Advanced JavaScript Frameworks (Angular)

Inventory Management System

INSTRUCTOR NAME: PALLAVI

INSTITUTION NAME: CHRIST UNIVERSITY

CIA-2

NAMES	REGISTRATION NUMBER	CHRIST EMAIL ID
BARATH C G	2462507	barath.cg@btech.christuniversity.in
N MUTHUKRISHNAN	2462519	n.muthukrishnan@btech.christuniversity.in
NEAL SEBASTIAN	2462520	neal.sebastian@btech.christuniversity.in
NOEL KOLANCHERY ROY	2462522	noel.kolanchery@btech.christuniversity.in

📃 ***Abstract:***

The Inventory Management System is a comprehensive web-based application developed using Angular 21 and TypeScript that enables organizations to efficiently manage their product inventory, supplier relationships, and purchase orders. The system provides a centralized platform for tracking stock levels, monitoring supplier information, managing order workflows, and generating real-time inventory insights. With a responsive user interface, robust component architecture, and RESTful service integration, the application delivers a scalable and maintainable solution for inventory operations across desktop, tablet, and mobile devices.

🎯 ***Objectives:***

1. Create a centralized inventory tracking system - Consolidate product stock information in a single, accessible platform.
2. Streamline supplier management - Maintain comprehensive supplier database with contact information and associated products
3. Implement order workflow management - Track both inbound and outbound inventory movements with status updates
4. Provide real-time visibility - Display up-to-date inventory metrics and alerts on a unified dashboard
5. Enable informed decision-making - Highlight low-stock items and pending orders requiring attention

📌 ***Scope of the Project:***

- Product inventory management (CRUD operations)
- Supplier information management (CRUD operations)
- Order tracking system (inbound/outbound)
- Dashboard with real-time metrics
- Responsive web interface
- Form validation (reactive and template-driven)
- Mock API integration with fallback
- Routing with child routes
- User documentation

🛠 ***Tools & Technologies Used:***

Tool / Technology	Purpose
Angular (v20)	Frontend framework
TypeScript	Application logic
Angular Material	UI components & styling
JSON Server	Mock backend
RxJS	Asynchronous data handling
HTML5	Page structure
CSS3	Styling and layout
Node.js	Runtime environment
Angular CLI	Project creation & management
VS Code	Code editor
Chrome DevTools	Testing & debugging

□ ***Application Structure Overview***

- The Inventory Management and Tracking System is structured using Angular's modular and component-based architecture. The application is divided into models, services, and components, each with a clearly defined responsibility.
- **Models** define the data structure for products, suppliers, and orders using TypeScript interfaces.
- **Services** handle business logic and communication with the backend API using Angular's HttpClient and dependency injection.
- **Components** manage the user interface and user interactions, such as displaying product lists, dashboards, and order tracking views.
- Angular routing is used to enable navigation between different sections of the application, while Reactive and Template-driven Forms are used for data input and validation. The application also integrates Angular Material components to provide a responsive and consistent user interface.
- This structured approach improves maintainability, scalability, and code reusability.

📓 ***Module Description:***

🏠 1 Dashboard Module 📊

The Dashboard Module serves as the central overview component of the Inventory Management and Tracking System. It provides a summarized view of key inventory metrics, enabling users to quickly understand the current status of stock and system activity.

