Contents

[1.0 Summary 1](#_Toc36929534)

[2.0 Requirement Analysis 1](#_Toc36929535)

[2.1 Functional Requirements 1](#_Toc36929536)

[2.2 Use Case Modelling 2](#_Toc36929537)

[2.2.2 Use Case Description 3](#_Toc36929538)

[2.2.3 Use Case Diagram 9](#_Toc36929539)

[3.0 System Design 9](#_Toc36929540)

[3.1 List of actions found at 4# Order Use Case 9](#_Toc36929541)

[3.2 Activity diagram for use case #4 order 10](#_Toc36929542)

**Online Shopping System**

# 1.0 Summary

The project aims to develop a web based online shopping system to help a business who primarily manufacture and sell computers. The system should allow customers to view, select and purchase products directly from the manufacturers website. All products need to be categorised in three groups and they are 1. Servers 2. Desktops 3. Laptops. Customers can either purchase his desired product as they are presented on the frontend, or they can customize their order using the system if they wish to get different configuration. The system should alter the price of the product after calculating price of newly added components.

To complete the ordering process, the customer will be asked to provide payment details along with delivery address. The system must verify these credentials and if they are correct the order gets recorded, then the system will issue a confirmation email to the customer. The confirmation email will hold order details to track its status. Acceptable payment methods are credit card and cheques. Verification of these two payment methods are different, as a result the system should have separate flow of works for them.

After the payment verification is completed, the salesperson then forwards the order to the warehouse and prints invoice.

# 2.0 Requirement Analysis

## 2.1 Functional Requirements

1. The customer views the standard configuration of the chosen server, laptop or computer on the manufacturers online shopping webpage along with the costs.
2. Customer wishes to view details of the configuration and may have the thoughts of purchasing it default or changing the configuration to match their satisfaction. The costs will be calculated upon the customers requirement.
3. Customer either orders the required essential online or they may request to speak to a salesperson, so they are able to get the relevant details provided about the order and discussing prices before an order is ready to take place. It is evident that the salesperson will need an interface to record the order on behalf of the customer.
4. To place an order, an online form must be filled out by the customer along with the invoice address, method of payment ( credit cards, cheques) and shipment.
5. As soon as the customer’s order gets recorded onto the system, the salesperson will send an electronic request to the warehouse with the details of the ordered configurations.
6. The relevant purchase details that will be required by the customer such as the order number and the account number is mailed to the customers email address so that they will be able to check their order status online.
7. The invoice the salesperson sent is received by the warehouse and the required essentials are shipped out to the customer.

## 2.2 Use Case Modelling

2.2.1 Actors

Primary actors of the system are customer, salesperson and the warehouse.

Customer: Anyone using the web system to purchase a product.

Salesperson: Company employees who have access to the system.

Warehouse: Responsible for managing inventories and shipping products.

2.2.1 Use Cases

|  |  |  |
| --- | --- | --- |
| **Use Case** | **Actor** | **Use case** |
| 1 | Customer, Salesperson | View products |
| 2 | Customer, Salesperson | Change components of the product |
| 3 | Customer | Request Call back |
| 4 | Customer | Order (submits without option to add discounts) |
| 5 | Customer, Salesperson | View order status |
| 6 | Salesperson | View orders |
| 7 | Salesperson | View call back requests |
| 8 | Salesperson | Record Order (has option to add discounts) |
| 9 | Salesperson | Verify Payment |
| 10 | Salesperson | Print Invoice |
| 11 | Salesperson | Send product request to warehouse |
| 12 | Salesperson, Warehouse | Update order status |
| 13 | Warehouse | View Orders from Salesperson |

## 2.2.2 Use Case Description

|  |  |
| --- | --- |
| **Use Case #1** | **View products** |
| Brief Description | Actor browse products and selects desired product and view its details. |
| Actors | Customer, Salesperson |
| Pre-Conditions | Actor visits the web page |
| Main Flows | 1. Products are displayed in a list. 2. Presses “view details” button of a product. 3. Single product page loads, this has full details and default components of the requested product. |
| Alternative Flows | 1. List products from a single category by selecting appropriate category from 3 given categories- a)Laptop b) Desktop c) Server |
| Post-Conditions | If the use case is successful, a database query will be sent to request details of a single product. |

|  |  |
| --- | --- |
| **Use Case #2** | **Change components of the product** |
| Brief Description | Actor modify component set of the product and system calculates price based on modifications. |
| Actors | Customer, Salesperson |
| Pre-Conditions | Actor is in the single product page. |
| Main Flows | 1. Add new components. 2. View updated product details. |
| Alternative Flows | 1. Replaces default components |
| Post-Conditions | If the use case is successful, then system will add new components or replace default components of the requested product and re-calculate the price. |
| **Use Case #3** | **Request Call back** |
| Brief Description | Request Call back |
| Actors | Actor enter his details and submits a form to request call back. |
| Pre-Conditions | Customer is on request call back page |
| Main Flows | 1. Customer fills out name and telephone number 2. Submits the form |
| Alternative Flows | 1. Customer does not complete required fields 2. Submits the form 3. System shows validation error. |
| Post-Conditions | If the use case is successful a database entry is made with the customers requests for a call back, otherwise validation error is shown to the customer. |

