

MALVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



SOFTWARE ENGINEERING PROJECT REPORT

Online Business Management System

Course - CST311: SOFTWARE ENGINEERING

Submitted To

Dr. Girdhari Singh

Submitted By – Divy Bansal

ID - 2017UCP1720

Table of Contents

1. *Objective*
2. *Overall Description*
3. *Software Requirements Specification Report*
 - a. *Functional Requirements*
 - b. *Non-Functional Requirements*
4. *Software Design Specification*
 - a. *Structured*
 - i. *Data Flow Diagrams*
 - ii. *ER Diagrams*
 - iii. *State Chart Diagram*
 - b. *Object Oriented*
 - i. *Use Case Diagram*
 - ii. *Class Diagram*
 - iii. *Sequence Diagram*
 - c. *Some Algorithms*
5. *Various Modules*
6. *Database Table Design*
7. *Testing*

OBJECTIVE

This Project is developed to serve as a platform for buyers and sellers to buy/sell commodities online.

If you have a physical store, you are limited by the geographical area that you can service. With an online website, the whole world is your playground.

Physical retail is driven by branding and relationships. In addition to these two drivers, online retail is also driven by traffic from search engines. It is not unusual for customers to follow a link in search engine results and land on an e-commerce website that they have never heard of. This additional source of traffic can be the tipping point for some small businesses.

E-commerce facilitates comparison shopping. There are several online services that allow customers to browse multiple e-commerce merchants and find the best prices.

Buyers and sellers of niche products can find it difficult to locate each other in the physical world. Online, it is only a matter of the customer searching for the product in a search engine. One example could be the purchase of obsolete parts. Instead of trashing older equipment for lack of spares, today we can locate parts online with great ease.

Overall Description

Product Perspective:

Online website based e-commerce is an outstanding way of bringing Sellers and Customers on an online platform to make purchase in a secured and efficient manner irrespective of distance between the two. The product is one stop for many products online.

Product Function:

- Provide a simple interface and platform to ease the process of buying as well as selling products online
- Include smooth functionality and efficiency that adds to buyers' confidence.
- Website keeps a constant focus on new category creation and expansion of products.
- Tracking feature of orders and pending deliveries for both sellers and buyers.
- Notifying order and delivery reports to customers and sellers.

User Classes and Characteristics:

Customer - He or she is a verified user of website who is intended to buy a product from the seller via the online platform. The customer must have a username and password to make a purchase.

Seller - He or she is a verified person who is allowed to sell items over the platform. Seller's details are stored on database and all the products are listed under him that he is ready to sell or are available. He is responsible to set products details, price, and quantity.

Administrator - He or she is responsible for monitoring functions and procedures on Platform. Administrator is responsible to provide valid information of a purchase to the concerned authority in case of any dispute between the customer and seller or in case of exchange.

Design Implementation:

Client-side Programming Language	
JavaScript	JavaScript is a lightweight, object-oriented, cross-platform scripting language, mainly used within web pages.
HTML5	HTML5 is the fifth revision of the HTML standard.
CSS	Cascading Style Sheets define style rules in a separate CSS file.

Operating Environment:

Recommended browsers are Chrome, Firefox, Safari and Internet Explorer 8 or higher.

articulars	Client System	Server System
Operating System	Windows/Linux/Android/iOS	Linux
Processor	Intel or AMD	Intel or AMD
Hard Disk	1 GB	1 TB
RAM	256	8GB

Assumptions and Dependencies

- The customer and seller must have basic knowledge of computers and English language.
- Each User must have a User ID and password.
- Each Seller must have Seller ID and password.
- There must be an Administrator.
- Internet connection is a must.
- Proper browsers should be installed in the user's system.

Specific Requirement Specification Report

Functional requirements:

This subsection contains the requirements for the website. These requirements are organized by the features discussed in the product functions. Features from there, they are then refined into use case diagrams and to sequence diagram to best capture the functional requirements of the system.

