**Mini ERP System Documentation**

**Overview**

The **Mini ERP System** is a backend solution built using **ASP.NET Web API** with **JWT Authentication** and **Role-Based Authorization**. This system manages various business operations including user management, product management, customer management, supplier management, sales orders, purchase orders, and automatic stock management.

**Project Features**

**1. Authentication & Authorization**

* JWT Token-Based Authentication
* Role-Based Authorization
* Three User Roles:
  + **Admin**: Manages users and products.
  + **Sales**: Manages customers and sales orders.
  + **Purchase**: Manages suppliers and purchase orders with stock management.

**2. Admin Functionalities**

* Add Users (Sales and Purchase Roles)
* Update Users
* Delete Users
* Manage Products (CRUD Operations)

**3. Sales User Functionalities**

* Add Customers
* View Customers
* Update Customers
* Delete Customers
* Create Sales Orders

**4. Purchase User Functionalities**

* Add Suppliers
* View Suppliers
* Update Suppliers
* Delete Suppliers
* Create Purchase Orders
* Automatic Stock Deduction

**Database Design**

**Tables:**

|  |  |
| --- | --- |
| **Table** | **Description** |
| Users | Stores User Information (Admin, Sales, Purchase) |
| Products | Stores Product Details with Stock |
| Customers | Stores Customer Information |
| Suppliers | Stores Supplier Information |
| SalesOrders | Manages Sales Orders linked with Customers and Products |
| PurchaseOrders | Manages Purchase Orders linked with Suppliers and Products |

**Relationships:**

* One-to-Many between **Customers** and **SalesOrders**
* One-to-Many between **Suppliers** and **PurchaseOrders**
* One-to-Many between **Products** and **PurchaseOrders**

**API Endpoints**

**1. Authentication**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Endpoint | Description | Role |
| POST | /api/auth/login | User Login | Admin, Sales, Purchase |

**2. Admin Module**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Endpoint | Description | Role |
| POST | /api/admin/add-user | Add User | Admin |
| GET | /api/admin/all-users | Get All Users | Admin |
| PUT | /api/admin/update-user | Update User | Admin |
| DELETE | /api/admin/delete-user/{id} | Delete User | Admin |

**3. Product Management**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Endpoint | Description | Role |
| POST | /api/admin/add-product | Add Product | Admin |
| GET | /api/admin/all-products | Get Products | Admin |
| PUT | /api/admin/update-product | Update Product | Admin |
| DELETE | /api/admin/delete-product/{id} | Delete Product | Admin |

**4. Sales Module**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Endpoint | Description | Role |
| POST | /api/sales/customers/add | Add Customer | Sales |
| GET | /api/sales/customers/all | View Customers | Sales |
| PUT | /api/sales/customers/update | Update Customer | Sales |
| DELETE | /api/sales/customers/delete/{id} | Delete Customer | Sales |
| POST | /api/sales/orders/create | Create Sales Order | Sales |

**5. Purchase Module**

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Endpoint | Description | Role |
| POST | /api/purchase/suppliers/add | Add Supplier | Purchase |
| GET | /api/purchase/suppliers/all | View Suppliers | Purchase |
| PUT | /api/purchase/suppliers/update | Update Supplier | Purchase |
| DELETE | /api/purchase/suppliers/delete/{id} | Delete Supplier | Purchase |
| POST | /api/purchase/orders/create | Create Purchase Order | Purchase |
| GET | /api/purchase/orders/all | View Purchase Orders | Purchase |

**Stock Management Flow**

|  |  |  |
| --- | --- | --- |
| Action | Table | Stock Behavior |
| Purchase | Product | Stock Reduces |
| Sales Order | Product | Stock Unchanged |
| Delete Purchase Order | Product | Stock Restored |

**JWT Authentication Flow**

1. Login with credentials via **/api/auth/login**.
2. JWT Token is generated and returned.
3. Token is passed in **Authorization Header** for every API request.
4. Server validates token and role before allowing access.