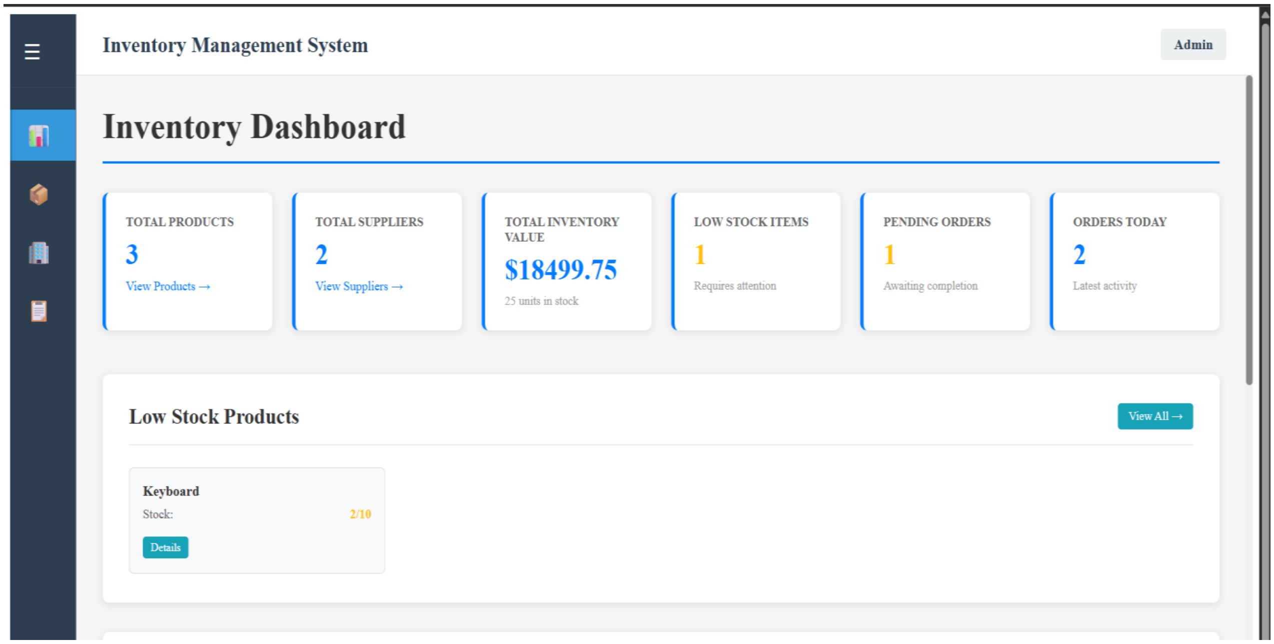
This module displays essential information such as the total number of products, low-stock alerts, and recent inventory updates. It retrieves data from the Product and Order services using Angular services and RxJS observables, ensuring that the displayed information remains up to date.

The Dashboard Module enhances usability by presenting critical data in a simple and intuitive format. By consolidating important inventory statistics into a single view, it helps users make faster and more informed inventory management decisions.

□ Key Functions:

- Displays a summarized overview of inventory status
- Shows total number of products in stock
- Highlights low-stock or reorder-level alerts
- Retrieves real-time inventory data from services
- Provides quick insights for decision-making

📸 Screenshot :