- *Displaying Related Items - Display available Items to the customer. Enable user to add different items to add to cart / buy*
- *Provide shopping cart facility - The system shall provide shopping cart during online purchase. The system shall allow user to add/remove products in the shopping cart. Later customer can confirm orders for purchase.*
- *Online tracking of Orders - The system shall allow user to enter the previous order section and display information for tracking. The system shall display the current tracking information about the order*
- *Create Sellable items on website - Provide an interface to all the sellers to add product, quantity available and price through their account.*
- *Update Quantity Of Item Available – Provide an interface to all the sellers to update quantity of the item available on the website*
- *Update Status Of Order – Provide an interface to all the sellers to update the status of order placed by the customer*





Non-Functional Requirements:

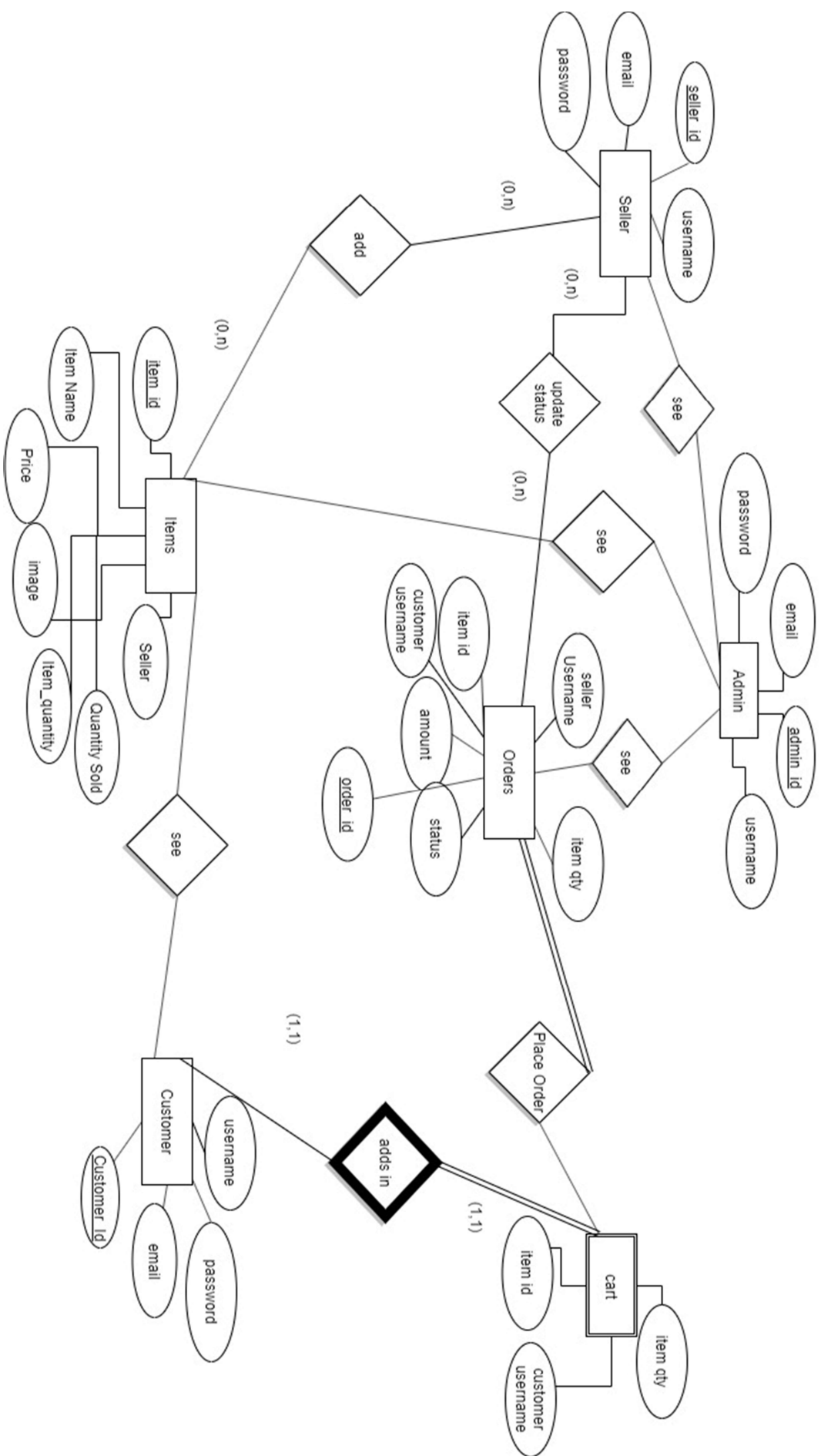
- Performance: The product shall be based on web and has to be run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. The performance shall depend upon hardware components of the client/customer.
- Security : The system shall use secure sockets in all transactions that include any confidential customer information like password
- Data Storage - The system's back-end servers shall never display a customer's password. The customer's password may be reset but never shown.
- Accessibility: The system's back-end servers shall only be accessible to authenticated administrators. The system's back-end databases shall be encrypted and within company's perimeter.
- Reliability: The system provides storage of all databases on redundant computers with automatic switchover. The reliability of the overall program depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.
- Maintainability: A commercial database is used for maintaining the database and the application server takes care of the site. Also the software design is being done with modularity in mind so that maintainability can be done efficiently

