Ideation Phase

Brainstorm & Idea Prioritization

Field	Details
Date	**02 November 2025**
Team ID	**NM2025TMID05827**
Project Name	**Medical Inventory Management System**
Maximum Marks	**4 Marks**

Medical Inventory Management System

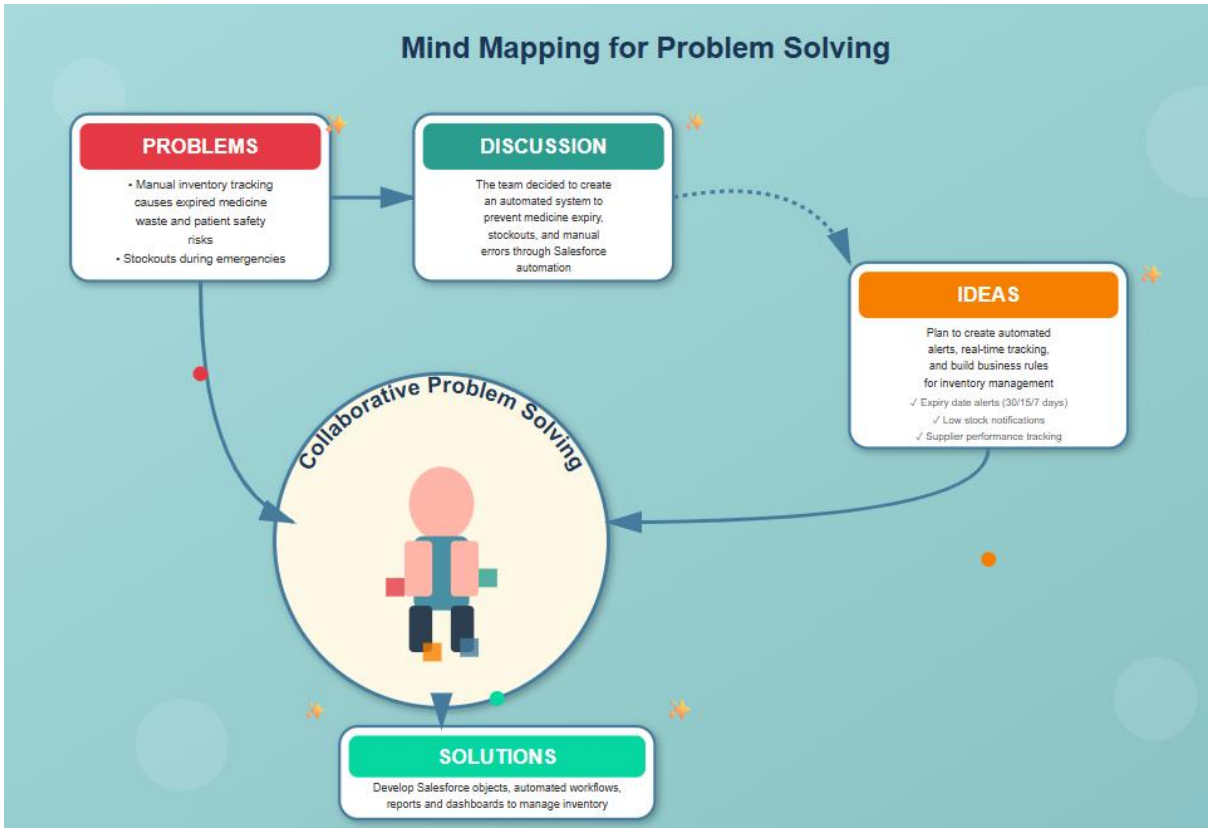
This guided project demonstrates how to build a comprehensive Salesforce-based medical inventory management system for healthcare facilities. It begins by creating custom objects for Products, Purchase Orders, Suppliers, and Inventory Transactions. A Lightning App is then created to provide a unified interface for managing all inventory operations.