Inventory Management System

Inventory Dashboard

TOTAL PRODUCTS
3

TOTAL SUPPLIERS
2

TOTAL INVENTORY VALUE
$18499.75
25 units in stock

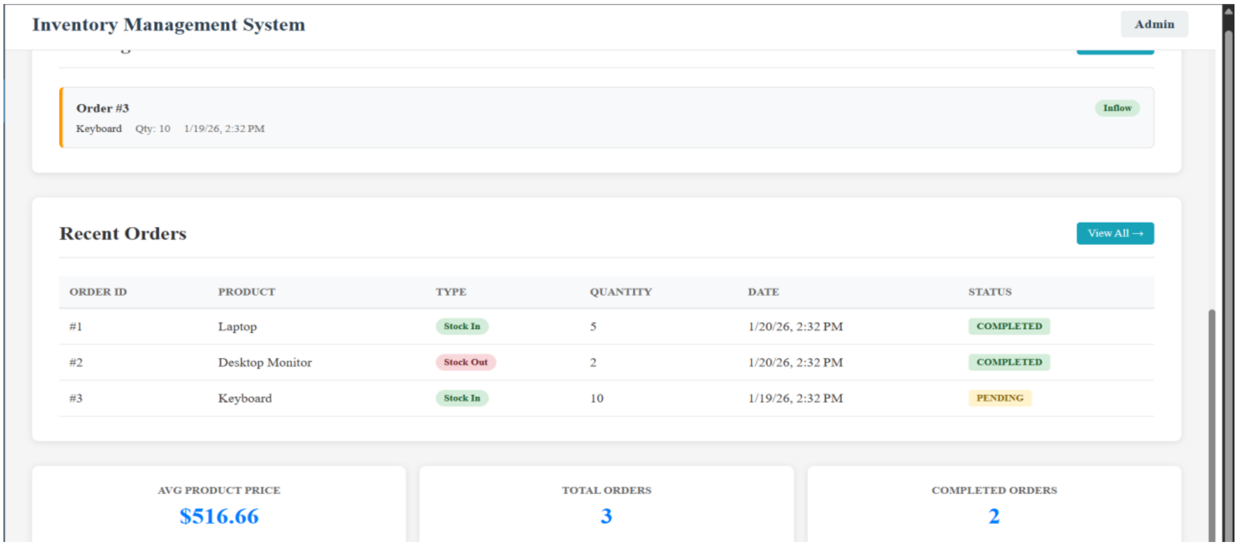
LOW STOCK ITEMS
1
Requires attention

PENDING ORDERS
1
Awaiting completion

ORDERS TODAY
2
Latest activity

Low Stock Products

Keyboard
Stock: 2/10


Inventory Management System

Order #3
Keyboard Qty: 10 1/19/26, 2:32 PM

Recent Orders

ORDER ID	PRODUCT	TYPE	QUANTITY	DATE	STATUS
#1	Laptop	Stock In	5	1/20/26, 2:32 PM	COMPLETED
#2	Desktop Monitor	Stock Out	2	1/20/26, 2:32 PM	COMPLETED
#3	Keyboard	Stock In	10	1/19/26, 2:32 PM	PENDING

AVG PRODUCT PRICE $516.66

TOTAL ORDERS 3

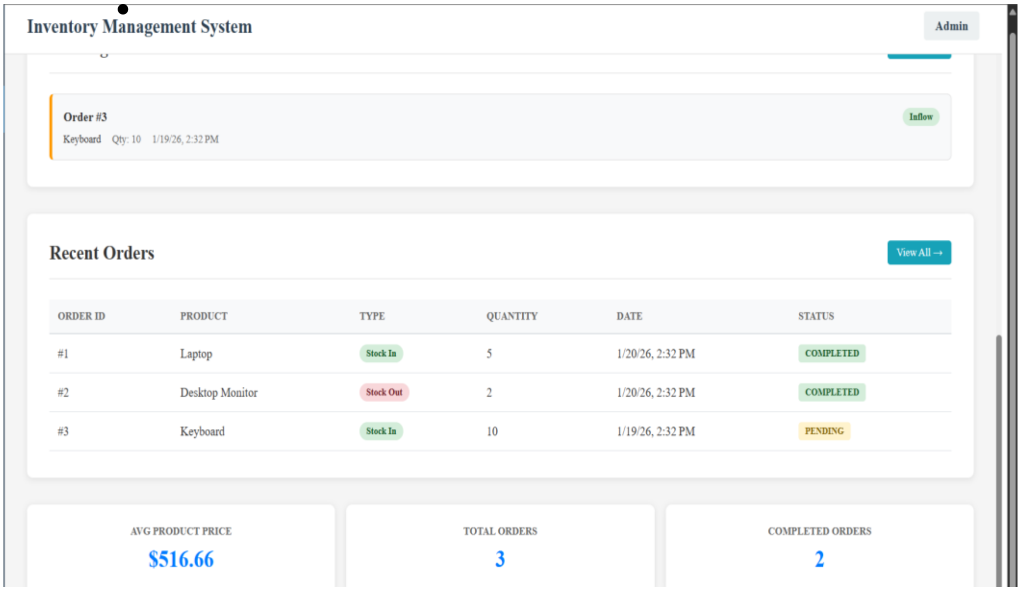
COMPLETED ORDERS 2

Products Module:

The Products Module is responsible for managing and displaying all inventory items within the system. It allows users to view detailed product information, including product name, price, available quantity, and reorder level. This module interacts with the Product Service to perform CRUD operations on product data through RESTful APIs. It uses Angular components and data binding to dynamically render product lists and ensure real-time updates. The Products Module plays a critical role in maintaining accurate inventory records and identifying low-stock items.

□ **Key Functions:**

- Displays a list of all available products
- Shows product details such as name, price, and quantity
- Allows adding, updating, and deleting product records
- Identifies products below reorder level
- Ensures accurate and up-to-date inventory information



Inventory Management System

Admin

Order #3
Keyboard Qty: 10 1/19/26, 2:32 PM

Inflow

Recent Orders

View All →

ORDER ID	PRODUCT	TYPE	QUANTITY	DATE	STATUS
#1	Laptop	Stock In	5	1/20/26, 2:32 PM	COMPLETED
#2	Desktop Monitor	Stock Out	2	1/20/26, 2:32 PM	COMPLETED
#3	Keyboard	Stock In	10	1/19/26, 2:32 PM	PENDING