Data flow Diagram (DFD):

A Data Flow Diagrams is a structured analysis and design tool that can be used for flowcharting in place of, or in association with, information-oriented and process-oriented systems flowcharts. A DFD is a network that describes the flow of data and the processes that change, or transform, data throughout a system. This network is constructed by using a set of symbols that do not imply a physical implementation. It has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design phase that functionality decomposes the requirement specifications down to the lowest level of detail.

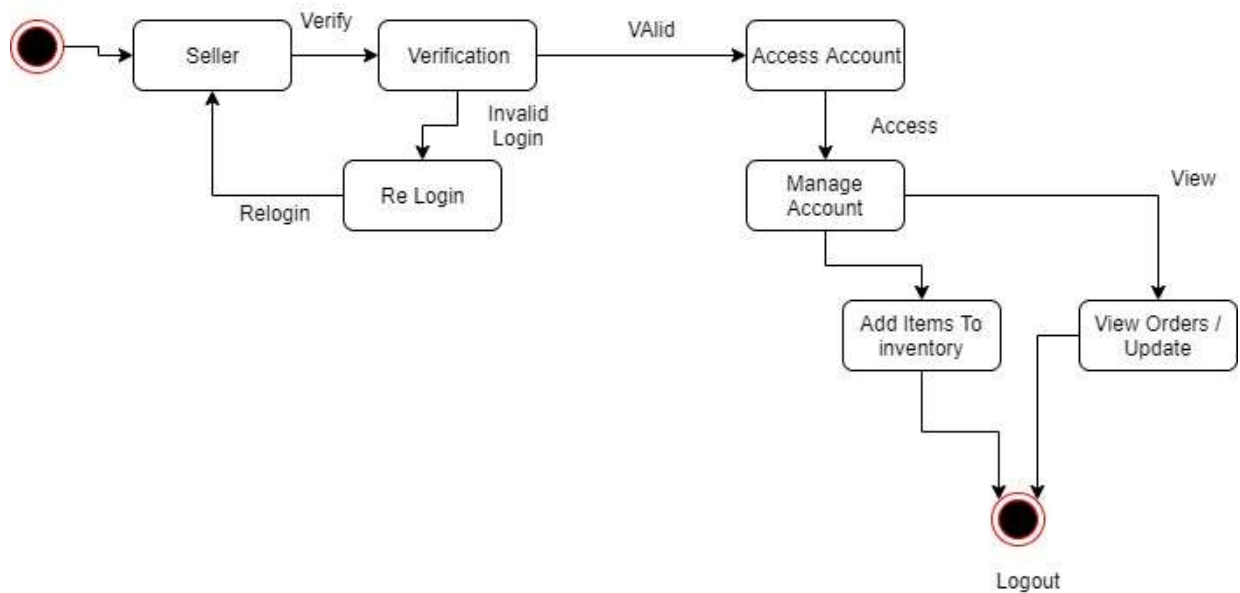
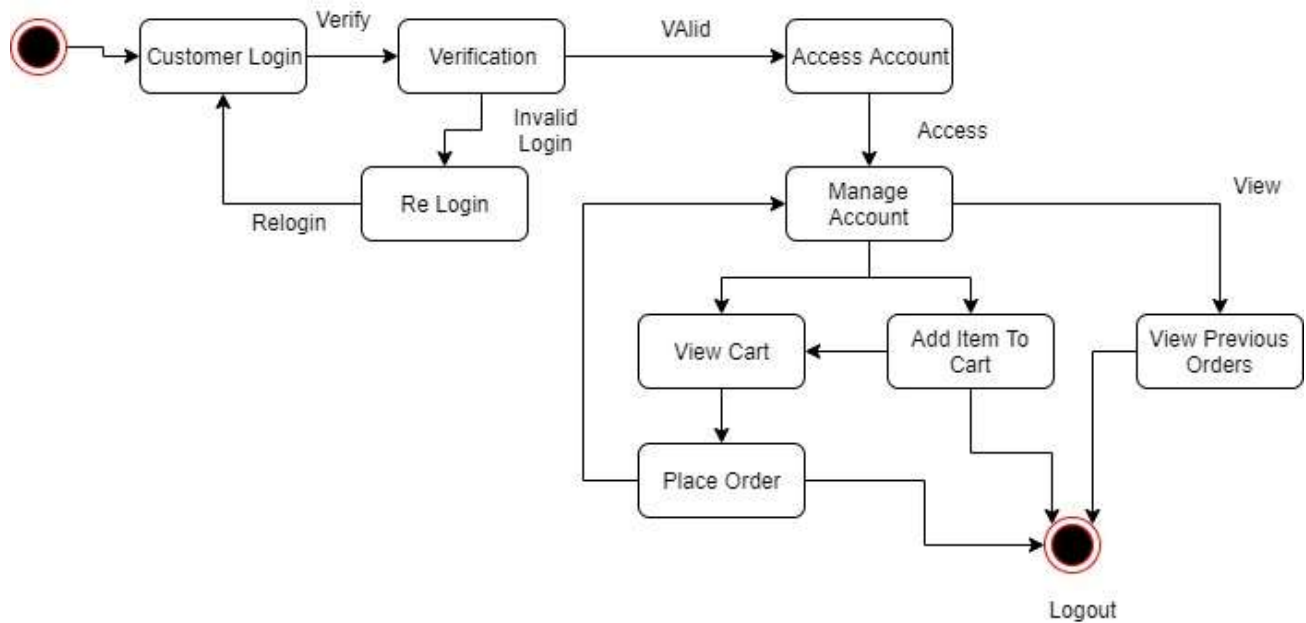
Data flow diagram symbol:

Symbols	Description
	Data Flow: Data flow are pipelines through the packets of information flow.
	Process: A Process or task performed by the system
	Entity: Entity are object of the system. A source or destination data of a system
	Data Store: A place where data to be stored




State Chart Diagram:

- It illustrates the in-trusting events and state of an object and behaviours of an object is reaction to an event. Transaction shows as allows labelled with their event. It is included with initial pseudo state and final end state.




Database Table Design:


Cart

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	cart_id 	int(11)			No	None
<input type="checkbox"/>	2	client_username	varchar(255)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	3	item_id	int(255)			No	None
<input type="checkbox"/>	4	item_qty	int(255)			No	None


Inventory

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	item_id 	int(10)			No	None
<input type="checkbox"/>	2	item	varchar(255)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	3	price	int(255)			No	None
<input type="checkbox"/>	4	salesman_username	varchar(255)	utf8mb4_general_ci		Yes	NULL
<input type="checkbox"/>	5	image	varchar(255)	utf8mb4_general_ci		Yes	NULL
<input type="checkbox"/>	6	qty_current	int(244)			Yes	NULL
<input type="checkbox"/>	7	qty_sold	int(244)			Yes	0

User

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id 	int(11)			No	None
<input type="checkbox"/>	2	username	varchar(255)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	3	email	varchar(255)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	4	password	varchar(255)	utf8mb4_general_ci		No	None

