5. System Deployment & Integration

1-Technology Stack

Frontend:

- ASP.NET Core MVC – Razor Views for UI
- Bootstrap 5 / Tailwind CSS – Responsive design
- Chart.js / Syncfusion / Telerik – Data visualization
- AJAX & SignalR – Real-time updates

Backend:

- ASP.NET Core – Web application framework
- Entity Framework Core – ORM for database management
- Hangfire / Quartz.NET – Background task processing

Database:

- SQL Server – Main relational database
- Stored Procedures & SQL Views – Optimized performance

Security & Authentication:

- ASP.NET Core Identity – User authentication & role management
- JWT Authentication & OAuth (Google, Microsoft, Facebook)
- CSRF & XSS Protection – Anti-forgery tokens
- SQL Injection Prevention – Parameterized queries

2- Deployment Diagram

Describes how software components are distributed across hardware.

Components:

- Client (User Devices) – Web browser accessing the ASP.NET MVC app
- Web Server (IIS / Azure App Service) – Hosts the ASP.NET Core MVC application
- Database Server (SQL Server / Azure SQL Database) – Stores system data
- Background Services – Hangfire/Quartz.NET for stock tracking
- Real-time Data Processing – SignalR for live updates

Deployment Flow:

User (Browser) → Web Server (ASP.NET MVC) → Database Server (SQL Server) → Background Services

3- Component Diagram

Shows system components and their dependencies.

Main Components:

✓UI Layer (Views & Controllers) → Razor Views, Bootstrap, Chart.js

✓Business Logic Layer (Services & Models) → Entity Framework, Business Rules

✓Data Access Layer (Repository Pattern & EF Core) → SQL Server

✓Authentication Module → ASP.NET Core Identity, JWT, OAuth

✓Background Services → Hangfire, Quartz.NET

✓Real-time Communication → SignalR for inventory synchronization