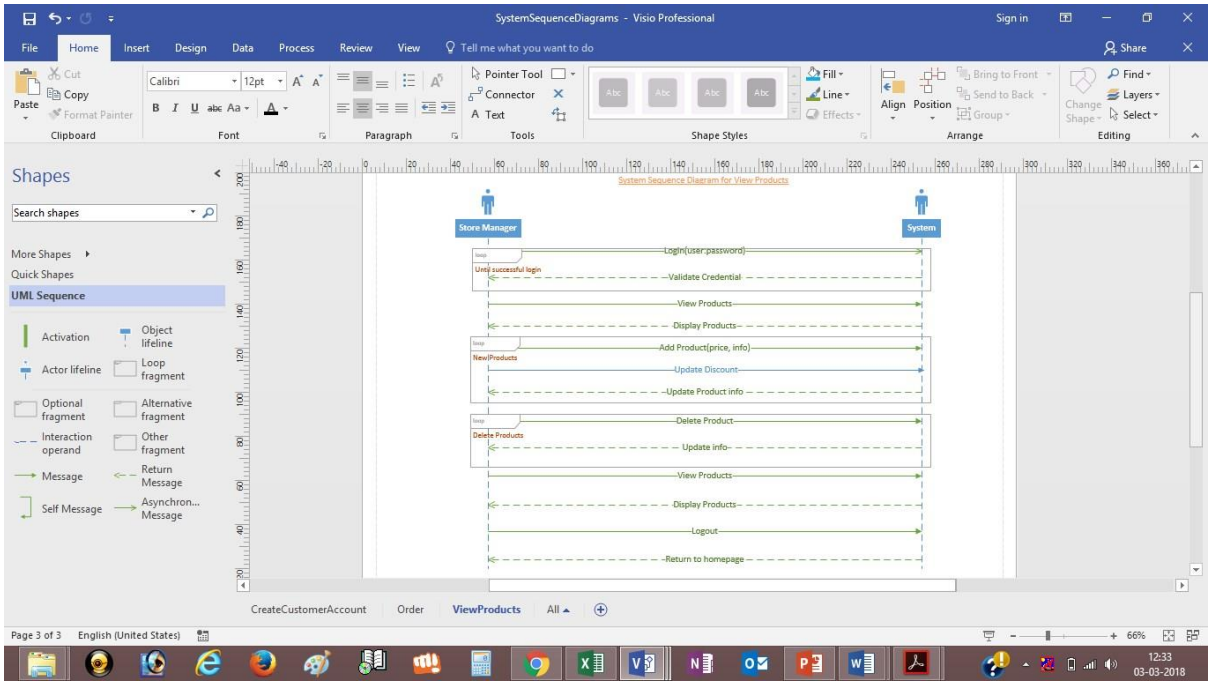

Payment
Existing Customer
<<boundary>>: OrderWindow
Order Controller
OD: Order
:OrderDA
:OrderDA
SubmitOrder()
Complete Order()
Complete Order()
ReturnAmount()
ReturnAmount()
Payment(payment info)
Payment(payment info)
Payment(payment info)
CreatePayment (amt,method,paymentInfo)
OT: OrderTransaction
SaveTxn
SaveOrder(OD)
Return Confirmation
Send Confirmation
Print reciept
If customer request for reciept
Generates reciept