AVG PRODUCT PRICE $516.66

TOTAL ORDERS 3

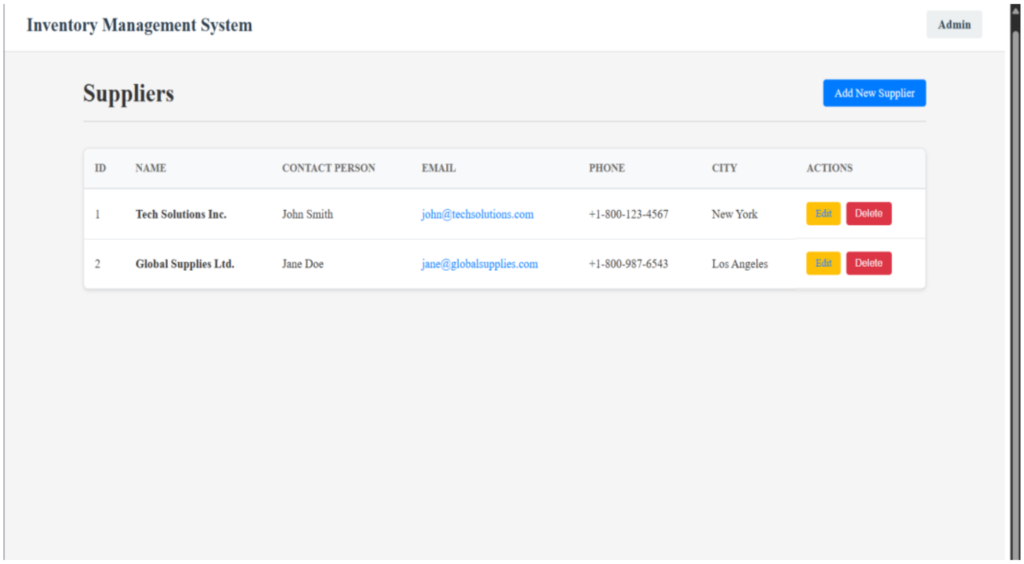
COMPLETED ORDERS 2

📝 ***Suppliers Module:***

The Suppliers Module manages supplier-related information required for inventory procurement. It enables users to add, view, and maintain details such as supplier name, contact information, and email address. The module uses template-driven forms for data input and validation and communicates with the backend through the Supplier Service. By maintaining structured supplier data, this module supports efficient coordination between inventory management and procurement processes.

□ Key Functions:

- Maintains supplier information and contact details
- Allows adding and viewing supplier records
- Validates supplier input data through forms
- Connects supplier data with products for procurement
- Supports efficient inventory sourcing and management



Inventory Management System

Suppliers

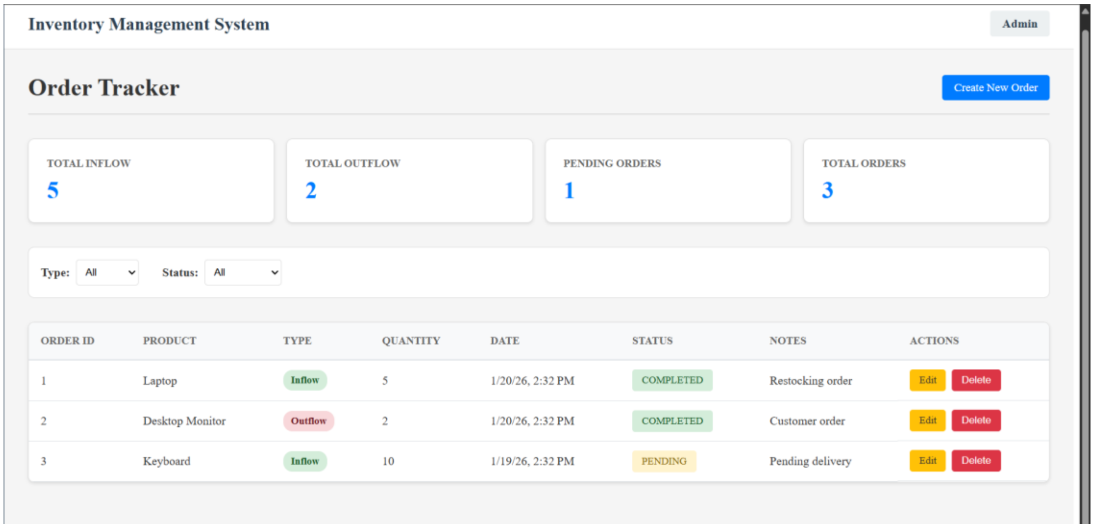
ID	NAME	CONTACT PERSON	EMAIL	PHONE	CITY	ACTIONS
1	**Tech Solutions Inc.**	John Smith	john@techsolutions.com	+1-800-123-4567	New York	Edit Delete
2	**Global Supplies Ltd.**	Jane Doe	jane@globalsupplies.com	+1-800-987-6543	Los Angeles	Edit Delete

