

# Revolutionizing Agriculture with AgriEdge Or-Mange Ltd: A Salesforce-Driven Order Management Solution

**Name:** H Rajanna Gari Anil Kumar

**Email:** ania08333@gmail.com

**Phone Number:** 9347785766

## Abstract:

The **AgriEdge Order Management System** is a Salesforce-based application designed to streamline the management of agricultural product orders, associated items, and shipment tracking. This project introduces a custom solution using Salesforce objects and Apex programming to automate and maintain data integrity across related records such as `AgriEdge_Order__c`, `AgriEdge_OrderItem__c`, and `AgriEdge_Shipment__c`.

The system automatically calculates the total price of orders whenever order items are created or updated and dynamically updates the payment status based on these totals. Additionally, it ensures consistency in data management through Apex triggers and helper classes such as `OrderTotalUpdater` and `OrderStatusUpdater`.

This solution enhances operational efficiency, reduces manual work, and provides real-time insights into the order lifecycle, making it highly useful for businesses in the agricultural domain. The project demonstrates the practical application of declarative and programmatic tools within the Salesforce ecosystem to solve real-world business problems.

## Objective:

The primary objective of this project is to develop a robust and efficient **Order Management System** within the Salesforce platform tailored for the agricultural domain. The system aims to automate and manage the lifecycle of orders, order items, and shipments through custom objects and Apex logic.

### **Specific objectives include:**

- To design and implement custom Salesforce objects:
  - AgriEdge\_Order\_\_c for capturing order details
  - AgriEdge\_OrderItem\_\_c for managing individual product entries in orders
  - AgriEdge\_Shipment\_\_c for tracking the shipment process
- To automate the calculation of total order amounts based on associated order items.
- To dynamically update the **Order Status** and **Payment Status** based on business rules.
- To handle shipment creation and status updates in accordance with the order lifecycle.
- To improve data integrity and minimize manual intervention using Apex triggers and helper classes.
- To provide a scalable and user-friendly interface for managing orders, products, and shipments in agricultural supply chains.

### **Technology Description :**

The AgriEdge Order Management System is built on the **Salesforce Platform**, leveraging its cloud-based architecture, declarative tools, and programmatic capabilities to deliver a scalable and automated solution for order tracking and shipment processing in the agricultural sector.

### Technologies Used:

Technology	Description
<b>Salesforce CRM</b>	Cloud-based CRM platform used to build and manage the entire application.
<b>Custom Objects</b>	AgriEdge_Order__c, AgriEdge_OrderItem__c, and AgriEdge_Shipment__c created to represent and store order-related data.
<b>Apex Classes</b>	Used to implement business logic for updating order totals, shipment status, and payment handling.
<b>Apex Triggers</b>	Invoked automatically after data changes (insert/update) to process related records and ensure data consistency.
<b>SOQL (Salesforce Object Query Language)</b>	Used to query Salesforce records in Apex code.
<b>Lightning UI</b>	Salesforce interface used by users to interact with records through standard or customized layouts.

### **Workflow Automation Tools:**

- **Invocable Apex** – Enables automation like Flow to call Apex methods.
- **Validation Rules** – Ensure data accuracy and enforce business rules.
- **Lookup Relationships** – Used to associate orders with order items and shipments.

The screenshot shows the Salesforce Object Manager interface. At the top, there are tabs for 'Recently Viewed' and 'Object Manager'. The main area displays a table of objects with columns for Label, API Name, Type, Description, Last Modified, and Deployed. A search bar at the top right contains the text 'agri'.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
AgriEdge Inventory	AgriEdge_Inventory__c	Custom Object	Manages product stock levels and locations	7/20/2025	✓
AgriEdge Order	AgriEdge_Order__c	Custom Object		7/20/2025	✓
AgriEdge Orderitem	AgriEdge_Orderitem__c	Custom Object		7/20/2025	✓
AgriEdge Shipment	AgriEdge_Shipment__c	Custom Object	Tracks order shipments and delivery status	7/20/2025	✓

### (Object Manager)

The screenshot shows the 'Fields & Relationships' section of the Object Manager for the 'AgriEdge Order' object. On the left, a sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, etc. The main area shows a table of fields with columns for Field Label, Name, Type, and Description. A search bar at the top right contains 'Quick Find'.