6. Screenshots

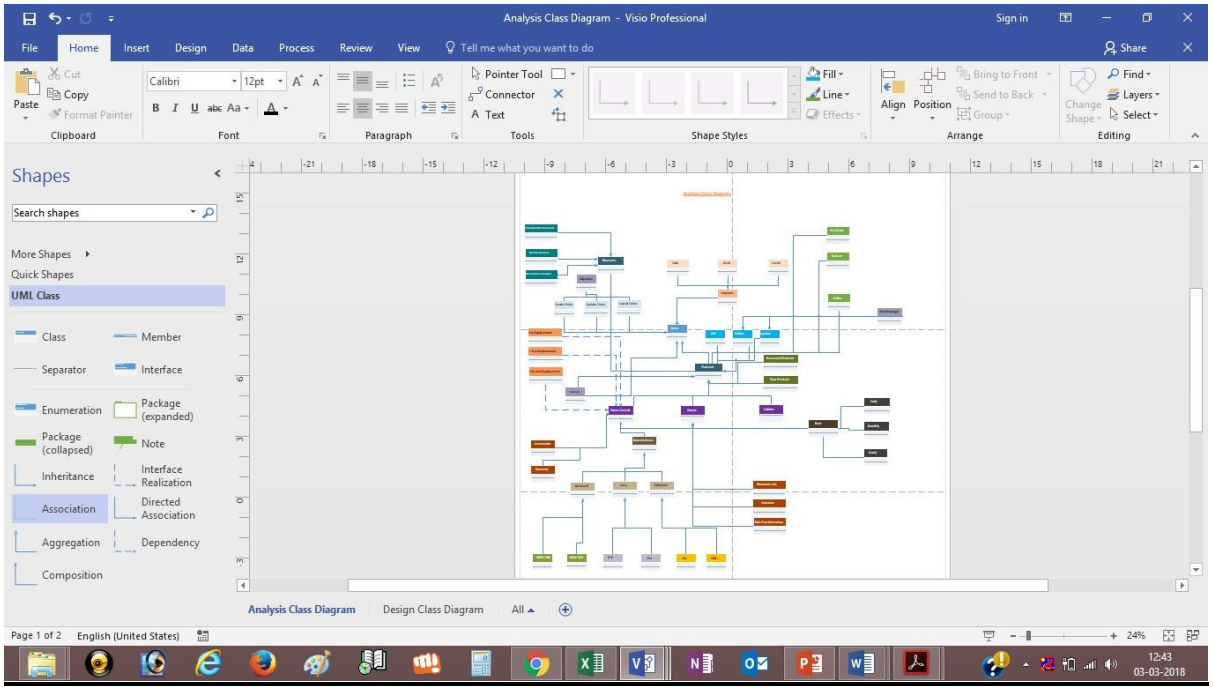
1. System Sequence diagram



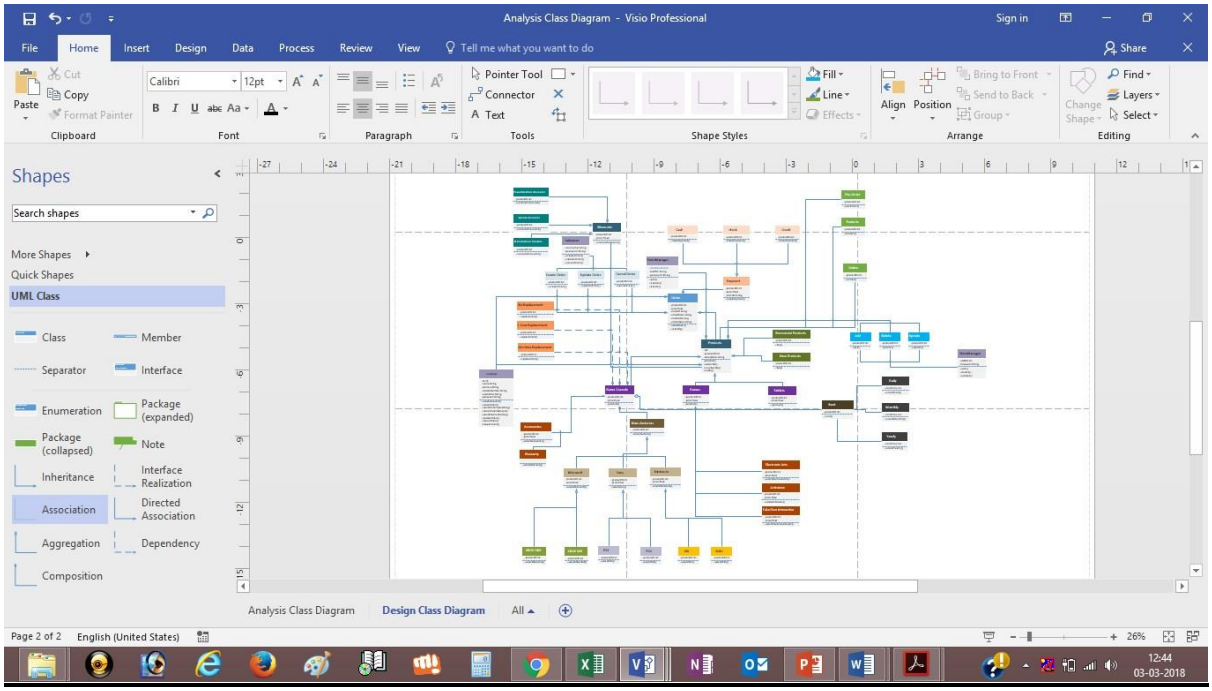




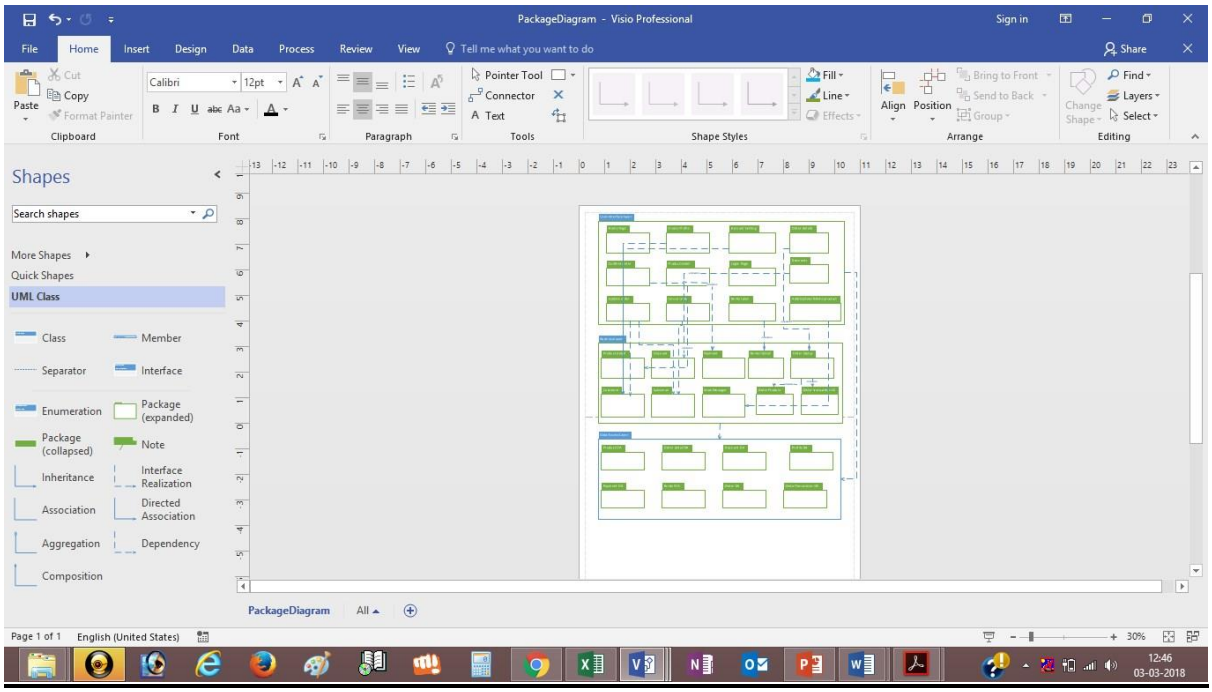
2. Class diagram for analysis model



3. Class diagram for design model