This ensures important medical supply data is tracked accurately and prevents critical issues such as expired medicine dispensing, stockouts during emergencies, and inefficient procurement processes. Healthcare workers can access real-time inventory levels, receive automated expiry alerts, and track supplier performance.

The workflow also includes automated processes to validate data entry, calculate order totals, update actual delivery dates, and generate comprehensive reports. First, objects and fields are created to store all inventory data. Then, validation rules ensure data integrity by preventing invalid entries. Finally, flows and triggers automate business processes, while reports and dashboards provide actionable insights. This process helps healthcare facilities maintain patient safety, optimize costs, and ensure regulatory compliance through complete audit trails.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Mind Mapping for Problem Solving



Step-2: Brainstorm, Idea Listing and Grouping

Idea Listing

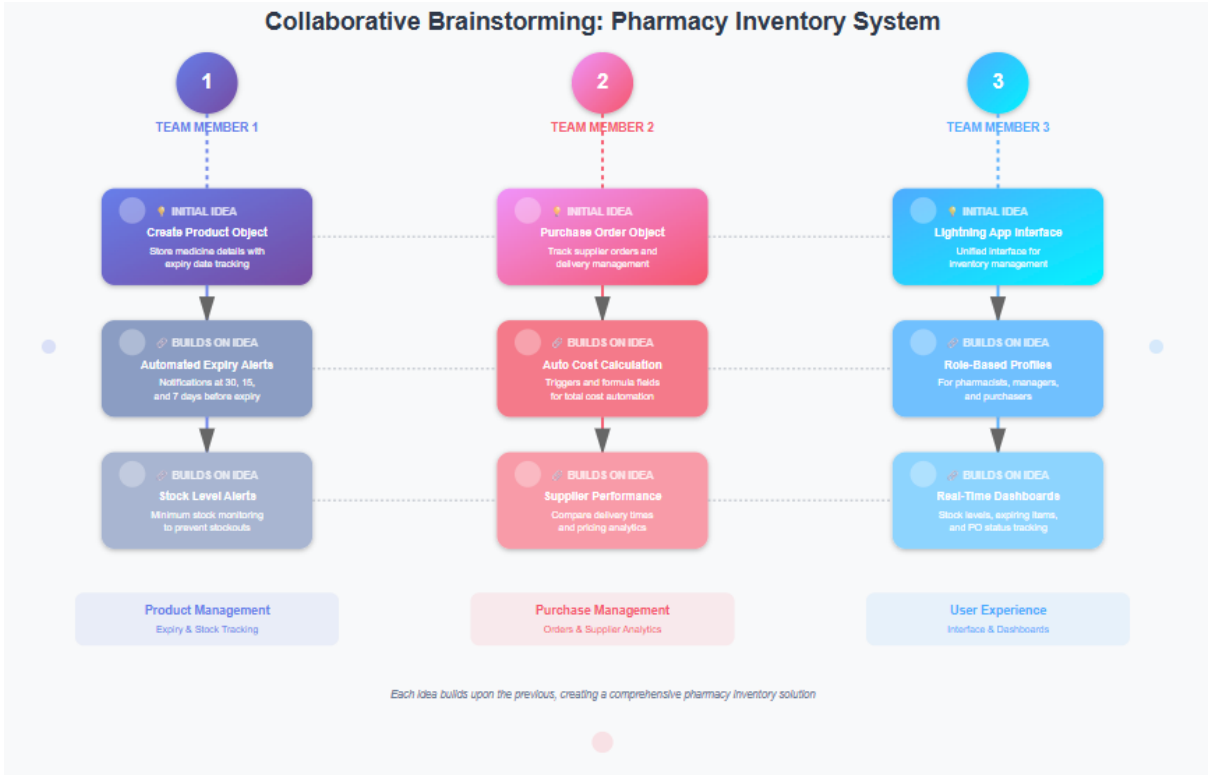


Fig2: Image that describes the work done by teammates

✅ **Brainstorm:**

Team members share ideas freely to explore solutions without judgment, encouraging creativity and participation.