FIELD LABEL	NAME	TYPE	DESCRIPTION
AgriEdge Order Name	Name	Auto Number	
Created By	CreatedBy	Lookup(User)	
Customer	Customer__c	Lookup(Account)	
Discounted Total	Discounted_Total__c	Formula (Currency)	
Last Modified By	LastModifiedBy	Lookup(User)	
Order Date	Order_Date__c	Date/Time	
Order Number	Order_Number__c	Auto Number	
Order Status	Order_Status__c	Picklist	
Owner	OwnerId	Lookup(User/Group)	
Payment Status	Payment_Status__c	Picklist	
Shipping Address	Shipping_Address__c	Text Area(255)	
Total Amount	Total_Amount__c	Currency	

### (AgriEdge Order Fields)

The screenshot shows the Salesforce Object Manager interface for the 'AgriEdge OrderItem' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Triggers. The main content area displays the 'Fields & Relationships' section, which contains 9 items sorted by Field Label. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
AgriEdge OrderItem	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Order	Order__c	Lookup(AgriEdge Order)	✓	
Owner	OwnerId	Lookup(User,Group)	✓	
Product	Product__c	Lookup(Product)	✓	
Quantity	Quantity__c	Number(18, 0)		
Total Price	Total_Price__c	Formula (Currency)		
Unit Price	Unit_Price__c	Currency(18, 0)		

### (AgriEdge OrderItem Fields)

The screenshot shows the Salesforce Object Manager interface for the 'AgriEdge Shipment' object. The left sidebar lists various setup options. The main content area displays the 'Fields & Relationships' section, which contains 8 items sorted by Field Label. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
AgriEdge Shipment Name	Name	Text(80)		✓
Carrier	Carrier__c	Picklist		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Order	Order__c	Lookup(AgriEdge Order)	✓	
Owner	OwnerId	Lookup(User,Group)	✓	
Status	Status__c	Picklist		
Tracking Number	Tracking_Number__c	Text(20)		

### (AgriEdge Shipment Fields)

**AgriEdge Inventory**

**Fields & Relationships**  
9 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
AgriEdge Inventory Name	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Product	Product__c	Lookup(Product)		✓
Reorder Level	Reorder_Level__c	Number(18, 0)		
Stock Quantity	Stock_Quantity__c	Number(18, 0)		
Stock Status	Stock_Status__c	Formula (Text)		
Warehouse Location	Warehouse_Location__c	Text(60)		

### (AgriEdge Inventory Fields)

**AgriEdge OrderItem** **anil**

**Details**

AgriEdge OrderItem anil	Owner <a href="#">Anil Kumar H Rajanna Gari</a>
Order ORD-0008	
Product potassium	
Quantity 10	
Unit Price \$12	
Total Price \$120.00	
AgriEdge Order	
Created By <a href="#">Anil Kumar H Rajanna Gari</a> , 7/27/2025, 8:07 PM	Last Modified By <a href="#">Anil Kumar H Rajanna Gari</a> , 7/27/2025, 8:07 PM

**Activity**

No activities to show.  
Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

### (New AgriEdge OrderItem)

**AgriEdge Order**  
ORD-0008

**Related Details**

AgriEdge Order Name	ORD-0008	Owner	Anil Kumar H Rajanna Gari
Customer	yash		
Order Status	Delivered		
Order Date	7/27/2025, 12:00 PM		
Total Amount	\$120		
Payment Status	Paid		
Shipping Address			
Discounted Total	\$108.00		
Order Number	ORD-0008		
Created By	Anil Kumar H Rajanna Gari, 7/27/2025, 8:07 PM	Last Modified By	Anil Kumar H Rajanna Gari, 7/28/2025, 8:38 AM