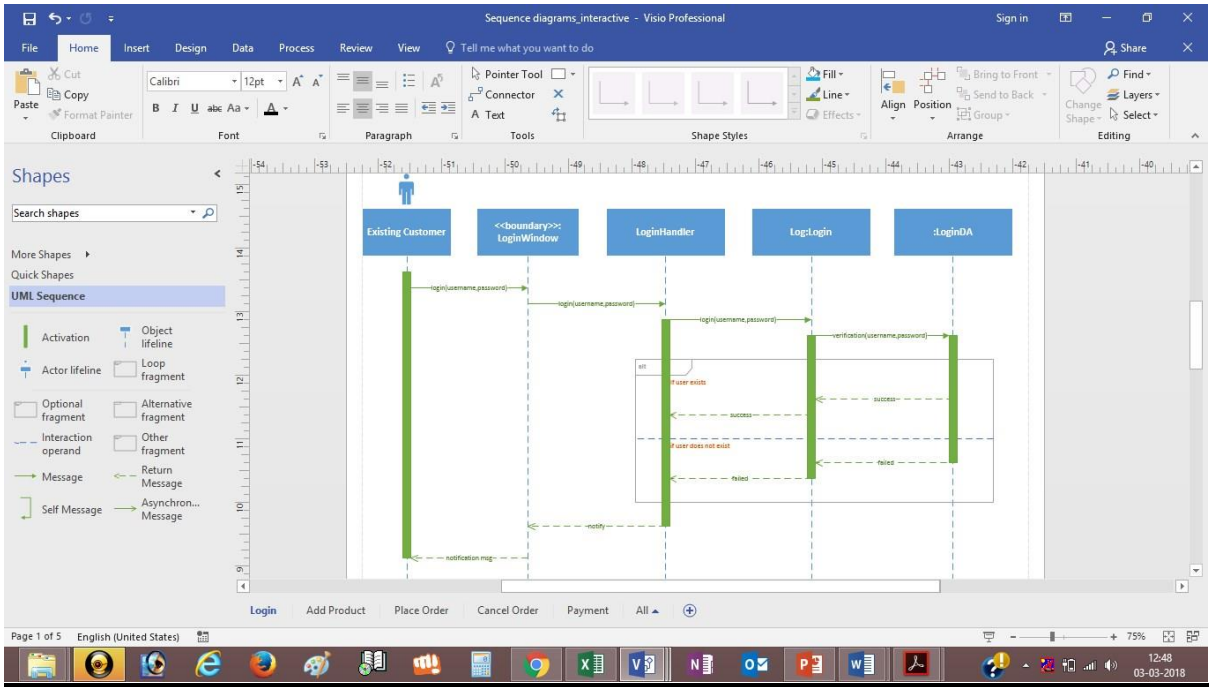


4. Package diagram

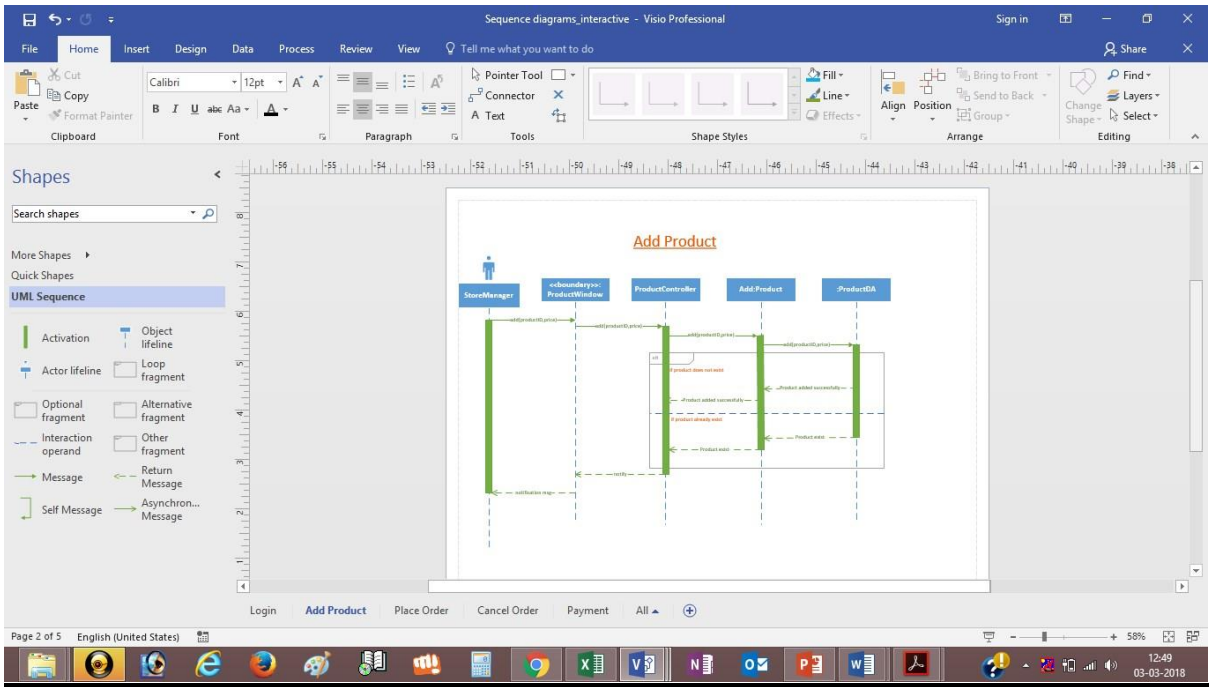


5. Sequence Interaction diagram

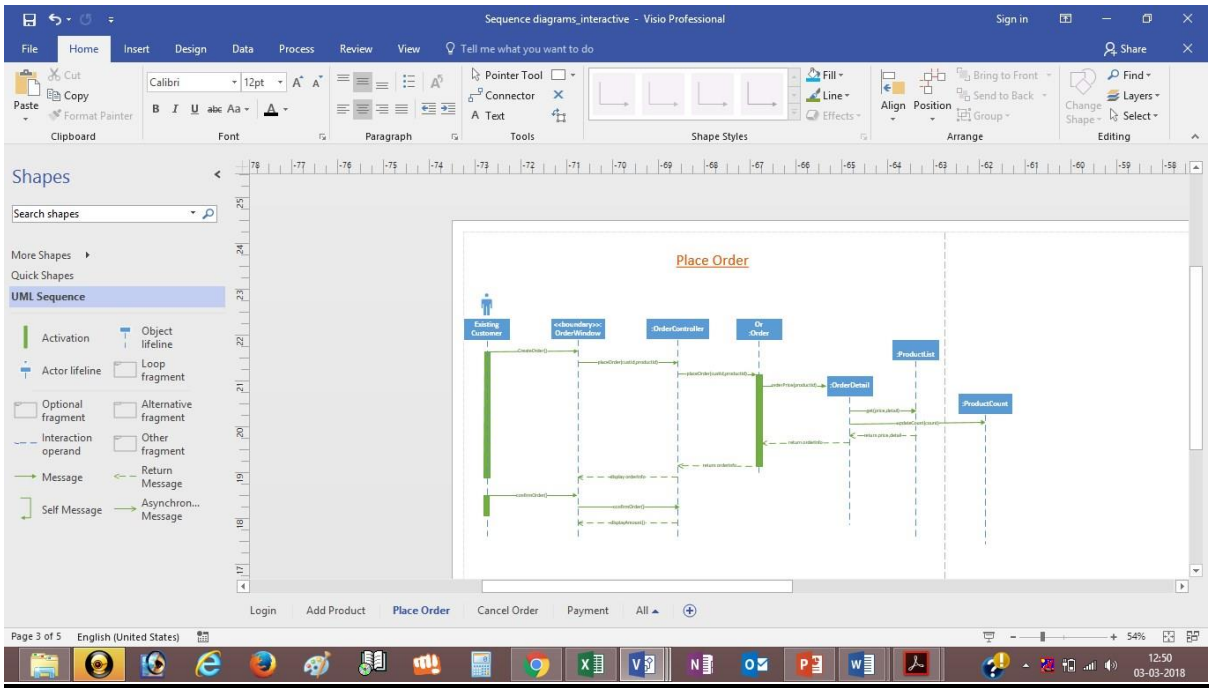
a. Login



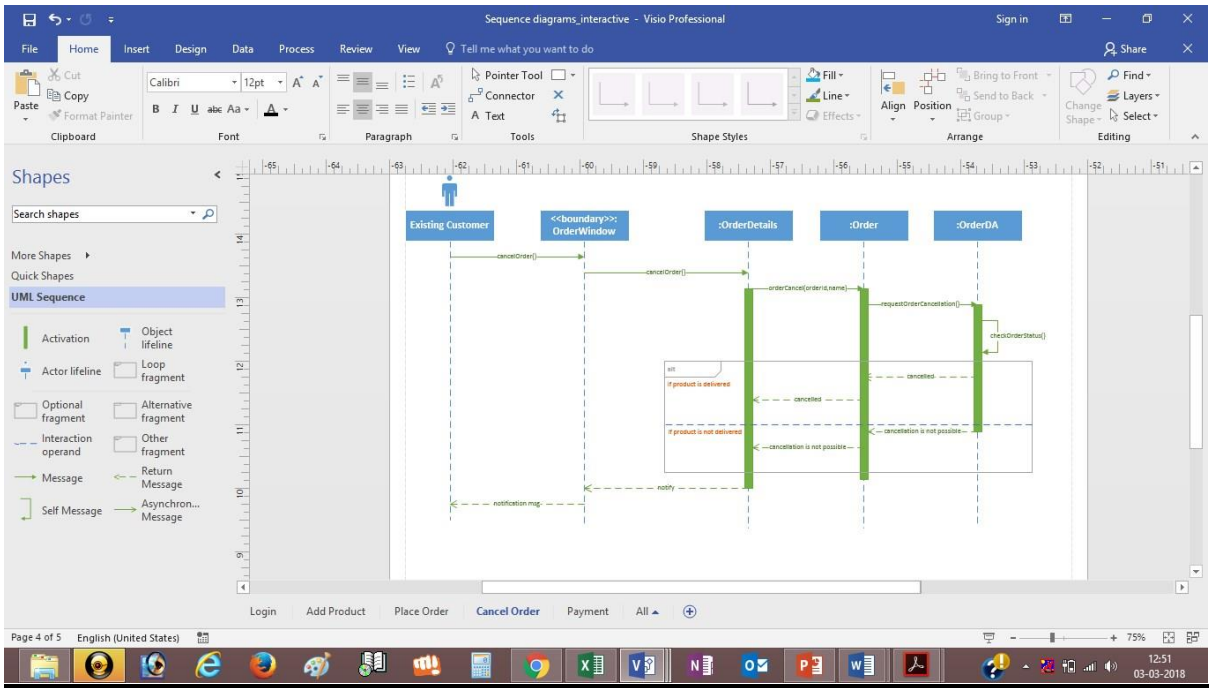
b. Add Product



