**System Modules & Functionalities**

**1.User Authentication Module**

**Functionality**:

* **User Login**: Secure login functionality using **ASP.NET Identity**.
* **Role Management**:
  + **Admin**, **User**, and **Manager**.
  + Role-based access control for different system functionalities.
* **Session Management**: Track and manage user sessions securely.

**Components**:

* **Login Page**: Allow users to authenticate.
* **Role-based Authorization**: Control access to features based on user roles.

**2. User Management Module (Admin Only)**

**Functionality**:

* **CRUD Operations for Users**:
  + Create, read, update, and delete users.
  + Assign roles to users (Admin, User, Manager)
* **User Profile**: Ability for users to manage their own profile details: update personal info, change password.

**Components**:

* **User List Page**: Admin views all users, with controls for editing and deleting users.
* **Create/Edit User Page**: Admin can add or edit user details and assign roles.
* **Profile Management Page**: Users can manage their own profile.

**3. Product Management Module (Admin & Manager)**

**Functionality**:

* **CRUD Operations for Products**:
  + Admin/Manager can create, read, update, and delete products.
* **Product Inventory Management:**
  + Track product stock levels.
  + Maintain cost per item.
  + Notifications for low-stock products.
  + Name of the item
  + Product in stock
  + Product sold
  + Number of products

**Components**:

* **Product List Page**: Admin/Manager views all products with current stock levels, stock sold, cost per item, name of stock and number of products.
* **Create/Edit Product Page**: Admin or Manager can add or modify product details.

**4. Invoice Management Module : Users/managers**

**Functionality**:

* **CRUD Operations for Invoices**:
  + Users can create and edit an invoice.
  + Managers can view and edit all invoices.
* **Add Items to Invoice**:
  + Users adding items to the invoice with real-time total calculation, this has to be handled with JavaScript.
* **Invoice Tracking**: Track which user created an invoice and the invoice's total.

**Components**:

* **Invoice List Page**: Users see their invoices; Managers see all invoices.
* **Create/Edit Invoice Page**:
  + Users create and edit invoices, adding/removing products with automatic total calculation (JavaScript).
  + Manager can edit any invoice.
* **Invoice Detail Page**: View detailed breakdown of items and costs.

**5. Reporting Module (Manager Only)**

**Functionality**:

* **Sales Reports**:
  + Number of items sold per product.
  + Total number of products sold.
  + Total number of products in stock.
  + Stock vs. sold product comparison.
* **Product Reports**:
  + Overview of all products (in-stock, sold, and low-stock).
* **Invoice Reports**:
  + Summary of all invoices, grouped by user and date.

**Components**:

* **Sales Reports Page**: Visual reports showing key metrics (e.g., number of products sold, stock remaining).
* **Product Inventory Reports Page**: Lists product details and stock status.
* **Invoice Reports Page**: Summary view of all invoices, sortable by date or user.

**6. Notification & Alerts Module (Manager Only)**

**Functionality**:

* **Low Stock Notifications**:
  + Alerts for products below the defined restock threshold (e.g., < 10 units).
  + Notify Managers when a product needs restocking.
* **Consolidated Notifications**:
  + Provide a single notification containing all low-stock products.

**Components**:

* **Notifications Dashboard**: Managers receive alerts within the system interface.
* **Email Notifications** (optional): Send low-stock alerts to Manager’s email.

**7. Frontend Module (All Roles)**

**Functionality**:

* **Client-Side Invoice Total Calculation**:
  + Use JavaScript to dynamically update invoice totals as items are added or removed.
* **Invoice Form (JavaScript)**: Automatically updates total without server communication as products are added.

**8. Database Module**

**Functionality**:

* **Data Storage**: Manage persistence of data for users, products, and invoices using **MS SQL**.
* **Database design:**
  + Create the following database table Users, Products, and Invoices.
  + Define relationships between Users, Products, and Invoices.
  + Maintain the integrity of product stock when invoices are created.

**Components**:

* **User Table**: Stores user information and roles.
  + User\_id
* **Product Table**: Stores product details.
  + Product\_id
  + Product\_Name
  + Product\_sold (number of items sold per product)
  + Product\_in\_stock
  + Costs\_per\_product\_item
  + Number\_of\_products
* **Invoice Table**: Stores invoice metadata
  + Invoice\_id
  + Date\_created (date invoice was created)
  + Created\_by (user created the invoice)
  + Invoice\_items
  + Total\_invoice

**Module Interaction & Workflow:**

1. **Authentication**: Users log in and access functionality based on their role.
2. **User/Role Management**: Admins manage users and assign roles.
3. **Product Management**: Admins/Managers add and manage products. Stock updates when invoices are processed.
4. **Invoice Creation**: Users create/edit invoices, Managers oversee all.
5. **Real-time Calculations**: Invoice totals are updated dynamically on the client side using JavaScript.
6. **Reporting & Analysis**: Managers generate reports on product sales, stock, and invoices.
7. **Notifications**: Alerts are generated for low-stock products to prompt restocking.

Grouping system **modules** and **functionalities**, will allow the system to remain structured, scalable, and easy to maintain. Each module has clearly defined responsibilities and connections with other modules, ensuring the system’s overall functionality is integrated and easy to understand.