|  |  |
| --- | --- |
| **Use Case #4** | **Order** |
| Brief Description | Actor submits order using the frontend interface. |
| Actors | Customer |
| Pre-Conditions | Customer has selected the product configuration. |
| Main Flows | 1. Add product to the cart. 2. Page refreshes and product is added to the cart. 3. Goes to **pay now** page. 4. Inserts shipping address 5. Selects payment method. 6. Inserts payment details 7. Submits the order. 8. Receives confirmation email with payment verification information and order number. |
| Alternative Flows | 1. Customer continues shopping. |
| Post-Conditions | If the use case is successful, the order is recorded in the database. If not, then the state does not change. |

|  |  |
| --- | --- |
| **Use Case #5** | **View order status** |
| Brief Description | Actor uses order number to view its status. Status are categorised in three categories and they are a) Order received b) On its way c) Delivered |
| Actors | Customer, Salesperson |
| Pre-Conditions | 1. Order must be recorded. 2. Order number must be present. 3. Payment must be verified. 4. Actor must be on the **Order status checker** page. |
| Main Flows | 1. Inserts valid order number in the form. 2. Submits the form 3. View order status |
| Alternative Flows | 1. Inserts invalid order number in the form 2. Submits the form 3. System displays error. |
| Post-Conditions | If the use case is successful, then Actor is presented with the status of the order. |

|  |  |
| --- | --- |
| **Use Case #6** | **View orders** |
| Brief Description | Actor views list or all recorded order. |
| Actors | Salesperson |
| Pre-Conditions | 1. System has at least one order. 2. Actor is logged in as salesperson. 3. Actor is on the order list page. |
| Main Flows | 1. View orders in a list 2. View individual orders by clicking view details button. |
| Post-Conditions | Actor views orders. |

|  |  |
| --- | --- |
| **Use Case #7** | **View call back requests** |
| Brief Description | Actor views list of all call back requests made using frontend request call back form. |
| Actors | Salesperson |
| Pre-Conditions | 1. There is at least one call back request in the system. 2. Actor is logged in as salesperson |
| Main Flows | 1. Visits the page call back request page 2. System produces list 3. View the list |
| Post-Conditions | If the use case is successful, then the system produces all call back requests in a list which contains name and phone number of each requester. |

|  |  |
| --- | --- |
| **Use Case #8** | **Record Order** |
| Brief Description | Record Order (has option to add discounts) |
| Actors | Salesperson |
| Pre-Conditions | 1. Customer has made a request call back 2. Actor is taking the order by calling the customer. 3. Actor is logged in as salesperson 4. Actor has already selected product configuration. |
| Main Flows | 1. Add product to the cart. 2. Page refreshes and product is added to the cart. 3. Goes to **Record payment** page. 4. Inserts shipping address. 5. Selects payment method. 6. Inserts payment details. 7. Submits the order. 8. Forward confirmation email to customer which has payment verification information and order number. |
| Alternative Flows | 3. Actor negotiate the price and adds discount  4. System re-calculates product price |
| Post-Conditions | If this use case is successful, a valid order is recorded in the system otherwise nothing changes. |

|  |  |
| --- | --- |
| **Use Case #9** | **Verify Payment** |
| Brief Description | Actor verifies payment and updates status using radio buttons a) Payment verified b) Payment pending |
| Actors | Salesperson |
| Pre-Conditions | 1. A valid order is in the system 2. Actor is inside the order detail page. 3. Actor is logged in as salesperson |
| Main Flows | 1. View automated payment status from credit card API for the order. 2. Update payment status. |
| Alternative Flows | 1. In case of cheque payment, the salesperson will have to manually verify the cheque by cashing the cheque in a bank. 2. Update payment status. |
| Post-Conditions | Payment is verified, status is updated. |

|  |  |
| --- | --- |
| **Use Case #10** | **Print Invoice** |
| Brief Description | Actor prints invoice for individual order. |
| Actors | Salesperson |
| Pre-Conditions | 1. Payment is verified. 2. Actor is logged in as salesperson 3. Actor is in the order details page of an individual order |
| Main Flows | 1. Press “Print invoice” button 2. Invoice is printed |

|  |  |
| --- | --- |
| **Use Case #11** | **Send product request to warehouse** |
| Brief Description | Actor sends product request to the warehouse using the system. |
| Actors | Salesperson |
| Pre-Conditions | 1. Payment verified 2. Actor logged in as salesperson. 3. Actor is in the order details page of an individual order. |
| Main Flows | 1. Press “forward order to warehouse” button Send order to warehouse. 2. System sends the order to warehouse. |
| Post-Conditions | If the use case is successful, then the order shows up on warehouse list of order which they need to process. |

|  |  |
| --- | --- |
| **Use Case #12** | **Update order status** |
| Brief Description | Actors uses Update order status |
| Actors | Salesperson, Warehouse |
| Pre-Conditions | 1. Logged in as either salesperson or warehouse staff. 2. Actor is in the Order details page. |
| Main Flows | 1. Selects the order. 2. Update order status a) Order received b) Order is on its way c) Delivered using radio button. 3. Save changes. |
| Post-Conditions | Order status of the order is updated. |

|  |  |
| --- | --- |
| **Use Case #13** | **View Orders from Salesperson** |
| Brief Description | Warehouse staff views all orders forwarded by salesperson |
| Actors | Warehouse |
| Pre-Conditions | 1. At least one order is in the list of orders sent by salesperson to be processed by the warehouse. 2. Actor is logged in as warehouse staff. |
| Main Flows | 1. Loads the “view all orders” page. 2. System loads all orders in a list. |