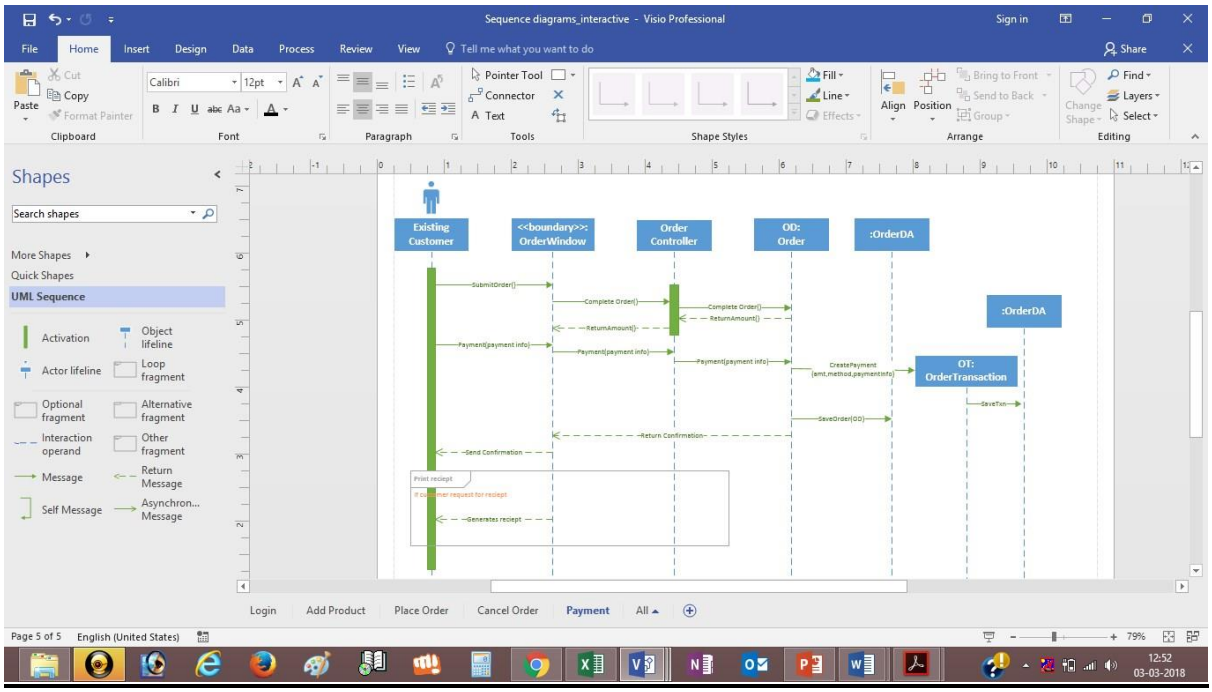
c. Place Order



d. Cancel Order



e. Payment


Sequence diagrams_interactive - Visio Professional
Shapes
Search shapes
More Shapes
Quick Shapes
UML Sequence
Activation
Object lifeline
Actor lifeline
Loop fragment
Optional fragment
Alternative fragment
Interaction operand
Other fragment
Message
Return Message
Self Message
Asynchron... Message
Existing Customer
<<boundary>>: OrderWindow
Order Controller
OD: Order
:OrderDA
:OrderDA
OT: OrderTransaction
SubmitOrder()
Complete Order()
ReturnAmount()
Payment(payment info)
CreatePayment
SaveOrder(OD)
SaveTxn
Return Confirmation
Send Confirmation
Print reciept
Generates reciept
Login
Add Product
Place Order
Cancel Order
Payment
All
Page 5 of 5
English (United States)
79%
12:52
03-03-2018