✅ **Idea Listing:**

All ideas from the session are written down to capture every suggestion and ensure no input is overlooked.

✅ **Grouping:**

Similar ideas are organized into categories to identify patterns, highlight priorities, and simplify decision-making.

✅ **Action Planning:**

Chosen ideas are turned into clear steps with assigned responsibilities and timelines.

Step-3: Idea Prioritization

Fig3: Image of Steps for Medical Inventory Management System

Idea Prioritization:

Idea prioritization helps break down complex projects into clear, focused components. In this project, the main goal is to build a comprehensive medical inventory management system in Salesforce that prevents expired medicine dispensing, stockouts, and manual errors while improving operational efficiency.

This approach ensures that patient safety and data integrity are maintained during critical healthcare workflows. By prioritizing ideas, we can separate core inventory tracking features from advanced analytics and reporting capabilities.

It also helps in highlighting the importance of automation (expiry alerts, reorder workflows), real-time visibility (dashboards), and regulatory compliance (audit trails). Each step, from object creation to dashboard development, becomes easier to plan and implement.

Clear visual representations like diagrams and flowcharts can simplify communication among team members, stakeholders, and management. Overall, idea prioritization strengthens project clarity, ensures systematic development, and supports smooth execution of the 16-milestone Medical Inventory Management System.

PRIORITIZED IDEAS TABLE:

Priority	Feature/Milestone	Description	Impact
01	Create Product Object	Foundation for storing medicine/supply data with expiry dates	Critical- Core data model
02	Define Custom Fields	Add fields for stock levels, pricing, expiry, suppliers	Critical- Data completeness
03	Build Purchase Order Object	Track orders from suppliers with delivery dates	High- Procurement tracking
04	Create Supplier Object	Manage supplier contacts and performance data	High- Vendor management
05	Develop Inventory Transaction Object	Record all stock movements with audit trail	Critical- Compliance
06	Create Page Layouts & Tabs	User-friendly interface for data entry	Medium- User experience
07	Build Lightning App	Unified interface for all inventory operations	High- Accessibility
08	Validation Rules & Data Integrity	Prevent invalid data entry and ensure quality	Critical- Data accuracy

09	Create Profiles and Roles	Role-based access control for security	High- Security compliance
10	Permission Sets Configuration	Granular access for specific users	Medium- Advanced security
11	Build Flows for Automation	Automate delivery date updates and alerts	High- Operational efficiency
12	Create Triggers for Order Totals	Automatic calculation of purchase amounts	High- Data automation
13	Develop Reports and Dashboards	Real-time visibility into inventory status	Critical- Decision making
14	Testing and Validation	Ensure all features work correctly	Critical- Quality assurance
15	User Training and Deployment	Onboard healthcare staff to new system	High- Adoption success
16	Monitor and Optimize	Continuous improvement based on usage	Medium- Long-term success

GROUPED IDEAS BY CATEGORY

Data Foundation (Milestones 1-5)

- Create core objects: Product, Purchase Order, Supplier, Inventory Transaction
- Define custom fields for all data points
- Establish object relationships (lookups, master-detail)

User Interface (Milestones 6-7)

- Design page layouts for optimal data entry
- Create custom tabs for easy navigation
- Build Lightning App as central hub

Security & Validation (Milestones 8-10)

- Implement validation rules to prevent errors
- Configure profiles for role-based access
- Set up permission sets for advanced controls

Automation (Milestones 11-12)

- Create flows for business process automation
- Develop triggers for complex calculations

- Automate alerts and notifications

Analytics & Insights (Milestone 13)

- Build reports for supplier performance
- Create dashboards for real-time visibility
- Design charts for trend analysis

Quality & Deployment (Milestones 14-16)

- Conduct comprehensive testing
- Train end users on system
- Monitor usage and optimize