Order Tracking Module:

The Order Tracking Module monitors the movement of inventory through stock inflow and outflow operations. It records order details such as product identifier, quantity, order date, and transaction type (inbound or outbound). This module retrieves and manages order data using the Order Service and displays it in a structured format for easy tracking. By maintaining a clear history of inventory transactions, the Order Tracking Module helps ensure transparency, accountability, and accurate stock management.

□ **Key Functions:**

- Tracks inventory inflow and outflow transactions
- Records order details including product, quantity, and date
- Differentiates between inbound and outbound orders
- Maintains a history of stock movements
- Helps ensure accurate stock level updates



Inventory Management System

Order Tracker

TOTAL INFLOW 5

TOTAL OUTFLOW 2

PENDING ORDERS 1

TOTAL ORDERS 3

Type: All

Status: All

ORDER ID	PRODUCT	TYPE	QUANTITY	DATE	STATUS	NOTES	ACTIONS
1	Laptop	Inflow	5	1/20/26, 2:32 PM	COMPLETED	Restocking order	Edit Delete
2	Desktop Monitor	Outflow	2	1/20/26, 2:32 PM	COMPLETED	Customer order	Edit Delete
3	Keyboard	Inflow	10	1/19/26, 2:32 PM	PENDING	Pending delivery	Edit Delete

🎨 ***UI & Styling Strategy***

• Angular Material components for consistent UI design.
• MatTable for displaying employee and attendance records.
• MatFormField, MatInput, and MatSelect for forms.
• MatSnackBar for notifications and alerts.
• Responsive layout using Material toolbar and side navigation.

⚡ ***Angular & TypeScript Overview:***

• TypeScript interfaces define structured models for Employee, Attendance, and Leave data.
• Services handle CRUD operations using HttpClient.
• RxJS Observables manage asynchronous data flow.
• Route guards restrict HR-only modules.
• Data binding and directives ensure dynamic UI updates.

🔍 ***Key Features:***

- Centralized management of products, suppliers, and orders
- Real-time display of product stock levels
- Dashboard summarizing key inventory metrics
- Product management with add, update, and delete operations
- Supplier information management
- Tracking of inventory inflow and outflow
- Responsive and user-friendly interface using Angular Materia

⚔ Challenges Faced & Solutions:

- **Environment setup issues:** Resolved by configuring PowerShell execution policies and properly installing Angular CLI.
- **Understanding project structure: S**olved by organizing code into models, services, and components.
- **Component generation errors:** Fixed by running Angular CLI commands inside the correct project directory.
- **Backend integration difficulties:** Addressed using JSON Server to simulate REST APIs.
- **UI consistency challenges:** Overcome by using Angular Material for a uniform and responsive interface.

✅ Outcome:

- Successfully developed a functional inventory management system using Angular
- Implemented product, supplier, and order tracking modules
- Achieved a responsive and user-friendly interface

📈 Future Enhancements:

- Add user authentication and role-based access control
- Integrate a real database and backend services
- Implement advanced reports and analytics
- Enable real-time low-stock notifications

🔚 Conclusion:

The Inventory Management and Tracking System was successfully designed and implemented using Angular and TypeScript. The application demonstrates effective use of component-based architecture, services, routing, and form handling to manage products, suppliers, and inventory transactions. By integrating Angular Material and a mock backend, the system provides a responsive and user-friendly interface. This project enhances practical understanding of modern front-end development concepts and serves as a strong foundation for scalable inventory solutions.

📚 References:

- Angular Documentation – https://angular.dev
- Angular Material Documentation – https://material.angular.io
- TypeScript Documentation – https://www.typescriptlang.org
- JSON Server Documentation – https://github.com/typicode/json-server
- Node.js Official Website – https://nodejs.org