## Use Case Diagram

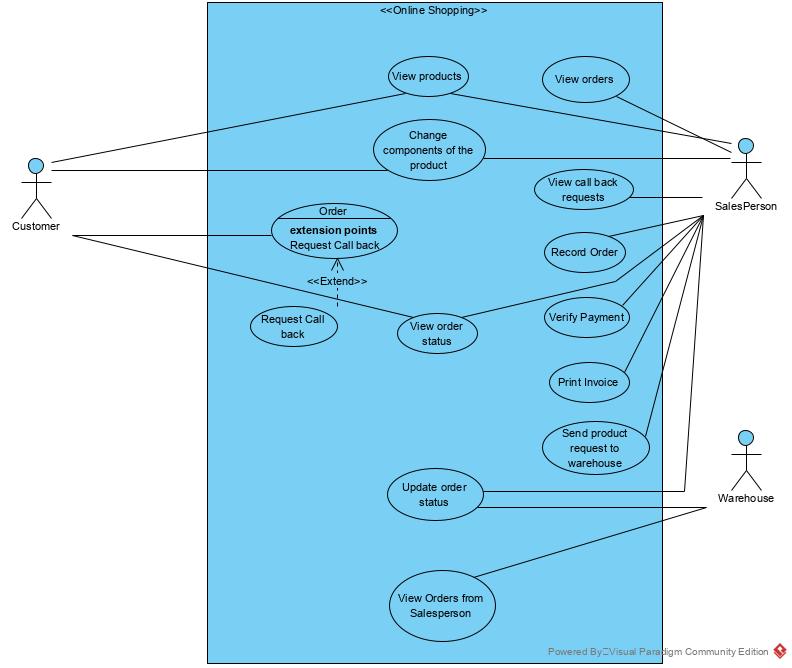


Figure 1 Use case model for the online shopping system

# 3.0 System Design

## List of actions found at #4 use case Order

Flow:

1. In order to aid the customer to send an item to the cart, the system will have a button beside the item. As soon as the customer presses the button, the system invokes an “add to cart” function which puts the item id, item price into a session array variable.
2. Customer clicks the checkout link at the cart, to visit the pay now page. The page contains all items details, total price based on all items added to the cart. This information is provided using a function called “checkout page”.
3. The checkout page should also contain a form that allows customer to insert shipping address, payment details, select payment method, and a button to submit the form, this invokes “get submit request” function. The system will validate the input and stores the order into the database.
4. System then sends a confirmation email containing order number and instruction to check its status.

From above flow, we have got following actions

1. Add to cart
2. Load checkout page request
3. Display purchase form
4. Get submit request
5. Validate inputs.
6. Store the order into database.
7. Send confirmation email.

## Activity diagram for #4 use case Order

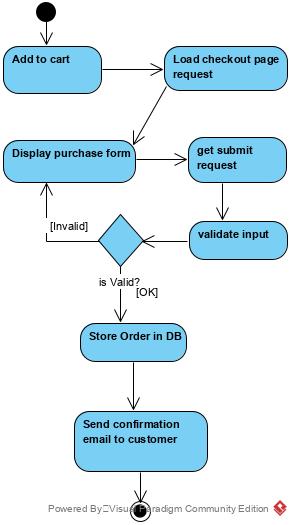


Figure 2: Activity Diagram

## Class Modelling

* 1. Entity classes

From the requirement analysis section, we have found following entity candidates. They are then divided into parent child relationship to make use of OOP inheritance.

1. Customer
2. Computer
   1. Default computer
   2. Configured computer
3. Order
4. Invoice
5. Payment
6. Configuration
7. Employee
   1. Salesperson
   2. Warehouse

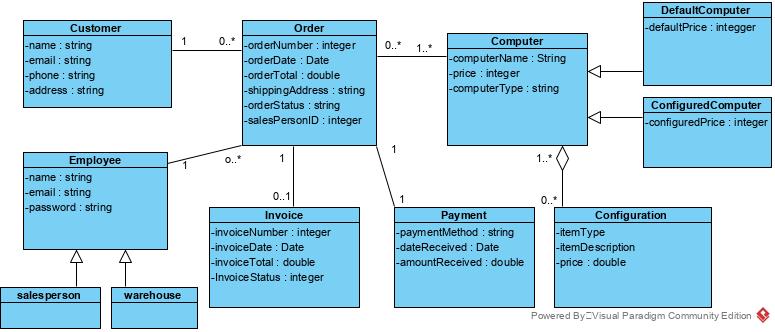


Figure Class Diagram