**Activity**

No activities to show.  
Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

## (AriEdge Order)

**AgriEdge Shipment**  
Test Order

**Related Details**

AgriEdge Shipment Name	Test Order	Owner	Anil Kumar H Rajanna Gari
Order	ORD-0002		
Tracking Number	12321		
Status	Delivered		
Carrier	FedEx		
AgriEdge Order			
Created By	Anil Kumar H Rajanna Gari, 7/25/2025, 9:58 AM	Last Modified By	Anil Kumar H Rajanna Gari, 7/25/2025, 9:58 AM

**Activity**

No activities to show.  
Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

## (AgriEdge Shipments)

# Project Explanation with Real-World Example

## Overview

This project simulates a **real-world agricultural supply chain management system** built on Salesforce. It automates order processing, shipment tracking, and order item management for a fictional agri-business called **AgriEdge** that delivers agricultural products like fertilizers, seeds, and equipment to farmers and retailers.

## Real-World Example Scenario

Let's say **AgriEdge Pvt. Ltd.**, an agriculture products supplier, receives an order from a farmer named **Ramesh** for:

- 10 bags of fertilizer
- 5 litres of organic pesticide

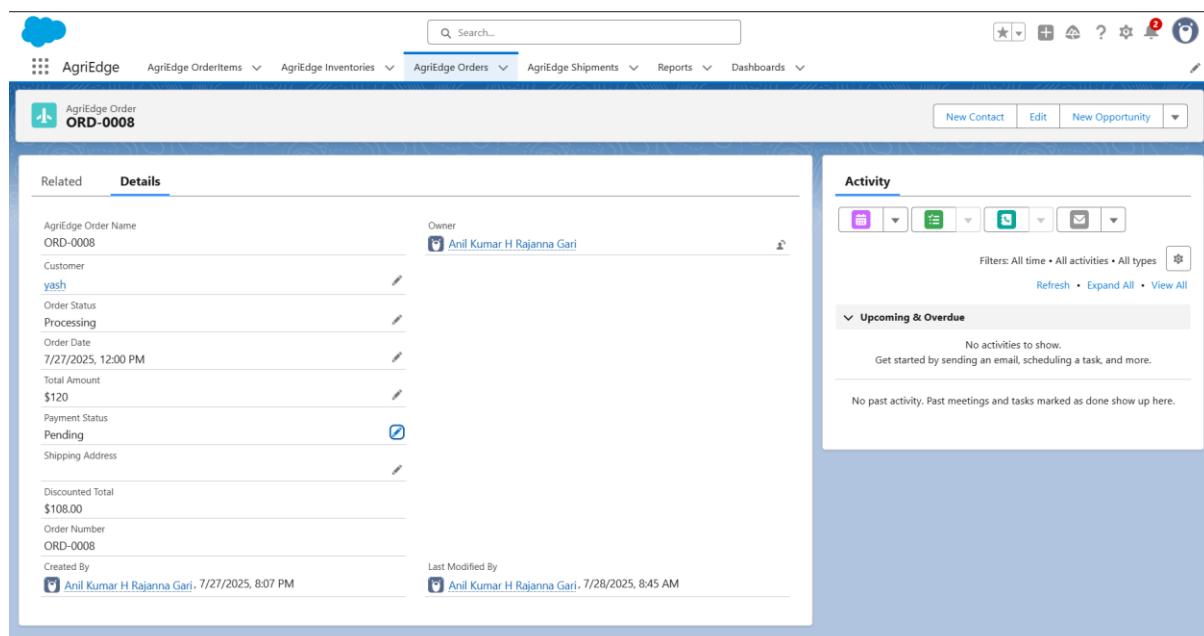
The company uses Salesforce to manage this order end-to-end. Here's how your system helps:

---

### Step 1: Order Creation (**AgriEdge\_Order\_\_c**)

A sales executive creates a new **AgriEdge Order** record:

- Customer: Yash
- Status: Pending
- Payment Status: Not Paid



The screenshot shows the Salesforce interface for creating a new order. The top navigation bar includes links for AgriEdge, AgriEdge OrderItems, AgriEdge Inventories, AgriEdge Orders, AgriEdge Shipments, Reports, and Dashboards. The main area displays the 'AgriEdge Order' record with ID 'ORD-0008'. The 'Details' tab is selected, showing fields for Order Name, Customer, Order Status, Order Date, Total Amount, Payment Status, and Shipping Address. The 'Owner' field is set to 'Anil Kumar H Rajanna Gari'. The 'Activity' sidebar indicates no upcoming or overdue activities.