Order Table

	#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	order_id 	int(10)			No	None
<input type="checkbox"/>	2	client_username	varchar(10)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	3	item_id	varchar(10)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	4	item_qty	varchar(10)	utf8mb4_general_ci		No	None
<input type="checkbox"/>	5	amount	int(100)			No	None
<input type="checkbox"/>	6	order_status	varchar(255)	utf8mb4_general_ci		No	Order is Placed
<input type="checkbox"/>	7	salesman_username	varchar(255)	utf8mb4_general_ci		Yes	NULL

UML Diagrams

UML is a standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems. UML diagrams are not only made for developers but also for business users, common people, and anybody interested to understand the system. The system can be a software or non-software system. Thus it must be clear that UML is not a development method rather it companies with processes to make it a successful system.

UML is a modelling language used to model software and non-software systems. Although UML is used for non-software systems, the emphasis is on modelling OO software applications.

Use Case Diagram:

The UML provides the use case diagram notation to illustrate the name of the use case actors and relationship between them. User case diagrams are used to model the functional interaction between users and system.

Class Diagram:

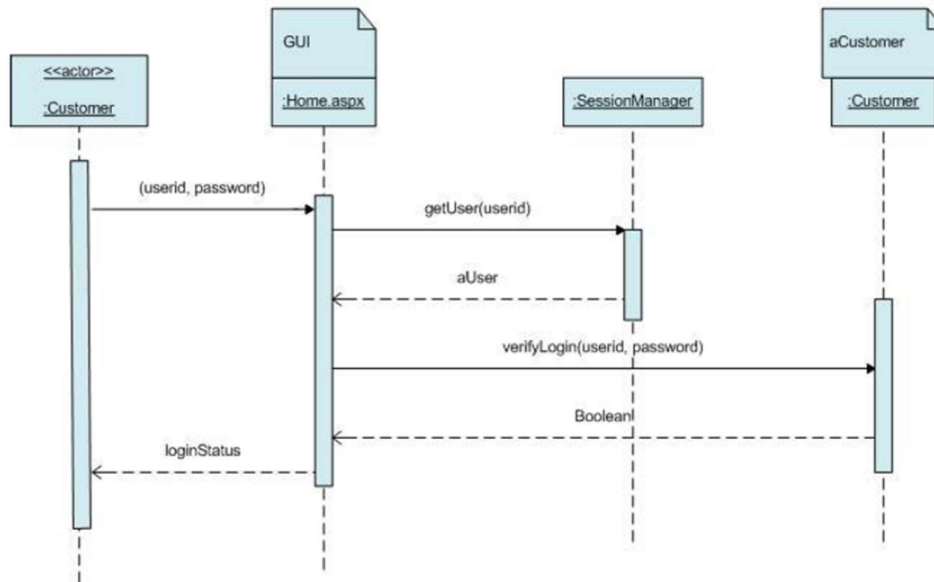
- Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.
- Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

Sequence Diagram:

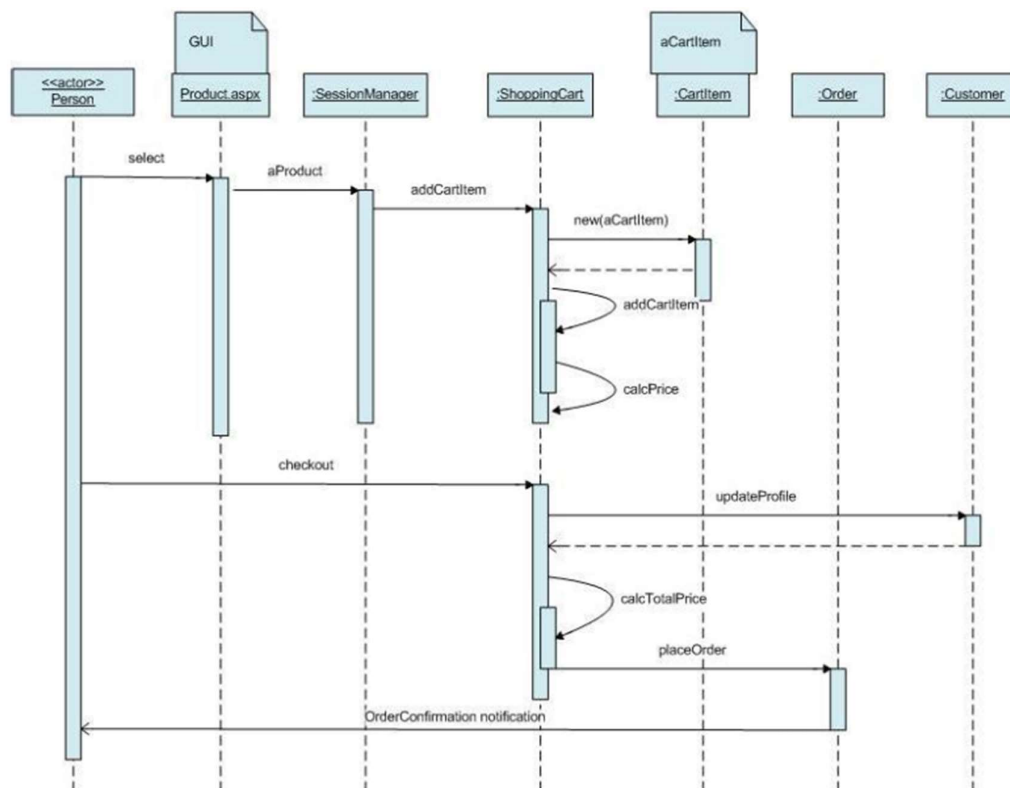
A sequence diagram illustrates in a kind of format in which each object interacts via Messages. It is generalization between two or more specification diagram.

Sequence diagram is an interaction over view diagram. It provides a big picture over view of now a set of interaction is related in terms of logic and process flow.

Login Sequence Diagram:



Buying (Add Product to Shopping Cart) Sequence Diagram:



TESTING

Features to be tested

This section outlines all the features that will be tested:

Type of User	Feature Identifier	Description
User	Case-1	System Register
	Case-2	Edit Shopping Cart
	Case-3	Add to Cart

CASE 1: System Register

Purpose: Test that users can register with the proper username and password

1. Visit Customer's Login web page
2. Enter Username
3. Enter Email
4. Enter Password
5. Click Signup button

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1	Enter empty value for <i>Name</i>	Display error message to enter some valid text.	Error: "Name is required, Please Enter Correct details".	✓		
2	Enter empty value for <i>User ID</i>	Display error message to enter some valid text.	Error: "This Field is required".	✓		

3.	Enter a Username not in use with other existing users.	User should be able to register with the website and directed to the secure Web page requested.	No Error	✓		
4.	Enter empty value for <i>Email Address</i>	Display error message "Enter a valid address"	Error message: "Enter a valid address"	✓		
5.	Enter empty value for either <i>Password</i> or <i>Confirm Password</i>	Display error message "Password does not match."	Error message: "Password does not match."	✓		

CASE 2: Edit Shopping Cart

Purpose: Test that clicking Update Quantities will update the cart summary accordingly.

1. After selecting a product, go to Shopping cart Web page.
2. Check after entering incorrect input, an appropriate message should be displayed.
3. If entered a valid number, check if the total quantity and relative price is updated after clicking update or delete button

S/N	Input data	Expected output	Actual Result	Pass	Fail	Remarks
1.	Negative input number or input other than integer number in "Quantity" field	Display error message to notify the given input is invalid	Error message: "Please input valid no. of items"	✓		
2.	Enter a Positive integer number in the "Quantity" field	The product quantity should be updated or deleted according to the specified input number	No Error. "The item is updated successfully"	✓		

CASE 3: Add to Cart

Purpose: Test that clicking add to cart button, product is getting added in the cart

1. Click Add to Cart button
2. Check whether the cart shows the product

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1.	Click "Add to Cart" Button for product not already in the cart.	The product should be moved to cart.	No Error. "The product is successfully added to Cart "	✓		
2.	Click "Add to Cart" Button for product already in the cart.	If product already exists then Display Error	Error. " Product Already Present in cart "	✓		

Modules

Login ()

Login Module Enables the User of the website either seller or the buyer to either register on the website or if already registered login to the website.

