



## Description

This continuous assessment (project) is a group project, and in each group, there should be four members.

## Project Specification

The purpose of this assignment is to create an end to end, E-commerce system. using client-server architecture. System should have **web application for back-office work** and **mobile application for the E-commerce Customers**. Here's an overview of the required functionality:

### a) Web Application:

#### a. User Management:

- i. Create web application users with distinct roles: Administrator, Vendor, and Customer Service Representative (CSR).
- ii. Only Administrator users should have access to functions b-ii and d

#### b. Product Management:

- i. Allow vendors to create, update, and delete product using a unique Product ID.
- ii. Allow for the activation and deactivation of product listings( categories).

#### c. Order Management:

- i. Facilitate the management of customer orders, including:
  - Creating new orders (with order status tracking from processing to delivery).
  - Updating order details (before the order is dispatched).
  - Canceling orders (before the order is dispatched).

#### d. Inventory Management:

- i. Manage inventory levels, including:
  - See how much of each product available in a stock.
  - Tracking inventory levels and generating alerts for low stock.
    - When alerting it should inform vendor as a notification
  - Removing stock entries (cannot remove products that are part of pending orders).
- ii. Vendors could create product items and stock could automatically managed based on number of items available.

**e. Vendor management**

- Creation of the vendor is a functionality under Administrator
- Customer could set the rankings and the average ranking should be stored and visible to user.
- Customer comments for the vendor and the rankings cannot be modified. Customer could change the comment which they have created.

**f. User Interface:**

- Develop user interfaces using Bootstrap for an elegant appearance.

**g. Customer Order Management**

- CSR, Administrator could be able to see the status of customer order and cancel it when customer request to cancel. It should define a note when cancelling the Customer Order. Once the order is cancelled, it should inform to customer as a notification.
- CSR, Administrator and Vendor could be able to see the status of customer order and mark it as delivered. Once it is marked as delivered, it should be informed to Customer.
- If the Order contains products of multiple Vendors, Particular vendor can mark their product is ready and set the Order status as partially delivered. Once all vendors set the order as delivered, Order will be automatically move into delivered state. Other than that, CSR and Administrator could mark the Order as Delivered directly ( Assuming that, they collect goods from Vendors and deliver from their end or they get stocks from warehouse).

**b) Mobile Application:**

**a. User Management:**

- Allow customers to create their own accounts (using their email as the primary key). Account should be activated by CSR or Administrator to login to mobile successfully.
- Enable customers to modify their accounts and deactivate them.
- Deactivated accounts can only be reactivated by a CSR.

**b. Product Browsing and Purchasing:**

- Allow customers to browse products by category, search for specific items, and view detailed product information.
- Enable customers to add products to their cart, place orders, and make payments.
  - It is not a real purchasing. Marked it as purchased in backend when the customer clicks on purchase button.
  - CSR or Administrator or Vendor could mark it as delivered.

**c. Order Tracking:**

- Enable customers to track the status of their orders and view order history.

**d. Vendor ranking and comment**

- Mobile user could rank the vendor based on the product purchase experience.

- Mobile user could comment on the vendor based on the product purchase experience.

### **c) Web Service:**

- This is the central web service that processes all client requests. ( C# based web service is preferred)
- It is a FAT service that handles all business logic.

### **System Requirements:**

- The system should include both a web and mobile application that connect to a centralized web service hosted on IIS.
- The server-side data storage should utilize a NoSQL database.
- The mobile application should be developed as a pure Android application without the use of frameworks
- Mobile application and the Web client should access the backend via a web service hosted in IIS.

**High-Level Diagram:** [Provide a high-level diagram illustrating the interaction between the web application, mobile application, web service, and NoSQL database.]

### **Project Scenario:**

The project involves the development of a robust E-Commerce Platform using client-server architecture. The platform will feature a web application for back-office activities, designed for three types of users: Administrators, Vendors, and Customer Service Representatives (CSRs). Administrators will manage the overall system, Vendors will handle product listings and inventory, while CSRs will focus on customer support, including handling order cancellations, Approval of User Accounts and Monitor Orders.

Additionally, a mobile application will be developed specifically for Android, allowing customers to search for products, make purchases, manage their accounts, and track orders. When customers create accounts via the mobile application, providing an email address will be mandatory, serving as the primary identifier for managing their account. Once the account is created from mobile, it should be alerted to CSR. If the account is approved, Customer can login to the Mobile app. ( meaningful message should be shown until it get approved)

Customers will not be able to cancel orders directly; they must request cancellations either through the mobile app or via a phone call. Upon receiving a cancellation request, the CSR will be Notified via notification and will have the authority to process the cancellation.

Vendors will have the capability to create and manage their own product listings. All items created by vendors will be visible to customers when they search for products. The search functionality will include filters based on item name, price, vendor, and ratings. Customers will also be able to sort the list of items based on these filters, including vendor-specific ratings and price.

Customer could add more than one items into shopping cart and purchase all of them at once. If the SC is purchased, Order is marked as purchased but it will not be delivered until relevant authority will deliver it. customer should be able to see the status of the Orders which he/she

has placed. CSR and Administrator should be able to Manage all Orders and Vendor should be able to manage Orders which are placed under them.

The platform will be integrated with a centralized web service that manages all business logic and data processing, ensuring seamless operation across both the web and mobile applications.

## Technology

- For the Web application : React
- For the Mobile Development : Android/Java
- For the web Service : Web API/C#

## What do you have to hand in?

1. Project Code AND a screen shot: Provide a single zipped file containing all directories/files in your project and a detailed report. You should also include a screen shot of the main opening menu/screen of your app. Include IT number in the zipped file name e.g. IT15895623.zip

**Note: Code that does not include the following will not be marked**

- **comment header block on each .cs file.**
  - **Inline comments at the beginning of each method.**
  - **Unique screenshots of the application.**
2. You should write a detail report that describes your app with following Criteria.
    - Screen shots of all UI's
    - Application high level diagram, Use case Diagram and DFD diagram ( DFD<sub>0</sub> and DFD<sub>1</sub>).
    - Database design
    - Source code (**not screen shots**). Copy and paste your codes.
    - All the references
    - Individual contribution
    - Challenges

## Marks Allocation

This group project assignment is worth **30%** of your overall mark in this module. Mark distribution is as follow.

Note: No exemptions are approved. It should clearly mention the contribution of each student. (Should this goes to the report or separate document?)

Project Report + Source code	15%
Demo/Viva(participation is compulsory)	15%

Should the total be 30%.

## **Due Date**

Please submit the project source code and report on or before **5<sup>th</sup> of October 2024 11.45 pm**.

## **What is Viva for the remaining assessment of the app?**

Viva session will test your knowledge in your own app. It will be done as a supervised test (After submission).

**If you are absent for the viva session, your assignment will not be marked.**

### Instructions

- All students are expected to submit assignments within the given deadline.
- late submissions are not accepted.
- Viva is treated as formal exam. No additional dates will be provided. You will get the date of viva at least a week before.
- Any code snippets that are not directly written by you (e.g., used from a tutorial) must be referenced as such within your code. You must directly comment the code to explain its source. **Failure to reference code that is not yours will be treated as plagiarism.**
- The app must be your own work. **Any plagiarism found will get zero marks**, and the plagiarism escalated as per DIT assessment regulations.