## (AgriEdge Order)

---

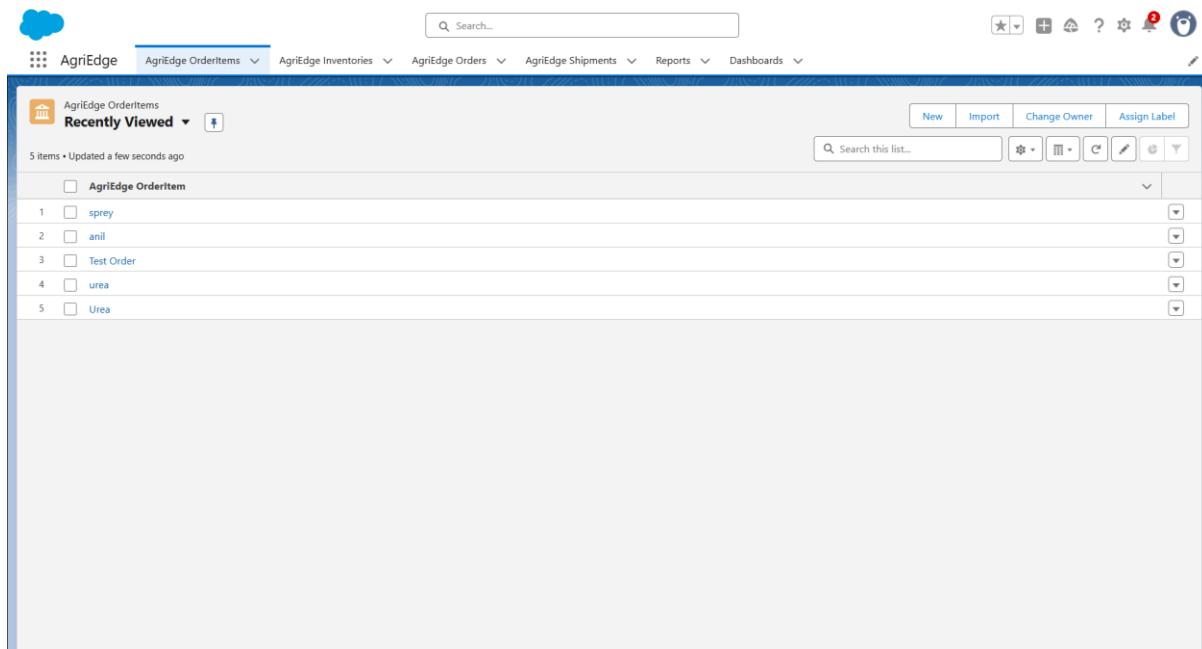
### Step 2: Adding Items (AgriEdge\_OrderItem\_\_c)

They add individual items to the order:

- Item 1: 10x Fertilizer @ ₹300 each
- Item 2: 5L Organic Pesticide @ ₹500/L

As soon as items are added:

- The system calculates the **total amount** using Apex (OrderTotalUpdater).
- If needed, order status is updated using trigger logic.



The screenshot shows the AgriEdge OrderItems page. At the top, there's a navigation bar with links for AgriEdge, AgriEdge OrderItems, AgriEdge Inventories, AgriEdge Orders, AgriEdge Shipments, Reports, and Dashboards. Below the navigation is a search bar labeled "Search...". The main area is titled "AgriEdge OrderItems Recently Viewed" and shows a list of 5 items updated a few seconds ago. The list includes:

- 1 sprey
- 2 anil
- 3 Test Order
- 4 urea
- 5 Urea

On the right side of the list, there are various buttons for managing the records, such as New, Import, Change Owner, and Assign Label. There's also a "Search this list..." field and some filter and sort options.