AddToCart ()

This Module facilitates the buyer to save the items he want to buy in the cart. The Buyer can login again on the website and see his/her cart

Order ()

Enables to order the items currently present in the module

AddToInventory ()

Enables the seller to add products on the website.

Update Inventory ()

Enables the seller to update the quantity of items currently in the inventory

UpdateOrderStatus ()

Enable The Seller to Update the Orders that he is responsible for

Some Algorithms:

User Validation:

```
$query_reg = "select * from client where username='$user_val' && password = '$pass_val' ";
$result = mysqli_query($conn,$query_reg);
$num_reg = mysqli_num_rows($result);
if($num_reg ==1){
    $_SESSION['user'] = $user_val;
    header('location:client_home.php');
}else{
    echo "<script>
        alert('Username / Password Incorrect Login Again');
        window.location.href='client_login.php';
    </script>";
}
```

ADD TO CART :

```
$query_reg = "select * from cart where client_username='$client_username' && item_id = '$item_id' ";

$result = mysqli_query($conn,$query_reg);
$num_reg = mysqli_num_rows($result);

        $query = "select qty_current from inventory where item_id='$item_id'";
        $result = mysqli_query($conn,$query);
        $value = mysqli_fetch_object($result);
        $qty_current = $value->qty_current ;

        if($quantity > $qty_current){
            echo "<script>
                alert('Quantity Entered Is Not available');
                window.location.href='client_home.php';
            </script>";
        }

else if($num_reg >0 or $quantity <= 0 ){

    echo "<script>
        alert('Item Is Already In Cart / Item cannot be added');
        window.location.href='client_home.php';
    </script>";
}

else{

        $query_reg = "insert into cart (client_username,item_id,item_qty) values
('$client_username','$item_id','$quantity)";
        mysqli_query($conn,$query_reg);
        echo "<script>
            alert('!! Item Added To Cart !! ');
            window.location.href='client_home.php';
        </script>";
}
```

Order :

```
while($num > 0)
{

$query = "SELECT * FROM cart where client_username='$client_username'";
$result = mysqli_query($conn, $query);
$row = mysqli_fetch_array($result);
$num=mysqli_num_rows($result);

$item_id=$row["item_id"];
$item_qty=$row["item_qty"];

$query = "select price from inventory where item_id='$item_id' limit 1";
$result = mysqli_query($conn,$query);
$value = mysqli_fetch_object($result);
$amount = $value->price*$item_qty ;

$query = "select salesman_username from inventory where item_id='$item_id' limit 1";
$result = mysqli_query($conn,$query);
$value = mysqli_fetch_object($result);
$salesman_username = $value->salesman_username ;

$query = "select qty_current from inventory where item_id='$item_id' limit 1";
$result = mysqli_query($conn,$query);
$value = mysqli_fetch_object($result);
$qty_current = $value->qty_current ;
$qty_current=$qty_current-$item_qty ;

$query = "select qty_sold from inventory where item_id='$item_id' limit 1";
$result = mysqli_query($conn,$query);
$value = mysqli_fetch_object($result);
$qty_sold = $value->qty_sold ;
$qty_sold=$qty_sold+$item_qty ;

$query_reg = "insert into orders (client_username, item_id, item_qty, amount,salesman_username)
values ('$client_username','$item_id','$item_qty','$amount','$salesman_username')";
mysqli_query($conn,$query_reg);

$query_reg = "update inventory set qty_current='$qty_current' where item_id='$item_id' ";
mysqli_query($conn,$query_reg);

$query_reg = "update inventory set qty_sold='$qty_sold' where item_id='$item_id' ";
mysqli_query($conn,$query_reg);

$query_reg = "delete from cart where client_username='$client_username' and item_id=$item_id ";
mysqli_query($conn,$query_reg);

$num=$num-1;

}
```