*(AgriEdge OrderItem)*

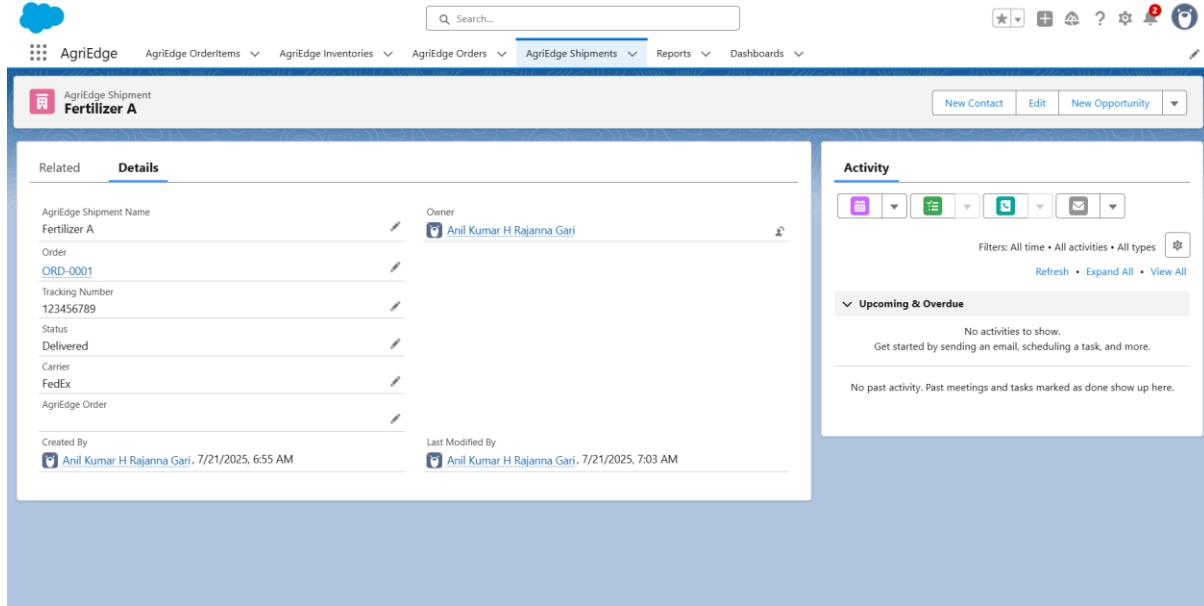
---

### Step 3: Shipment Creation (AgriEdge\_Shipment\_\_c)

Once payment is received:

- A shipment record is created with:
  - Tracking Number (auto-populated or manually added)
  - Carrier (e.g., FedEx)
  - Shipment Status: “In Transit” or “Delivered”

If no tracking number is added, validation logic prevents saving incomplete data.



*(shipment form with tracking and carrier fields.)*

## 🤖 Automation Using Triggers and Apex

- **Trigger** on AgriEdge\_OrderItem\_\_c: Automatically updates order total and order status whenever items are added/updated.
- **Helper Class**: Handles creation or update of shipment when order status changes to "Delivered".
- **Validation**: Ensures data integrity (e.g., tracking number is required for completed orders).

Class	Method	Duration	Result	Errors	Stack Trace
AgriEdgeOrderTests	testAgriEdgeOrderShipment	0:00	Pass		
AgriEdgeOrderTests	testAgriEdgeOrderTrigger	0:00	Pass		
AgriEdgeOrderTests	testAgriEdgeOrderTrigger...	0:00	Pass		
AgriEdgeOrderTests	testOrderStatusUpdater	0:01	Pass		
AgriEdgeOrderTests	testOrderTaskCreator	0:01	Pass		
AgriEdgeOrderTests	testOrderTotalUpdater	0:00	Pass		
AgriEdgeOrderTests	testSendOrderEmail	0:01	Pass		

*(code Coverage)*

The screenshot shows two separate instances of the Salesforce Apex class editor. The top instance displays the code for `OrderTaskCreator.apxc`, which contains logic for creating tasks for new orders. The bottom instance displays the code for `OrderItemTrigger.apxt`, which is a trigger for the `AgriEdge_OrderItem__c` object. Both instances show the code editor, a related entities sidebar, and a navigation bar at the top.

```

1 * public with sharing class OrderTaskCreator {
2
3     @InvocableMethod
4     public static void createTaskForNewOrder(List<Id> orderIds) {
5
6         // Validate input list
7         if (orderIds == null || orderIds.isEmpty()) {
8             System.debug('No order IDs received.');
9             return;
10        }
11
12        // Fetch users with Profile Name 'Platform 1'
13        List<User> platformUsers = [
14            SELECT Id
15            FROM User
16            WHERE Profile.Name = 'Platform 1'
17            AND IsActive = TRUE
18            LIMIT 10
19        ];
20
21        if (platformUsers.isEmpty()) {
22            System.debug('No active users found with Platform 1 profile.');
23        }
24    }
}

```

```

1 * trigger OrderItemTrigger on AgriEdge_OrderItem__c (after insert, after update) {
2     Set<Id> orderIds = new Set<Id>();
3
4     // Collect Order IDs from inserted/updated OrderItem records
5     for (AgriEdge_OrderItem__c orderItem : Trigger.new) {
6         if (orderItem.Order__c != null) {
7             orderIds.add(orderItem.Order__c);
8         }
9     }
10
11    if (!orderIds.isEmpty()) {
12        OrderStatusUpdater.updateOrderStatus(orderIds);
13        OrderTotalUpdater.updateOrderTotal(orderIds);
14    }
15 }

```

*(trigger code and class snippet in Salesforce Apex class editor.)*

## Result

Ramesh can now receive timely deliveries, track shipments, and AgriEdge's internal team can manage all orders, items, and logistics efficiently in one platform.

## Conclusion

The **AgriEdge Order and Shipment Management System** effectively demonstrates how Salesforce can be used to streamline agricultural product ordering, order item management, and shipment tracking in a structured and automated way. Through the use of **custom objects** like AgriEdge\_Order\_\_c, AgriEdge\_OrderItem\_\_c, and AgriEdge\_Shipment\_\_c, and supporting automation via **Apex Triggers and Classes**, the project successfully:

- Automates the calculation of order totals.
- Updates order statuses dynamically based on order item and payment activity.
- Ensures valid shipment creation with necessary data such as tracking numbers.
- Prevents data inconsistencies through validation rules and field checks.

This system provides **clear visibility** into the sales and delivery process for both business users and customers. It reduces manual intervention and increases operational efficiency—key benefits for any organization involved in product delivery and logistics.

---

### Future Scope

The current project lays the foundation for a robust agri-business order management system. There are several possibilities to enhance and scale the system in the future:

#### 1. Real-Time Inventory Integration

- Connect the system with **AgriEdge\_Inventory\_\_c** to automatically deduct stock when orders are placed.
- Show real-time product availability to avoid over-selling.

#### 2. Customer Self-Service Portal

- Enable customers (farmers, retailers) to **log in**, place orders, view order history, and track shipments.
- Use **Salesforce Experience Cloud** to build the portal.

#### 3. Mobile-Optimized Interface

- Use **Salesforce Mobile App** or **custom Lightning Web Components (LWC)** for on-the-go access to order and shipment data.

## 4. Advanced Reporting and Dashboards

- Build **dynamic dashboards** for admin users to monitor total orders, pending shipments, total revenue, etc.
- Add charts showing top-selling products and regional order patterns.

## 5. AI-Powered Recommendations

- Use **Salesforce Einstein** to suggest frequently ordered items or delivery time estimates based on historical data.

## 6. Third-Party Carrier Integration

- Integrate with APIs from logistics companies (e.g., FedEx, DHL) to auto-fetch tracking updates and display shipment progress inside Salesforce.

## 7. Enhanced Role-Based Security

- Fine-tune **profile and permission sets** for different users: admin, sales rep, warehouse staff, etc.